



THE HONG KONG
POLYTECHNIC UNIVERSITY

香港理工大學

Pao Yue-kong Library

包玉剛圖書館

Copyright Undertaking

This thesis is protected by copyright, with all rights reserved.

By reading and using the thesis, the reader understands and agrees to the following terms:

1. The reader will abide by the rules and legal ordinances governing copyright regarding the use of the thesis.
2. The reader will use the thesis for the purpose of research or private study only and not for distribution or further reproduction or any other purpose.
3. The reader agrees to indemnify and hold the University harmless from and against any loss, damage, cost, liability or expenses arising from copyright infringement or unauthorized usage.

If you have reasons to believe that any materials in this thesis are deemed not suitable to be distributed in this form, or a copyright owner having difficulty with the material being included in our database, please contact lbsys@polyu.edu.hk providing details. The Library will look into your claim and consider taking remedial action upon receipt of the written requests.



The Hong Kong Polytechnic University
Department of Electrical Engineering

**INVESTIGATION OF SIMULTANEOUS
LOCALIZATION AND MAPPING
(SLAM) IN DYNAMIC SETTINGS**

ZHANG Xinzheng

A thesis submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy

February 2009

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____ (Signed)

ZHANG Xinzheng (Name of student)

To my parents and my wife

Abstract

A crucial characteristic of an indoor autonomous mobile robot is its ability to determine its whereabouts and make sense of its environments. Simultaneous localization and mapping (SLAM) is also regarded as an essential behaviour for realization of other advanced tasks such as exploration and autonomous navigation. SLAM in static environments whereby the mobile robot is the sole moving object has been exhaustively studied in the last three decades. Real world, however, is generally dynamic, and the states of the objects are versatile over time. In this context, it is imperative to study SLAM in such environments. This thesis reports research carried out on a robust mapping methodology via robust statistics theory and develops a new set of SLAM strategies based on sensor fusion viewpoint by distributed fusion technology, Bayesian inference, and information theory. Additionally, the data association problem in SLAM is also considered.

At first, occupancy grid-based and segment-based maps are studied along with three perception systems involving only ultrasonic sonar, only laser rangefinder and sonar plus monocular camera and are validated in both static and dynamic environments. The main purpose is to decide an effective map model and sensor configurations to be applied in subsequent studies. These studies led to adoption of the segment-based map integrated with laser rangefinder and its fusion with vision and ultrasonic sensors. This arrangement has been treated as the experimental framework for the rest of studies in the thesis.

Estimation algorithms based on the ordinary least square fail to extract the features in the presence of moving objects. In this thesis, a robust regression model is utilized via a robust estimate called MM-estimate that fits the blurred data well, and provides a reasonable segment prediction. This robust regression model is

embedded into the Extended Kalman Filter (EKF) SLAM to remove the dynamic features which correspond to the moving objects and sensor noise, and enhances the performance of SLAM procedure. In the EKF-SLAM, the data association problem is revisited and an optimal graph theory based approach is proposed. It is mathematically proved that optimally solving the minimum weighted bipartite graph matching problem is equivalent to optimally resolve the data association problem.

For some special cases, however, the robust regression model does not function properly. When the dynamic objects move slowly or momentarily pause for a while, they are erroneously regarded as line segments. Hence, an indirect sensor fusion strategy is presented, which consists of two aspects. The first is a feature fusion based on the Bayesian inference which synthesizes line segments generated by a robust regression model from laser rangefinder with static line features extracted from a monocular camera. This policy eliminates any pseudo segments that will appear from momentary pausing of dynamic objects in laser data. The second is a modified multi-sensor point estimation fusion that amalgamates two individual EKF-SLAM algorithms: monocular and laser SLAM. It is mathematically proved that the covariance of the state variables in fused SLAM is reduced compared with those of individual SLAM, and the accuracy of localization is improved. Particularly, for monocular SLAM the thesis suggests another data association technique based on the homography transformation that relaxes the pleonastic computation.

The indirect feature fusion procedure only makes the hypothesis test on the removal of pseudo segments. Considering this deficiency, a modified feature fusion process named direct fusion management is proposed, which immediately combines the homogeneous parameters of segments extracted from sonar, laser rangefinder and camera by means of information theory. Also the associative features from different sensors are determined by information entropy. The fusion

algorithm is a simple and general framework, which borrows the idea of information entropy weight in decision analysis discipline. Furthermore, the entropy weight is introduced into the parameter covariance fusion processing. That is allocating the weight to the related covariance matrix and applying the covariance intersection algorithm for these weighted covariance matrices to derive an amalgamated reduced covariance matrix. The fused features contribute to the EKF-SLAM and decrease the error of robot position compared with the results without fusion.

In essence, the thesis presents novel methodologies for SLAM in dynamic settings. The above solutions are validated by extensive simulation and experimental studies.

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to my supervisors Dr. Y.K. Wong and Prof. A.B. Rad for their invaluable guidance and supervision, and kind and continuous encouragement throughout this research project.

Special thanks to Prof. A.B. Rad, who provided me an opportunity to visit the Simon Fraser University as a visiting student and gave me number of valuable comments and inspirations through my extended study there.

I would also like to thank my friends Mr. Paul, Quoquan HUANG and Mr. Yan LIU who have given useful suggestions, criticisms, and warmly helped my research.

The financial support by the award of the studentship from The Hong Kong Polytechnic University for the three-year study is also acknowledged.

Finally but not the least, I would like to thank my wife Iris, Jianfen ZHANG who always encourages me to work hard and spends her much time to take care of my life so that I can concentrate on my research and study.

Table of Contents

Chapter 1 Introduction	1-1
1.1 Background and Motivation.....	1-2
1.2 Research Outline	1-4
1.3 Organization of the Thesis	1-6
1.4 Statement of Originality.....	1-8
1.5 Current Outcomes	1-8
Chapter 2 Literature Review	2-1
2.1 Introduction	2-1
2.2 Perception Systems	2-2
2.3 Simultaneous Localization and Mapping.....	2-5
2.4 Sensor Fusion	2-14
2.5 Data Association.....	2-16
2.6 Summary	2-19
Chapter 3 Comparison of Mapping Methodologies	3-1
3.1 Introduction	3-1
3.2 Perception Systems and Map Representation	3-2
3.3 Experimental Results	3-4
3.4 Discussion	3-8
Chapter 4 Robust Regression Model for SLAM	4-1
4.1 Introduction	4-1
4.2 Robust Regression Model	4-3
4.3 SLAM Algorithm with Robust Regression Mapping	4-8
4.3.1 Sensor readings grouping based on Robust Regression Model	4-8
4.3.2 Computation of Segment Parameters.....	4-12
4.3.3 Extended Kalman Filter based SLAM (EKF-SLAM)	4-13
4.4 Graph Theory based Data Association.....	4-17
4.4.1 Formulation of IP Problem.....	4-17
4.4.2 Formulation of LP Relaxation.....	4-20
4.4.3 Minimum Weight Bipartite Perfect Matching.....	4-21
4.4.4 Algorithm based on Weighted Bipartite Matching.....	4-24
4.5 Experimental Studies	4-25

4.5.1 Validation of Graph Theoretic Data Association Algorithm	4-25
4.5.2 Validation of Robust Regression Model for SLAM.....	4-31
4.6 Discussion	4-36
Chapter 5 Sensor Fusion for Pseudo Features Elimination	5-1
5.1 Introduction	5-1
5.2 Line based EKF Monocular SLAM	5-3
5.2.1 Motion Model of the Monocular Camera/Robot	5-4
5.2.2 Line Extraction and Measurement Model.....	5-5
5.2.3 EKF based Monocular SLAM	5-8
5.3 Homography Transformation based Data Association.....	5-10
5.4 Sensor Fusion Techniques	5-14
5.4.1 Line Features Fusion	5-14
5.4.2 Modified Multi-sensor Point Estimation Fusion.....	5-15
5.5 Experimental Studies	5-17
5.5.1 Testing the Feature Fusion Strategy	5-18
5.5.2 Testing the Fusion SLAM based on Modified MPEF.....	5-20
5.5.3 Testing the HTDA	5-22
5.6 Discussion	5-23
Chapter 6 Direct Sensor Fusion Management	6-1
6.1 Introduction	6-1
6.2 Feature Extraction from Rangefinder Sensors	6-2
6.3 Feature Extraction from Vision Systems.....	6-6
6.3.1 Representation for Lines Obtained from Monocular Camera.....	6-6
6.3.2 Representation for Lines Obtained from Stereo Camera	6-10
6.4 Direct Sensor Fusion Management Based on Information Theory.....	6-13
6.4.1 Information Entropy Based Associated Features Determination ..	6-13
6.4.2 Entropy Weight Fusion Management for Sensor Measurements and Variance.....	6-17
6.4.3 Practical considerations.....	6-20
6.5 Experimental Studies	6-21
6.5.1 Validation on Virtual Rangefinder Model	6-22
6.5.2 Validation on Direct Sensor Fusion Management.....	6-23
6.6 Discussion	6-34

Chapter 7	Conclusions	7-1
7.1	Major Contributions	7-1
7.2	Suggestions for Future Research.....	7-4
7.3	Closing Remarks	7-5
Appendix	A-1
A.	Coordination Reference Systems and Transformation.....	A-1
B.	The Derivation of Modified Multi-sensor Point Estimation Fusion	A-6
C.	Information Entropy	A-12
Bibliography	B-1

List of Figures

Figure 1.1 Research outline	1-5
Figure 2.1 Fully autonomous capabilities	2-3
Figure 3.1 Pioneer 2DX and loaded sensors	3-4
Figure 3.2 Hand-measured blueprint of the corridor	3-5
Figure 3.3 Results of the segment-based map in the static corridor.	3-6
Figure 3.4 Results of FTGBM in the dynamic corridor.....	3-8
Figure 4.1 Fitted line determined by MM-estimate versus OLS estimate for the data with one outlier	4-4
Figure 4.2 Fitted line determined by MM-estimate versus OLS estimate for the data after removal of the outlier.....	4-8
Figure 4.3 The process of building segment set.....	4-9
Figure 4.4 Regression parameters catastrophe.....	4-10
Figure 4.5 Measurement model and the geometric relationship between the robot and features	4-14
Figure 4.6 Simulation setup.	4-27
Figure 4.7 Estimation errors of robot pose in the environment with sparse landmarks	4-28
Figure 4.8 Estimation errors of robot pose in the environment with dense landmarks	4-29
Figure 4.9 Estimation errors of landmarks in the environment with sparse landmarks	4-30
Figure 4.10 Estimation errors of landmarks in the environment with dense landmarks	4-31
Figure 4.11 Hand-measured map of the corridor	4-31
Figure 4.12 A sequence of local maps for static corridor.....	4-32
Figure 4.13 Final result of SLAM in static environment.....	4-33
Figure 4.14 A sequence of local raw laser data and local maps for dynamic corridor	4-34
Figure 4.15 Global raw laser data and final result of SLAM in dynamic environment	4-35
Figure 4.16 Fitted lines estimated by RANSAC and MM-estimator.....	4-36
Figure 4.17 Fitted lines estimated by RANSAC, MM-estimator and OLS	4-38
Figure 4.18 Pseudo segment extraction case	4-38

Figure 5.1 The global and camera reference systems	5-5
Figure 5.2 Line feature extraction from image.	5-6
Figure 5.3 Matched points determined by SIFT descriptors.....	5-11
Figure 5.4 HTDA algorithm.....	5-13
Figure 5.5 Modified MPEF algorithm.	5-17
Figure 5.6 Local mapping results at the 33rd sample time.	5-19
Figure 5.7 The fusion SLAM results.	5-20
Figure 5.8 The estimate covariance of fused and individual robot state.....	5-21
Figure 5.9 The estimate covariance of fused and individual features.....	5-21
Figure 5.10 The example of HTDA.	5-22
Figure 5.11 Difference between the actual and estimated location of end 2 from 40th to 60th sample time.....	5-23
Figure 6.1 Segment extraction process for sonar data	6-5
Figure 6.2 An example of segment extraction from sonar data.	6-6
Figure 6.3 Geometric relationship on various coordinate references.	6-10
Figure 6.4 An example of segment extraction from stereo camera.	6-12
Figure 6.5 Identical sensor data in different sampling time.....	6-15
Figure 6.6 Positions of measured points in control lab.....	6-22
Figure 6.7 Obtained lines from the monocular camera outside the control lab....	6-23
Figure 6.8 EKF SLAM based on proposed sensor fusion strategy in simulated environment.	6-25
Figure 6.9 Comparison on the covariance of robot pose and feature parameters, and consistent validation for robot pose when using proposed sensor fusion management.	6-27
Figure 6.10 The sonar and laser readings at the 54th laser sampling time.	6-28
Figure 6.11 The entropy comparison bar.	6-28
Figure 6.12 Segments obtained from sonar and laser.	6-30
Figure 6.13 Associative features	6-31
Figure 6.14 A wide glass wall in front of the robot in the wide corridor, and can not be detected by camera but can be done by sonar.	6-32
Figure 6.15 The final map after combing the measurements of sonar, laser and camera.	6-32
Figure 6.16 Comparison on the covariance of robot pose.	6-33
Figure A.1 The coordinate systems of global, robot and rangefinders.	A-1

Figure A.2 Ideal pinhole camera model.....	A-3
Figure A.3 Various coordinate systems for Pioneer 3DX.....	A-5
Figure A.4 Transformation from camera frame to virtual local frame	A-5

List of Tables

Table 3.1 Comparative results of different map representation and sensor systems	3-9
Table 4.1 The process for detecting the special case via VROS	4-11
Table 5.1 Intrinsic Parameters of Canon VCC4.....	5-18
Table 5.2 The hypothesis test of feature fusion.....	5-19
Table 6.1 Range and Bearing Comparison.....	6-23
Table 6.2 Lines parameters comparison.....	6-23
Table 6.3 Parameters in experimental studies	6-24
Table 6.4 Results before and after sensor fusion	6-25
Table 6.5 Errors between sonar segment 5 and all laser segments	6-28
Table 6.6 Parameters for the segments from different sensors	6-30
Table 6.7 Fused parameters.....	6-31

Chapter 1 Introduction

It is widely believed that Karel Capek, the celebrated Czech playwright coined the term “Robot” in 1920. Almost ninety years passed on, the realm of robotics has allured public at large and robots have captivated humans of all walks of life. They are used in kindergartens and primary schools, designed in universities and research institutions, and employed in many industrial applications. The public fascination with robots has generated huge interest in entertainment industry as well. The recent Hollywood blockbuster “*I, Robot*” was a technophobic display of a not too distant future where robots were intelligent and could even commit crimes. It is plausible that future robots are indeed going to demonstrate human-level intelligence and be part of our social settings and even refuse to comply with our agreed social norms. However, this thesis objective is far-less unassuming and just addresses the relatively less complex problems of “where am I?” and “how to make sense of an unknown environment?”. The successful solution to these two problems is the precondition to more advanced functions and capabilities. The last two decades has witnessed a remarkable body of work devoted to provide viable solutions to accomplish these tasks. The literature refers to the problem as “SLAM” which stands for Simultaneous Localization and Mapping. A particular class of robots identified as Autonomous Mobile Robots (AMR) is often employed as a platform for development and research in this area. The ideas and designs, however, can readily be applied to humanoid robots as well.

The research reported in this thesis is the contribution of the author in this exciting field. In a nutshell, the main objectives of this PhD research are:

- To compare several perception systems and map representations and determine a rational configuration for the whole research;

- To contrive robust algorithms for handling moving objects in order to improve the performance of the mapping process as well as the SLAM in dynamic environments;
- To suggest and develop sensor fusion systems for fusing the information of vision and rangefinder sensors including ultrasonic sonar and laser, and hence to develop a set of new sensor-fusion-based SLAM strategy;
- To propose and prove an optimal data association algorithm, and introduce a new method of data association in vision-based SLAM.

1.1 Background and Motivation

Autonomous mobile robots have already taken up ranks in the human daily life. Domestic service robots such as vacuum-cleaners and garden robots relieve us of such tedious chores as cleaning, ironing, and peeling vegetables. Another notable application of domestic robots has been in the area of home surveillance. Not only can robots watch the house like guard dogs, but they can sound an alarm in the event of a break-in and take appropriate action in high-risk situations like gas leaks or flooding. Especially in Hong Kong, there is much room for applications of robots in indoor environments. The first robot service restaurant opened its doors at Tseung Kwan O Center in July 2006¹. The restaurateur hired three robots *waiters*, one for welcoming customers, one for dance show, and one for moving in the restaurant, taking charge of order dishes and delivering the menus. The extended development of AMR focuses on the autonomous vehicles, i.e. smart cars that have the capability of correcting the driver error and even taking control of the wheel.

The birth of the first mobile robot can be traced back to World War II as a result of technical advances on a number of relatively new research fields like computer science and cybernetics. However, they were mostly flying bombs. Examples are smart bombs that only detonate within a certain range of the target, the use of

¹ It is available at http://news.xinhuanet.com/tai_gang_ao/2006-07/18/content_4846755.htm, and the related videos are on <http://hk.youtube.com/watch?v=r0C7aFzXlq8&feature=related>

guiding systems and radar control. The ancestor of the real mobile robots is the autonomous guided vehicle system developed by Barrett Electronics Co. USA in the early 1950s, but it had limited functions and only could navigate in a small and within a specified range. In 1995, the programmable mobile robot *Pioneer* that produced by MobileRobots Co. became commercially available at an affordable price, triggering an upsurge in robotics research and university study over the next decades as mobile robotics became a standard part of the engineering and computer science curriculum.

Various amazing abilities of the AMR can be established on the basis of perception of environments, localization of the robot position and many other intelligent autonomous tasks. However, the primary challenge facing robotics today is that of situation awareness. We can build very capable hardware that can perform useful tasks under human control, but the hardware can not give the robot the same level of awareness as that of a human. Similarly, we can mount sophisticated cameras to a robot and send the signal into a computer; however, the ability to convert the pixel data into an accurate 3D world model seems to involve substantial real-world knowledge. This manifests that nowadays the major problem is not the hardware devices but the software design and the embedded intelligence.

To design and implement these different competences in mobile robots require the synergy of technologies borrowed from diverse research disciplines. Numerous researchers in robotics community devoted themselves to solving the challenging problems encountered in the implementation of autonomous tasks. Map building, localization, and SLAM are the basic abilities of AMR for advanced capabilities such as exploration, navigation, path planning, and even the cooperation among multiple robots.

SLAM is defined as the problem of building a map while at the same time localizing the robot within that map. In practice, these two problems cannot be

solved independent of each other. Before a robot can answer the question of what the environment looks like given a set of observations, it needs to know from which location those observations have been made. At the same time, it is hard to estimate the current position of a vehicle without a map. Therefore, SLAM is often referred to as a chicken and egg problem: A good map is needed for localization while an accurate pose estimate is needed to build a map, which makes the SLAM as an open issue and attracts abundant researchers. The prime study on SLAM focused on the operation in static environments where only the AMR was the moving object. It is obvious that this assumption is not practical. Recently, the research of SLAM permeates the area of dynamic environments based on the mature work on static environments. Many researchers have developed various methods to solve the problem and enhance the performance of SLAM, but to the best of our knowledge few of them have been involved in a more general algorithm to concurrently handle miscellaneous moving objects with different dynamic degrees, such as rarely moved trash bins, frequently walking people in the office, etc.

In this thesis, the author would not attempt finding a general SLAM framework for covering very complex dynamic environments but address selected problems in an interesting way. The focus is mainly on the SLAM and its associated problems in dynamic environments.

1.2 Research Outline

The studies undertaken in this thesis can be classified into three phases as articulated in Figure 1.1. In Phase I, the proper perception system and map representation is determined through comparison of several common sensors and methods of map description. To address the moving objects in the environments, a robust regression model for mapping is proposed and this model is able to tackle most dynamic objects.

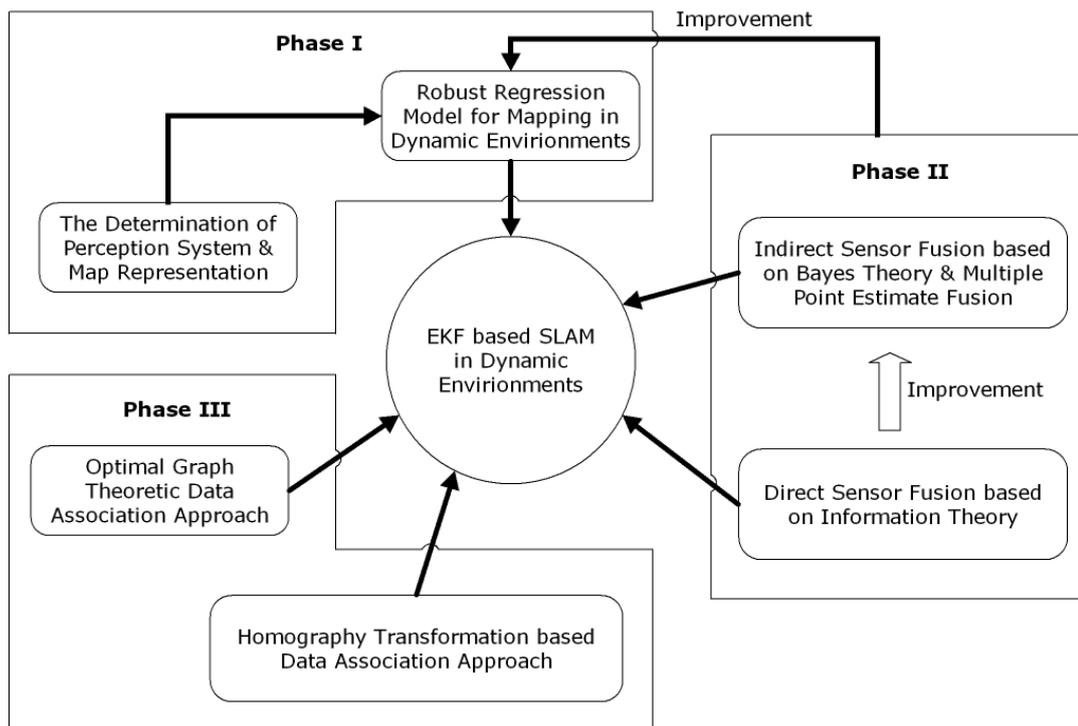


Figure 1.1 Research outline

On the basis of primary work in Phase I, Phase II concentrates on the application of sensor fusion technology for SLAM. It consists of two aspects: indirect and direct paradigms. The indirect sensor fusion employs the Bayes theory to remove the potential pseudo segments related to dynamic objects and multi-sensor point estimation fusion technique to incorporate two individual SLAM procedures so as to reduce the errors generated in any single SLAM. The reason why it is called *indirect* is that the parameters of map features are not fused but used for removal of erroneous features. However, direct sensor fusion remedies this disadvantage. It integrates the feature parameters as well as their covariance via information theory. In addition, associative features extracted from different sensors are confirmed also by the information entropy.

Phase III focuses on another important problem in SLAM: data association. This problem is reviewed from the viewpoint of graph theory and an optimal approach based on the minimum weight bipartite perfect matching graph is mathematically

proved. As for data association in vision based SLAM, a homography transformation based data association method is designed to release the computational complexity.

1.3 Organization of the Thesis

The rest of the thesis is organized in a sequential fashion and report the findings of each phase of the project in a logical yet chronological manner.

Chapter 2 lays the background for the rest of the thesis. It presents a detailed yet highly selective literature review on the subject areas relevant to the topics of this study. It incorporates various perception systems for autonomous mobile robots, the state of art of SLAM in both static and dynamic environments, data association, and sensor fusion.

In Chapter 3, on the basis of our group previous work, we compare several perception systems and map representation for the map building process in the static and dynamic environments. The purpose is to make a rational sensor systems and environment representation and to lay a solid foundation for the studies in the entire thesis. The determined perception system and mapping method will be employed in the SLAM and related topics of following chapters.

A robust regression model is proposed for segment-based mapping in Chapter 4. It adopts the MM-estimate to consider the noise of sensor data and the outliers that correspond to moving objects in dynamic environments. The MM-estimates are interesting as they combine high efficiency and high breakdown point in a simple and intuitive way. This robust regression model is integrated with the extended Kalman filter (EKF) to implement the SLAM in dynamic circumstances. Additionally, we formulate the data association that is a critical aspect in SLAM as an equivalent minimum weight bipartite perfect matching problem that can be

optimally solved. We also mathematically prove the optimality of the graph theoretic approach.

Chapter 5 presents an indirect sensor fusion strategy applied for SLAM in dynamic environments. The algorithm design consists of two aspects. One is a feature fusion which synthesizes line segments generated by a robust regression model from laser rangefinder with static line features extracted from monocular camera. This policy eliminates any pseudo segments that will appear from momentary pausing of dynamic objects in laser data. The other is a modified multi-sensor point estimation fusion (MPEF) that amalgamates two individual EKF-SLAM algorithms: monocular and laser SLAM. We mathematically prove that the covariance of the state variables in fused SLAM is reduced compared with those of individual SLAM, and the accuracy of localization is improved. Also, a new data association technique based on the homography transformation for monocular SLAM is suggested, which relaxes the pleonastic computation.

We present a novel direct sensor fusion management in Chapter 6. It is formulated by concepts in information theory and incorporated into EKF-SLAM for reducing the uncertainty of map features and the robot position. We borrow the idea of the entropy weight from the decision analysis area and design a simple and general entropy weight method which considers the measurements and measurement variance to integrate the sensor information from various devices. It directly combines the feature parameters represented in the similar form. Similarly, the entropy weight is introduced into the covariance intersection (CI). A modified CI technique is proposed for fusing measurements covariance matrices of different sensors. In addition, an information theoretic algorithm via computing the error entropy is developed to confirm the associative features obtained from different sensor devices. The proposed method is general, has no extra postulated conditions, and its implementation is simple.

Chapter 7 encapsulates the findings of this research work and draws conclusions. It also provides some suggestions for further work in this area of research.

1.4 Statement of Originality

The original contributions or important findings by the author in this PhD research are elaborated in the following statements:

- ❖ Design of a robust regression model by MM-estimate for segment based mapping and SLAM procedures to remove the features corresponding to different dynamic objects;
- ❖ Design of an indirect and a direct sensor fusion methods respectively through Bayes theory & MPEF method and information theory;
- ❖ Formulation and proof of an optimal data association scheme via the graph theoretic approach, and modified MPEF;
- ❖ Design of a homography transformation based data association for monocular based SLAM.
- ❖ Design of a virtual rangefinder model for line extraction from a monocular camera.

1.5 Current Outcomes

At the time of writing this thesis and based on the findings reported here, two international journal papers have been published, another three journal papers have been submitted to international robotics journals, and two conference papers were presented at IFAC World Congress, one conference paper is under review. These papers are listed as:

Journal Papers

1. Zhang, Xinzheng, Rad, A., Wong, Y., Huang, G., Ip, Y., & Chow, K. (2007). A Comparative Study of Three Mapping Methodologies. *Journal of Intelligent and Robotic Systems*, 49(4), 385-395.
2. Zhang, Xinzheng, Rad, Ahmad, & Wong, Yiu-Kwong (2008). A Robust Regression Model for Simultaneous Localization and Mapping in Autonomous Mobile Robot. *Journal of Intelligent and Robotic Systems*, 53(2), 183-202.

Journal Papers under Review

1. Zhang Xinzheng, Rad Ahmad, & Wong Yiu-Kwong (2008). Sensor fusion of monocular camera and laser rangefinder for line-based SLAM task in autonomous mobile robots. Submitted to *Autonomous Robots*.
2. Zhang Xinzheng, Rad Ahmad, & Wong Yiu-Kwong (2008). Sensor Fusion for SLAM based on Information Theory. Submitted to *Journal of Intelligent and Robotic Systems* after revision.
3. LIU, Yan, Zhang, Xinzheng, Rad, A.B., Ren, Xuemei & Wong, Yiu-Kwong (2008). Entropy Based Robust Estimator and its application to Line-based Mapping. Submitted to *Robotics and Autonomous Systems*.

Conference Papers

1. Zhang Xinzheng, Rad Ahmad, & Wong Yiu-Kwong (2008). A Virtual Range Finder based on Monocular Vision System in Simultaneous Localization and Mapping. *Proceeding of the 17th IFAC World Congress*, July 6-11, 2008, Seoul, Korea, 2369-2341.
2. Huang Guoquan P., Zhang Xinzheng, Rad Ahmad, & Wong Yiu-Kwong (2008). An Optimal Graph Theoretic Approach to Data Association in SLAM. *Proceeding of the 17th IFAC World Congress*, July 6-11, 2008, Seoul, Korea, 14669-14674.

3. Liu Yan, Rad Ahmad, Ren Xuemei and Zhang Xinzheng (2009). Simultaneous Localization and Mapping with Entropy Based Robust Estimator. Submitted to *48th IEEE Conference on Decision and Control*, Shanghai, P.R.China

Chapter 2 Literature Review

2.1 Introduction

With the purpose of setting the scene for the rest of this thesis and an intention to present the background and insight to the core of the studies endeavored in this research, this chapter aims to convey a thorough yet highly selective review of related work by other researchers.

How can an autonomous mobile robot navigate unsupervisedly in an unknown and uncertain real-world environment to fulfill its tasks? How does the robot learn the environment that it is supposed to navigate? Leonard and Durrant-Whyte (1991a) summarized these general problems of mobile robot navigation. Such navigation strategies began with wire-guidance in the 1970s and progressed in the early 2000s to beacon-based triangulation. Current commercial robots autonomously navigate based on sensing natural features. The realization of simultaneous localization and mapping (SLAM) as the basic ability of an autonomous mobile robot is a crucial step for meaningful implementation of navigation, exploration and other advanced autonomous tasks. Many researchers devoted themselves to reaching the solution for this basic capability. In the early stages, the studies focused on the map building or localization separately. Those works assumed that some part of SLAM was completed manually. For example, the localization algorithm usually used the pre-acquired maps; as well the mapping process was achieved by human-guided localization. The explosive research occurred after the pioneering work on SLAM (Randall Smith, Self, & Cheeseman, 1988) had been presented. Various methodologies have been proposed to address the SLAM problem. The underlying difficulties in SLAM were ignored in earlier studies but have been pursued relentlessly in the last two decades. Furthermore, some SLAM methodologies are incorporated with other advanced autonomous behaviors such as exploration in

order to accomplish multiple tasks concurrently. Most recently, with the development of computer vision techniques, the algorithms in this field open a new area for SLAM and the corresponding research topic is referred to as Structure from Motion (SFM) (Dellaert, Seitz, Thorpe, & Thrun, 2000). It attempts the process of finding a three-dimensional structure by analyzing the motion of an object over time. Figure 2.1 illustrates the essential capabilities required in a fully autonomous mobile robot and the associative technologies. It also depicts an overview of the main scope of this thesis.

As there is a vast amount of research and writing which is relevant to autonomous behaviors in robotics community, in this chapter a few aspects interrelated with this thesis will be mainly concentrated on: Section 2.2 describes various perception systems applied in robotic area. Section 2.3 states the SLAM implemented in static and dynamic environments, as well the SLAM with vision system is discussed. A brief literature review in sensor fusion is included in Section 2.4. Section 2.5 introduces another important problem in SLAM—data association. Finally, Section 2.6 summarizes this chapter.

2.2 Perception Systems

Perception systems that are the key requirements for any but the simplest behaviors for autonomous mobile robot consist of the diverse sensor devices for acquiring the knowledge about the environments. These sensors accompanied by algorithms provide the ability of estimating about the current state of the circumstances. With different characteristics, they are categorized as two predominant paradigms: proprioceptive and exteroceptive sensors which provide the interior and exterior information respectively about the robot itself and its ambient environments.

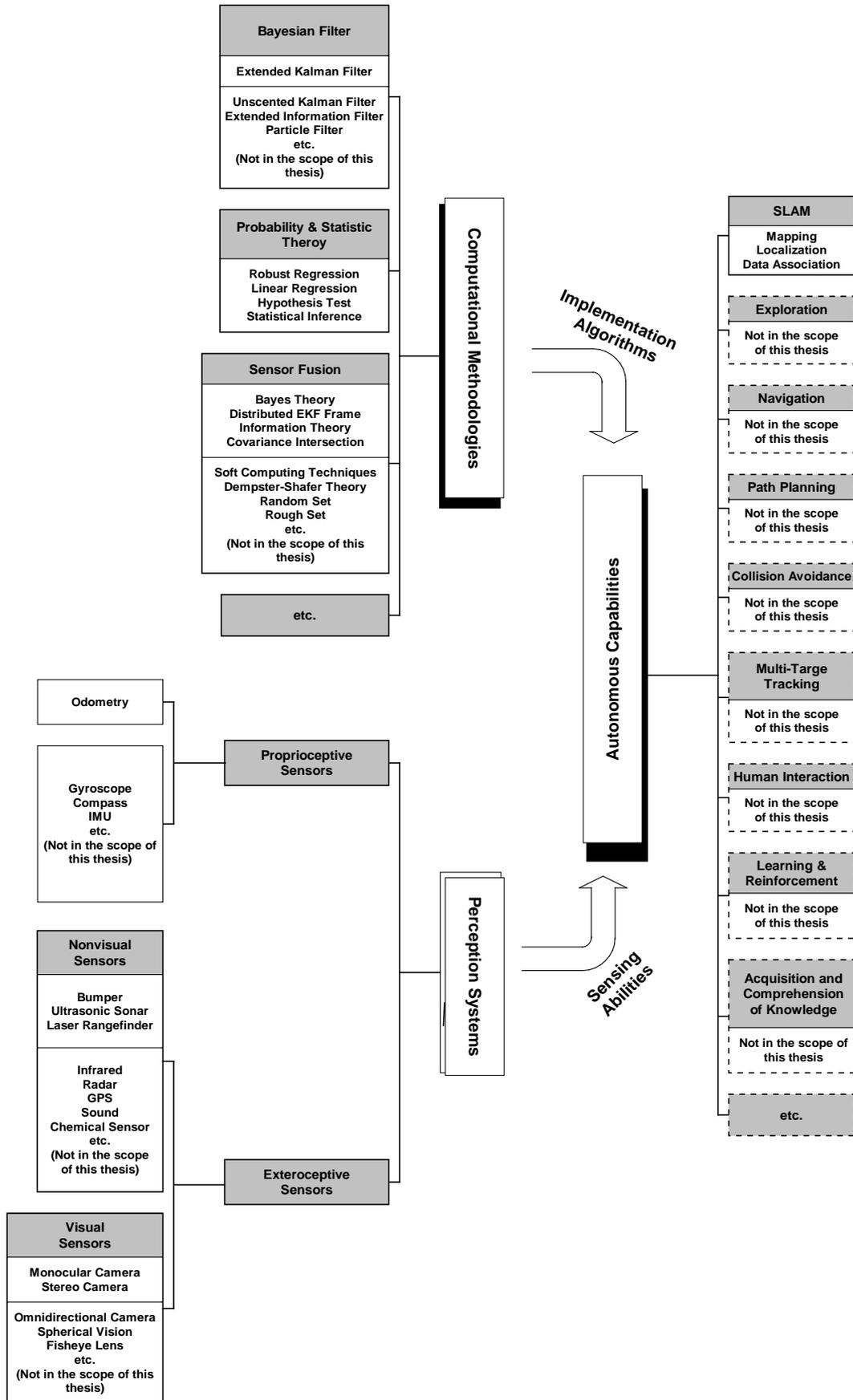


Figure 2.1 Fully autonomous capabilities

There are many proprioceptive sensors such as gyroscope, compass, inertial measurement unit (IMU), etc. A popular class, odometry measures the information of the wheel translation and rotation velocity, acceleration and robot heading; and estimates the robot pose by dead reckoning algorithm (Bowditch, 2002). It is a simple mathematical procedure of estimating robot current pose based upon a previously determined pose, and advancing that pose based on the known speed, elapsed time, and the course. However, the main disadvantage induced by dead reckoning is its unbounded accumulated errors because of the inherent property of integration process. Typical dead-reckoning errors will become so large that the robot's internal position estimate may be unacceptably wrong (Gourley & Trivedi, 1994). Therefore, exteroceptive sensors are considered to continually assist and gradually decrease such errors.

Exteroceptive sensors provide the robot with the information on the surrounding environment allowing the robot to interact with the world - roughly similar to human perception systems. Tactile sensors like bumpers and whiskers, as the name implies, sense direct physical contact between the robot and an object of interest. They also provide information about forces and torques transferred between the robot and other objects. Time of Flight system, encapsulating typical ultrasonic sonar and laser rangefinder, regarded as proximity sensors supplies the estimates of the distance (range) to objects in the environment. Ultrasonic sonar is the most common technique applied in the mobile robotics applications due to the ready availability, low cost, and easy interface. It has been used for various tasks in robotic field, for example localization, SLAM, exploration and so on. In comparison with ultrasonic sonar, laser rangefinder exhibits a much more accurate performance and repeatability. It is an effective alternative to other sensing technologies for robot pose estimation and the related scan-matching algorithm (Diosi & Kleeman, 2005) has been proposed. Humans can acquire massive information through their eyes and are able to make exceptionally good estimates about their local environment before and while navigating in known and unknown

environments. Hence, it is obviously natural to consider the vision system as an alternative sensor for a mobile robot. Vision refers to processing data from any modality which uses the electromagnetic spectrum to produce an image. The sensors for vision usually are CCD or CMOS cameras. These cameras relate measurements to scene structure and the computer vision or robot vision techniques examine the task of building computer representations of the environment from the captured images. Following the development of computer vision technology, vision sensors have become strikingly powerful tools for variety of autonomous behaviors. Much more presentations on many different sensors are explained in the essay (Everett, 1995).

Many of the above sensors either alone or through fusion with others can serve for the basic ability of SLAM. In this thesis, we apply the odometry as the proprioceptive sensor and ultrasonic sonar, laser rangefinder and monocular & stereo cameras as exteroceptive sensors for the experimental studies.

2.3 Simultaneous Localization and Mapping

SLAM is a crucial characteristic for a truly autonomous mobile robot. It allows a robot navigating from an unknown origin, building the maps of the unknown environment from the sensor perception systems and concurrently estimating their pose with respect to a certain coordinate system by using those constructed maps. This is not as straightforward as it might sound due to inherent uncertainties in discerning the robot's relative movement from its various sensors. If the measured distance and direction traveled has a slight inaccuracy at the next iteration of map building, then any features being added to the map will contain corresponding errors. If unchecked, these positional errors build cumulatively, grossly distorting the map and therefore the robot's ability to know its precise location. A seminal work in SLAM to discuss these challenges is the research of R.C. Smith and P. Cheeseman (1987; 1988; 1990) on the representation and estimation of spatial

uncertainty. In those works, a feasible solution for SLAM was proposed. Other pioneering work in this field was conducted by the research group of Hugh F. Durrant-Whyte (1991b) in the early 1990s.

Mapping

As its name implies, SLAM consists of two operations: mapping and localization. An appropriate map representation can actively help implementing the localization. There are three major mapping approaches: occupancy grid-based maps, topological maps and feature-based maps. Thrun (2003) provides a comprehensive survey on these various mapping techniques. The occupancy grid mapping employed in the previous work (Chow, Rad, & Ip, 2002) and other research work (Thrun, 1998b) describes a map of the environment as an evenly spaced field of binary random variables each of which represents the presence of an obstacle at that location in the environment. It computes approximate posterior estimates for these random variables, therefore this algorithm is rather easy to construct and maintain. Topological maps (Remolina & Kuipers, 2004) such as Voronoi diagrams (Garrido, Moreno, Abderrahim, & Martin, 2006) are graph-like spatial representations. These maps lack scale, and distance and direction are subject to change and variation, but the relationship between points is maintained. Feature-based maps describe the environment with distinct features including lines (Ip, Rad, Chow, & Wong, 2002), landmarks (Thrun, 1998a), arcs (Lee, Cho, Chung, Lim, & Kang, 2005), etc. Especially for the indoor environments, the feature-based mapping method is much easier than other two paradigms to represent the objects. In order to gain insight into attributes of different mapping techniques, we compare three mapping algorithms with different configurations of the perception systems in Chapter 3.

Among the primitives of numerous features, segment/line is perhaps the simplest and can expediently describe most structured indoor environments. Many

algorithms have been proposed to extract the line segments from the raw sensor data. Arras (2003) presented an algorithm based on linear regression theory. He first transformed the line extraction problem into a search problem in model space (line parameter domain). Subsequently, he applied the Agglomerative Hierarchical Clustering algorithm to construct adjacent line segments. Furthermore, he introduced a new feature type: multi-segment lines that exploited the high degree of co-linearity of indoor environments and overcame the problems with low feature discrimination. Ip, et al. (2002) suggested an enhanced adaptive fuzzy clustering (EAFC) algorithm to build a segment-based map in which the noise was treated as a special cluster using noise clustering (NC) technique. Split-and-Merge originated from computer vision is probably the most popular line extraction algorithm. It proceeds iteratively by gradually adding successive sensor readings to a line defined by the first few observations. A least square algorithm is used to detect whether the point lies on the postulated line. If the error is greater than the threshold error, the line is terminated and another line would be initiated (D. C. K. Yuen & B. A. MacDonald, 2003; David C. K. Yuen & Bruce A. MacDonald, 2003). Xu, Liu and Xiang (2003) developed a modified version named split-merge-split-merge phase to build a local map. The line fitting algorithm which belongs to another simple strategy referred to incremental algorithm has been suggested (Pfister, Roumeliotis, & Burdick, 2003; Xavier, Pacheco, Castro, Ruanot, & Nunes, 2005). Hough transform (Fichtner & Grobmann, 2004; Muñoz-Salinas, Aguirre, & García-Silvente, 2006; Prez Lorenzo, Vazquez-Martin, Nunez, Perez, & Sandoval, 2004) has been successfully applied to detect lines on images and is significantly robust. However, it is often influenced by illumination, asymmetric background, etc. The most difficult problem in generating the Hough transform is selecting the quantization levels for segments parameters. Log Hough implementation (Alempijevic & Dissanayake, 2004) has addressed the quantization problem whereby the r space is quantized with a log distribution utilizing the minimum range r as the shifting parameter. In addition, performing the operation in log space also considerably reduces the computational cost of the

classic Hough algorithm. Weingarten and Siegwart (2006) adopted the 3-Dimensional laser sensor to extract the segments which are described by SPmodel with the help of Random Sample Consensus (RANSAC) (A. F. Martin & Robert, 1981) segmentation paradigm. As a survey study, Nguyen et al. (2005) compared six popular line extraction algorithms above including split-and-merge, linear regression, incremental algorithm, RANSAC, Hough transform, and Expectation-Maximization (EM) algorithm. The results of the comparison indicated that the split-and-merge and incremental algorithms were more efficient for their superior speed and accuracy. However, correct selection of various thresholds such as errors, line length and the inherent recursive nature for these two algorithms are among their drawbacks. The common segmentation methodology for dynamic environments, RANSAC, is available for robust fitting of models in the presence of data outliers, while it takes unbounded time for RANSAC to compute the model parameters and this would force the algorithm towards a suboptimal model. To address these disadvantages, we propose a robust regression model in Chapter 4 to build the segment based maps for both static and dynamic environments and integrate this model into the SLAM procedure.

Localization

Localization task is essentially an attempt by the robot to determine where it is located in the map built from the mapping process. It is an important part of the autonomous navigation for mobile robots. In a typical indoor environment with a flat floor plane, localization becomes a matter of estimating robot pose. It has two types: local localization and global localization.

Local localization or pose tracking provides the mobile robot position a new pose estimate given the previous information. Probabilistic localization algorithms are variants of the Bayes filter. With natural landmarks or artificial beacons, the straightforward application of Bayes filters to localization problem is named

Markov Localization (D. Fox, Burgard, & Thrun, 1999). It handles uncertainty by multi-modal probability densities that are allowed and propagated through the motion model and are maintained over the space of all locations of a robot in its environment. The extended Kalman filter (EKF) localization algorithm (John J. Leonard & Durrant-Whyte, 1992) is a special case of Markov localization, where the probability density is uni-modal Gaussian and only mean and covariance need to be propagated. Moving away from point features, Lu & Milios (1997) as well as Gutmann & Schlegel (1996) introduced the scan matching algorithm which made it possible to localize a robot with remarkable accuracy. Bengtsson and Baerveldt (2003) presented a new scan-matching algorithm based on the IDC (iterative dual correspondence) algorithm, which showed a good localization performance even in dynamic environments.

Global localization algorithm considers the case where the robot locates itself without knowing the start pose or capitalizing upon information about where it was before. In such a situation robot interrupts its normal operation and starts wandering through the environment trying to find its pose. Besides Markov localization that can be applied, Monte Carlo Localization (MCL) (Dieter Fox, Burgard, Dellaert, & Thrun, 1999) and its modified versions (Milstein, Sánchez, & Williamson, 2002; Yee & Vermaak, 2005) are in the form of an efficient data-dependent proposal that can be used both for initialization and re-initialization after tracking failure or robot kidnapping. The global localization also can be achieved by matching distinctive scale invariant visual landmarks in the current frame to a database map by RANSAC algorithm (Stephen, Lowe, & Little, 2002). Different from the passive form above, the active approach makes the robot actively interact with the environment and choose the optimal action which minimizes the expected future uncertainty. Therefore the active global localization strategies are born, including heuristic (Gasparri, Panzieri, Pascucci, & Ulivi, 2007; Jensfelt & Kristensen, 2001), geometric (O'Kane, 2006; Rao, Dudek, & Whitesides, 2007) and entropy-based approaches (Dieter Fox, Burgard, & Thrun, 1998; Porta,

Verbeek, & Kröse, 2005).

In this thesis, we would not study the localization problem independently but design it within SLAM where only the local localization will be handled.

SLAM

Durrant-Whyte and Bailey (Bailey & Durrant-Whyte, 2006; Durrant-Whyte & Bailey, 2006) provided an overview of SLAM, which is a comprehensive introduction including history, the problem structure, computational solution, convergence of algorithm, data association, implementation, and open resource. The important points of their reports focus on the development and challenges which make the SLAM as an interested open problem and captivate the researchers of the whole world.

Application of EKF algorithm to the SLAM was first presented in the pioneering research work of Smith & Cheeseman and implemented in Leonard and Durrant-Whyte (1991b; 2001). After that the EKF algorithm became de-facto in many SLAM implementations. Noting the limitations of EKF, many researchers investigated modified or alternative techniques for maintaining the accurate pose estimates during SLAM. SEIF SLAM (Thrun, et al., 2004) is the use of the sparse extended information filter (SEIF) to solve SLAM by maintaining a posterior over the robot pose and the map. Similar to GraphSLAM (Thrun & Montemerlo, 2006), the SEIF SLAM is a fully solution but an online algorithm. It derives the creation of information links between pairs of features from a filtering perspective. Another alternative of SLAM algorithms is called FastSLAM which was first proposed by Montemerlo et al. (2002). FastSLAM algorithms were developed based on Particle Filter (PF). Condition on these particles the mapping problem is factored into separate problems. Therefore, one EKF for each feature is used to update the feature estimate. The basic algorithm can be implemented in time logarithmic in

the number of landmarks. Hence, FastSLAM offers computational advantages over plain EKF implementations and many of its descendants. Furthermore, the improved FastSLAM algorithms have been proposed such as FastSLAM 2.0 (Montemerlo, Thrun, Koller, & Wegbreit, 2003), DP-SLAM (Eliazar & Parr, 2004), Unscented FastSLAM (C. Kim, Sakthivel, & Chung, 2008), etc.

Extension of SLAM to dynamic environments has been studied in recent years. Burgard et al. (1999) updated a given static map using the most recent sensor information to deal with the effect caused by moving people in the environment. Fox et al. (1999) proposed a filtering technique to identify range measurements that do not correspond to the given world model, and then to update the robot position using only those measurements. Montemerlo Thrun and Whittaker (2002) employed a Rao-Blackwellized particle filter to solve the simultaneous localization and people tracking problem based on a prior accurate map of the corresponding static environment, which is similar to FastSLAM. These approaches, however, all depend on the *existing* maps to detect the influence of state changes, and then provide respective action. In contrast, EM algorithm (Biswas, Limketkai, Sanner, & Thrun, 2002; Hähnel, Triebel, Burgard, & Thrun, 2003) interleaves the identification of measurements that correspond to dynamic objects with a mapping and localization algorithm which can isolate the dynamic elements to improve the estimate about spurious measurements and enhance localization accuracy in different kinds of dynamic environments. Andrade-Cetto and Sanfeliu (2002) combined the landmark strength validation and Kalman filtering for map updating and robot position estimation to learn moderately in dynamic indoor environments. Hähnel, Schulz and Burgard (2003) presented a Sample-based Joint Probability Data Association Filters (SJPDFs) to map populated environments and filter out the people tracking results of which the data are obtained with the laser range scanners. Wolf and Sukhatme (2005) proposed an online algorithm for SLAM in the dynamic environment which is robust to detect dynamic entities both when they move in and out robot's field of view, although

the limitations of moderately dynamic indoor environments and the narrow assumption of localization implementation. Wang (2004) established a mathematical framework to integrate SLAM and DATMO (Detection and Tracking Moving Objects). The idea is to identify and keep track of moving objects in order to improve the quality of the map. Also from the viewpoint of multi-target tracking (MTT), we addressed the SLAM in dynamic environments in the previous works. One approach employed Sequential Monte Carlo methods (G. Q. Huang, Rad, Wong, & Ip, 2004) as robust and computationally efficient algorithm, where the SLAM procedure made use of Rao-Blackwellized particle filter to estimate robot pose and EKF to estimate feature location; and a Hybrid Independent/Coupled Sample-based Joint Probability Data Association Filter (Hyb-SJPDAF) was applied to solve MTT and data association problem. The other method used a hierarchical hybrid method (G. Q. Huang, Rad, & Wong, 2005) to solve SLAM locally by Maximum Likelihood with occupancy grid map and globally by EKF with feature-based map, and suggested a straightforward Nearest Neighborhood (NN) algorithm based on Euclidean metric to address MTT.

Most recently, advances in computer vision have provided robotics researchers with efficient and powerful techniques that can be employed in SLAM. Davison and his group proposed a real time monocular SLAM algorithm that generated a 3D trajectory of a previously unknown scene (Civera, Davison, & Montiel, 2008; J. Civera, A. J. Davison, & J. M. M. Montiel, 2007a; J. Civera, A. Davison, & J. Montiel, 2007b; Andrew J. Davison, Reid, Molton, & Stasse, 2007; Andrew. J. Davison, 2003; Montiel, Civera, & Davison, 2006). The core of their approach was the online creation of a sparse but persistent map of natural landmarks within a probabilistic framework. The 3D position of the landmarks was described by a unified inverse depth parameterization algorithm that allowed efficient and accurate representation of uncertainty during un-delayed feature initialization procedure. Their research made a valuable contribution and opened up a new direction in robotics research. Following this pioneering work on monocular

SLAM, other researchers studied line-based algorithms. Eade and Drummond (2006) proposed an edge-let landmark to depict the line features in images. This work, which is their extension of the so-called scalable monocular SLAM (E. Eade & T. Drummond, 2006), avoids regions of conflict and deals with multiple matches through robust estimation. Gee and Mayol-Cuevas (2006) used fast conic extraction to obtain the 2D edges and then estimated the 3D segments with the Unscented Kalman filter (UKF). Also Smith, Reid and Davison (2006) applied FAST corners to quickly verify that there was an edge between two corners by bisecting checks. Besides, other researchers conducted similar studies on line based SLAM with a single camera. Lemaire and Lacroix (2007) introduced the Plücker coordinates for 3D line description and considered constraints associated with Plücker representation during the updating stage of Kalman filter. Folkesson, Jensfelt and Christensen (2005) suggested a M-space feature representation similar to SP-model. This feature representation is a general and systematic technique that makes it possible to change sensors and features without any variation to SLAM implementation. Additionally, lines and points can be merged to enhance the performance of visual SLAM. Jeong and Lee (2006) constructed the 3D line and corner maps. The reconstructed 3D line landmarks improved the performance of the robot navigation when robot's pose remained uncertain with only the corner information. A six degrees of freedom Hierarchical SLAM (Marzorati, Matteucci, Migliore, & Sorrenti, 2007) was used to unify a framework of uncertain projective geometry to describe, combine, and estimate various types of geometric elements. This SLAM strategy also improved mapping accuracy and pose estimation. To enhance the precision of the map, Diosi and Kleeman (2004) combined measurements from a laser rangefinder with those from an advanced sonar array. In that configuration, the sonar aided laser segmentation, the laser helped good sonar point feature selection, and the measurements corresponding to the same object in laser and sonar were fused.

Most relevant research on vision SLAM, however, have been implemented in

static space or the environments with few moving objects. Additionally, the traditional rangefinder sensors, for example laser and sonar, sometimes fail to detect the dynamic objects, which may lead to inappropriate robot actions that ultimately results in failure to complete the expected tasks. In this study, combining with the robust regression model for map representation, we will consider the problem of SLAM in dynamic scenarios from a sensor fusion viewpoint which incorporates the sensor information of the camera and rangefinder sensors in Chapters 5 and 6.

2.4 Sensor Fusion

Sensor fusion is also known as multi-sensor data fusion and is a subset of information fusion. It is the combination of sensory data from disparate sources such that the resulting information is in some sense better than would be possible when these sources were used individually. By an efficient scheme, the significant profits of sensor fusion concentrate on improving confidence in decisions due to the use of complementary and redundancy information, performance to decision making and performance in adverse environmental conditions.

With advantages over the use of a single sensor alone, sensor fusion technique has frequently been applied to robotic community. Castellanos, Neira and Tardos (2001) described a probabilistic technique for validating the individual and joint compatibility of measurements acquired from laser rangefinder and monocular camera, the benefits of which for SLAM problem is providing partial redundancy between sensor observations and increasing both reliability and precision from early stages of the processing. EKF as a very prevalent sensor fusion scheme has been used to solve several problems such as localization (Lizarralde, Nunes, Liu, & Wen, 2003; Roumeliotis & Bekey, 2002) and SLAM in robotics research. This classical method is extended by many researchers. Ahn et al. (2008) applied EKF-SLAM framework to fuse the features extracted respectively from ultrasonic

sonar and stereo camera, which achieved correct data association via the object recognition and high frequency update via the sonar features. Tsai, Lin and Lai (2005) used the same method to incorporate the information of inertial and ultrasonic sensors. Similar to EKF, particle filter, another kind of Bayesian filter, is adopted for sensor fusion (Moreira, Machado, Mendonca, & Pereira, 2007; Vadakkepat & Jing, 2006), which has ability to deal with the multi-modal distribution. The multi-rate EKF and multi-rate UKF (Gemeiner, Einramhof, & Vincze, 2007) were used to implement the fusion of inertial and vision data for SLAM, which overcame the inherent drawbacks of EKF. On the basis of Bayesian theory and EKF framework we also suggest an indirect sensor fusion strategy in Chapter 5 to avoid a few of disadvantages existed in EKF so that the performance of EKF-SLAM can be improved and additionally some pseudo map features can be removed.

The probabilistic method and the series of Bayesian filter belong to the statistical fusion techniques while other fusion methodology of uncertainty inference is applicable. Dempster-Shafer rules based sensor fusion strategy is proposed to integrate two omnidirectional images for solving the problem of sensorial construction in mapping procedure (Drocourt, Delahoche, Marhic, & Clerentin, 2002). The work of fuzzy decision (Matía & Jiménez, 1998) employed an adaptive fusion of reactive behaviors to provide emergent behaviors, which has indicated that the fusion of heterogeneous sensors in a mobile robot is capable of reducing uncertainty in position estimation. A sensor fusion algorithms for mapping was presented to evaluate the grid map obtained from a camera, an array of ultrasonic sensors and a laser rangefinder by an adaptive fuzzy logic fusion algorithm (Kapach & Edan, 2007). Similarly, Huang, Rad, Wong & Ip (2007) also exploited the fuzzy system as the heterogeneous multi-sensor fusion algorithm for mapping in dynamic environments. Furthermore, Kim et al. (2007) proposed a hierarchical hybrid sensor fusion algorithm for map building process, which incorporates three fusion methods into three level fusion modules: weighted average of probability,

expert rules and Bayesian method.

Sometimes, the prior knowledge required for the methods represented above is difficult to obtain. Therefore, a general fusion approach based on information theory (Manyika & Durrant-Whyte, 1994) was suggested. In that work, the fusion algorithm was a distributed information filter and the measures of the fusion were posterior entropy, prior entropy and mutual information. The sensor management method made use of information-based utility functions. With the advantage of generality of Shannon entropy, a Bayes-Maximum Entropy (Beckerman, 1992) formalism for multi-sensor data fusion was introduced, and applied to the fusion of ultrasound and visual sensor data robot. A minimum entropy approach (Yifeng & Leung, 1997) was presented for multi-sensor data fusion in non-Gaussian environments. Also the self-entropy and conditional entropy were merged with Markov chain as a robust algorithm named entropy based Markov chain to aggregate multiple sensory observations into a consensus output. Recently, an Entropy Fusion Model (Fassinut-Mombot & Choquel, 2004) was defined, the purpose of which was to reduce the combination space by explicitly resending the notions of source redundancy and source complementarity in the form of entropy measures. Unfortunately, this model has been mainly applied to object recognition and not spread to SLAM. Following the concept of information theory and borrowing the idea of decision analysis, Chapter 6 presents and develops a direct sensor fusion management based on information entropy weight, which fuses the sensor measurements and related variance to reduce the feature uncertainty.

2.5 Data Association

As one of most critical and challenging problems in SLAM, data association consists of relating sensor measurements to the features in the existing map. It is essential to establish correct correspondences between the sensed and mapped features for building a consistent map, because any single mismatching may cause

the estimator such as EKF to diverge.

In stochastic mapping, the problem of data association is resolved by widely employing the gated nearest neighbor (NN) algorithm (J. J. Leonard & Durrant-Whyte, 1991b). The normalized squared innovation test is used to determine the compatibility, and the Mahalanobis distance is calculated to select the best matching. The most appealing characteristics of NN is its $O(mn)$ computational complexity besides its conceptual simplicity. Here m is the number of measurements and n is the number of existing landmarks in the map. It performs well in the environments with sparse landmarks. However, in the surroundings with high density of landmarks, the innovations of matching different observations obtained at same locations are correlated, thus, the NN algorithm may accept wrong matching, which leads to divergence of the estimator. Dezert and Bar-Shalom (1993) proposed a better solution, i.e., joint probabilistic data association (JPDA). JPDA associates all of the measurements falling inside a suitably chosen validation region of a track to itself by a probabilistic weighting procedure and performs relatively well when spurious measurements are relatively moderate. The limitation however is that it can be computationally prohibitive in terms of calculating weighting probabilities, and the process may corrupt the feature recognition or discrimination information. Neira and Tardos (2001) proposed using a joint compatibility test based on the branch and bound (JCBB) search with an acceptable computational cost in indoor environments. JCBB takes groups of feature observation associations into consideration in the context of searching for the hypothesis with the maximum number of compatible pairs. However, the resultant exponential search space, despite the branch and bound pruning, renders the method computationally intensive for real-time implementation. Nieto, Guivant, Nebot and Thrun (2003) proposed a real-time data association method for FastSLAM by applying the multiple hypotheses tracking (MHT) method in a variety of outdoor environments. MHT is the most structured approach employing the idea of delay decision for multi-target tracking

and data association (Reid, 1979). It forms a number of hard association hypotheses from several scans of data, and delays the association decision to a later time when more information becomes available. Bailey et al. (2000) considered relative distances and angles between points and lines in two laser scans and used graph theory to find the largest number of compatible pairings between the measurements and existing features. More recently, Zhang, Xie and Adams (2005) formulated the problem of data association in SLAM as a linear programming (LP) relaxation and thus obtained the suboptimal correspondences by solving the LP problem. In the same fashion of formulating data association into an optimization problem, Wijesoma, Perera and Adams (2006) introduced a multidimensional assignment based method to resolve the problem. To the best of our knowledge, however, most methods for solving the data association above are suboptimal. In Chapter 4, we revisit this problem by proposing an optimal graph approach.

Computer vision technology makes the data association methodology feasible for visual SLAM algorithm. One of the methods is based on Scale Invariant Feature Transformation (SIFT) (Lowe, 2004). Landmarks are identified by SIFT and represented by keypoint descriptors (Miro, Dissanayake, & Weizhen, 2005; Sim, Elinas, Griffin, & Little, 2005). These landmarks subsequently are treated as the ideal candidates to the robust data association. Gil et al. (2006; 2007) managed the data association with the SIFT features from the pattern classification viewpoint, and the Mahalanobis distance was established by the average SIFT descriptors and a high dimensional covariance matrix. Similarly with pattern recognition technology, object-based SLAM (Ahn, Choi, Choi, & Chung, 2006) combined advantages of multi-scale Harris corner as a detector and the SIFT descriptor for natural object recognition, which provides a correct data association.

In the reference works (Civera, et al., 2007a; Civera, et al., 2007b; Andrew J. Davison, et al., 2007; Andrew. J. Davison, 2003; Montiel, et al., 2006; P. Smith, et

al., 2006) samples in a window region are used to match the predicted features and calculate the innovation. However, the computation pixel by pixel in the predefined region is a little bit repetitious. In Chapter 5, we also use SIFT and suggest a data association scheme based on the homography transformation matrix. With the help of SIFT descriptors, this matrix is estimated by the matched points between two images. Compared with the related work above, instead of directly applying SIFT descriptors as the natural features, our method emphasizes on using SIFT mechanism to determine the matched points between any two images and then with these matched points estimates the homography transformation matrix.

2.6 Summary

In this chapter, the background materials related to various sensor systems, SLAM including mapping and localization, sensor fusion and data association have been reviewed, which provides the foundation on the problems to be addressed in the subsequent chapters.

The survey on perception sensor systems gives the information on the instruments acquiring the environmental knowledge. Some of them will be taken as the fundamental devices for our research. The literature on the central topic related to this thesis, SLAM including mapping and localization especially in dynamic environments, has been revisited. The probable disadvantages in existing algorithms were discussed and we will attempt to tackle these problems through the robust methodology and sensor fusion viewpoint after checking the essays corresponding to the sensor fusion application. Another critical problem in SLAM, data association, was also considered. By analysis of the data association methods in literature, the sub-optimal characteristic is stated. We will propose and prove an optimal approach to handle data association problem. Additionally, a special method to address the data association in visual SLAM procedure will also be suggested which could release the computational complexity that is induced by

some of techniques in literatures.

We will first start by determining a proper perception system and map representation for our study in the next chapter.

Chapter 3 Comparison of Mapping Methodologies

3.1 Introduction

Map building procedure is generally regarded as one of the most important problems in the pursuit of realization of a truly autonomous mobile robot. In recent years, this field has matured to a point where detailed maps of complex environments can be built in real-time, specifically for indoors environments. Many existing techniques are robust to noise and can cope with variety of structured static and dynamic environments. The integration of localization and mapping has led to development of SLAM algorithms which have gained considerable attention in the last two decades. Thrun (2003) provided a comprehensive introduction and compared various probabilistic mapping techniques. Following this extensive survey, the predecessors in our group proposed the Enhanced Adaptive Fuzzy Clustering (EAFC) integrating Noise Clustering (NC) to build the segment-based map (Ip, et al., 2002) and Fuzzy Tuned Grid-Based Map (FTGBM) (Chow, et al., 2002). As for dynamic map building, similar to Oriolo's work (Oriolo, Ulivi, & Vendittelli, 1997; Oriolo, Ulivi, & Vendittelli, 1998), by using FTGBM Huang, Rad & Wong (2006) suggested a sensor fusion method which used sonar temporal difference (STD) and statistical background subtraction (SBS) to detect and track moving objects. Oriolo's algorithm was classified under consistency-based methods realized by monitoring a sequence of temporal lattice maps for a certain number of measurement periods to detect moving objects by using sonars. In contrast, we employed a background subtraction technique and adopted an EM that learned 3-class mixture of Gaussians to model the non-stationary background relied on sufficient update during mapping process. After finding the moving objects, we proposed a fuzzy-tuned integration (FTI) to incorporate the results of motion detection into the mapping process.

To make a rational environment representation and to lay a solid foundation for the rest of this thesis, we devote our attention in this chapter to the mapping problem that is the essential element in the process of implementing any localization or SLAM algorithm. We compare two predominant mapping paradigms: segment-based map and FTGBM with three different perception systems and ascertain a proper mapping methodology including the map representation and sensor configuration which will be employed in the thesis research. In the following section, the perception systems and two map representations are reviewed. Section 3.3 describes the experimental results with the distinct mapping methodologies. Finally, the comparative results as well as the determined map representation and sensor systems are discussed in Section 3.4.

3.2 Perception Systems and Map Representation

Mobile robots acquire knowledge about their environments through taking measurements using various sensors and then extract meaningful information from those measurements. The most common sensors used in mobile robots are optical encoders which are classified as proprioceptive sensors and measure the robot internal position and its speed by advanced dead-reckoning. To acquire information on the robot environment, vision sensors and rangefinders such as ultrasonic sonar and laser classified as exteroceptive sensors are generally employed. These sensors were mounted on PIONEER II and III mobile robots and constitute the basis of experimental studies of this research work. In this chapter, we will test the properties of several of these sensors including encoders, sonar, laser and integration of various sensors, and formulate the perception systems which serve for the whole research.

A comprehensive review of map representation and methods was included in Chapter 2. Here, we only briefly present our previous works on map building:

segment-based and fuzzy-tuned grid-based map. Interested readers may refer to Chow et al. (2002) and Ip et al. (2002) for further information.

Segment-based Map

The fuzzy clustering algorithms have been used in many applications involving data segmentation, and it is suitable for feature detection of linear shapes. Therefore, fuzzy clustering facilitates direct extraction of a line segment within the data set space. Based on the fundamental fuzzy c-means (FCM) clustering method, enhanced adaptive fuzzy clustering (AFC) algorithm was developed in our segment-based mapping in which noise is treated as another separate cluster, Noise Clustering (NC). The enhanced AFC algorithm is divided into two phases. The function of *phase I* is to select the cluster centers within the data set by the standard FCM algorithm. After obtaining the cluster centers in the data set, the *phase II* (AFC) is used to calculate the line segments. The previously calculated cluster centers are used for initialization of the fuzzy partition matrix of AFC to obtain the cluster with linear prototypes, i.e. line segments. After extracting and grouping the line segments, with the compatible line segment merging technique, we merged the similar basic segments together to form a single line segment. The basic segments are discontinuous outlines of the object boundaries in the vicinity of the mobile robot trajectory. In order to improve the quality of the map, the number of line segments representing the same boundary is merged up by some reasonable conditions. Practically, sonar interference definitely affects the detection, however, two techniques are considered to eliminate most of interferences. One is setting a sonar range cut-off limit (1500mm). All sonar data which are over this limit are discarded and not used for feature extraction. The other is applying the NC. This special cluster aims to alleviate arbitrary distribution of noise in ultrasonic sonar including the interferences.

Fuzzy-Tuned Grid-Based Map (FTGBM)

In FTGBM, the probability distribution function (pdf) of the sensor model is tuned

by a set of fuzzy rules based on the maximum probability of the grid cell within the sensor cone. Similar to traditional approaches, the occupancy grid probabilities of the state $s(C_i)$ of grid cell C_i for the environmental map $P[s(C_i)=occ|x]=0.5$ means unknown or unexplored region. $P[s(C_i)=occ|x]=1$ means that the grid cell C_i is occupied and vice-versa. The fuzzy-tuned sensor model—pdf—considers the example of a range sensor characterized by Gaussian uncertainty in both range and bearing measurements where some elements in pdf are tuned by fuzzy theory under some conditions. This FTGBM alleviates the disadvantage that specular reflection and sonar interferences are modeled by traditional Gaussian function. It should be noted that the measurement noise is assumed to be white and augmented into sensor data as additive noise. After obtaining the sensor model, we used Bayesian update rule to update the occupancy probabilities of the grid cells.

3.3 Experimental Results

The experimental platform in this thesis was a Pioneer 2DX mobile robot equipped with an un-calibrated Canon VCC4 monocular camera with a fixed angle, a SICK LMS200 laser rangefinder, a PNI vector module, a pair of bumper and sixteen ultrasonic sonar sensors: six at front locations, six at rear, separated by 20° each, and four locating at each side. The driving mechanism was by means of two reversible DC motors with wheel encoders to update the location. Figure 3.1 shows the real Pioneer 2DX mobile robot and its diverse sensor systems. This platform is the main device employed in this thesis and will not be described more in the following chapters.

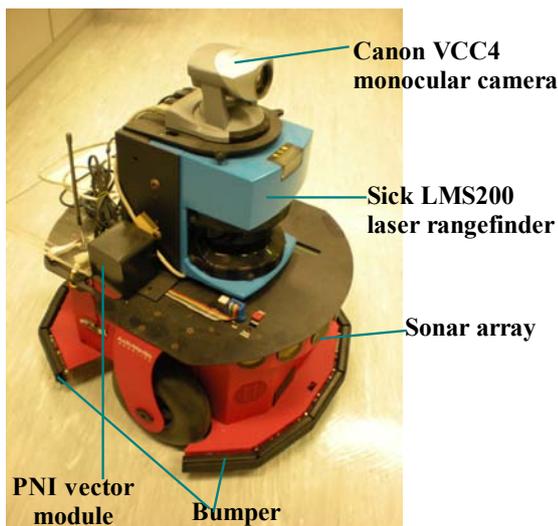


Figure 3.1 Pioneer 2DX and loaded sensors

We intend to illustrate the mapping results of three configurations of perception systems: only sonar, only laser, and combination of sonar plus camera with the segment-based and grid-based mapping techniques. The environment including static and dynamic ones is the corridor outside the Control Research Laboratory within the Department of Electrical Engineering building. Figure 3.2 gives the hand-measured blueprint of the experimental corridor. The software is written in C/C++ language, and Saphira API libraries have been used to obtain the sonar and laser data and estimate the current pose of the robot. The navigation is not autonomous in the present implementation. The localization procedure applied the dead-reckoning mechanism because the mapping process was a short-term. Meanwhile, some landmarks were placed on the ground for minimizing the localization errors. In all experiments, the robot traveled with an average speed of 150 mm/s and all sensor data have been transformed into the same coordinate frame (Please refer to the Appendix A for coordinate transformation). To compare the results conveniently, all mapping processes are implemented offline.

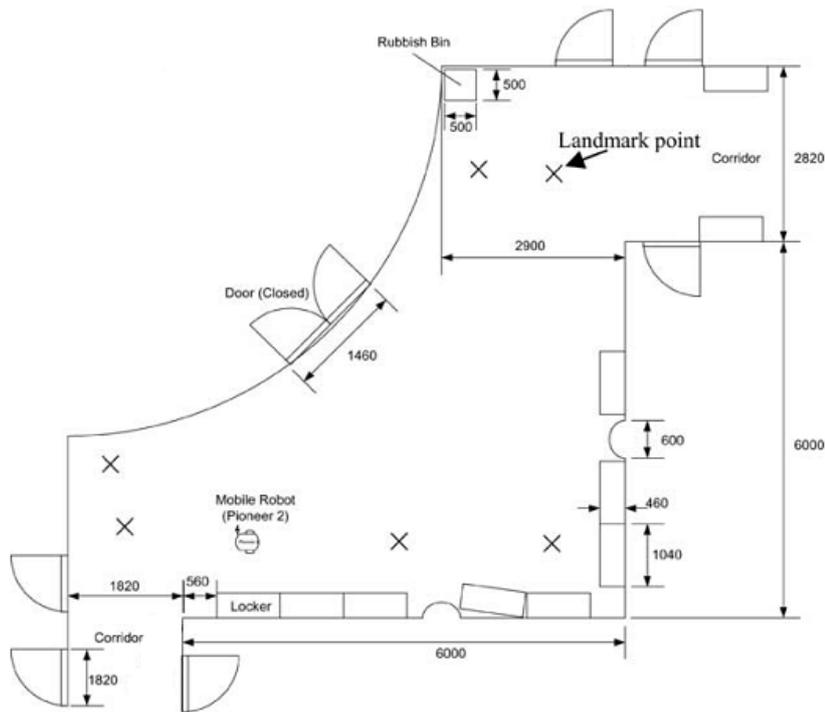


Figure 3.2 Hand-measured blueprint of the corridor

Case 1: Mapping with segment-based technique in the static environment

In this case, no moving objects appeared in the corridor except the mobile robot. We built the segment-based maps by EAFC algorithm for sonar and laser readings, and through sensor fusion techniques for the incorporated measurements of sonar and monocular camera. The results are illustrated in Figure 3.3. Figure 3.3(a) and (c) look like similar because there is no moving object in the corridor so that the performance of fusing information of sonar and camera is not obvious. As numerous data are obtained from laser sensor, the map in Figure 3.3(b) is more accurate than those in Figure 3.3(a) & (c).

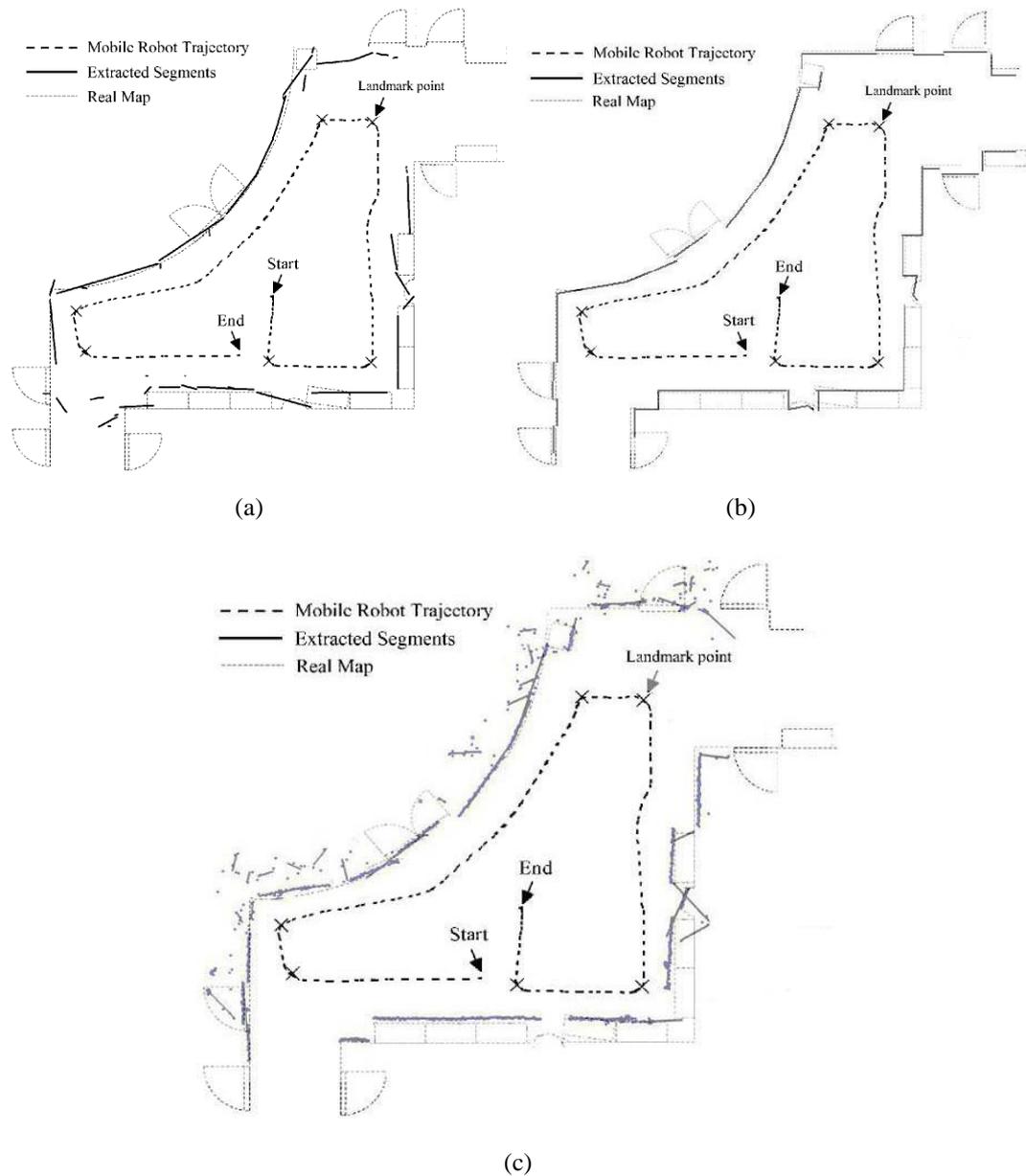
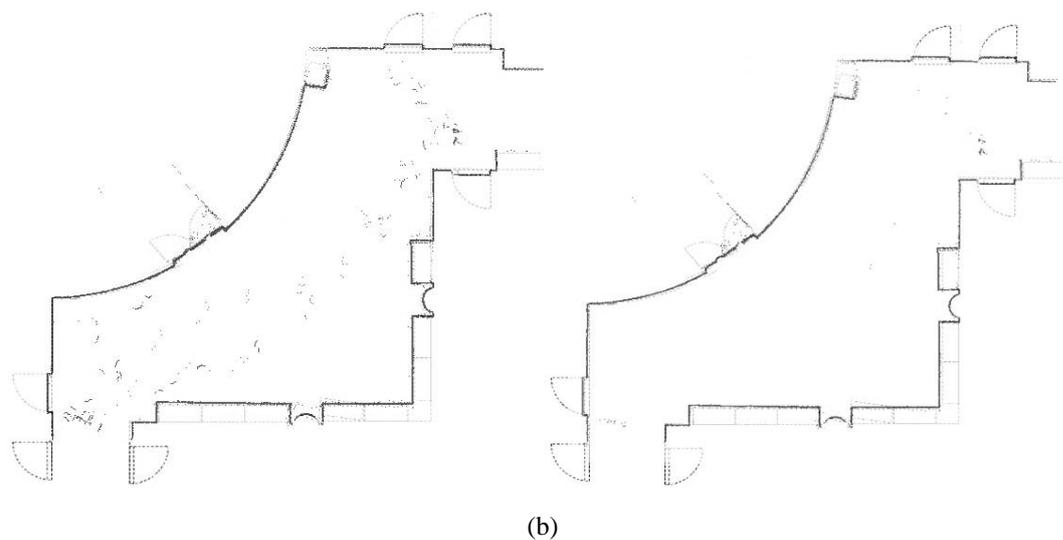
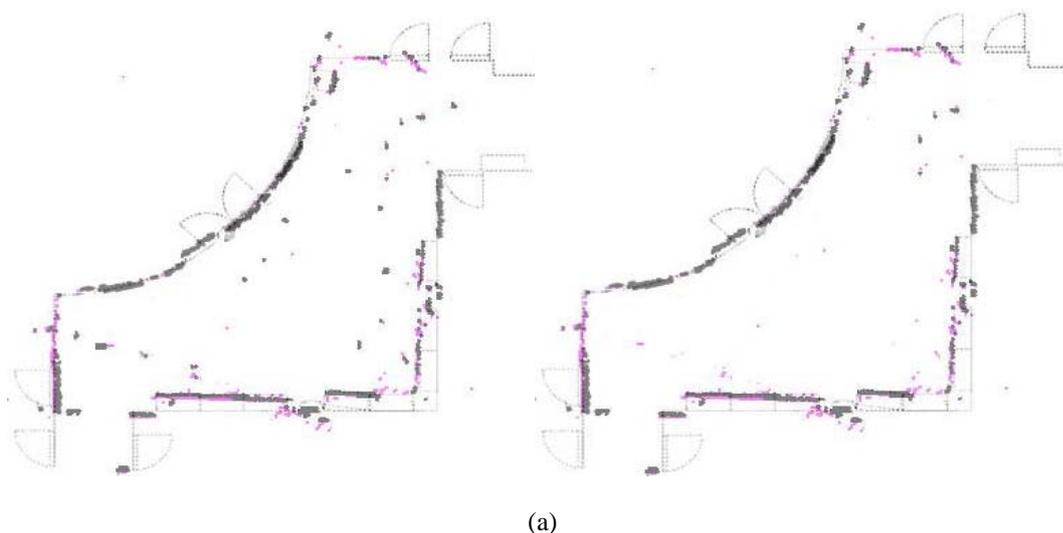


Figure 3.3 Results of the segment-based map in the static corridor. (a) Sonar sensor only; (b) Laser sensor only; (c) Combination of sonar and monocular camera.

Case 2: Mapping with FTGBM in the dynamic environment

There were several persons walking through the corridor where the robot was navigating. The size of grid cell was 50 mm. We also made use of our proposed sensor fusion algorithm to implement mapping for integration of sonar and camera. When using other sensor configurations including sonar and laser only, we applied the EM algorithm similar to Biswas et al. (2002) and Hahnel et al. (2003) to deal with the spurious data in MATLAB platform. Figure 3.4 shows the final maps

without (left column) and with (right column) handling the outliers. In Figure 3.4(a) & (c), one person stopped near the robot for a moment, and in Figure 3.4(b) two persons walked through the corridor: one stood in front of a door and opened it (top-right part), the other pushed another door and entered the room (bottom-left part). Note that the effect of moving objects on the final map can not be eliminated thoroughly by sonar and laser only since the person is treated as a static object when he stopped temporarily (right plots of Figure 3.4(a) & (b)). However, with the combination of sonar and camera perception system, moving objects can be detected, tracked and then filtered out so that the quality of the final map is improved (cf. Figure 3.4(c)).



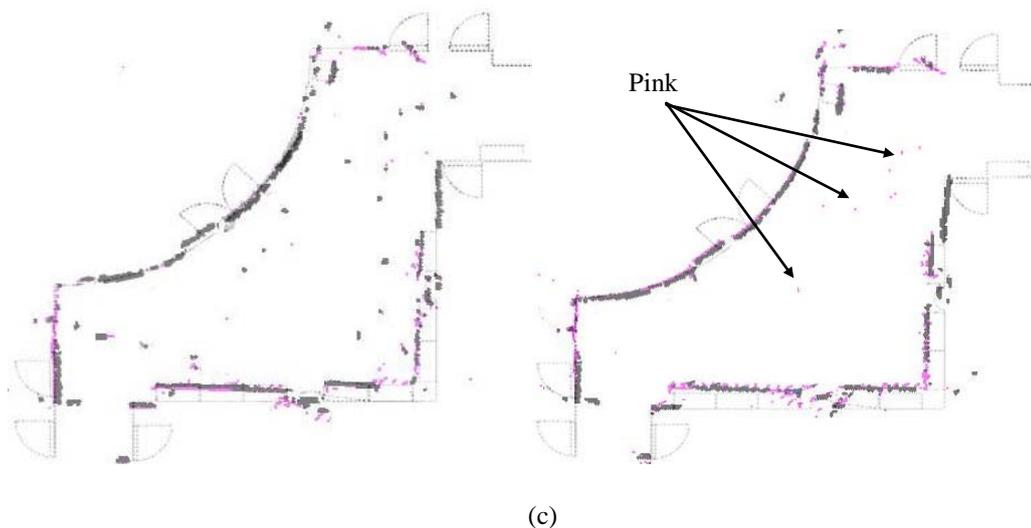


Figure 3.4 Results of FTGBM in the dynamic corridor. Black: final map; Pink: preprocessed data
 (a) Sonar sensor only; (b) Laser sensor only; (c) Combination of sonar and monocular camera.

3.4 Discussion

Table 3.1 summarizes and compares the representation of map and sensor configurations employed for the experiments above in different environments. Here we do not consider the efficiency of the mapping algorithms but take only three indices as criteria of the comparison: one is the map quality, one is the memory cost for storing the map features, and the other one is the quantity of sensor information. It seems from Table 3.1 that the quality of the map built from laser data is better in both static and dynamic environments while the application of incorporating the sonar and monocular camera can provide much better maps in dynamic situation. It is further demonstrated that the segment-based mapping costs less memory space than FTGBM especially when the volume of map increases rapidly. Additionally, within any map representation the integration of sonar and camera perception system consumes much more memory than any of the others. This is because processing image data need more memory space than managing simple rangefinder data. The perception system of sensor integration has the most quantity of information, laser takes the second place and the information reflected in sonar is the least. Considering all the factors manifested in the experimental

results, we selected the segment-based mapping as the method of environment representation, and laser rangefinder and multi-sensor integration as the perception systems for the rest of the thesis. Segment-based mapping easily describes most indoor environments and requires less memory to save map features. The laser rangefinder supports the precise range measurements and sensor integration guarantees correct map description in dynamic environments.

Table 3.1 Comparative results of different map representation and sensor systems

PERCEPTION SYSTEMS	SEGMENT-BASED MAP (STATIC ENVIRONMENT)			FTGBM (DYNAMIC ENVIRONMENT)		
	Quality	Memory Cost	Information Quantity	Quality	Memory Cost	Information Quantity
	Ultrasonic Sonar Sensor	+	+	-	-	-
Laser Rangefinder	++	+	+	+	-	+
Sonar and Monocular Camera	+	-	++	+ / ++	--	++

+: good; -: poor

The main objective of this chapter is to select a proper mapping methodology for the studies of SLAM in the next chapters. From this viewpoint, we applied a sensible strategy to deal with the localization problem even though it is not flawless. To smooth over these disadvantages and make mapping quality as good as possible, we placed some landmarks on the turning points where most errors occurred. When the mobile robot traveled across these known landmarks, its pose was adjusted. The performance of the localization strategy above is equivalent to the SLAM with known data association. When implementing mapping for dynamic environments, the algorithms for filtering out the outliers corresponding to moving objects are not taken as the primary issue. The methods in our previous and the work by other researchers are introduced directly to ensure an acceptable mapping process. As the core topic of the thesis is the SLAM in dynamic environments, we devote much more attention to this problem in the following chapters on the basis of the determined map representation and perception system in this chapter.

Chapter 4 Robust Regression Model for SLAM

4.1 Introduction

It is imperative that autonomous mobile robots are outfitted with the capability of making sense of the dynamic world around them. Following the mapping strategies outlined in the last chapter, we adopt the segment-based map which is perhaps the simplest and can expediently describe most structured indoor environments. Many algorithms have been proposed to extract the line segments from the raw sensor data that contain intrinsic and extrinsic noise patterns. Ip et al. (2002) suggested an enhanced adaptive fuzzy clustering (EAFC) algorithm to build a segment-based map in which the noise was treated as a special cluster using noise clustering (NC) technique. Nguyen et al. (2005) compared six popular line extraction algorithms including Split-and-Merge, linear regression, incremental algorithm, RANSAC, Hough transform, and EM algorithm. The results of the comparison indicated that the Split-and-Merge and incremental algorithm were more efficient for their superior speed and accuracy. However, it was also commented that the appropriate selection of various parameter thresholds such as errors, line length, and the inherent recursive nature for these two algorithms were regarded as the main drawbacks of them. In addition, if the data obtained from laser scan is noisy or different objects obscure the planes defining the map, then the line assembly would fail. In dynamics environments and in the presence of moving objects, algorithms based on linear regression theory fail, but RANSAC is available for robust fitting of models in the presence of data outliers. However, the upper bound on the time that it takes to compute the model parameters can not be prefixed in RANSAC as this would force the algorithm towards a suboptimal model. Additionally, if more than one model in a particular data set exists, RANSAC could fail to converge to either one. Another disadvantage of RANSAC algorithm is its dependence on the threshold ϵ , which

decides on how close a point-correspondence must be to the computed model to be considered as inliers. EM algorithm also has been employed for dynamic environments because it provides an iterative procedure to compute a series of easier log-likelihood for an incomplete data set as well as the missing information. Nevertheless, it sometimes fails to converge or converges very slowly. Especially, in some problems the E- or M-steps may be analytically intractable.

In the proposed approach, the concept of robust regression (Maronna, Martin, & Yohai, 2006; Rousseeuw & Leroy, 1987) is introduced to extract the segments from the dynamic environment, which is referred to as robust regression model. This model exploits a MM-estimate for segment parameter estimations, and it concurrently removes the sensor noise and the moving objects from the sensor data. The MM-estimate combines a high breakdown point (50%) with good efficiency as linear regression. With this estimate, the robust regression model quickly converges and alleviates the disadvantages of RANSAC and EM algorithms. Moreover, this model can separate the distinct models in a group of sensor data. In comparison with Split-and-Merge and incremental algorithm, the robust regression model adopts less thresholds, which relieves the overall difficulty of thresholds selection.

As another critical and challenging problem in SLAM, the data association consists of relating sensor measurements to the features in the existing map. It is essential to establish correct correspondences between the sensed and mapped features for building a consistent map, because any single mismatching may cause the estimator such as EKF to diverge. It is intuitive to consider data association as a search problem in the space of measurement-feature correspondences. However, it is usually intractable to do exhaustive searching, because the complexity of finding correspondences between the measurements and the mapped features is exponential on the number of measurements.

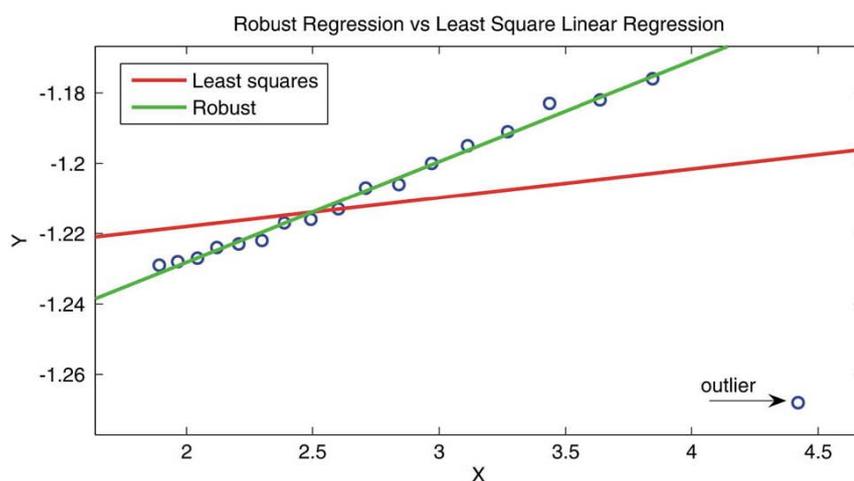
To the best of our knowledge, almost all the existing methods for solving the data association is suboptimal, such as the gated nearest neighbor (NN) algorithm (J. J. Leonard & Durrant-Whyte, 1991b), joint probabilistic data association (JPDA) (Dezert & Bar-Shalom, 1993), joint compatibility test based on the branch and bound (JCBB) search (Neira & Tardos, 2001), and multiple hypotheses tracking (MHT) method (Nieto, et al., 2003). In this chapter, we revisit the problem by proposing an optimal graph approach. Specifically, the data association in SLAM is first formulated as a 0-1 integer programming (IP) problem. It is well known that optimally solving IP problem generally is NP-hard. Therefore, the relaxation technique is usually adopted, for example linear programming relaxation (Zhang, et al., 2005), to obtain suboptimal solutions. We algorithmically prove that the IP problem is equivalent to a minimum weight bipartite perfect matching problem. Hence, we are able to optimally solve the bipartite matching problem and thus equivalently optimally resolve the IP problem (i.e., data association).

The remainder of this chapter is structured as follow: After presenting some basic definitions and formulations about the robust regression model in Section 4.3, we describe the mapping method based on the robust regression model and the related EKF-SLAM algorithm. Section 4.4 analyzes the relationship between the data association problem and minimum weight bipartite perfect matching and mathematically proves the optimality of the suggested algorithm. The experimental studies are depicted in Section 4.5. We discuss the test results and list some remarks in Section 4.6.

4.2 Robust Regression Model

Ordinary least squares (OLS) estimates for regression models are highly sensitive to outliers. There are several fundamental assumptions that have to be fulfilled for the regression model. When those assumptions are not met, the prediction and estimation of the model may become biased. Residuals that may be very large can

seriously distort the prediction. Those extremely large residuals are referred to as outliers that would inflate the error variance, make the estimation less asymptotically consistent, and bias the parameter estimates. The robust approach aims at deriving methods that produce reliable parameter estimates even when the data set does not exactly follow a prescribed distribution. A more informal data-oriented characterization of robust methods is that they fit the bulk of the data well. If the data contains no outliers the robust method gives approximately the same results as the OLS; whereas if a small proportion of outliers are present, the robust method gives approximately the same results as the classical method applied to the *typical* data. As a consequence of fitting the bulk of the data well, robust methods provide a very reliable approach of detecting outliers, even in high-dimensional multivariate situations. Figure 4.1 shows the fitted line on sensor data containing the outliers that correspond to the dynamic object or noise by OLS estimate and a robust estimate called MM-estimate respectively. In contrast, the robust line fits the bulk of the data well, and is expected to provide a reasonable prediction. In this section, a few concepts and formulae applied for mapping are discussed.



Least squares:	$Y = -1.2344 + 0.0082029 \cdot X$	RMS error = 0.021935
Robust:	$Y = -1.28546 + 0.0286407 \cdot X$	RMS error = 0.00949807

Figure 4.1 Fitted line determined by MM-estimate versus OLS estimate

for the data with one outlier.

The general linear regression model is formulated as follows:

$$\mathbf{y} = \mathbf{x}\boldsymbol{\beta} + u \quad (4.1)$$

where \mathbf{y} is the response, $\mathbf{x}=[1 \ x]$ is the independent variable, $\boldsymbol{\beta}=[b_0 \ b_1]^T$ is the unknown regression parameters, and u is the random error and independent of \mathbf{x} and \mathbf{y} . The fitted value and residuals corresponding to the vector $\boldsymbol{\beta}$ are defined respectively as $\hat{\mathbf{y}}(\boldsymbol{\beta}) = \mathbf{x}\boldsymbol{\beta}$ and $\mathbf{r}(\boldsymbol{\beta}) = \mathbf{y} - \hat{\mathbf{y}}(\boldsymbol{\beta})$. Assume model (4.1) with fixed \mathbf{x} (i.e. no noise exists in \mathbf{x}) where u has a probability density

$$\frac{1}{\sigma} f\left(\frac{u}{\sigma}\right) \quad (4.2)$$

where σ is a scale parameter. For the linear model (4.1), \mathbf{y} is independent but not identically distributed, and \mathbf{y} has the probability density

$$\frac{1}{\sigma} f\left(\frac{y - \mathbf{x}\boldsymbol{\beta}}{\sigma}\right) \quad (4.3)$$

The likelihood function for $\boldsymbol{\beta}$ assuming a fixed value of σ is

$$L(\boldsymbol{\beta}) = \frac{1}{\sigma^n} \prod_{i=1}^n f\left(\frac{y_i - \mathbf{x}_i\boldsymbol{\beta}}{\sigma}\right) \quad (4.4)$$

Calculating the Maximum Likelihood Estimate (MLE) means maximizing $L(\boldsymbol{\beta})$, which is equivalent to finding $\hat{\boldsymbol{\beta}}$ such that

$$\min \left(\frac{1}{n} \sum_{i=1}^n \rho\left(\frac{r_i(\hat{\boldsymbol{\beta}})}{\hat{\sigma}}\right) + \log \sigma \right) \quad (4.5)$$

where $\rho = -\log f$. Define regression M-estimate as solutions $\hat{\boldsymbol{\beta}}$ to

$$h(\boldsymbol{\beta}) = \min \left(\sum_{i=1}^n \rho\left(\frac{r_i(\hat{\boldsymbol{\beta}})}{\hat{\sigma}}\right) \right) \quad (4.6)$$

where $\hat{\sigma}$ is an error scale estimate. Differentiating (4.6) yields the equation

$$\sum_{i=1}^n \psi\left(\frac{r_i(\hat{\boldsymbol{\beta}})}{\hat{\sigma}}\right) \mathbf{x}_i = 0, \text{ where } \psi = \rho' \quad (4.7)$$

In most situations, $\hat{\sigma}$ is computed previously, but it can also be computed

simultaneously through a scale M-estimating equation (Maronna, et al., 2006). In many practical situations, the choice of ψ function is not critical to obtaining a good robust estimate, and many selections will give similar results that offer improvements in terms of efficiency and bias over classical estimates in the presence of outliers. Theoretically, re-descending ψ functions are to be preferred, and Tukey's biweight (also known as bi-square) function is a popular choice. The main advantage of monotone estimates is that all solutions of (4.7) are solutions of (4.6).

The approach to robust regression estimates where both \mathbf{x} and the \mathbf{y} may contain outliers is to use the M-estimate $\hat{\boldsymbol{\beta}}$ defined by (4.6) with a bounded ρ -function and a high breakdown point preliminary scale $\hat{\sigma}$. The further information on the requirements of this $\hat{\sigma}$ is explained in the reference (Maronna, et al., 2006). If ρ has a derivative ψ , it definitely follows (4.7), where ψ has re-descending form. Consequently the estimating (4.7) may have multiple solutions corresponding to multiple local minima of the function on the right-hand side of (4.6), and generally only one of them (the optimal solution) corresponds to the global minimized $\hat{\boldsymbol{\beta}}$ defined by (4.6). ρ and $\hat{\sigma}$ may be chosen in order to attain both a high breakdown point and a high efficiency. If there is a single stationary point i.e. the optimal solution $\hat{\boldsymbol{\beta}}_o$, then the solution of (4.7) converges to it so that $\hat{\boldsymbol{\beta}}_o$ is also the solution of (4.6). Please refer to Rousseeuw and Leroy (1987) for the proof in detail. The method to compute the approximated $\hat{\boldsymbol{\beta}}$ defined by (4.6) with a bounded ρ -function is called an MM-estimate, which refers to the fact that more than one M-estimation procedure is used to calculate the final estimate. MM-estimate attempts to retain the robustness and resistance of S-estimation, whilst gaining the efficiency of M-estimation. The first M means to find a highly robust and resistant S-estimate that minimizes an M-estimate of the scale of the residuals, and the second M indicates that the estimated scale is held constant

whilst a close-by M-estimate of the parameters is located.

The main difference between linear regression and robust regression is that in linear regression the parameters β are predicted by OLS algorithm, however for robust regression with the MM-estimate, the iteratively re-weighted least squares (IRLS) algorithm (Rousseeuw & Leroy, 1987) is employed to find the final estimate. The weights accompany with the points $[x_i, y_i]$ at each iteration are calculated by applying a ρ function to the residuals from the previous iteration. This IRLS algorithm gives lower weight to points that do not fit well. The results are less sensitive to outliers as compared with linear regression based on OLS. Concerning the properties of ψ function stated above, in this study we selected the bi-square family of functions as the bounded ρ function

$$\rho(x) = \begin{cases} -[1 - (x/k)^2]^3, & \text{if } |x| \leq k \\ 1, & \text{if } |x| > k \end{cases} \quad (4.8)$$

and the derivative $\rho'(x) = 6\psi(x)/k^2$ where

$$\psi(x) = x[1 - (x/k)^2]^2, \quad |x| \leq k \quad (4.9)$$

The relevant weight function is chosen as

$$W(x) = \begin{cases} \psi(x)/x = [1 - (x/k)^2]^2, & |x| \leq k, x \neq 0 \\ \psi'(x), & x = 0 \end{cases} \quad (4.10)$$

Considering the advantages of the robust regression, we apply it to our segment-based mapping algorithm which will be interpreted in the following section. Figure 4.1 shows that the MM-estimate is more efficient than OLS and Figure 4.2 illustrates that the result of the MM-estimate almost coincides with that of OLS estimate after removal of the outlier data point.

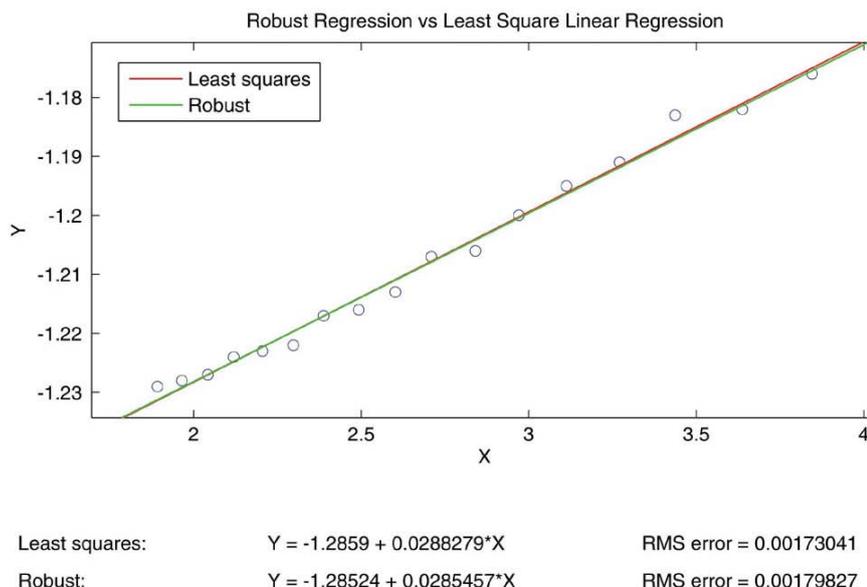


Figure 4.2 Fitted line determined by MM-estimate versus OLS estimate

for the data after removal of the outlier.

4.3 SLAM Algorithm with Robust Regression Mapping

4.3.1 Sensor readings grouping based on Robust Regression Model

The idea of grouping algorithm is similar to incremental algorithm, but with robust regression fewer thresholds are used. The equation of the line adopts the Hesse standard form. The laser readings with the angular resolution of 1° are processed to extract the segment parameters: p and a which are the perpendicular distance of the line from the origin and the orientation of this perpendicular line with respect to (w.r.t.) robot framework respectively. Besides these two parameters for measurement representation, we introduce other two extra parameters: unit eigenvector and the variance ratio of slope. Their role will be interpreted in more details later. The laser rangefinder provides N positions of points by Cartesian coordinates $[x_p, y_p]^T$ in a frame of sensor data. We group these sensor readings into a certain segment set L_s through two partition processes. The former is a rough one which builds the segment set iteratively, and the latter is a refined one which deletes the points with low weight values from the segment set built through the

rough process. As the variance of laser reading described in its manual is valid when the range is less than or equal to 8 meters, we filtered out the measurements which were larger than 8 meters before processing.

In rough grouping, the segment set $L_{s(i)}$ is initialized by the first two points P_1 and P_2 in a frame of data and the related regression parameters β in (4.1) are obtained by OLS based linear regression. When a new point P_{newj} is coming into the current segment set, β is recomputed with this new point. However, now the computational strategy is not linear regression but the robust regression with the MM-estimate. After re-computation, the regression parameters as well as another important value, the weight W for each point, are obtained. If the $w_{p(j)}$ approximates to zero (in this study we use $w_{p(j)}=0$), then the point P_j is removed from the current segment set; otherwise P_j is assigned to current segment set, where the P_j can be the new point P_{newj} or the one existing in the segment set. Particularly, when the point whose $w_{p(j)}$ closes to zero is the last point, except deleting it we concurrently put it into a temporal segment set. If such N_p (we selected 4 in this study) points appear continuously, we build the new segment set $L_{s(i+1)}$ and in the next step the regression parameters and weight values are computed based on this new segment set by MM-estimate. This process is illustrated in Figure 4.3.

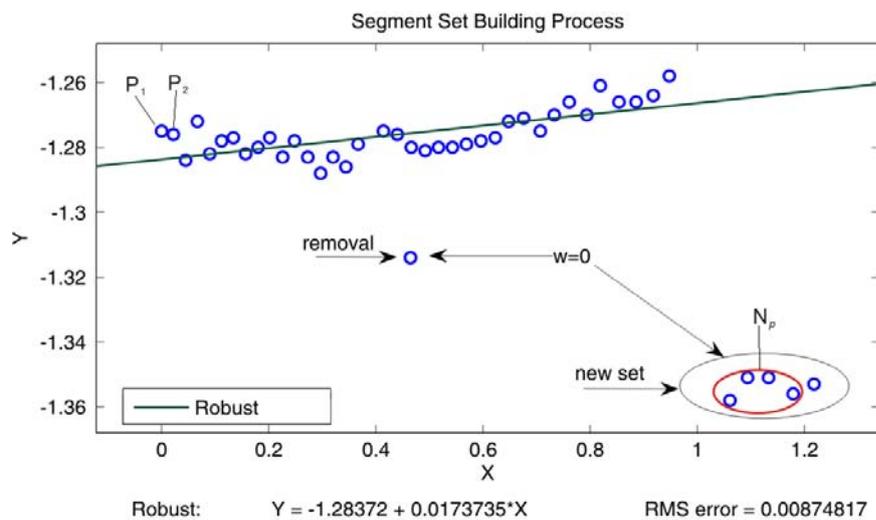


Figure 4.3 The process of building segment set.

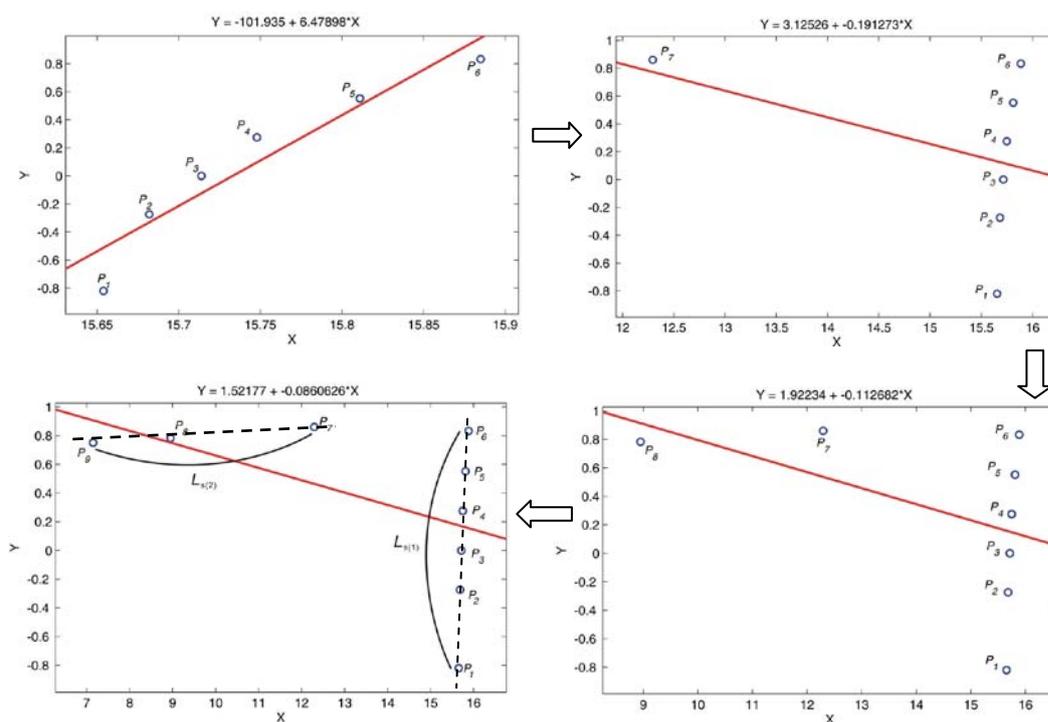


Figure 4.4 Regression parameters catastrophe

Since the laser rangefinder scans the objects in counterclockwise direction, several points have the approximate x or y coordinates. They may probably belong to different segments but are wrongly grouped into one segment by robust regression. For example in Figure 4.4, P_1 to P_6 fit the segment represented in red line shown in the upper left subfigure. When more points P_7 , P_8 and P_9 that have approximate y coordinates as P_6 are added, they cannot be treated as the outliers and misidentified same group (red line) as P_1 to P_6 , which is illustrated in the upper right, lower right and lower left subfigures of Figure 4.4. Actually it appears that P_1 to P_6 fit the segment $L_{s(1)}$ and P_7 to P_9 match the segment $L_{s(2)}$ (cf. lower left subfigure in Figure 4.4). For this case, it is not sufficient to category the readings and extract the segment parameters only by using robust regression. It can be seen from Figure 4.4 that the regression parameter β_1 (i.e. slope) has a serious change before and after P_7 inserted but when other two points P_8 and P_9 are added the variance ratio of β_1 tends to be smooth. Therefore, we design a parameter called variance ratio of slope (VROS) to detect the special case stated above. The

expression for VROS is

$$VROS(j-1) = \frac{|\beta_{1(j-1)} - \beta_{1(j)}|}{\Delta t_{resp}} \quad (4.11)$$

where $\beta_{1(j-1)}$ and $\beta_{1(j)}$ are slope parameters before and after the j th point P_j is added into computation respectively, Δt_{resp} is the response time interval between obtaining the measurements of P_{j-1} and P_j . Here we use 13ms which is the response time of SICK LMS200 laser. The $(j-1)$ on the left side means the j th point P_j is not the real element of the current segment set and only used to checked whether it belongs to the current data set. The process for detecting the special case by VROS is stated in Table 4.1.

Table 4.1 The process for detecting the special case via VROS

Step	Process
1°	Compute $\beta_{1(j-1)}$ and $\beta_{1(j)}$ before and after the checked point P_j added into the current segment set, and then compute $VROS(j-1)$;
2°	If $VROS(j-1)$ is larger than Th_{slop} then go to step 4°;
3°	If $Flag_{VROS}$ is TRUE then go to step 4° else go to step 7°
4°	$nBuf = nBuf + 1$, store $VROS(j-1)$ into VROS buffer and let $Buf_{VROS}(nBuf)$ equal to $VROS(j-1)$. Let $Flag_{VROS} = TRUE$;
5°	If $nBuf$ equals to 3 then go to step 6° else go to step 7°
6°	In VROS buffer, $Buf_{VROS}(1)$ is surely larger than Th_{slop} and if $Buf_{VROS}(2)$ and $Buf_{VROS}(3)$ are both below Th_{slop} then a new segment set is built i.e. $L_{news} = \{j, j+1, j+2\}$. If not both $Buf_{VROS}(2)$ and $Buf_{VROS}(3)$ are less than Th_{slop} , then it means there exists outliers and they can be removed by another condition: $w_{p(j)}$ closes to 0. Reset $nBuf$ to 0 and $Flag_{VROS}$ to FALSE. Stop process.
7°	Let $\beta_{1(j-1)}$ equal to $\beta_{1(j)}$ and go to step 1°

where $nBuf$ is the number of VROS buffer, $Flag_{VROS}$ is the flag which identifies the event that the severe change of slope occurred. Th_{slop} is determined through testing on numerous sensor data which were obtained in different corridors.

For example in Figure 4.4, before P_7 is coming, $\beta_1(6)$ is 6.479; when P_7 is checked, $\beta_1(7)$ is -0.1913, so the $VROS(6)$ is 513. Assuming Th_{slop} is 10^2 , $VROS(6)$ is over Th_{slop} hence store it into $Buf_{VROS}(1)$. Similarly, $VROS(7)=6.05$ and $VROS(8)=2.05$ are also saved into $Buf_{VROS}(2)$ and $Buf_{VROS}(3)$. Obviously, $VROS(7)$ and $VROS(8)$ are less than Th_{slop} , therefore P_7, P_8 and P_9 consist of the new segment set $L_{s(2)}$.

When a frame of sensor data was grouped into different segment sets L_s (some data were removed as the outliers), it is necessary for these sets to undergo another refined process in order to ensure that the points with low weights do not influence the parameter extraction in the next phase. This refinement is also the robust regression process similar to the rough partition; but the implementation is only one step not an iteration routine.

4.3.2 Computation of Segment Parameters

After the segment set L_s has been identified, the segment parameters p and a w.r.t. local framework are computed by all points that belong to L_s . The cost function of parameters extraction is same as the formula described by Garulli et al. (2005).

Denoting the j th points in the i th set $L_{s(i)}$ as $P_j^i = [x_j^i, y_j^i]^T$, the cost function is

$$[p_i, a_i] = \arg \min_{p, a} E(p, a) \quad (4.12)$$

$$\text{where } E(p, a) = \sum_{j=1}^{n_i} (p - x_j^i \cos a - y_j^i \sin a)$$

n_i is the number of points in segment set $L_{s(i)}$. The covariance of \mathbf{R} for parameters p and a is also provided, which is calculated as Garulli's method (Garulli, et al., 2005).

Besides these two parameters for implementing SLAM, we also compute another parameter referred to as unit eigenvector for merging the compatible segments. We borrowed the idea of weight computation in split-merge-split-merge mapping method (Xu, et al., 2003) to compute a modified covariance matrix. From this matrix we derived the unit eigenvector. For the point P_j of each segment set $L_{s(i)}$, define D_{for} and D_{bac} as the distances from the j th point to $(j-1)$ th point and from the j th point to $(j+1)$ th point respectively, then take $(D_{for}+D_{bac})$ as the weight w_j^i of P_j . Note that the first and last points only have the D_{bac} and D_{for} which are

regarded as the related weights. All these weights consist of a weight vector $\mathbf{W}_{L_{s(i)}}$ for segment $L_{s(i)}$. After standardizing this weight vector, we compute modified covariance matrix as follows

$$\begin{aligned}
 m'_x &= \sum_{j=1}^{n_i} w_j^i x_j^i, \quad m'_y = \sum_{j=1}^{n_i} w_j^i y_j^i, \quad S'_{xx} = \sum_{j=1}^{n_i} w_j^i (x_j^i - m'_x)^2 \\
 S'_{yy} &= \sum_{j=1}^{n_i} w_j^i (y_j^i - m'_y)^2, \quad S'_{xy} = \sum_{j=1}^{n_i} w_j^i (x_j^i - m'_x)(y_j^i - m'_y) \\
 C_m &= \begin{pmatrix} S'_{xx} & S'_{xy} \\ S'_{xy} & S'_{yy} \end{pmatrix} \tag{4.13}
 \end{aligned}$$

From modified covariance matrix C_m , the maximal eigenvalue and corresponding unit eigenvector labeled as λ_m and ϕ_m can be determined. With the unit eigenvector obtained from (4.13), we merged compatible line segments, of which the technique is the same as the predecessor work (Ip, et al., 2002). Note that after this merging process, the parameters of new segment should be re-computed via robust regression process.

4.3.3 Extended Kalman Filter based SLAM (EKF-SLAM)

As the mobile robot does not have prior knowledge of the environment, the research on SLAM seeks to enable the robot to move through its environment and build a consistent map of its surroundings as well as an estimate of its own trajectory using only onboard sensors. In this thesis, the most influential and popular algorithm, Extended Kalman filter (EKF) (Thrun, Burgard, & Fox, 2005), is applied to SLAM with graph theory based data association which will be presented in the next section. Let the robot position at time zero as the origin of the world coordinate system, i.e. $\mathbf{x}_R(0)=[x_R(0), y_R(0), \phi_R(0)]^T$. Figure 4.5 gives the geometric relationship between the robot and segment in local and global framework.

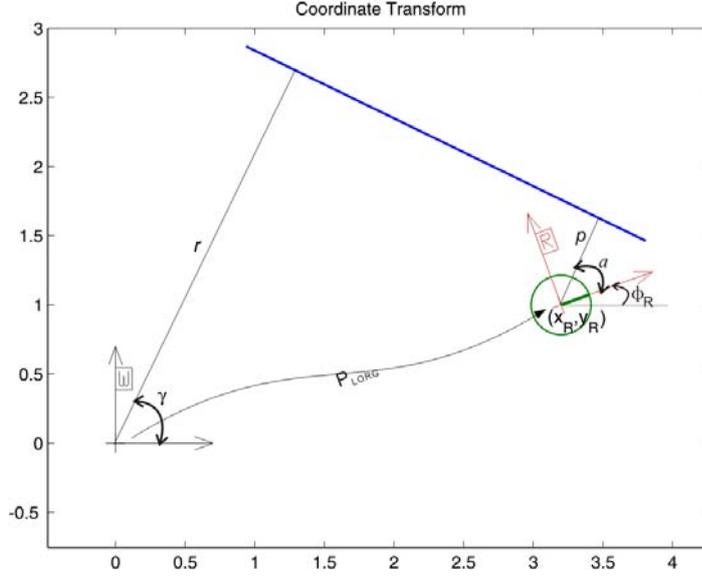


Figure 4.5 Measurement model and the geometric relationship between the robot and features

Given at time k , the kinematical equation describing the robot motion w.r.t. the global framework is

$$\begin{aligned} \mathbf{x}_R(k+1) &= f(\mathbf{x}_R(k), u(k)) + v(k) \\ &= \begin{bmatrix} x_R(k) \\ y_R(k) \\ \phi_R(k) \end{bmatrix} + \begin{bmatrix} v_{tr}(k)\Delta t \cos \phi_R(k) \\ v_{tr}(k)\Delta t \sin \phi_R(k) \\ \omega_{rot}(k)\Delta t \end{bmatrix} + v(k), \quad v(k) \sim N(0, Q(k)) \end{aligned} \quad (4.14)$$

where $v_{tr}(k)$ is the translational velocity and $\omega_{rot}(k)$ is the angular velocity, and they contribute to control variable $u(k)=[s(k), a(k)]^T$, where $s(k)=v_{tr}(k)\Delta t$, $a(k)=\omega_{rot}(k)\Delta t$. $u(k)$ is assumed to be corrupted by a zero-mean and variance $\text{var}(u(k))$ Gaussian noise process. There are three types of movement errors: k_R is range error factor (unit: mm^2/m), k_θ is turn error factor (unit: deg^2/deg) and k_D is drift error factor (unit: deg^2/m). The variance of $u(k)$ is

$$\begin{aligned} \text{var}(u(k)) &= \text{diag}([\text{var}(s(k)), \text{var}(a(k))]) \\ &= \text{diag}([k_R | s(k)|, k_\theta | a(k)| + k_D | s(k)|]) \end{aligned}$$

Hence the covariance for the process noise $v(k)$ is

$$Q(k) = \begin{bmatrix} k_R s(k) \cos^2 \phi_R(k) & k_R s(k) \sin \phi_R(k) \cos \phi_R(k) & 0 \\ k_R s(k) \sin \phi_R(k) \cos \phi_R(k) & k_R s(k) \sin^2 \phi_R(k) & 0 \\ 0 & 0 & k_\theta a(k) + k_D s(k) \end{bmatrix}$$

When the robot is moving, it is able to measure the information about the segment

from its current pose $\mathbf{x}_R(k)$. Let $\mathbf{z}_i(k)=[p_i(k), a_i(k)]^T$, $i=1, \dots, n$ denote the parameter of the i th feature in the local reference frame. Then, from Figure 4.5, it can be expressed as a function of the current robot pose $\mathbf{x}_R(k)$ and the parameters $l_{fi}=[r_{fi}, \gamma_{fi}]^T$ of the sensed feature in the global reference frame. The measurement model is

$$\begin{aligned} \mathbf{z}_i(k) &= h(\mathbf{x}_R(k), l_{fi}) + w_i(k) \\ &= \begin{bmatrix} r_{fi} - x_R(k) \cos \gamma_{fi} - y_R(k) \sin \gamma_{fi} \\ \gamma_{fi} - \phi_R(k) \end{bmatrix} + w_i(k), \quad w_i(k) \sim N(0, R_i(k)) \end{aligned} \quad (4.15)$$

where $w_i(k)$ represents measurement noise. It is also assumed to be a zero-mean Gaussian with covariance $R_i(k)$ that is calculated from last subsection.

With the motion model (4.14), measurement model (4.15) and initial conditions: $\mathbf{x}_R(0)$, $\mathbf{P}(0)$, $\mathbf{Q}(0)$ and $\mathbf{R}(0)$, the time update equations are

$$\hat{\mathbf{x}}_k^- = f(\mathbf{x}_{k-1}, u_k) = \begin{bmatrix} x_{R_{k-1}} \\ y_{R_{k-1}} \\ \phi_{R_{k-1}} \end{bmatrix} + \begin{bmatrix} s_k \cos \phi_{R_{k-1}} \\ s_k \sin \phi_{R_{k-1}} \\ a_k \end{bmatrix} \quad (4.16)$$

$$\mathbf{P}_k^- = \mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^T + \mathbf{W}_k \mathbf{Q}_{k-1} \mathbf{W}_k^T \quad (4.17)$$

where \mathbf{A}_k and \mathbf{W}_k are the process Jacobian matrix w.r.t. robot pose and control variable respectively at step k . They are given as

$$\mathbf{A}_k = \begin{bmatrix} \frac{\partial f_x}{\partial x_{R_{k-1}}} & \frac{\partial f_x}{\partial y_{R_{k-1}}} & \frac{\partial f_x}{\partial \phi_{R_{k-1}}} \\ \frac{\partial f_y}{\partial x_{R_{k-1}}} & \frac{\partial f_y}{\partial y_{R_{k-1}}} & \frac{\partial f_y}{\partial \phi_{R_{k-1}}} \\ \frac{\partial f_\phi}{\partial x_{R_{k-1}}} & \frac{\partial f_\phi}{\partial y_{R_{k-1}}} & \frac{\partial f_\phi}{\partial \phi_{R_{k-1}}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -s_k \sin \phi_{R_{k-1}} \\ 0 & 1 & s_k \cos \phi_{R_{k-1}} \\ 0 & 0 & 1 \end{bmatrix}$$

$$W_k = \begin{bmatrix} \frac{\partial f_x}{\partial s_k} & \frac{\partial f_x}{\partial a_k} \\ \frac{\partial f_y}{\partial s_k} & \frac{\partial f_y}{\partial a_k} \\ \frac{\partial f_\phi}{\partial s_k} & \frac{\partial f_\phi}{\partial a_k} \end{bmatrix} = \begin{bmatrix} \cos \phi_{R_{k-1}} & 0 \\ \sin \phi_{R_{k-1}} & 0 \\ 0 & 1 \end{bmatrix}$$

Once measurements \mathbf{z}_k become available the Kalman gain matrix \mathbf{K}_k is computed and used to incorporate the measurement into the state estimate. The state error covariance for the updated state estimate \mathbf{P}_k is also computed using the following measurement update equations

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \quad (4.18)$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - h(\hat{\mathbf{x}}_k^-, l_f)) \quad (4.19)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \quad (4.20)$$

where \mathbf{z}_k is the observation obtained from laser readings, and $h(\hat{\mathbf{x}}_k^-, l_f)$ is the predicted features. \mathbf{H}_k is the measurement Jacobian matrix w.r.t. robot pose and map features at step k . It is

$$H_k = \begin{bmatrix} \left(\begin{array}{ccc} \frac{\partial h_{p_1}}{\partial \hat{x}_{R_k}^-} & \frac{\partial h_{p_1}}{\partial y_{R_k}^-} & \frac{\partial h_{p_1}}{\partial \hat{\phi}_{R_k}^-} \\ \frac{\partial h_{a_1}}{\partial \hat{x}_{R_k}^-} & \frac{\partial h_{a_1}}{\partial y_{R_k}^-} & \frac{\partial h_{a_1}}{\partial \hat{\phi}_{R_k}^-} \\ \vdots & \vdots & \vdots \\ \frac{\partial h_{p_n}}{\partial \hat{x}_{R_k}^-} & \frac{\partial h_{p_n}}{\partial y_{R_k}^-} & \frac{\partial h_{p_n}}{\partial \hat{\phi}_{R_k}^-} \\ \frac{\partial h_{a_n}}{\partial \hat{x}_{R_k}^-} & \frac{\partial h_{a_n}}{\partial y_{R_k}^-} & \frac{\partial h_{a_n}}{\partial \hat{\phi}_{R_k}^-} \end{array} \right) & \left(\begin{array}{cc} \frac{\partial h_{p_1}}{\partial r_{f_1}} & \frac{\partial h_{p_1}}{\partial \gamma_{f_1}} \\ \frac{\partial h_{a_1}}{\partial r_{f_1}} & \frac{\partial h_{a_1}}{\partial \gamma_{f_1}} \\ \vdots & \vdots \\ \frac{\partial h_{p_n}}{\partial r_{f_n}} & \frac{\partial h_{p_n}}{\partial \gamma_{f_n}} \\ \frac{\partial h_{a_n}}{\partial r_{f_n}} & \frac{\partial h_{a_n}}{\partial \gamma_{f_n}} \end{array} \right) \\ \left(\begin{array}{ccc} -\cos \gamma_{f_1} & -\sin \gamma_{f_1} & 0 \\ 0 & 0 & -1 \\ \vdots & \vdots & \vdots \\ -\cos \gamma_{f_n} & -\sin \gamma_{f_n} & 0 \\ 0 & 0 & -1 \end{array} \right) & \left(\begin{array}{ccc} 1 & \hat{x}_{R_k}^- \sin \gamma_{f_1} - y_{R_k}^- \cos \gamma_{f_1} & 0 \\ 0 & 1 & 0 \\ \vdots & \vdots & \vdots \\ 0 & \hat{x}_{R_k}^- \sin \gamma_{f_n} - \hat{y}_{R_k}^- \cos \gamma_{f_n} & 1 \\ 0 & 0 & 1 \end{array} \right) \end{bmatrix}$$

It should be noted that we still applied the fuzzy tuning strategy (Ip, Rad, & Wong, 2004) to cope with the process covariance matrix $\mathbf{Q}(k)$ of which the k_R , k_θ and k_D are adjusted by a series of fuzzy rules. For invisible segments, p will be negative and a is added by π ; further the related elements in \mathbf{H}_k have to be affixed with a *minus*.

4.4 Graph Theory based Data Association

Data association in SLAM is a decision process of associating measurements with existing features in the stochastic map. We start by formulating the problem as a 0-1 integer programming (IP) or 2D assignment problem.

4.4.1 Formulation of IP Problem

At time step k , denote a set of measurements collected in the latest scan by $\mathbf{Z}(k)$ and a set of features (here we use landmarks to make an explanation) existing in the map so far by $\mathbf{F}(k)$, i.e.,

$$\mathbf{Z}(k) \square \{z_i(k) : i = 1, 2, \dots, n_k\} \quad (4.21)$$

$$\mathbf{F}(k) \square \{l_j(k) : j = 1, 2, \dots, m_k\} \quad (4.22)$$

where n_k is the number of actual measurements at time k and m_k the number of existing landmarks in the map up to time k . Note that $z_0(k)$ and $l_0(k)$ are the dummy elements in the case of a false alarm or new landmark is detected. Next, we introduce a 0-1 decision variable.

$$x_{ij} \square \begin{cases} 1, & \text{if } z_i(k) \text{ associated with } l_j(k) \\ 0, & \text{otherwise} \end{cases} \quad (4.23)$$

Two special cases shall be emphasized here: $x_{i0}^k = 1$ stands that i th measurement can not be assigned to any of the existing landmarks in the map and therefore assigned with a dummy one which may be false alarm or new landmark; $x_{0j}^k = 1$

implies that j th landmark in the map does not have any possible measurement associated with it in the current scan. It should be noted that in tracking typically we do not define the $j=0$ case. We do so in order to get all equality constraints which will become much clearer after the discussion below.

There are two important physical constraints imposed on the data association problem as Li, Luo, Wong and Bosse (1999) discussed. (i) *Single source constraint*: Each actual measurement $z_i(k)$ ($i=1,2,\dots,n_k$) can be assigned to at most one landmark. However, the dummy measurement $z_0(k)$ can be assigned to multiple landmarks in the case of false detection or new detected landmark. We therefore set it free to have the following equality constraint.

$$\sum_{j=0}^{m_k} x_{ij}(k) = 1, \forall i = 1, 2, \dots, n_k \quad (4.24)$$

and (ii) *Single return constraint*: Each landmark $l_j(k)$ ($j=1,2,\dots,m_k$) can produce at most one measurement in current scan. Clearly not all landmarks can return measurements. In other words, some existing landmarks are undetected in current scan and hence we assign these undetected landmarks dummy reports.

$$\sum_{i=0}^{n_k} x_{ij}(k) = 1, \forall j = 1, 2, \dots, m_k \quad (4.25)$$

Our objective is to exactly match the sensor observations with the existing landmarks in the map. Similar to the multi-target tracking problem(Li, et al., 1999), the cost of a feasible association of measurement with existing landmark or new landmarks (or false alarm) is as the negative logarithm of the normalized joint probability of such an association. Define the set of all possible association pairs at current time step $\Omega(k) \square \{(i, j) : z_i(k) \in \mathbf{Z}(k); l_j(k) \in \mathbf{F}(k); i, j \neq 0\}$, and a partition of the set $\omega = \{\omega_T, \omega_F\}$, where ω_T denotes the set of measurements associated with existing landmarks in the map while ω_F is the set of measurements associated with new landmarks or false alarms. Note that we do not distinguish the cases of new landmark and false alarm. We instead assume all measurements not associated in

current scan are new landmarks, because if spurious landmarks are detected, they will be removed from the map if they do not appear in the following scans. Thus, ω_F only stands for the case of association with new landmarks. Therefore, the likelihood of detecting new landmarks is simply approximated with 1, i.e. $\Lambda(\omega_F)=1$. For the true associations, we have

$$\Lambda(\omega_T) = \prod_{(i,j) \in \omega_T} q(z_l) \quad (4.26)$$

where z_l denotes the true association of $z_i(k)$ and $l_j(k)$, and $q(z_l)$ is a Gaussian probability density function of the measurement variable $z_i(k)$, i.e.

$$q(z_l) \square N(z_i(k) : \hat{z}_j(k | k-1), S_i(k)) \quad (4.27)$$

where $\hat{z}_j(k | k-1)$ is the estimate of landmark $l_j(k)$, and $S_i(k)$ is the covariance matrix of the residual $z_i(k) - \hat{z}_j(k | k-1)$. Both quantities are obtained in the update step in EKF-SLAM. Therefore, the likelihood of the partition ω can be calculated as follows (the time index is temporarily dropped to preserve the clarity of the presentation):

$$\begin{aligned} \Lambda(\omega) &= \Lambda(\omega_T)\Lambda(\omega_F) = \prod_{(i,j) \in \omega_T} q(z_l) \\ &= \prod_{(i,j) \in \omega_T} \frac{1}{\sqrt{\det(2\pi S_i)}} \exp\left\{-\frac{1}{2}(z_i - \hat{z}_j)^T S_i^{-1}(z_i - z_j)\right\} \end{aligned} \quad (4.28)$$

In order to ensure the likelihood is consistent, normalization of $\Lambda(\omega)$ is necessary. Our objective is rephrased to find the one with maximum likelihood $\Lambda(\omega)$ among all possible partitions ω , which is equivalent to minimizing the negative log-likelihood $-\ln\Lambda(\omega)$, i.e. $J(\omega) \square -\ln\Lambda(\omega) = -\sum_{(i,j) \in \omega} \ln q(z_l)$. By defining the cost coefficient

$$c_{ij}(k) \square \begin{cases} 0, & (i,j) \in \omega_F \\ -\ln q(z_l), & (i,j) \in \omega_T \end{cases} \quad (4.29)$$

the data association of SLAM can be formulated as the following 0-1 IP problem:

$$\begin{aligned} \mathbf{\Pi}_1 : \quad & \min \sum_{(i,j) \in \omega} c_{ij}(k) x_{ij}(k) \\ & \text{s.t. Equation (4.23), (4.24) and (4.25)} \end{aligned} \quad (4.30)$$

Notice that any solution to this IP corresponds to a matching and therefore this is a valid formulation of the minimum weight perfect matching problem in bipartite graphs (West, 2001), which will be elaborated in subsection 4.4.3.

4.4.2 Formulation of LP Relaxation

Consider now the linear programming (LP) obtained by simply dropping the integrality constraints:

$$\begin{aligned} \mathbf{\Pi}_2 : \quad & \min \sum_{(i,j) \in \omega} c_{ij}(k) x_{ij}(k) \\ & \text{s.t. } x_{ij} \geq 0, \text{ Equation (4.24) and (4.25)} \end{aligned} \quad (4.31)$$

This is the LP relaxation of the above IP problem ($\mathbf{\Pi}_1$), which has been explored by Zhang et al. (2005). In an LP, the variables can take fractional values and therefore there are many feasible solutions to the set of constraints above which do not correspond to matching. But we only care about the optimum solutions. The set of feasible solutions to the constraints in $\mathbf{\Pi}_2$ forms a polytope, and when we optimize a linear constraint over a polytope, the optimum will be attained at one of the corners or extreme points of the polytope. In general, even if all the coefficients of the constraint matrix in an LP are either 0 or 1, the extreme points of an LP are not guaranteed to have all coordinates integral (This is of no surprise since the general IP problem is NP-hard, while LP is polynomially solvable). As a result, there is no guarantee that the optimum solution of IP ($\mathbf{\Pi}_1$) is equal to optimum solution of its LP relaxation ($\mathbf{\Pi}_2$). However, $\mathbf{\Pi}_2$ provides a lower bound of $\mathbf{\Pi}_1$. Moreover, the following lemma is easy to prove.

Lemma 1. If an optimum solution to $\mathbf{\Pi}_2$ is integral, then it must also be an optimum solution to $\mathbf{\Pi}_1$.

Proof. The integral optimum solution to $\mathbf{\Pi}_2$ satisfies all the constraints of $\mathbf{\Pi}_1$. ■

4.4.3 Minimum Weight Bipartite Perfect Matching

By assigning infinite costs to the edges not present, we assume that the bipartite graph is complete. The minimum cost (or weight) perfect matching problem is often described by the following account: There are n jobs to be processed on n machines and one would like to process exactly one job per machine such that the total cost of processing the jobs is minimized. Analogue to this story, with help of the dummy variables, the IP formulation of data association in SLAM, Π_1 , can be considered as a minimum weight bipartite perfecting matching problem. Instead of using optimization method directly to solve the data association problem, we shall employ graph approaches to solve the equivalent bipartite matching problem. In the case of the perfect matching problem, the constraint matrix has a very special form and one can show that the optimality of the solutions can be preserved. To do so, we start by stating a very crucial result—*Lemma 2*—in the following, and also describe the purely algorithmic proof.

Lemma 2. Any extreme point of the polytope of Π_2 is a 0-1 vector and, hence, is the incidence vector of a perfect matching.

Proof. To prove algorithmically, we construct a primal-dual algorithm for solving the minimum weight perfect matching problem. Suppose in a specific case of the bipartite matching problem, we have measurement u_i and landmark v_j such that $u_i + v_j \leq c_{ij}$. The dual of the LP relaxation, Π_2 , can be obtained as follows:

$$\begin{aligned} \Pi_3 : \quad & \max \sum_{(i,j) \in \omega} (u_i + v_j) \\ & \text{s.t. } u_i + v_j \leq c_{ij} \end{aligned} \tag{4.32}$$

The dual constraints can be interpreted as $w_{ij} \geq 0$, where $w_{ij} = c_{ij} - u_i - v_j$. If, for any instance, we could always find a feasible solution u, v to the dual Π_3 and hence a perfect matching such that equalities in Equation (4.23) hold (i.e. the cost of the perfect matching is equal to the value of the dual solution). Thus, we would know that the matching found is optimum. Given a solution u, v to the dual, a perfect matching would satisfy equality if it contains only edges (i, j) such that

$w_{ij}=c_{ij}-u_i-v_j=0$. This is what is referred to as complementary slackness. However, for a given u & v , we may not be able to find a perfect matching among the edges with $w_{ij}=0$. The algorithm performs a series of iterations. It always maintains a dual feasible solution and tries to find an almost primal feasible solution satisfying complementary slackness. The fact that complementary slackness is imposed is crucial in any primal-dual algorithm.

More precisely, the algorithm works as follows. It first starts with any dual feasible solution, say $u_i=0$ for all i and $v_j=\min_{i \in \omega} c_{ij}$ for all j . In a given iteration, the algorithm has a dual feasible solution (u, v) or say (u, v, w) . Imposing complementary slackness means that we are interested in matchings which are subgraphs of $B=\{(i, j): w_{ij}=0\}$. If B has a perfect matching then the incidence vector of that matching is a feasible solution in Π_2 and satisfies complementary slackness with the current dual solution and, hence, must be optimal. To check whether B has a perfect matching, one can use the cardinality matching method. If the maximum matching output is not perfect, then the algorithm will use information from the optimum vertex cover C^* to update the dual solution in such a way that the value of the dual solution increases. Recall that we are maximizing the dual.

Let the set L (for labeling) of vertices which can be reached by a directed path from an exposed vertex in measurement set Z . In particular, there is then no edge of B between $Z \cap L$ and $F \cap L$, where we remind that F is the mapped feature set. In other words, for every $i \in (Z \cap L)$ and every $j \in (F - L)$, we have $w_{ij} > 0$. Let $\delta = \min_{i \in (Z \cap L), j \in (F - L)} w_{ij}$. By the above argument, $\delta > 0$. The dual solution is updated as follows:

$$u_i = \begin{cases} u_i, & i \in Z - L \\ u_i + \delta, & i \in Z \cap L \end{cases} \quad (4.33)$$

$$v_j = \begin{cases} v_j, & j \in F - L \\ v_j - \delta, & j \in F \cap L \end{cases} \quad (4.34)$$

One easily check that this dual solution is feasible, in the sense that the corresponding vector \mathbf{w} satisfies $w_{ij} \geq 0$ for all i and j . The difference between the values of the new dual solution and the old dual solution is equal to:

$$\begin{aligned} \delta(|Z \cap L| - |F \cap L|) &= \delta(|Z \cap L| + |Z - L| - |Z - L| - |F \cap L|) \\ &= \delta(n - |C^*|) \end{aligned} \quad (4.35)$$

where Z has size of n and C^* is the optimum vertex cover for the bipartite graph with edge set B . But by assumption $|C^*| < n$, implying that the value of the dual solution strictly increases.

This procedure is repeated until the algorithm terminates. At that point, we have an incidence vector of a perfect matching and also a dual feasible solution which satisfy complementary slackness. They must therefore be optimal and this proves the existence of an integral optimum solution to Π_2 . Since, by carefully choosing the cost function, one can make any extreme point be the unique optimum solution to the linear program. Now we need to prove that the algorithm indeed terminates. Notice that at least one more vertex of F must be reachable from an exposed vertex of Z (and no vertex of F becomes unreachable), since an edge $e=(i, j)$ with $i \in (Z \cap L)$ and $j \in (F - L)$ now has $w_{ij}=0$ by our choice of δ . This also gives an estimate of the number of iterations. In at most n iterations, all vertices of F are reachable or the matching found has increased by at least one unit. Therefore, after $O(n^2)$ iterations, the matching found is perfect. ■

Now we reach the core of our findings, which is the equivalence between IP problem and minimum weight bipartite perfect matching problem.

Lemma 3. Solving Π_1 is equivalent to solve a corresponding minimum weight bipartite perfect matching.

Proof. With Lemma 1 and 2, the optimum solution of the minimum weight

bipartite perfect matching is also optimum solution to Π_1 . ■

Therefore, instead of solving the original IP problem (i.e. Π_1) directly, we resolve the minimum weight bipartite perfect matching problem to obtain the optimum solution to the data association.

4.4.4 Algorithm based on Weighted Bipartite Matching

In this subsection, we focus on finding the minimum weight matching in the bipartite matching. The general idea is straightforward: start with any empty matching, and repeatedly discover augmenting paths. The weight is computed from the Mahalonobis distance that is

$$d_M = [\mathbf{z}_k - h(\hat{\mathbf{x}}_k, l_f)]^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} [\mathbf{z}_k - h(\mathbf{x}_k^-, l_f)]$$

Several essential definitions are first delivered (West, 2001). Given a matching M in a bipartite graph $G=(V, E)$, a simple path in G is called an *augmenting path* w.r.t. M if its two vertices are both unmatched and its edges are alternative in $E-M$ and M . Let p_{ap} be an augmenting path w.r.t. M , and P denote the set of edges in path p_{ap} , then $M \oplus P \square (M - P) \cup (P - M)$ is called *symmetric difference* of M and P . One can verify the following properties of $M \cup P$: (i) it is a matching, and (ii) $|M \cup P| = |M| + 1$. The total weight of matching M is $w(M) = \sum_{e \in M} w(e)$. Suppose M' be a set of edges. An incremental weight $\Delta M'$ is defined as $\Delta M' = w(M' \cap M) - w(M' - M)$. From this definition, for an augmenting path p_{ap} w.r.t. M , ΔP gives the net change in the weight of the matching after augmenting p_{ap} , i.e.

$$w(M \cup P) = w(M) + \Delta P \quad (4.36)$$

The minimum weight matchings are found iteratively. Specifically, the matching M is initialized to be empty. At each iteration, M is increased by finding an augmenting path of minimum weight. The procedure stops till no augmenting path w.r.t. M can be found. Johnson and Mcgeoch (1993) already proved that the

process yields a minimum weight matching if repeatedly performing augmentations by using augmenting paths of minimum incremental weight. In order to search augmenting paths w.r.t. matching M systematically and efficiently, a search starts by constructing alternating paths from the unmatched points. As an augmenting path must have one unmatched endpoint in Z and the other in F , in general, the search starts by growing alternating paths only from unmatched vertices of Z , and may search for all possible alternating paths from unmatched vertices of Z simultaneously in a breadth-first manner. In this work, the approach proposed by Johnson and Mcgeoch (1993) is employed to compute the minimum weight matching in the bipartite graph, which consists of two basic steps: (i) finding a shortest path augmentation from a subset of vertices in Z to a subset of vertices in F , and (ii) performing the shortest augmentation.

4.5 Experimental Studies

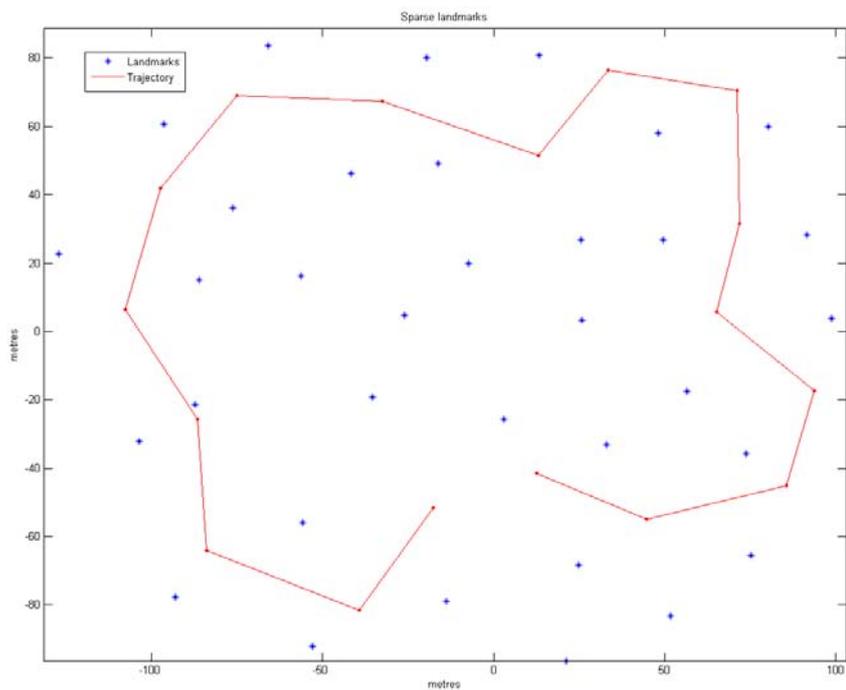
We evaluate our proposed data association approach through simulation and the robust regression model for EKF-SLAM via the experiments in static and dynamic environments. Notice that the latter experiments have incorporated the data association algorithm validated in former simulations.

4.5.1 Validation of Graph Theoretic Data Association Algorithm

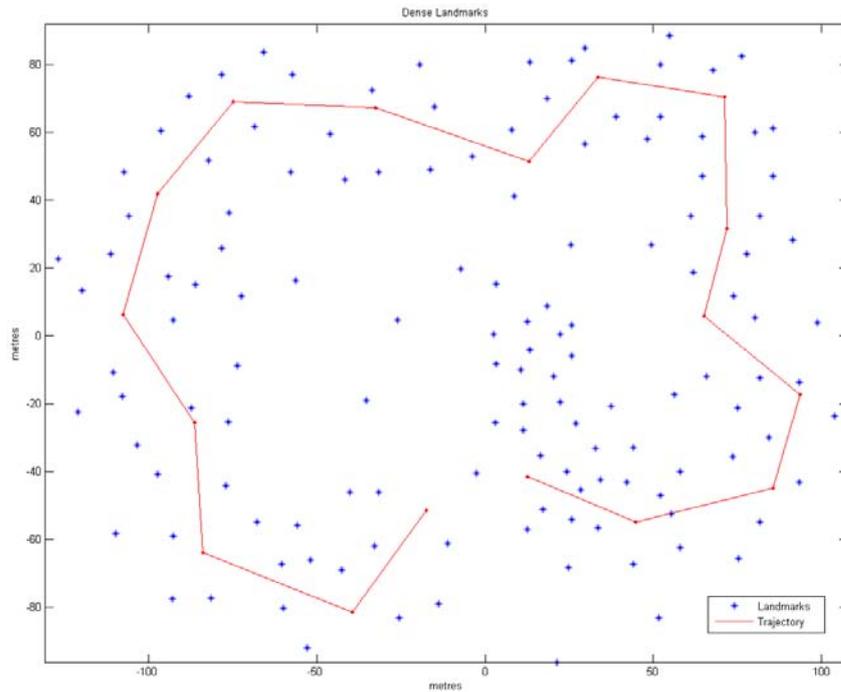
We implemented the simulation experiments based on the simulator written by Bailey¹. To demonstrate the capability of the proposed graph approach to improve the accuracy of the data association and thus the estimation, we particularly compared the performance with NN data association, which is one of most widely used methods in EKF-SLAM. Two different scenarios were considered: one is with sparse landmarks, the other has relatively denser landmarks (cf. Figure 4.6). The velocity of the robot is kept constant at $v_{tr}=2\text{m/sec}$, while its rotational velocity is obtained by calculating the changing rate of the orientation from the

¹ It is available online http://www-personal.acfr.usyd.edu.au/tbailey/software/slam_simulations.htm

current location to the next one in order to best fit the generated trajectory. The robot pose has zero initial uncertainty. The standard deviation of the velocity measurement noise is $\sigma_{v_t}=0.1\text{m/sec}$ and the standard deviation of the errors in the orientation estimates is $\sigma_{\omega}=0.0524\text{rad/sec}$. Similarly, the standard deviations of the exteroceptive measurement noise (i.e., range and bearing) are $\sigma_r =0.1\text{m}$ and $\sigma_b=0.0524\text{rad}$. The maximum sensing range of the sensor is set to 5m.



(a)

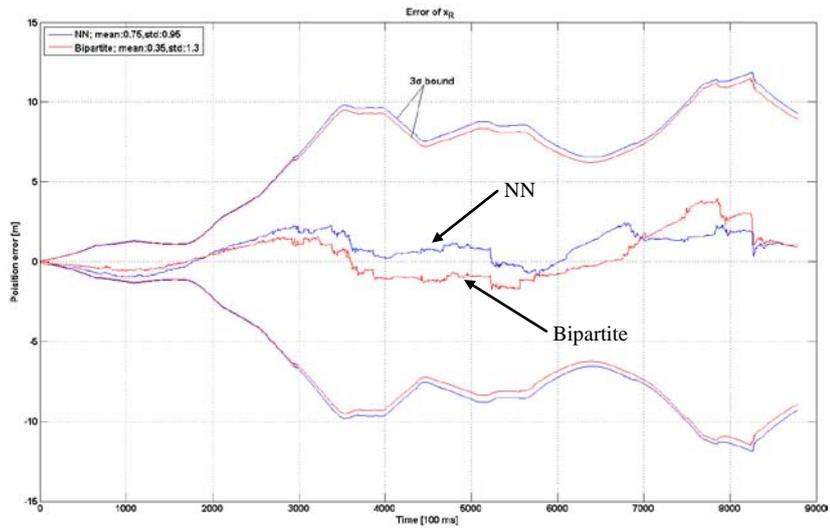


(b)

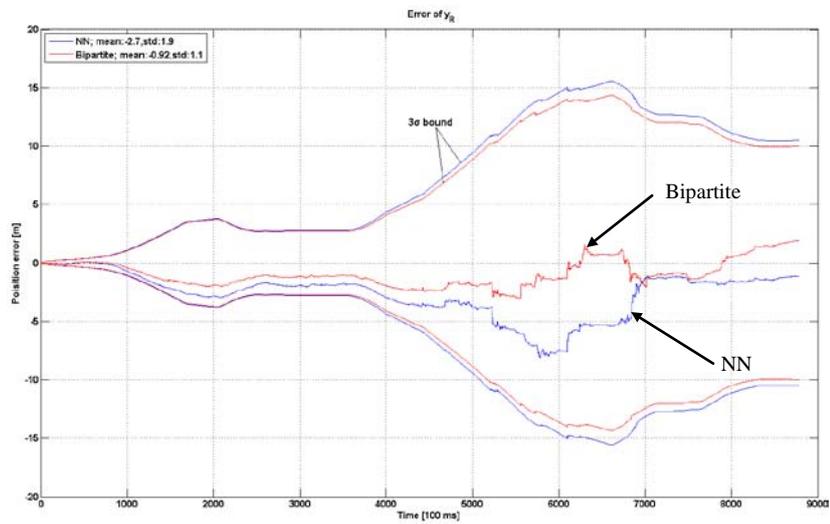
Figure 4.6 Simulation setup. A robot equipped with range bearing sensor moves on the planned trajectory at constant velocity of $v = 2$ m/sec.

(a) Sparse landmarks; (b) Dense landmarks.

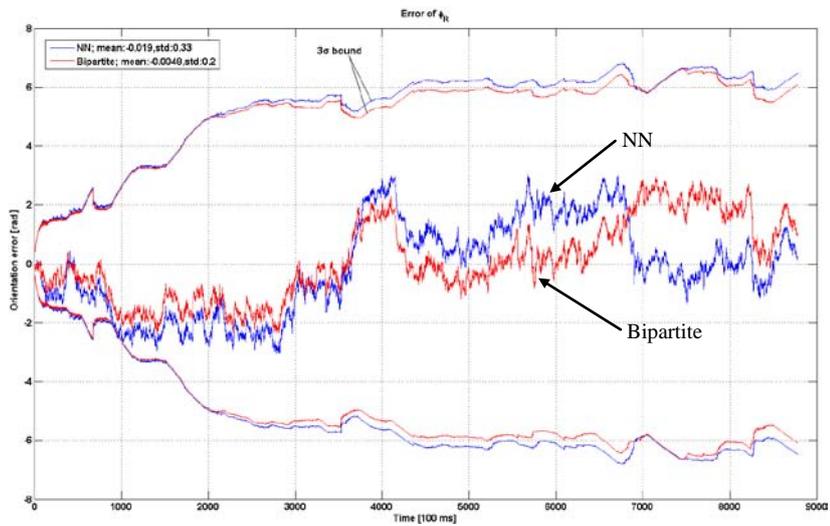
The resulting estimation errors of robot pose are shown in Figure 4.7 and 4.8, respectively. Figure 4.9 and 4.10 depict the estimation errors of the landmarks. As seen from these figures, the bipartite matching data association performs consistently, since the estimation errors are all well bounded within the 3σ regions, thus validating the effectiveness of the proposed algorithm. Moreover, in terms of the accuracy, the proposed graph approach attains better results, in that it has smaller covariance than NN, especially in the dense environment.



(a) Error of x_R



(b) Error of y_R



(c) Error of ϕ_R

Figure 4.7 Estimation errors of robot pose in the environment with sparse landmarks

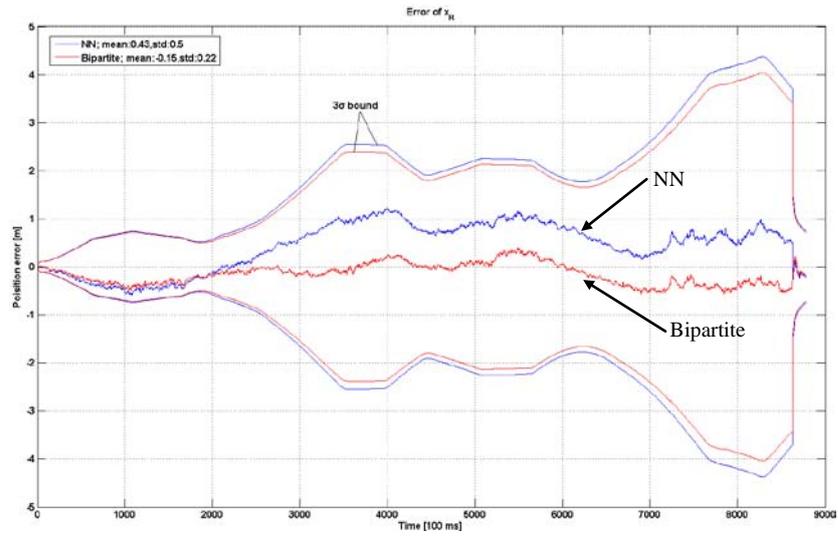
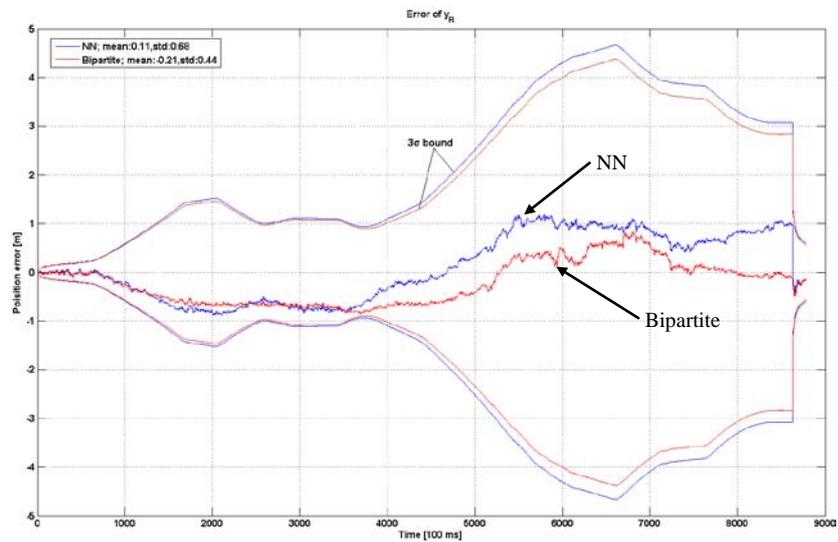
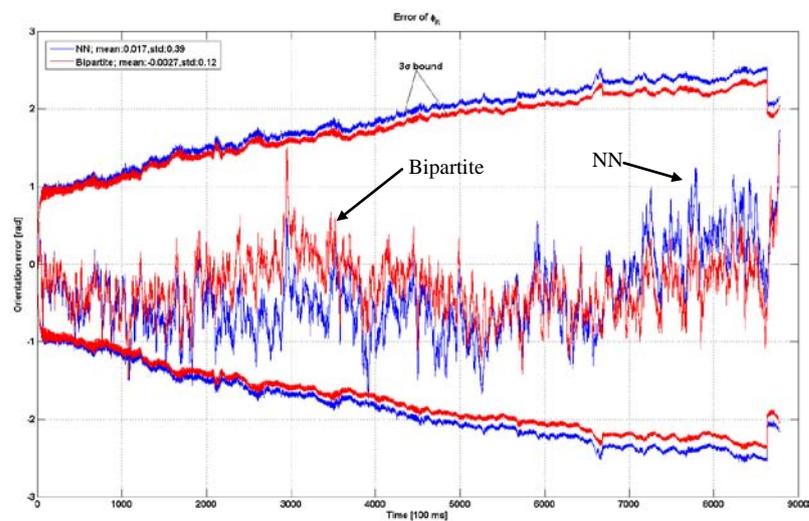
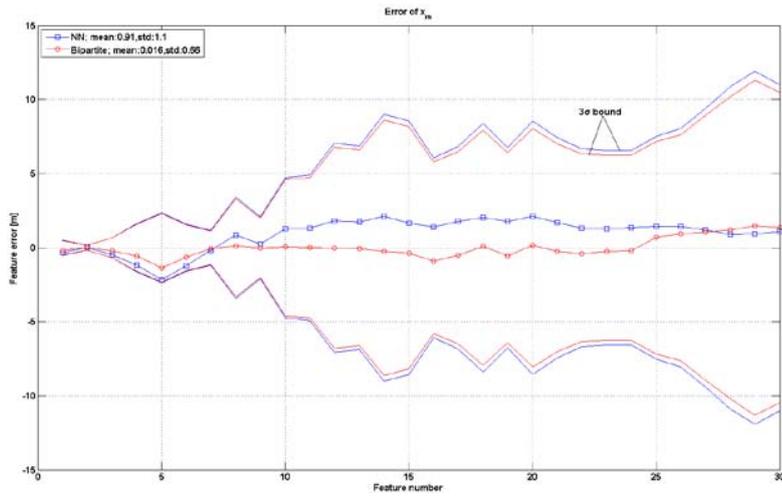
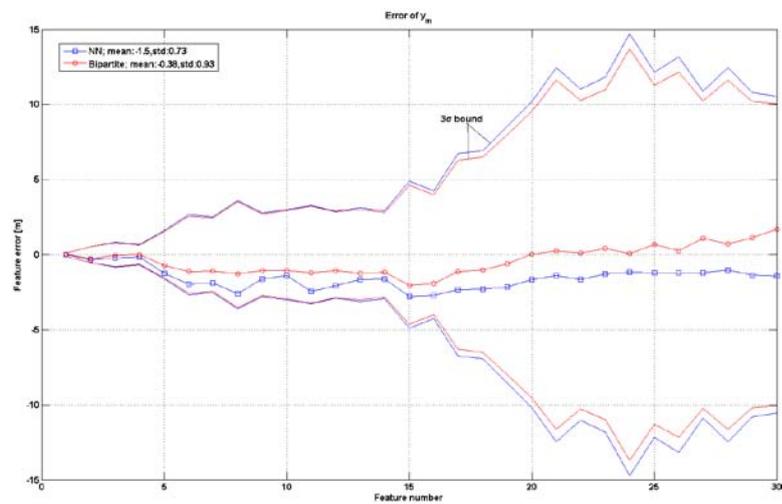
(a) Error of x_R (b) Error of y_R (c) Error of ϕ_R

Figure 4.8 Estimation errors of robot pose in the environment with dense landmarks

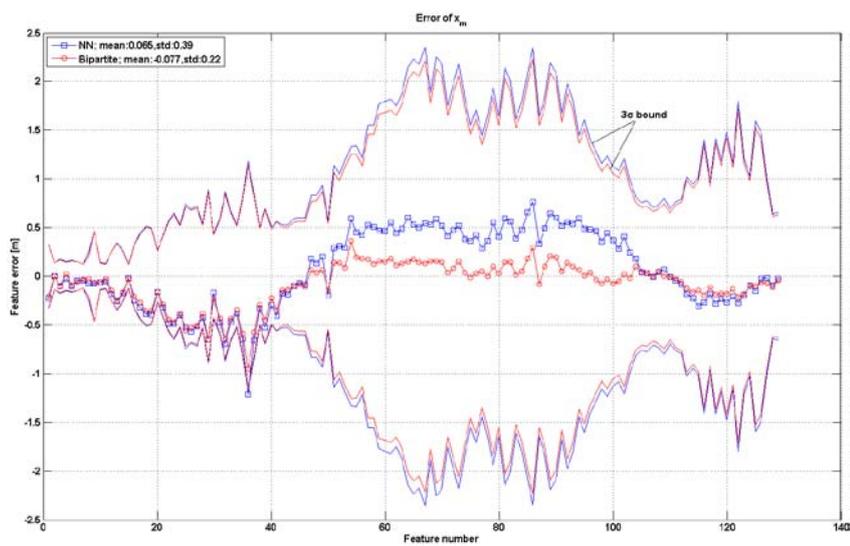


(a) Error of x_m



(b) Error of y_m

Figure 4.9 Estimation errors of landmarks in the environment with sparse landmarks



(a) Error of x_m

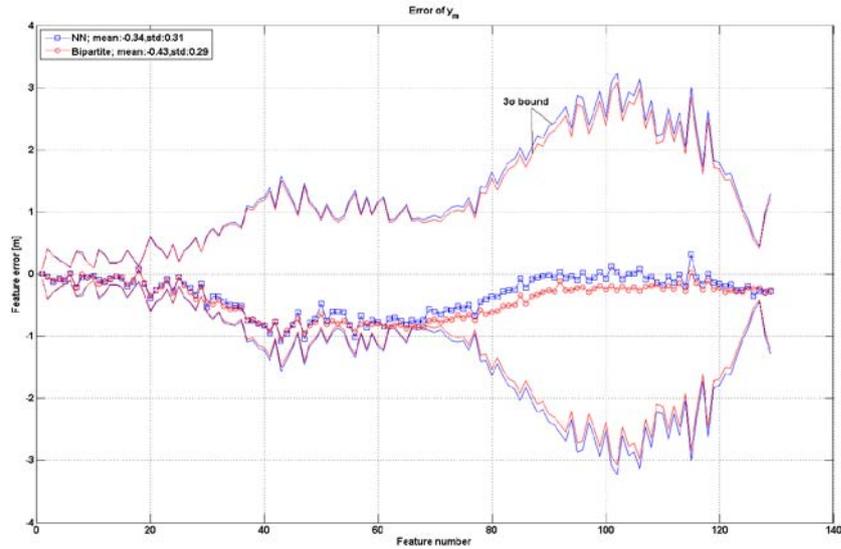
(b) Error of y_m

Figure 4.10 Estimation errors of landmarks in the environment with dense landmarks

4.5.2 Validation of Robust Regression Model for SLAM

To evaluate the performance of the robust regression model for SLAM, we conducted extensive experiments in the corridor outside the Control Research Laboratory. We tested our proposed algorithm in static as well as dynamic environments. Figure 4.11 shows a hand-measured map of the corridor. We collected the laser sensor data when the robot traveled every 300 mm with an average speed of 100 mm/s. After that, the algorithm was implemented on MATLAB with the help of the CAS Robot Navigation Toolbox¹.

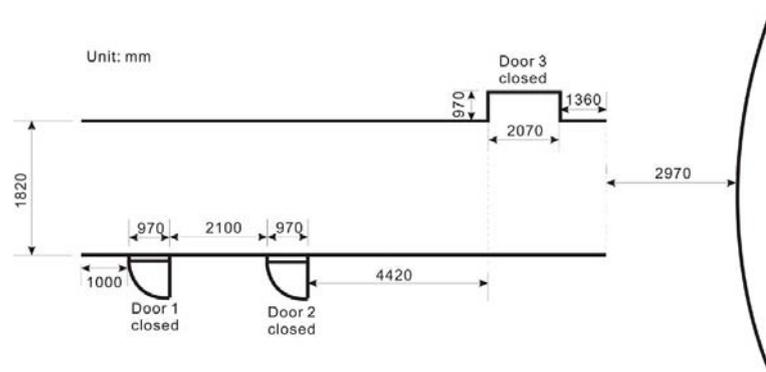


Figure 4.11 Hand-measured map of the corridor

¹ It is available online <http://www.cas.kth.se/toolbox/>

In static environment, there were no other moving objects except the mobile robot. The local maps including $1T_s$, $8T_s$, $14T_s$, $20T_s$ and $32T_s$ are shown in Figure 4.12. Circles indicate the laser raw readings, and T_s denotes sample time. Figure 4.13 gives the result of SLAM, which elucidates that robust regression model suits for static environments.

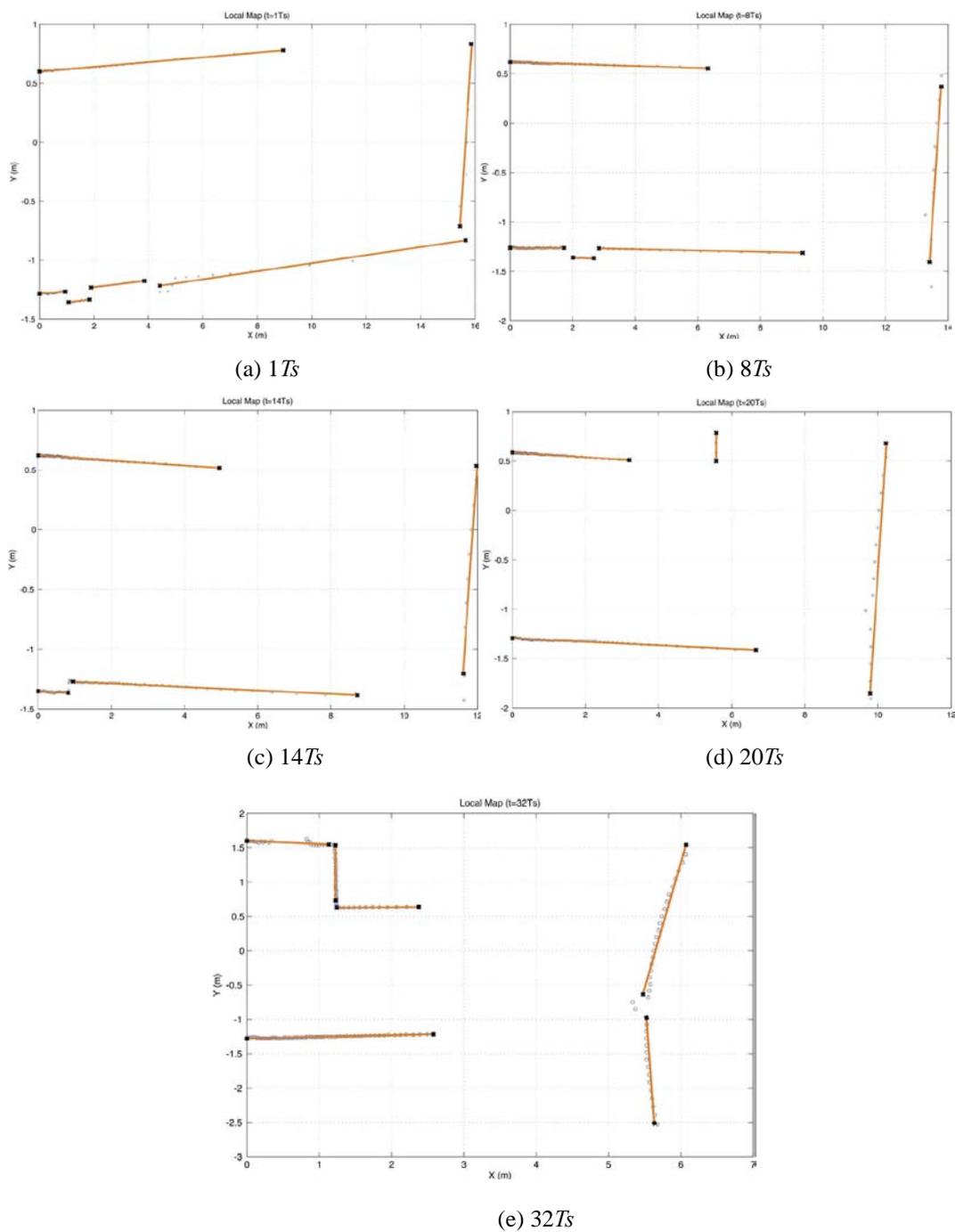


Figure 4.12 A sequence of local maps for static corridor

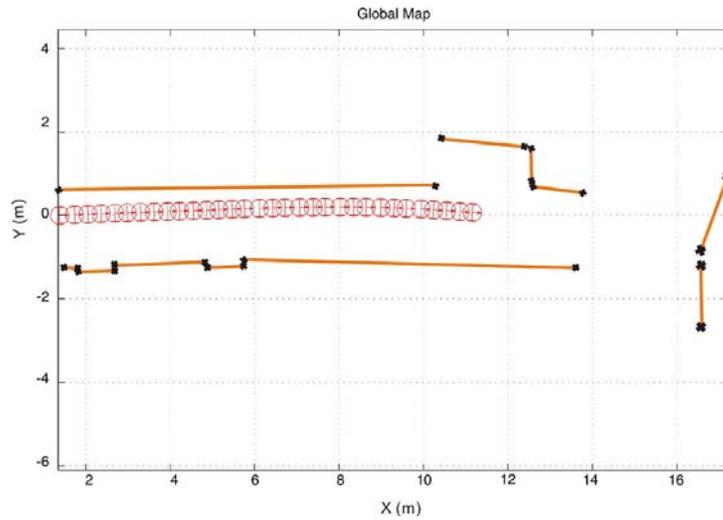
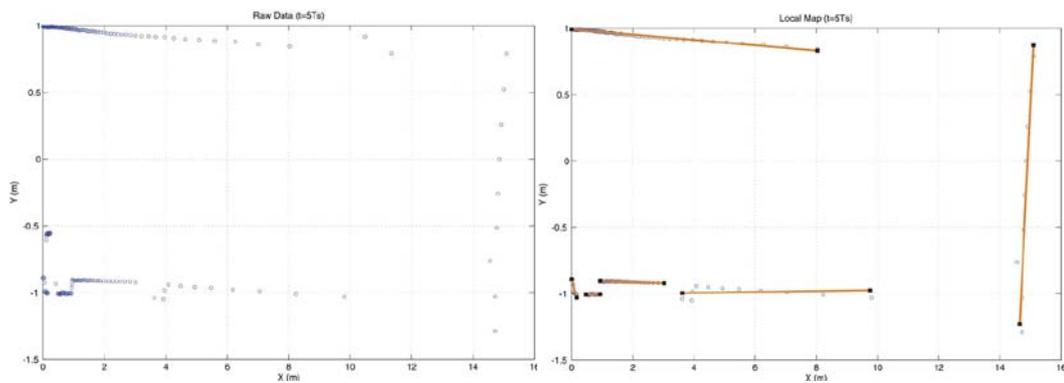
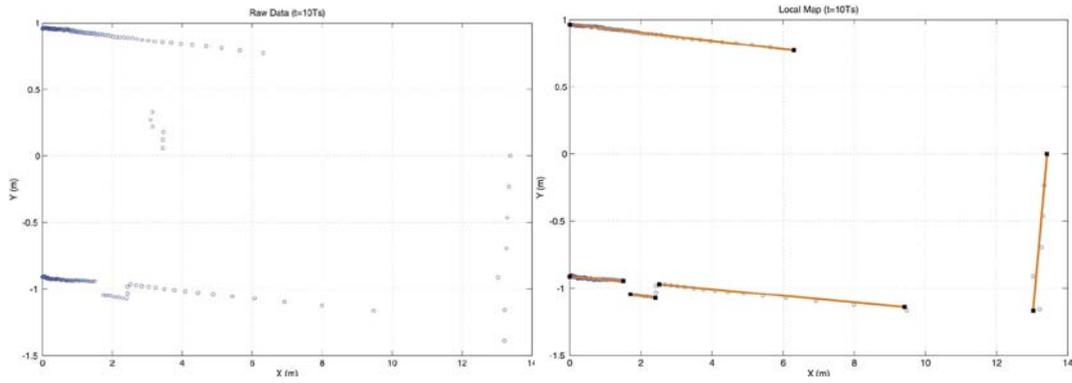


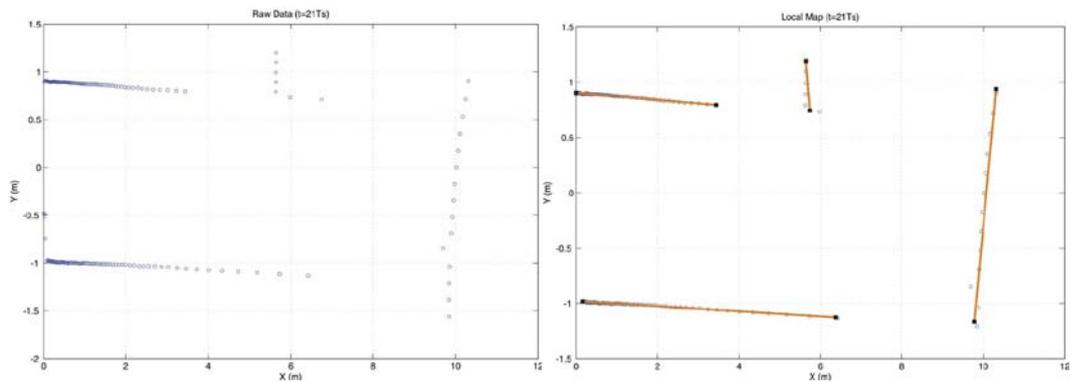
Figure 4.13 Final result of SLAM in static environment

As for dynamic environment, there was one person walking through the corridor with a normal speed. Figure 4.14 illustrates the mapping process for the local results of $5Ts$, $10Ts$, $21Ts$, $28Ts$ and $36Ts$. The subplots on the left side are the raw data and those on the right side are the extracted segment local maps. The raw data w.r.t. global framework and final results of SLAM are displayed in Figure 4.15. The laser data representing the walking person in Figure 4.14 (b), (d) and (e) are removed from the local maps and it is clear from Figure 4.15 that the results of SLAM in dynamic environment is similar to those obtained in static environment. This also states that the robust regression model can deal with the outliers corresponding to the dynamic objects as well as noise.

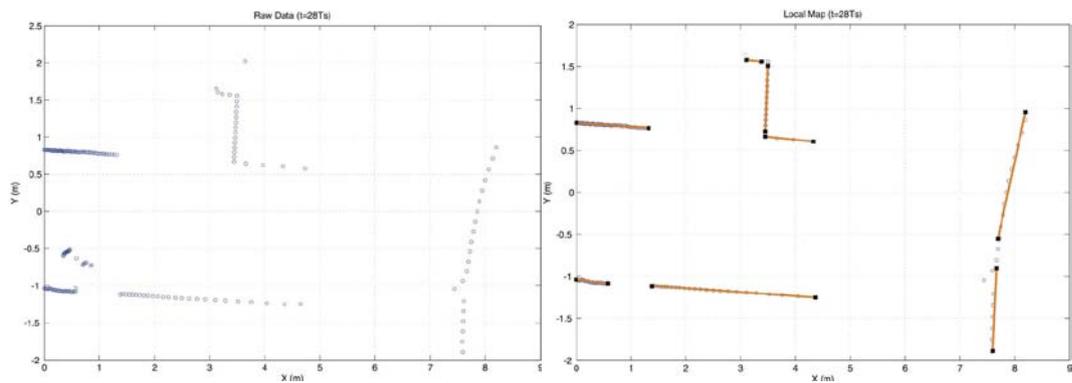
(a) $5Ts$



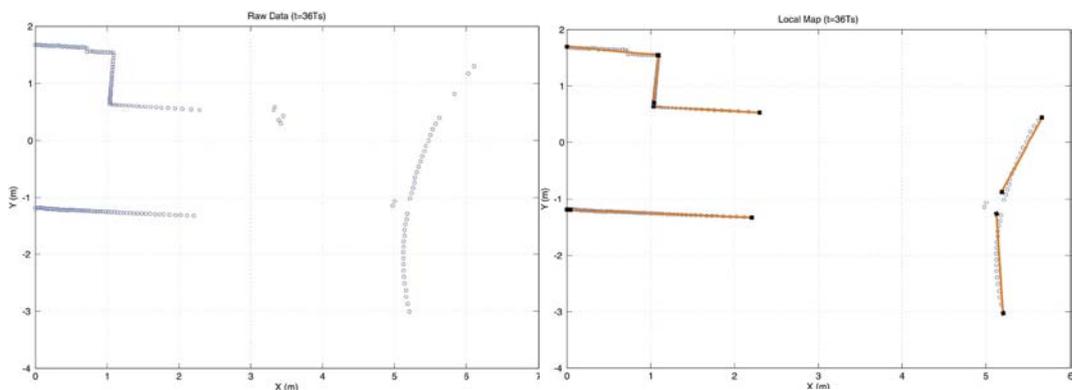
(b) $10T_s$



(c) $21T_s$



(d) $28T_s$



(e) $36T_s$

Figure 4.14 A sequence of local raw laser data and local maps for dynamic corridor

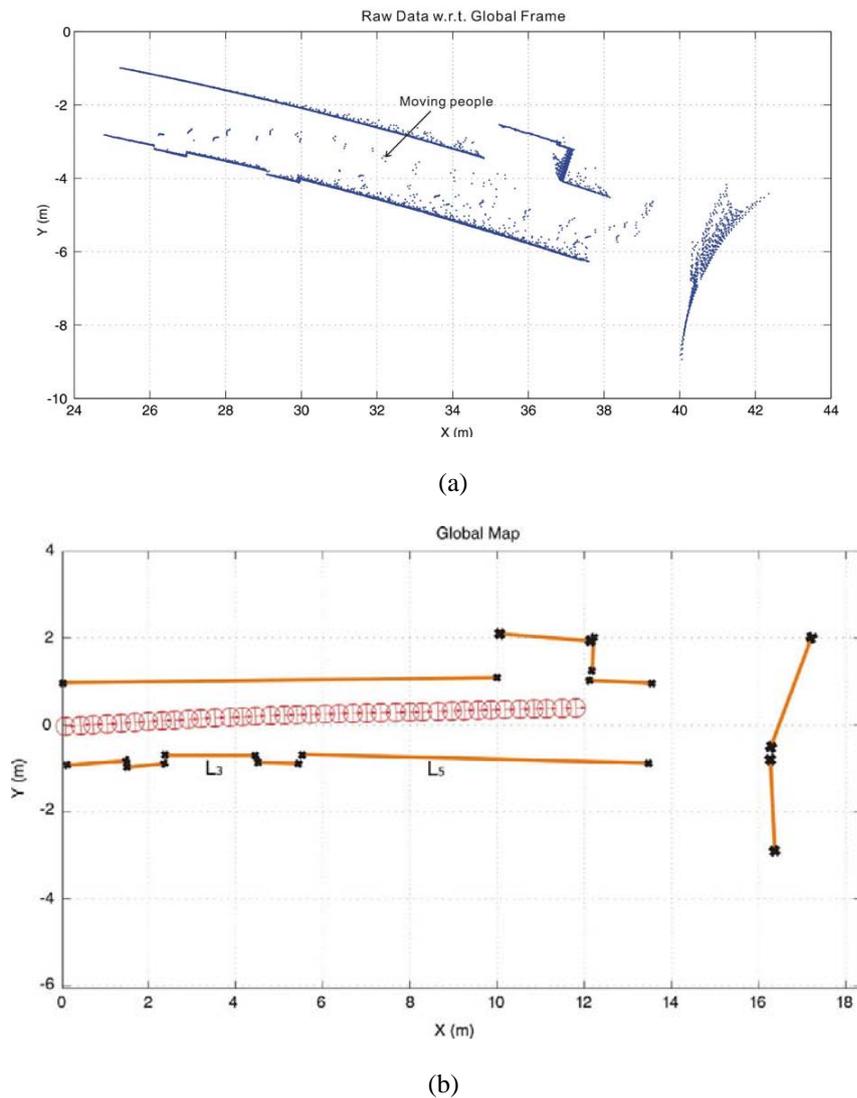


Figure 4.15 Global raw laser data and final result of SLAM in dynamic environment

Compared with RANSAC, the robust regression model has a similar accuracy (cf. Figure 4.16) but requires less time to acquire the estimate of the regression parameters. We tested the computational time for RANSAC and MM-estimate by the computer with Pentium P4 3.20GHz CPU and 1G RAM. The normal elapsed time was 0.848280 seconds for RANSAC and 0.098378 seconds for MM-estimate. The minimum elapsed time was 0.630992 and 0.022483 seconds respectively. It is evident that even the minimum computational time for RANSAC is much larger than MM-estimate for parameter estimation. This is because that the subset of the original data for hypothesis test in RANSAC is selected randomly.

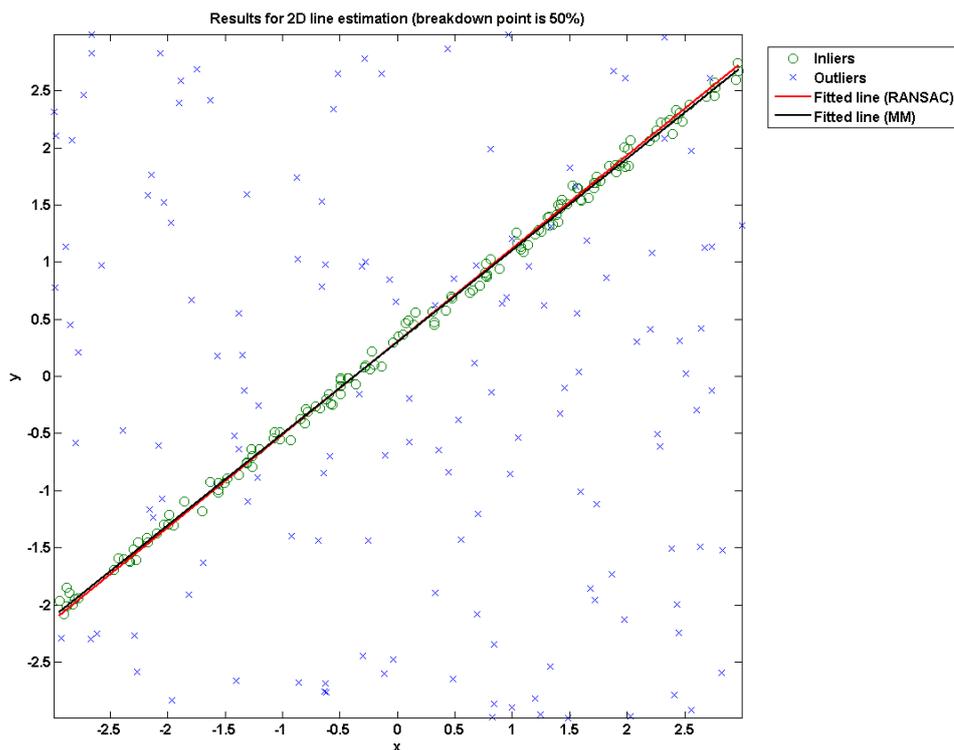


Figure 4.16 Fitted lines estimated by RANSAC and MM-estimator

where breakdown point is 50%

4.6 Discussion

In this chapter, we proposed a robust regression model for mapping that discards the dynamic objects, and incorporated this model with the EKF-SLAM so that the moving objects can be avoided in the extracted map features. Experimental results have demonstrated the performance of outlier detection for not only in the presence of sensor data noise but also the dynamic environment. In our previous work, we designed a fuzzy strategy to represent the sensor noise as a separate cluster, noise cluster, in static environment. However this cluster did not consider the moving objects and related fuzzy membership functions did not suit for the dynamic environments. The robust regression model in this chapter addresses the sensor noise and dynamic items concurrently without requiring any special functions for noise and dynamic properties. Compared with the Split-and-Merge or incremental algorithm, the number of thresholds applied in those two methods has

been reduced to only two (i.e. Th_{stop} and N_p), so that the difficulty of thresholds selection is significantly alleviated. As well, with this robust regression model it takes less time to extract the feature parameters than RANSAC and can obtain the optimal estimate. Furthermore, when there are several similar segment models, for example L_3 and L_5 in Figure 4.15 (b), fitting the laser readings, it is easy to separate them by our proposed method but it may not be possible by using RANSAC. Despite their superior performance over least squares estimation in many situations, robust methods for regression are still not widely used. One possible reason is that the robust estimate is much more computationally intensive than the least squares estimation. In recent years however, this objection has become less relevant as computing power has drastically increased.

Unfortunately, there is an inherent restraint that is the maximum breakdown point is 50%. If the proportion of incorrect observations preponderates over 50%, as is shown in Figure 4.17, the robust regression model fails to represent the correct segment features. The fitted line described by MM-estimate is similar to the result of OLS. The reason is there is not useful information enough to support the robust regression for feature extraction. We hardly encounter the case that the proportion of outliers exceeds 50% in this thesis study. From this viewpoint, we may claim that the robust regression model can apply to most of dynamic environments. Additionally, only with robust regression models, momentary stopping of moving people or other robots in the vicinity of the autonomous robot probably are treated as the segments, for example in Figure 4.18 there was a person standing in front of the robot, and the segment 4 was the extracted segment that related to the stopped person. It is apparent that this segment is a pseudo feature and will weaken the performance of the mapping even for SLAM. To counter this problem, we suggested a sensor fusion method in the next chapter to combine the lines extracted from the monocular vision system with the segments from laser rangefinder to eliminate those pseudo features.

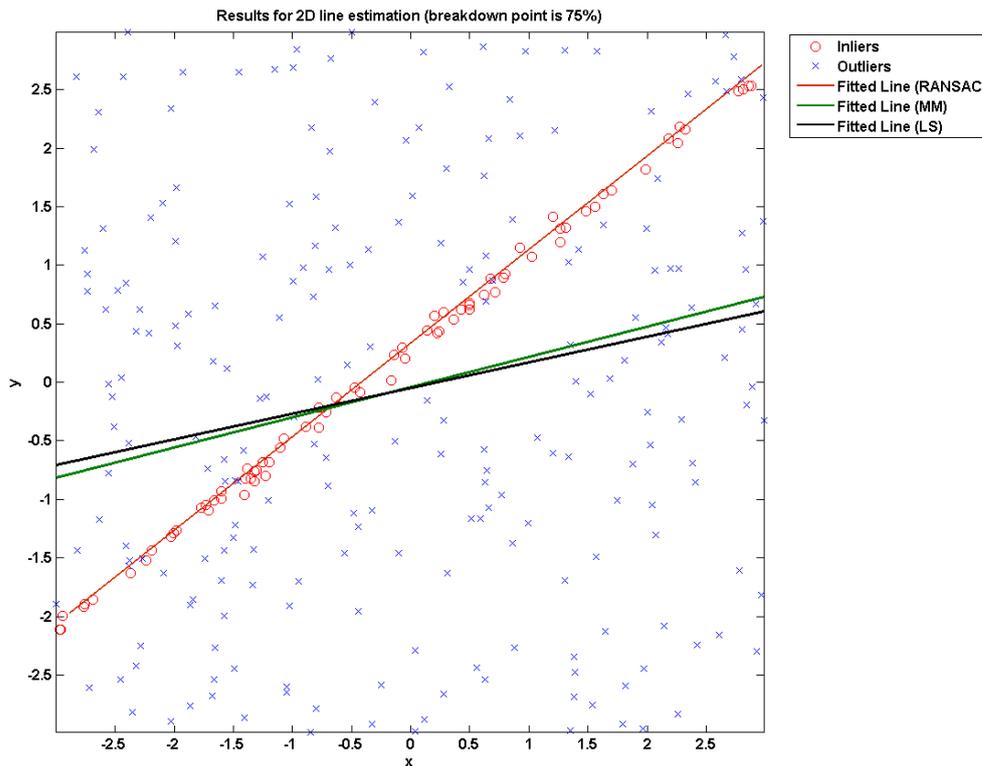


Figure 4. 17 Fitted lines estimated by RANSAC, MM-estimator and OLS where breakdown point is 75%

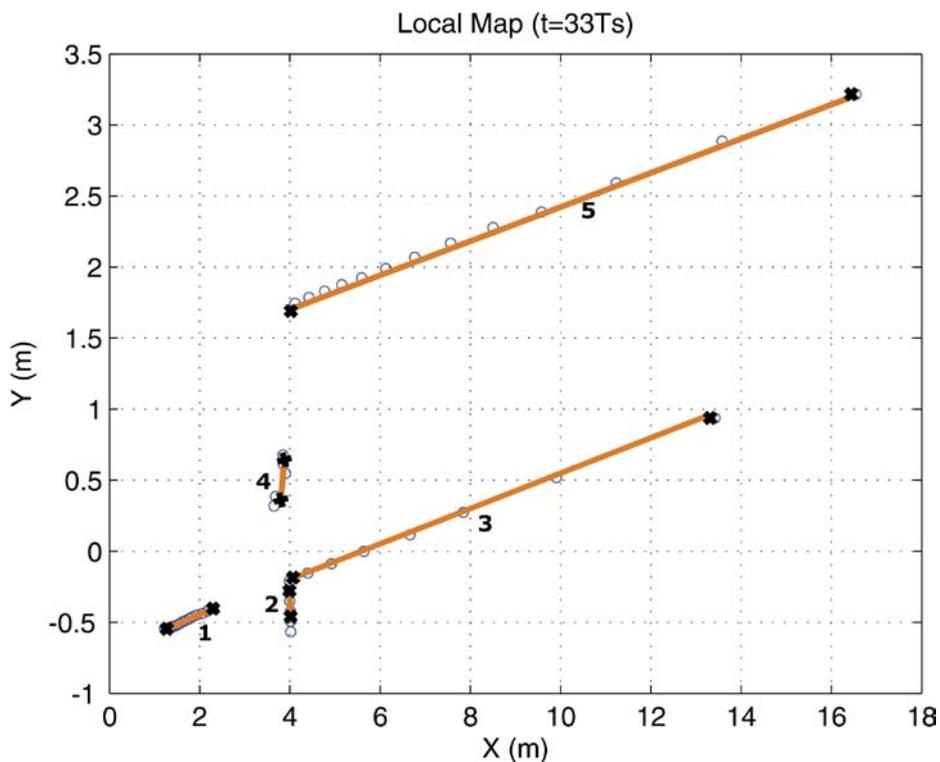


Figure 4.18 Pseudo segment extraction case

Concerning the data association problem, we formulated it as an equivalent minimum weight bipartite perfect matching problem which can be optimally solved, thus obtaining the optimal solution to the data association problem. We mathematically prove the optimality of the graph theoretic approach. Compared with the prevalent NN method, the proposed method has smaller errors, and reduces covariance for robot pose and feature positions. Obviously, it is limited to analyze the performance between the NN method and graph approach and more thorough comparison studies with the existing methods in the literature are to be studied. In this chapter we expected providing an optimal data association approach, therefore we tentatively discarded the performance of the algorithm for weighted bipartite matching and we applied the classical implementation. The computational complexity for bipartite approach is $O(V^2 \log V + VE)$ where V is the number of the vertex and E is the number of the edges. It is much more complex than that of NN algorithm of which the computational complex is $O(mn)$.

Chapter 5 Sensor Fusion for Pseudo Features Elimination

5.1 Introduction

The robust regression model eradicates most features corresponding to dynamic objects provided that these objects keep the movement status. Unfortunately, this assumption can not be satisfied in practice. A general scenario in many indoor dynamic environments includes moving objects such as people walking around or standing at any place. When moving people momentarily stop for a while, the robust regression model can not remove these dynamic elements and may misidentify them as the static segments. Obviously, these are pseudo features and can not be introduced into the SLAM procedure. In this chapter, we dispose the temporary stationary objects with the assistance of a monocular camera.

Monocular camera as a low cost sensor has been widely used in numerous robotic applications in recent decades. It provides the autonomous mobile robot with abundant information that facilitates intuitive interpretation and comprehension of the environment better than other scanning sensors. Advances in computer vision have provided researchers with efficient and powerful techniques that can be employed in variety of autonomous tasks. Davison and his group proposed a real time monocular SLAM algorithm that generated a 3D trajectory of a previously unknown scene (Civera, et al., 2007a; Civera, et al., 2007b; Andrew J. Davison, et al., 2007; Andrew. J. Davison, 2003; Montiel, et al., 2006). The core of their approach was the online creation of a sparse but persistent map of natural landmarks within a probabilistic framework. The 3D position of the landmarks was described by a unified inverse depth parameterization algorithm that allowed efficient and accurate representation of uncertainty during un-delayed feature

initialization procedure. Their research made a valuable contribution and opened up a new direction in robotics research. Following this pioneering work on monocular SLAM, other researchers studied line-based algorithms. Eade and Drummond (2006) proposed an edge-let landmark to depict the line features in images. This work, which is the extension of the so-called scalable monocular SLAM (E. Eade & T. Drummond, 2006), avoids regions of conflict and deals with multiple matches through robust estimation. Gee and Mayol-Cuevas (2006) used fast conic extraction to obtain the 2D edges and then estimated the 3D segments with the Unscented Kalman filter (UKF). Also Smith Reid and Davison (2006) applied FAST corners to quickly verify that there was an edge between two corners by bisecting checks.

Much of relevant research above, however, implemented SLAM in static space or within environments with few moving objects. The dynamic objects induce spurious features and make it difficult to obtain the correct estimates of the state variables. Furthermore these dynamic objects may lead to inappropriate robot actions that ultimately results in failure to complete the expected tasks.

To discard the dynamic objects correctly, we present a sensor fusion strategy for line-based SLAM applied in dynamic environments. The algorithm fuses the sensor information of a monocular camera and a laser rangefinder. It includes two modules: One is a feature fusion that integrates the lines extracted respectively from a single camera and a laser to remove the erroneous features corresponding to dynamic objects; the other is referred to as a modified multi-sensor point estimation fusion (MPEF) which incorporates two separate EKF-SLAM frameworks: (a) monocular SLAM and (b) laser SLAM represented in Chapter 4. By this modified MPEF, the robot pose covariance is reduced compared with the corresponding value in each individual SLAM.

Computer vision technology also makes feasible to address the data association

problem based on visual information. One of methods is on the basis of Scale-invariant feature transform (SIFT) algorithm (Lowe, 2004). Landmarks are identified by SIFT and represented by keypoint descriptors (Miro, et al., 2005; Sim, et al., 2005). These landmarks subsequently are treated as the ideal candidates to the robust data association. Gil et al. (2006; 2007) managed the data association with the SIFT features from the pattern classification viewpoint, and the Mahalanobis distance was established by the average SIFT descriptors and a high dimensional covariance matrix. Similarly with pattern recognition technology, object-based SLAM (Ahn, et al., 2006) combined advantages of multi-scale Harris corner as a detector and the SIFT descriptor for natural object recognition, which provides a correct data association. Different from these works, without using the SIFT descriptor directly, the suggested data association technology in this chapter employs the homography transformation matrix (Hartley & Zisserman, 2003) estimated by the matched points located in two images. These matched points are determined by SIFT descriptors

The structure of this chapter is as follow. A description of the modeling and measurements equations of line based EKF monocular SLAM is introduced in Section 5.2. In Section 5.3, the homography transformation based data association algorithm is stated. The overall structure of sensor fusion scheme is presented in Section 5.4. A detailed description of experimental studies is included in Section 5.5. Finally, the discussion is expressed in Section 5.6.

5.2 Line based EKF Monocular SLAM

As the laser SLAM procedure is same as the presentation in Chapter 4, it suffices here to discuss the outline of the line-based monocular SLAM framework. We simply convert the camera model proposed in Davison's and his extended work to 2D formation, and embed the parameters of line features extracted from images into state vectors for the SLAM.

5.2.1 Motion Model of the Monocular Camera/Robot

The Canon VCC4 camera is fixed on the robot platform which moves in a 2D plane, and the translational and rotational velocity can be directly achieved from the odometry sensors. For convenience, we assume the origins of the robot and the camera reference systems are identical. Therefore we obtain a simplified camera/robot motion model

$$\mathbf{x}_{v_k} = \begin{pmatrix} \mathbf{r}_k^{WR} \\ \mathbf{q}_k^{WR} \end{pmatrix} = f_v(\mathbf{x}_{v_{(k-1)}}, \mathbf{u}_k) + v_k = \begin{pmatrix} \mathbf{r}_{k-1}^{WR} + \mathbf{v}_{tr_k}^W \Delta t \\ \mathbf{q}_{k-1}^{WR} \times \mathbf{q}(\boldsymbol{\omega}_k^W \Delta t) \end{pmatrix} + v_k \quad (5.1)$$

where

$$\mathbf{r}_k^{WR} = (x_{R_k}, a_h, z_{R_k})^T, \mathbf{v}_{tr_k}^W = (v_{x_k}, 0, v_{z_k})^T, \boldsymbol{\omega}_k^W = (0, \omega_{y_k}, 0)^T, \mathbf{u}_k = (\mathbf{v}_{tr_k}^W, \boldsymbol{\omega}_k^W)^T$$

$$\mathbf{q}_k^{WR} = \begin{pmatrix} q_0 \\ q_x \\ q_y \\ q_z \end{pmatrix} = \begin{pmatrix} \cos \frac{\|\boldsymbol{\omega}_k^W \Delta t\|}{2} \\ \sin \frac{\|\boldsymbol{\omega}_k^W \Delta t\|}{2} \cdot \frac{\omega_k^W \Delta t}{\|\boldsymbol{\omega}_k^W \Delta t\|} \end{pmatrix} = \begin{pmatrix} \cos \frac{|\omega_{y_k}|}{2} \\ 0 \\ \sin \frac{|\omega_{y_k}|}{2} \\ 0 \end{pmatrix}$$

a_h is the constant height of the camera from the ground plane; v_{xk} and v_{zk} are the sub-translation velocity along x - and z -axis at time k ; ω_{ky} is exact rotational velocity at time k and \mathbf{q}^{WR} is the orientation quaternion similar to the definition of Davison's work (Civera, et al., 2007a; Civera, et al., 2007b; Andrew J. Davison, et al., 2007; Andrew. J. Davison, 2003; Montiel, et al., 2006). $v(k)$ is the process noise with zero mean and covariance \mathbf{Q} . In Figure 5.1(a), the global and the camera/local references are shown in red and black respectively. It is important to note that the local reference has a little difference from the traditional one displayed in the lower plot of Figure 5.1(b). The camera pose is represented in z - x , which is plotted in upper subfigure of Figure 5.1(b), not x - y coordinates. This stems from the original application domain in computer vision, where the z axis of coordinate frames are aligned with the optic axis of cameras and for a robot with a forward-facing camera, this makes z horizontal.

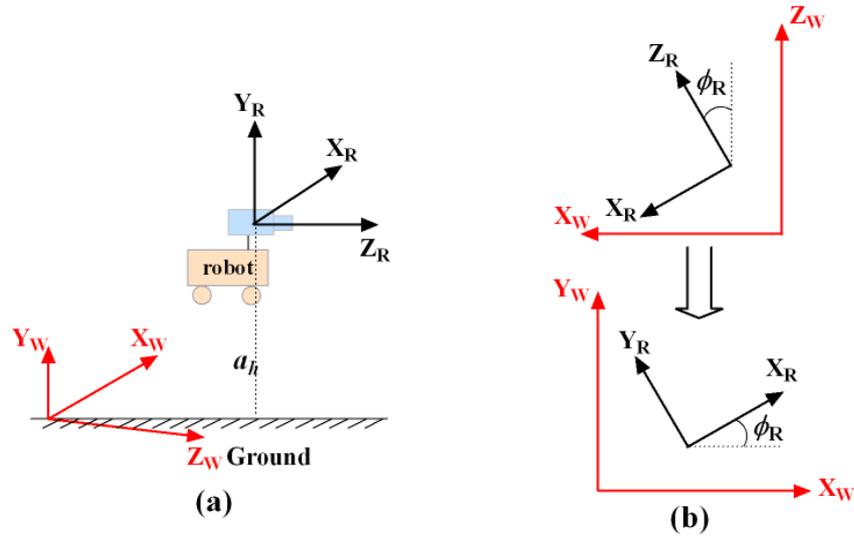


Figure 5.1 The global and camera reference systems

Since the laser based SLAM uses the traditional coordinates to predict and update the robot pose, to fuse both state variables rationally, it is necessary to convert the pose coordinates from our definition to the traditional representation. The upper plot of Figure 5.1(b) is the top view of Figure 5.1(a). Triple ${}^{WV}[z_R, x_R, \phi_R]^T$ specifies the Cartesian coordinates of our reference system. The relationship between our and conventional reference system is

$${}^{WC}[x_R, y_R, \phi_R]^T = {}^{WV}[z_R, x_R, \phi_R]^T \quad (5.2)$$

The superscripts WC and WV mean the traditional and our vision system frames represented in global coordinate system respectively. It should be noted that the angle ϕ_R is computed from the angular velocity ω^W and is restricted within the range $-\pi < \phi_R \leq \pi$

$$\phi_{R_k} = \phi_{R_{(k-1)}} + \arccos \left\| \frac{d\phi_k^W}{dt} \Delta t \right\| \quad \phi_{R_{(k-1)}} + \arccos(\omega_{y_k}^W \Delta t) \quad (5.3)$$

5.2.2 Line Extraction and Measurement Model

Line extraction is actually an edge detection operation in the image processing terminology. Most of the edge features in the literature are extracted by using Canny operator (Ethan Eade & Tom Drummond, 2006; Gee & Mayol-Cuevas,

2006). In this current study, we employed another first-order edge detector: Sobel operator combined with thresholding technique for edges extraction in a region of interest (ROI). The ROI we selected is a rectangle window defined as $40 < v \leq 240$ and $0 < u \leq 320$ (cf. Figure 5.2(a)). This is because

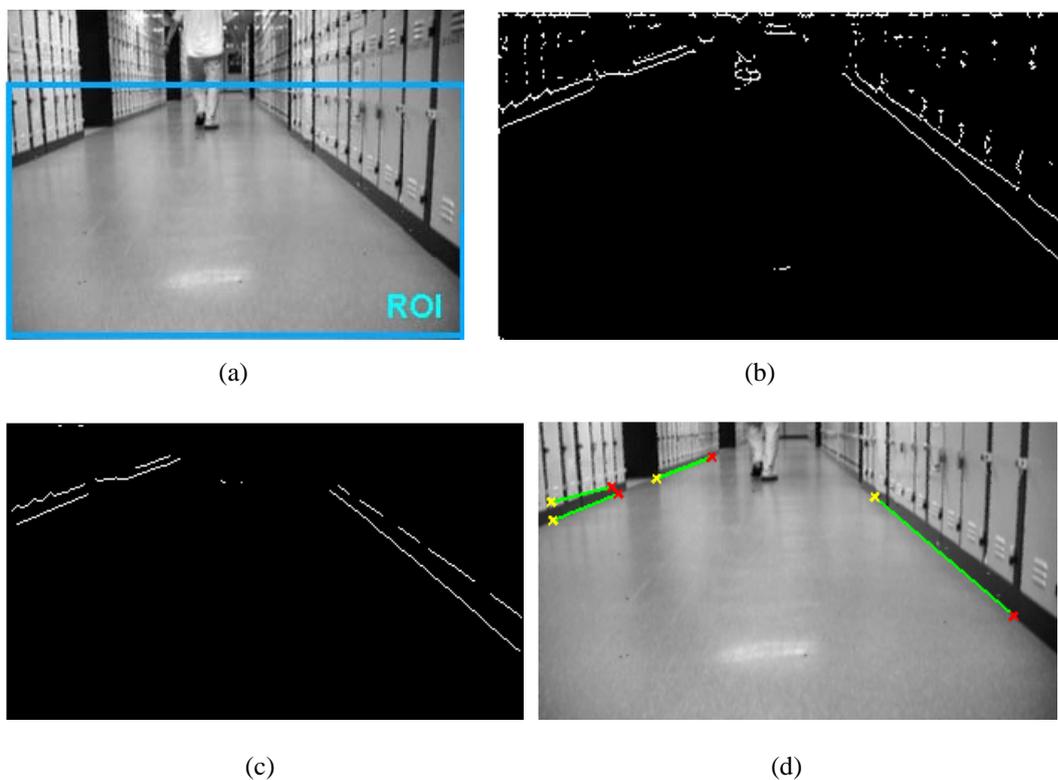


Figure 5.2 Line feature extraction from image. (a) The region of interest (ROI); (b) Detected edges without morphological operation; (c) Detected edges after morphological operation; (d) Selected line features whose length is greater than the threshold.

we just consider the horizontal static edges in this ROI, nor focus on tracking the dynamic targets. However, not all of the moving objects can be eliminated from the selected region, which is illustrated in Figure 5.2(b). It seems that some extracted edges probably correspond to the dynamic objects (i.e. the person here). To withdraw these potential outliers, we firstly carried out the *shrink* and *clean* morphological operations on all edges. The shorter and thinner edges, which usually relate to the parts of dynamic objects, are taken out; and for these processed edges, if the length of anyone is less than 35 pixels we also rejected it. This operation ensures to wipe away spurious edges that were not removed in

shrink and *clean* process. Finally we did the *thicken* operation to recover the interested edges as displayed in Figure 5.2(c)&(d), which will prepare for edge parameter extraction in the next step.

We divided the parameters for edge representation into two parts. One was used in measurement model; the other was for data association and sensor fusion. In this subsection, we mainly discuss the parameters for measurement model. Similar to Gee and Mayol-Cuevas (2006) and Smith et al. (2006), we employed image coordinates of the lines ends to build a couple representation, i.e. $[p_s, p_e]^T$. We borrowed the idea of Davison's work and the measurement model is

$$\mathbf{z}_{ik} = h_i(\mathbf{x}_{vk}, \mathbf{y}_i) \quad (5.4)$$

$$\mathbf{h}_L^R(\mathbf{x}_{vk}, \mathbf{y}_i) = R_k^{RW}(\mathbf{y}_i^W - \mathbf{r}_k^W) \quad (5.5)$$

$$h_i(\mathbf{x}_{vk}, \mathbf{y}_i) = (u, v)^T = \left(u_o - fk_u \frac{h_{L_x}^R}{h_{L_z}^R}, v_o - fk_v \frac{h_{L_y}^R}{h_{L_z}^R} \right)^T \quad (5.6)$$

$$R_k^{RW} = \begin{bmatrix} q_0^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y - q_0 q_z) & 2(q_x q_z + q_0 q_y) \\ 2(q_x q_y + q_0 q_z) & q_0^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_0 q_x) \\ 2(q_x q_z - q_0 q_y) & 2(q_y q_z + q_0 q_x) & q_0^2 - q_x^2 - q_y^2 + q_z^2 \end{bmatrix} \quad (5.7)$$

where $\mathbf{y}_i = [x_i, y_i, z_i, \theta_i, \phi_i, \rho_i]^T$ is the 3D position of the line ends, which is depicted in inverse depth parameters (Montiel, et al., 2006) (\mathbf{y}_i is the element of state variable); R_k^{RW} is the rotation matrix associated with quaternion \mathbf{q}_k^{WR} ; Equation (5.5) is the position of the ends relative to the camera and equation (5.6) is the position $[u, v]$ at which the ends would be expected to be found in the image; fk_u , fk_v , u_o and v_o are the standard camera calibration parameters. Note that the ends initialization is same as Montiel et al. (2006).

Besides the ends for measurement model, we also considered several additional parameters including mid-point p_{mp} of the lines and the line descriptor in Hough space (Wong, Shi, & Chan, 1997) $[d_H, \alpha_H]^T$ (usually it is written as $[\rho, \theta]^T$ and here we use a different notation just to distinguish it with the components in \mathbf{y}_i). They

are applied as the auxiliary parameters for our proposed data association and sensor fusion strategies, and we will concentrate on these topics in the following sections.

A step-by-step procedure for the complete line extraction algorithm follows:

Step1: Pre-process the acquired image to filter out different noise signals;

Step2: Select the region of interest (ROI);

Step3: In the ROI, detect the horizontal edges by the Sobel operator combined with thresholding;

Step4: *shrink* and *clean* morphological operations on all edges to eliminate the ones corresponding to dynamic objects;

Step5: Remove the edges whose length is less than 35 pixels;

Step6: *thicken* operation to recover the interested edges;

Step7: Extract the image coordinates of line ends and mid-point, and descriptors in Hough space.

5.2.3 EKF based Monocular SLAM

The extended Kalman filter algorithm is also considered as the dominant approach for monocular SLAM. The time update and measurement update procedures are similar to Chapter 4. Hence, we only explain the essential differences in this section.

Current estimates of the state of the robot and the scene features which are known about are stored in the system state vector \hat{x} and the uncertainty of the estimates in the covariance matrix P . \hat{x} and P will change in size dynamically as features are added to or deleted from the map. They are partitioned as follows:

$$\hat{\mathbf{x}} = \begin{pmatrix} \hat{\mathbf{x}}_v \\ \hat{\mathbf{y}}_1 \\ \mathbf{y}_2 \\ \vdots \\ \hat{\mathbf{y}}_i \end{pmatrix}, \mathbf{P} = \begin{pmatrix} \mathbf{P}_{x_v x_v} & \mathbf{P}_{x_v y_1} & \mathbf{P}_{x_v y_2} & \cdots & \mathbf{P}_{x_v y_i} \\ \mathbf{P}_{y_1 x_v} & \mathbf{P}_{y_1 y_1} & \mathbf{P}_{y_1 y_2} & \cdots & \mathbf{P}_{y_1 y_i} \\ \mathbf{P}_{y_2 x_v} & \mathbf{P}_{y_2 y_1} & \mathbf{P}_{y_2 y_2} & \cdots & \mathbf{P}_{y_2 y_i} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{y_i x_v} & \mathbf{P}_{y_i y_1} & \mathbf{P}_{y_i y_2} & \cdots & \mathbf{P}_{y_i y_i} \end{pmatrix} \quad (5.8)$$

The predicted covariance of the state vector is

$$\mathbf{P}_k^- = \begin{bmatrix} \frac{\partial f_v}{\partial \mathbf{x}_v} \mathbf{P}_{xv xv(k-1)} \left(\frac{\partial f_v}{\partial \mathbf{x}_v} \right)^T + \mathbf{Q}_{k-1} & \frac{\partial f_v}{\partial \mathbf{x}_v} \mathbf{P}_{xv y_1(k-1)} & \frac{\partial f_v}{\partial \mathbf{x}_v} \mathbf{P}_{xv y_2(k-1)} & \cdots \\ \mathbf{P}_{y_1 xv(k-1)} \left(\frac{\partial f_v}{\partial \mathbf{x}_v} \right)^T & \mathbf{P}_{y_1 y_1(k-1)} & \mathbf{P}_{y_1 y_2(k-1)} & \cdots \\ \mathbf{P}_{y_2 xv(k-1)} \left(\frac{\partial f_v}{\partial \mathbf{x}_v} \right)^T & \mathbf{P}_{y_2 y_1(k-1)} & \mathbf{P}_{y_2 y_2(k-1)} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (5.9)$$

where $\mathbf{Q}_k = \left(\frac{\partial f_v}{\partial \mathbf{u}_k} \right) \mathbf{P}_n \left(\frac{\partial f_v}{\partial \mathbf{u}_k} \right)^T$, \mathbf{P}_n is the covariance of control vector \mathbf{u}_k . $\partial f_v / \partial \mathbf{x}_v$ and $\partial f_v / \partial \mathbf{u}_k$ are Jacobian matrices whose calculations are much more complicated but a tractable matter of differentiation; and we do not present the derivation results here. Analogous to time update process, the measurement update equations on Kalman gain are

$$\mathbf{K}_k = \mathbf{P}_k^- \frac{\partial h_i}{\partial \mathbf{x}} \mathbf{S}_i^{-1} = \begin{pmatrix} \mathbf{P}_{xx} \\ \mathbf{P}_{y_1 x} \\ \mathbf{P}_{y_2 x} \\ \vdots \end{pmatrix} \frac{\partial h_i}{\partial \mathbf{x}_v} \mathbf{S}_i^{-1} + \begin{pmatrix} \mathbf{P}_{xy_i} \\ \mathbf{P}_{y_1 y_i} \\ \mathbf{P}_{y_2 y_i} \\ \vdots \end{pmatrix} \frac{\partial h_i}{\partial \mathbf{y}_i} \mathbf{S}_i^{-1} \quad (5.10)$$

$$\mathbf{S}_i = \frac{\partial h_i}{\partial \mathbf{x}_v} \mathbf{P}_{xx} \left(\frac{\partial h_i}{\partial \mathbf{x}} \right)^T + \frac{\partial h_i}{\partial \mathbf{x}_v} \mathbf{P}_{xy_i} \left(\frac{\partial h_i}{\partial \mathbf{y}_i} \right)^T + \frac{\partial h_i}{\partial \mathbf{y}_i} \mathbf{P}_{y_1 x} \left(\frac{\partial h_i}{\partial \mathbf{x}_v} \right)^T + \frac{\partial h_i}{\partial \mathbf{y}_i} \mathbf{P}_{y_1 y_i} \left(\frac{\partial h_i}{\partial \mathbf{y}_i} \right)^T + \mathbf{R} \quad (5.11)$$

Other formulae are same as those in Chapter 4. \mathbf{S}_i , the covariance of innovation, is a complicated calculation. \mathbf{S}_i has a further role in active search; it is a measure of the information content expected of a measurement. Each feature stores an 11×11 image patch as a descriptor, and correlation is performed between this patch and the pixels in the search region to determine the actual feature measurement in the

image. Feature searches with high S_i will provide more information about estimates of camera and feature positions. In an implementation of vision-based SLAM for a robot with maneuverable cameras this led directly to active control of the viewing direction towards profitable measurements. This is the idea of Davison. Actually, in our case the active vision system is invalid and the matching algorithm on S_i may not be exploited. Additionally, the search engine is a little bit redundant. To concern these problems, we suggest a data association method to be presented in next section.

5.3 Homography Transformation based Data Association

Sampling is considered very important in nearest neighbor data association methods. In the reference works (Civera, et al., 2007a; Civera, et al., 2007b; Andrew J. Davison, et al., 2007; Andrew. J. Davison, 2003; Montiel, et al., 2006; P. Smith, et al., 2006) samples in a window region are used to match the predicted features and calculate the innovation. However, the computation pixel by pixel in the predefined region is a little bit repetitious. Since each 3D point has only one projection in the image plane, to determine the matched points in 3D space is equivalent to find the identical 2D image points. From this viewpoint, in this section, we suggest a data association scheme based on the homography transformation matrix (HTDA). With the help of SIFT descriptors, this matrix is estimated by the matched points between two images. Compared with the related work stated in Section 5.1, instead of directly applying SIFT descriptors as the natural features, our method emphasizes on using SIFT mechanism to determine the matched points between any two images and then with these matched points estimates the homography transformation matrix. Figure 5.3 shows the matched points. Obviously they are unsusceptible to the moving object (the person). Therefore it is reasonable to treat them as the stable points to determine the homography matrix \mathbf{M} and its covariance $\Sigma_{\mathbf{M}}$ by MLE technique (Hartley & Zisserman, 2003). Our data association algorithm is implemented on the basis of

\mathbf{M} and $\Sigma_{\mathbf{M}}$.

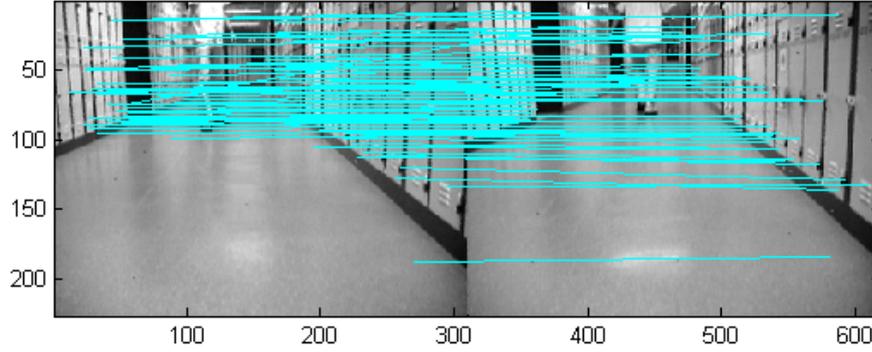


Figure 5.3 Matched points determined by SIFT descriptors

The predicted pixel coordinates of line ends and mid-point are expressed as

$$\hat{p}_f = \mathbf{M}p_m \quad (5.12)$$

where p_m is the pixel coordinates of the line ends and mid-point stored in the map (note that mid-point is not the component of the state variable). The observation of the captured feature in the image is marked as p_f , and the definition of Mahalanobis distance is

$$d_m = (p_f - \hat{p}_f)^T \Sigma_{\mathbf{M}}^{-1} (p_f - \hat{p}_f) \quad (5.13)$$

Compared with the formula in Gil's work (A. Gil, et al., 2006), the main differences are the Mahalanobis distance in (5.13) is constructed by $\Sigma_{\mathbf{M}}$ and the pixel coordinates without any SIFT descriptor. As any two images used for \mathbf{M} and $\Sigma_{\mathbf{M}}$ estimation imply the information of the camera motion, we regard the uncertainty induced by motion as the potential elements involved in $\Sigma_{\mathbf{M}}$. Also for calculation of \mathbf{M} and $\Sigma_{\mathbf{M}}$, we have considered the pixel error in both images as well as the propagation in equation (5.12). Therefore the covariance in observation and prediction can be regarded as being led by the covariance of \mathbf{M} . That is why here we use $\Sigma_{\mathbf{M}}$ for the distance computation. If at least two of d_m values which belong to same line features are less than a threshold (usually a χ^2 distribution value), then the observed line is associated with the one stored in the map, labeled as 1, otherwise the line is a new feature, marked as 0. Note that once the

corresponded 2D image points have been ascertained the matched 3D points are also known. Therefore, those 3D points not 2D image points have to be used for innovation computation in measurement update procedure of EKF.

Practical considerations on data association

Sometimes the predicted line ends may locate outside the image range, and we probably cannot match the lines via two ends or one-end-one-midpoint. For this particular situation, we can employ the auxiliary Hough space parameters designed in previous section for data association. We adopt an alternative way to test whether the predicted ends lie on the observed lines. The homogeneous representation for observed lines represented by the Hough space parameters is $l_m = (\cos\alpha_H, \sin\alpha_H, -d_H)^T$, and homogenous coordinate for predicted ends is $\hat{p} = (\hat{p}_u, p_v, 1)^T$. The ends lie on the lines if and only if $p^T l_m = 0$. We relax this condition practically as

$$p^T l_m < \varepsilon \quad (5.14)$$

where ε is an arbitrarily small positive quantity.

It is impractical and time consuming to compute all \mathbf{M} and $\Sigma_{\mathbf{M}}$ between the most recent image and all the previous ones. In this study, we captured an image per second and calculated \mathbf{M} , $\Sigma_{\mathbf{M}}$ by using the newest grabbed image and the four latest ones, because the robot moves in 300mm/s and after 5 seconds some features stored in the map could probably disappear in current image. Hence, it is not reasonable to use these features for the purpose of feature prediction. Figure 5.4 illustrates our HTDA algorithm.

HTDA ALGORITHM

```

// INPUT: observed lines parameters, the 5 most recent images

// OUTPUT: DA matrix

[ desCur, locCur ] = sift(CurrentImg ); // Find SIFT keypoints for each image. The outputs
                                         // are des: descriptor for the keypoint;
                                         // loc: keypoint location

for each observed line i
    for k = 4:-1:1
        [ desK, locK ] = sift( Img( k ) );
        // Estimating M and  $\Sigma_M$ 
        [ M( k ),  $\Sigma_M( k )$  ] = HomographyEstimation( locCur, locK,  $\sigma_C$  );
        // Observation prediction
        for each line feature j stored in map
            EndsPred ( j ) = M( k )EndsMap( j );
            dm = ( EndsObs( i ) - EndsPred( j ) ) · ( $\Sigma_M( k )$ )-1 · (EndsObs( i ) - EndsPred( j ))T;
            if ( dm <=  $\chi^2$  value )
                DA( i, j, k ) = 1;
            else if isPredEndsOnObsLine( EndsPred( j ), ObsLineHoughPara( i ) )
                DA( i, j, k ) = 1;
            else
                DA( i, j, k ) = 0;
            end
        end
        if ~isZero( DA( i, :, : ) )
            continue;
        end
    end
end

```

Figure 5.4 HTDA algorithm

5.4 Sensor Fusion Techniques

As was mentioned in Chapter 4, we used the robust regression model to extract the segments from the raw laser rangefinder data (we call them as laser segments), and eliminated most of the outliers related to moving objects. However, if these dynamic objects momentary start and stop several times, they could probably be treated as segment features which deteriorates the performance of SLAM. Since the lines extracted from the monocular camera are almost static features, we combine these lines with laser segments and adopt Bayesian decision as the feature fusion strategy to remove those pseudo segments. Furthermore, we suggest a modified MPEF to incorporate the individual state estimates of the monocular and laser SLAM. With modified MPEF, the covariance of the robot pose is reduced so that the accuracy of the localization can be improved.

5.4.1 Line Features Fusion

As the limit of the horizontal field of view (FOV) of the monocular camera is about 48° , it is feasible to extract the laser segments within 67° to 114° , i.e. -24° to 24° converted into the robot reference. After receiving a line feature from an image, by parameter vector \mathbf{y}_i we can respectively compute approximate angles of the two ends from robot head (i.e. the z -axis of robot frame) as well as determine the angular interval $[\zeta_1, \zeta_2] \subseteq [-24^\circ, 24^\circ]$ for this line. Within this angle boundary, we extract the laser segments and then employ the Bayesian decision (Zhu, 2003) to fuse the features. The decision rule is

$$\frac{p(\mathbf{y} | \mathbf{H}_1) \frac{\mathbf{H}_1}{\mathbf{H}_0} p(\mathbf{H}_0)(C_{10} - C_{00})}{p(\mathbf{y} | \mathbf{H}_0) \frac{\mathbf{H}_0}{\mathbf{H}_1} p(\mathbf{H}_1)(C_{01} - C_{11})} \square LR(\mathbf{y}) \frac{\mathbf{H}_1}{\mathbf{H}_0} \eta \quad (5.15)$$

where C_{ij} , $i=0,1, j=0,1$, represents the cost of declaring \mathbf{H}_i true when \mathbf{H}_j is actually true. Generally, the monotonically increasing natural logarithm rule is considered, that is

$$\ln LR(\mathbf{y}) \underset{\mathbf{H}_0}{\overset{\mathbf{H}_1}{\gtrless}} \ln \eta \quad (5.16)$$

In this study we choose $[\mathbf{z}_C, \mathbf{z}_L]^T = [s_C + v_C, v_L]^T$ as the null hypothesis \mathbf{H}_0 and relevant alternative hypothesis \mathbf{H}_1 is $[\mathbf{z}_C, \mathbf{z}_L]^T = [s_C + v_C, s_L + v_L]^T$. v_C and v_L are mutually independent additional sensor noises of the camera and laser. They have zero mean and covariance σ_C^2 and σ_L^2 . s_C and s_L are the lines parameters extracted from the camera and laser with the covariance R_C and R_L . Noted that the parameter s_C for lines representation in images is $[d_H, \alpha_H]^T$ of Hough space, and R_C is the covariance of $[d_H, \alpha_H]^T$. As for s_L and R_L , they are the parameters of laser segments and relevant covariance, the calculation of which are stated in Chapter 4.

Suppose that $p(\mathbf{H}_0) = p(\mathbf{H}_1) = 0.5$, $C_{01} = C_{10} = 1$ and $C_{00} = C_{11} = 0$, which means that the cost for mistaken decision is much more than that for correct decision. Let $\mathbf{y} = [\mathbf{z}_C, \mathbf{z}_L]^T$, then $p(\mathbf{y}|\mathbf{H}_0) \sim \mathcal{N}(\mathbf{0}, \Sigma_0)$ and $p(\mathbf{y}|\mathbf{H}_1) \sim \mathcal{N}(\mathbf{0}, \Sigma_1)$. Here

$$\Sigma_0 = \begin{pmatrix} R_C + I\sigma_C^2 & 0 \\ 0 & \sigma_L^2 \end{pmatrix}, \Sigma_1 = \begin{pmatrix} R_C + I\sigma_C^2 & 0 \\ 0 & R_L + I\sigma_L^2 \end{pmatrix} \quad (5.17)$$

and the decision rule (5.16) is equal to

$$-\frac{1}{2} \left(\begin{bmatrix} \mathbf{z}_C \\ \mathbf{z}_L \end{bmatrix}^T \Sigma_1^{-1} \begin{bmatrix} \mathbf{z}_C \\ \mathbf{z}_L \end{bmatrix} - \begin{bmatrix} \mathbf{z}_C \\ \mathbf{z}_L \end{bmatrix}^T \Sigma_0^{-1} \begin{bmatrix} \mathbf{z}_C \\ \mathbf{z}_L \end{bmatrix} \right) \underset{\mathbf{H}_0}{\overset{\mathbf{H}_1}{\gtrless}} \ln \frac{|\Sigma_1|}{|\Sigma_0|} \quad (5.18)$$

With rule (5.18), we validate all the laser segments located in $[\zeta_1, \zeta_2]$. If \mathbf{H}_0 is accepted then the laser segment is the outlier, otherwise if \mathbf{H}_1 is accepted then it is the real static feature.

5.4.2 Modified Multi-sensor Point Estimation Fusion

A framework of MPEF for Kalman filter was proposed by Zhu (2003). It led to a lower covariance for fused state estimates compared with each individual one, as well maintaining the optimal estimation. We have extended the idea of MPEF in

this subsection to EKF-SLAM and mathematically proved that the covariance of the fused state variables decreased even though the fused estimation could not be kept at an optimal value when the fused state variables were propagated backward to the individual EKF-SLAM. Please refer to Appendix B for more details on theoretical derivation, and here we directly apply the derivational results. The purpose of modified MPEF is to improve the accuracy of localization. We sketched our fusion SLAM algorithm in Figure 5.5. The superscript i indicates the type of the sensor, 1 for monocular camera and 2 for laser; f means fusion and b stands for back propagation.

FUSION SLAM BASED ON MODIFIED MPEF ALGORITHM

```

// Robot pose initialization
[  $\mathbf{x}_{v0\_C}$ ,  $P_{v0\_C}$  ] = PoseInitialization( Camera );
 $\sigma_C$  = getSensorError( Camera );
[  $\mathbf{x}_{v0\_L}$ ,  $P_{v0\_L}$  ] = PoseInitialization( Laser );
[  $\sigma_{range}$ ,  $\sigma_{bearing}$  ] = getSensorError( Laser );
Q = createQ( $\sigma_{tra}$ ,  $\sigma_{rot}$ );
// Line Feature initialization
SegC = HorizontalEdge( image ); // Line extraction
[  $\mathbf{y}_{0\_C}$ ,  $P_{y0\_C}$ ,  $R_C$  ] = initializeNewFeature( SegC, camPar,  $\mathbf{x}_{v0\_C}$ ,  $P_{v0\_C}$ ,  $\sigma_C$  );
 $\mathbf{X}_{0\_C}$  = createX(  $\mathbf{x}_{v0\_C}$ ,  $\mathbf{y}_{0\_C}$  );  $P_{0\_C}$  = createP(  $P_{v0\_C}$ ,  $P_{y0\_C}$  );
SegL = LineExtraction( laserdata ); // Line extraction
[  $\mathbf{y}_{0\_L}$ ,  $P_{y0\_L}$ ,  $R_L$  ] = initializeNewFeature( SegL,  $\mathbf{x}_{v0\_L}$ ,  $P_{v0\_L}$ ,  $\sigma_{range}$ ,  $\sigma_{bearing}$  );
 $\mathbf{X}_{0\_L}$  = createX(  $\mathbf{x}_{v0\_L}$ ,  $\mathbf{y}_{0\_L}$  );  $P_{0\_L}$  = createP(  $P_{v0\_L}$ ,  $P_{y0\_L}$  );
// Fused robot pose initialization
 $\mathbf{x}_{v0}^f = \mathbf{x}_{v0\_L}$ ;  $P_{v0} = P_{v0\_L}$ ;
// Main loop
k = 1;
while isRobotRunning()
     $\mathbf{u}_k = \text{getControl}( k );$ 
    [  $\mathbf{X}_{k|k\_C}$ ,  $P_{k|k\_C}$ ,  $\mathbf{X}_{k|k-1\_C}$ ,  $P_{k|k-1\_C}$  ] = MonoSLAM(  $\mathbf{X}_{k-1\_C}$ ,  $P_{k-1\_C}$ ,  $\mathbf{u}_k$ , Q,  $R_C$ , ...,
                                                    getObservation( imagek ) );

```

```

[  $\mathbf{X}_{k|k\_L}$ ,  $\mathbf{P}_{k|k\_L}$ ,  $\mathbf{X}_{k|k-1\_L}$ ,  $\mathbf{P}_{k|k-1\_L}$  ] = LaserSLAM(  $\mathbf{X}_{k-1\_L}$ ,  $\mathbf{P}_{k-1\_L}$ ,  $\mathbf{u}_k$ ,  $\mathbf{Q}$ ,  $\mathbf{R}_L, \dots$ 
                                     getObservation( laserk ) );

// Do MPEF procedure

 $\mathbf{x}_{k|k}^f = \mathbf{x}_{k|k-1}^f + P_{k|k}^f \sum_{i=1}^2 (P_{k|k}^i)^{-1} (\mathbf{x}_{k|k}^i - \mathbf{x}_{k|k-1}^i)$ 
 $= \mathbf{W}[\mathbf{x}_{k|k-1}^1, \mathbf{x}_{k|k}^1, \mathbf{x}_{k|k-1}^2, \mathbf{x}_{k|k}^2, \mathbf{x}_{k|k-1}^2]^T$ 

 $(P_{k|k}^f)^{-1} - (P_{k|k-1}^f)^{-1} = \sum_{i=1}^2 [(P_{k|k}^i)^{-1} - (P_{k|k-1}^i)^{-1}]$ 

//  $\mathbf{W}$  is the weight matrix, and  $\mathbf{W} = [I, P_{k|k}^f (P_{k|k}^1)^{-1}, -P_{k|k}^f (P_{k|k}^1)^{-1},$ 
 $P_{k|k}^f (P_{k|k}^2)^{-1}, -P_{k|k}^f (P_{k|k}^2)^{-1}]$ 

// Propagate backward the MPEF results to each individual SLAM

 $\mathbf{x}_{k|k-1}^i = \mathbf{x}_{k|k-1}^f; P_{k|k-1}^i = P_{k|k-1}^f; // i = 1, 2$ 

// Update individual covariance

 $P_{k|k}^{i-f} = \text{update}(P_{k|k}^i, P_{k|k-1}^i, P_{k|k-1}^f); // P_{k|k}^f = P_{k|k}^b \leq P_{k|k}^i, i = 1, 2$ 

k = k + 1;

end

```

Figure 5.5 Modified MPEF algorithm.

5.5 Experimental Studies

The camera was calibrated by the Calibration Toolbox written by Bouguet¹ and the intrinsic parameters are listed in Table 5.1. By ARIA and OpenCV class library a sequence of images as well a frame of laser data were collected when the mobile robot was moving with an average speed of 300mm/s. The environment is the corridor outside the Control Lab. There were several people walking through the corridor with normal speed around the robot. Sometimes they slowed down or stopped completely at some place. After obtaining sensor data, we implemented the SLAM and sensor fusion offline in MATLAB environment. The experiments were designed to validate our sensor fusion strategy and data association algorithm.

¹ It is available online. http://www.vision.caltech.edu/bouguetj/calib_doc/index.html

Table 5.1 Intrinsic Parameters of Canon VCC4

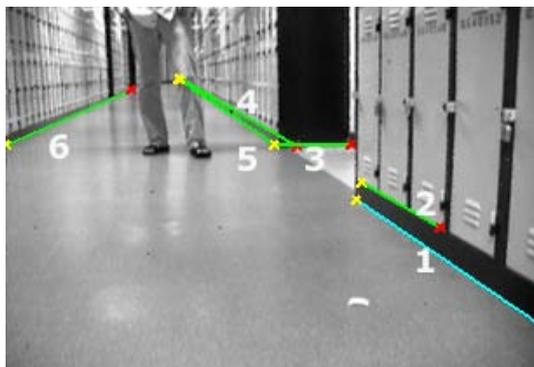
Item	Value
Focal length	$fc = [365.12674 \quad 365.02905]$
Principal point	$cc = [145.79917 \quad 114.50956]$
Screw factor	$\alpha_c = 0.000$
Distortion factor	$kc = [-0.22776, 0.36413, -0.00545, -0.00192, 0.000]$
Pixel std	$err = [0.10083 \quad 0.10936]$

5.5.1 Testing the Feature Fusion Strategy

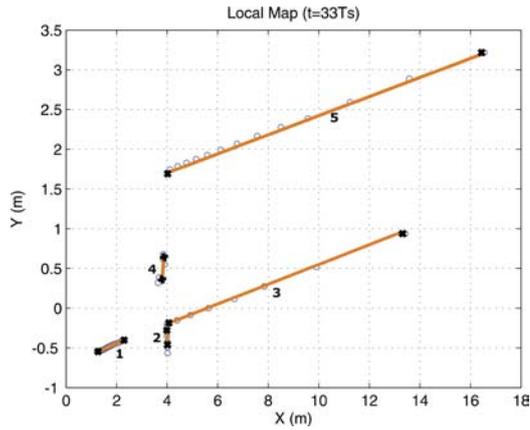
In this experiment, a person stood in front of the robot for few minutes, and only with laser sensor some pseudo segments could not be removed. As it is displayed in Figure 5.6(c), segment 4 is the one corresponding to the person. Using our line feature fusion method, we incorporated the image lines, which were extracted from the raw image Figure 5.6 (a) and labeled in numbers in Figure 5.6 (b), with the laser segments and listed the hypothesis test results of fusion in Table 5.2. Segment 4 did not match any image line and it can be eliminated from the laser segment map shown in Figure 5.6 (d). Segment 3 correlated with line 4 and 5 because it located respectively in the angle interval determined by line 4 and 5. However, segment 3 only related to line 5 after feature fusion. With this experiment, our feature fusion method is competent for disposing pseudo and confused features.



(a)



(b)



(c)

(d)

Figure 5.6 Local mapping results at the 33rd sample time. (a) The original captured image. A person stood in front of the robot for a moment. (b) The extracted static lines. (c) There is a pseudo segment (segment 4) related to standing person only by using laser sensor. (d) After integrating the lines information extracted from images, the incorrect feature was removed.

Table 5.2 The hypothesis test of feature fusion

Number of Segments from laser	Number of lines in image					
	1	2	3	4	5	6
1	H_1	×	×	×	×	×
2	×	×	H_1	×	×	×
3	×	×	×	H_0	H_1	×
4	×	×	×	×	×	×
5	×	×	×	×	×	H_1

×: outside the angle boundary

The SLAM result after feature fusion is shown in Figure 5.7. The map drawn by the software of Mobilerobots Co. is overlaid in light gray color for comparison. Except those lines disappearing in the camera view, most of segments are identical; however, few of them are lost. This is because the assumptions of our Bayesian fusion rule are slightly tight so that a segment related to the real static object is deleted as the pseudo one.

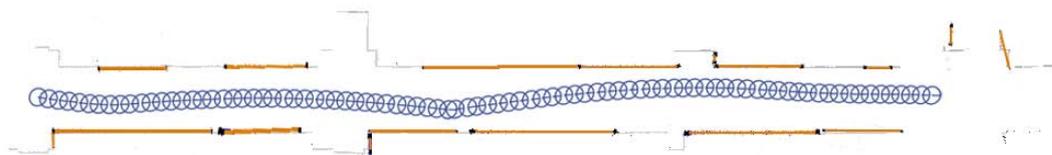


Figure 5.7 The fusion SLAM results. Orange segments are the map of the fusion SLAM, And gray lines are drawn by the software of Mobilerobots Co.

5.5.2 Testing the Fusion SLAM based on Modified MPEF

We ran two individual EKF-SLAM procedures in parallel, computed the values of the fused state variables, and then propagated these fused state variables back to monocular and laser SLAM. Figure 5.8 illustrates the covariance of the fused and individual robot states. It can be seen that the covariance of the position: x_R and y_R is obviously reduced after fusion. However, the value of the orientation is similar to the covariance of laser SLAM, but it is more efficient than that of the monocular SLAM. In this point, the modified MPEF method could decrease the covariance of robot orientation. Figure 5.9 gives the results of covariance on one of the line features. Noted that these line features are those existed in a sequence images. In this experiment, we selected 5 lines which always appear in 40 consistent images, and plotted covariance for the end of one feature in Figure 5.9. It seems that the covariance after sensor fusion is also reduced. With these experiments, we may state that the modified MPEF makes the covariance of state variables less and increase the accuracy of localization.

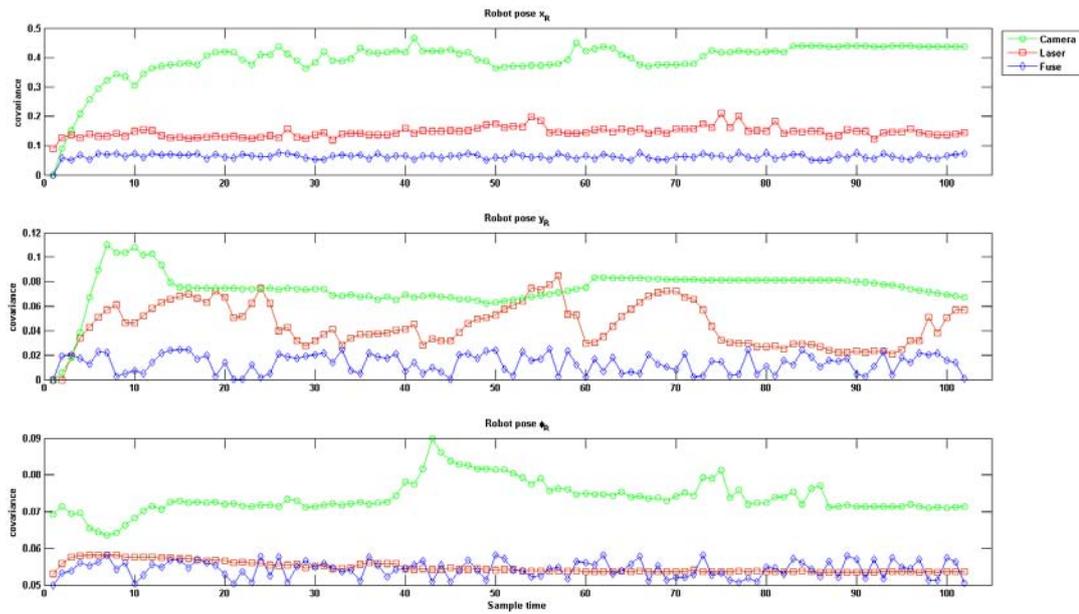


Figure 5.8 The estimate covariance of fused and individual robot state. Red line: covariance predicted in laser SLAM; Green line: covariance predicted in monocular SLAM; Blue lines: fused covariance by MPEF.

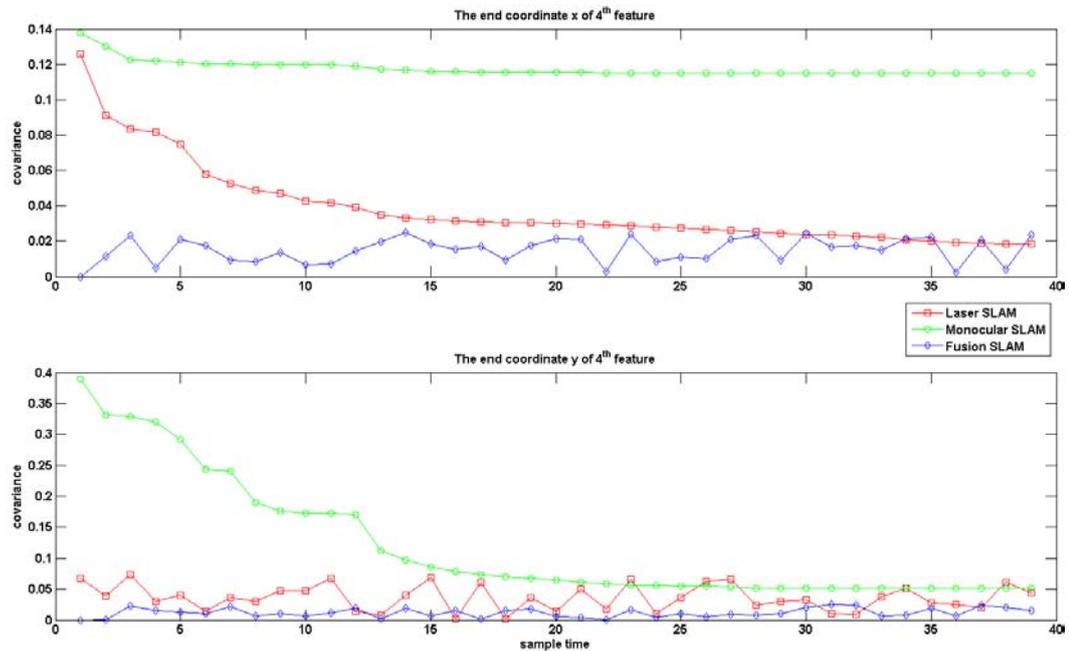


Figure 5.9 The estimate covariance of fused and individual features. Red line: covariance predicted in laser SLAM; Green line: covariance predicted in monocular SLAM; Blue lines: fused covariance by MPEF.

5.5.3 Testing the HTDA

We first compute \mathbf{M} and $\Sigma_{\mathbf{M}}$ from the current captured image (labeled as image 2) and the 4th image (labeled as image 1) stored in image sequence buffer. After that we selected one pair of lines to demonstrate our data association method. As shown in Figure 5.10, we marked ends and midpoint of the existed line of the map as 1, 2, 3, and those of captured line as 1', 2', 3'. According to (5.12), we obtained the predictions of 1', 2' and 3', and stressed in red cross in image 2. To make it clear, as is shown in image 1 of Figure 5.10 we also stressed the corresponding stored points in red cross. In this case, the prediction of 2' almost coincides with the 2', but the prediction of 3' is far from 3', the reason of which is for the lines with different lengths the midpoints are probably not identical. In addition, the prediction of end 1' is out of the bound of image 2. For these two special situations, we used the alternative method stated in Section 5.3 and tested whether the predictions lie on the captured line by (5.14), and obviously they did. Therefore, we can decide the predicted line in image 2 matched the stored line in image 1.

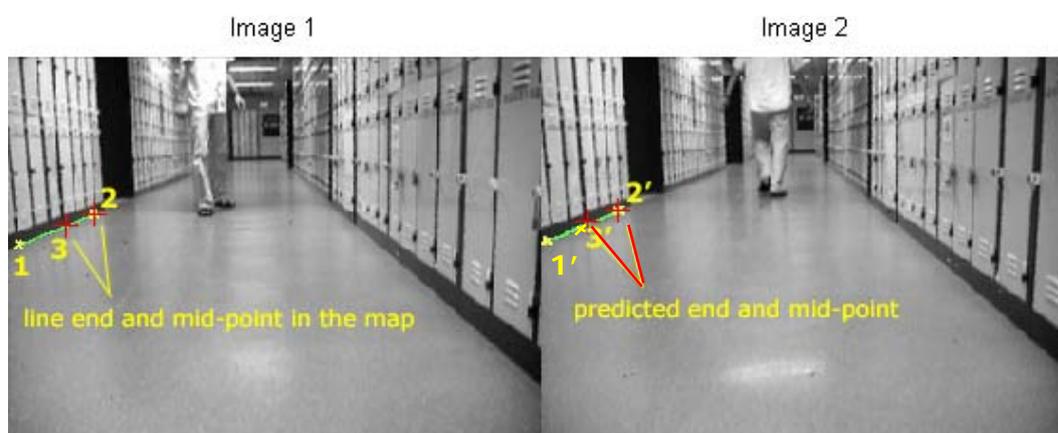


Figure 5. 10 The example of HTDA. The image on the left is captured at the 57th sample time and the right one is at the 58th sample time.

Figure 5.11 shows the true errors of HTDA for the known line end 2 shown in the

left image of Figure 5.10. Because there is no device in our present experimental conditions for detecting the ground truth of the features, we provisionally measured the end of a line by hand as accurate as possible, which follows the similar process presented by Wijesoma, Perera and Adams (2006). This end appears in 20 sequential images. As displayed in Figure 5.11, the actual feature estimation errors are bounded within the 3σ bound, demonstrating the effectiveness and consistency of the HTDA.

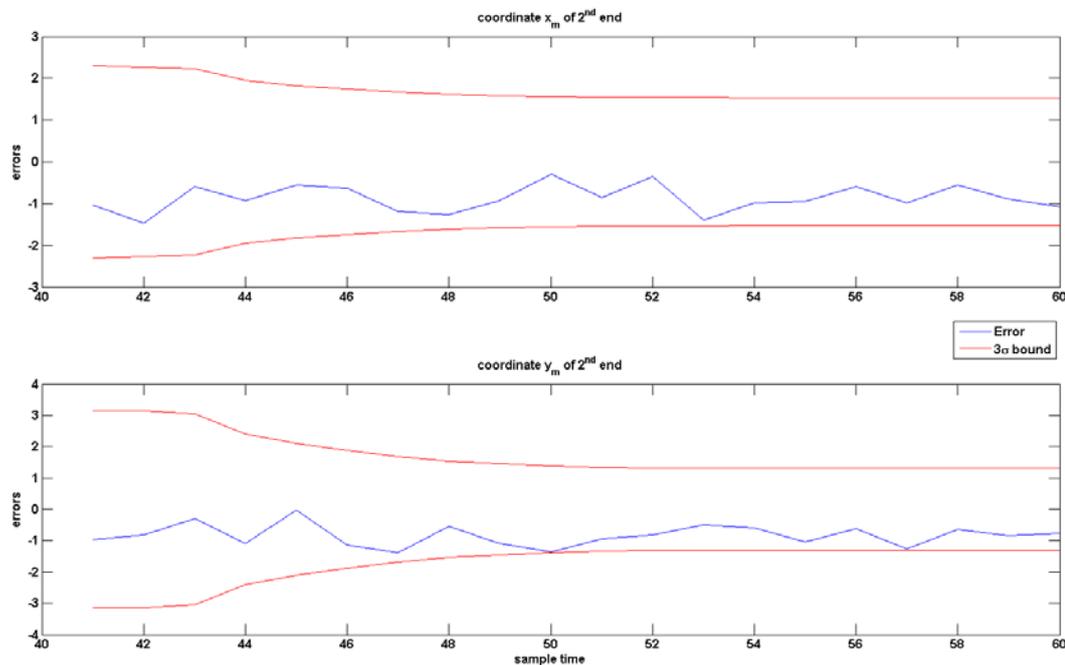


Figure 5.11 Difference between the actual and estimated location of end 2 from 40th to 60th sample time. The 99% confidence limit is shown in red line.

5.6 Discussion

The sensor fusion method suggested for the SLAM in dynamic environments consists of the feature fusion and modified MPEF components. Feature fusion policy incorporates the static line features extracted from monocular camera with the segments represented by robust regression model from laser sensor, the purpose of which is to remove the potential pseudo segments corresponding to the moving objects in laser information. In addition, the modified MPEF, which

combines distinct state estimates of the individual SLAM procedure (monocular and laser SLAM), reduces the covariance of the state variables and improves the accuracy of localization.

In current study, the lines extracted from the camera are limited in a ROI. In the predefined ROI, objects almost maintain the static status. Therefore, in the proposed feature fusion module the lines detected from the images play a primary role for removing the pseudo segments generated from the laser data. Actually, even without the ROI, the camera also can detect the moving objects with the target tracking technique, which is presented in our previous work. However, in this research we do not use the target tracking to detect moving objects but mainly focus on applying the static elements existing in the image to help to find the spurious features and then remove them. The main advantage of this way is that the cost of time on image processing is less than that consuming in target tracking.

Other sensors, to the best of our knowledge, could not be used for detecting the moving objects when these sensors are employed to extract the segment in dynamic environments, because there is no clear difference between the measurements of dynamic and static objects. However, if the measurements from these sensors such as laser rangefinder are not considered to present the segment based map, they can be adopted to detect the moving objects through efficient algorithms. Some research work have implemented the SLAM in dynamic environment with non-vision sensor systems, in which the maps are not segment based map but other type of map for example grid based map.

Compared with other non-vision sensors, the dominant advantages of vision sensors are it has abundant information and with these sufficient information it is easy to implement various tasks such as target tracking. So far the disadvantages of vision system are on the information processing and information understanding. The image processing needs more resource of the computer. Fortunately, with the

development of the computer techniques, it is not a serious problem. The challenge nowadays is on the software design for implementing the information understanding just like the ability of human beings.

Additionally, for monocular SLAM we presented a data association method based on homography transformation between any two images. It relaxes pleonastic computation. However, the proposed sensor fusion just makes use of the feature parameters to eliminate the pseudo segments according to Bayesian criteria, which is an indirect policy and does not combine the parameters of line features extracted from laser and camera but fuses the state variables existed in each individual EKF-SLAM only. In a sense, we call this strategy as indirect sensor fusion or semi-direct sensor fusion. When verifying the validation of the features, we limited the laser data within the FOV of the camera. Most of laser information is derelict and the useful laser measurements are not considered for fusion. As analysis in experimental study, the suppositions for Bayesian fusion rule are so tight that few of static segments are deleted erroneously as the pseudo ones. Additionally, the parameters for image lines are based on the ends and it is poor for this form of representation that the volumes of the state vector and covariance matrix become enormous when the number of the features grows.

We will suggest a direct sensor fusion management based on the information theory, which integrates the parameters of the segments extracted respectively from camera and rangefinder sensors in a straightforward style in the next chapter. The parameters for describing the segments are not the coordinates of the end but a compact form similar to Hesse standard form. The purpose of all these schemes above is to cover the shortages existed in the sensor fusion presented in this chapter.

Chapter 6 Direct Sensor Fusion Management

6.1 Introduction

The sensor fusion strategy presented in Chapter 5 is an indirect method which only applies the feature parameters to eliminate the pseudo segments according to Bayesian criteria without integrating each individual segment measurement. In this chapter, a direct sensor fusion management is to be suggested.

A very prevalent direct sensor fusion scheme EKF and its extension have been used to solve several problems in robotics research (Ahn, et al., 2008; Lizarralde, et al., 2003; Roumeliotis & Bekey, 2002; Tsai, et al., 2005). Similar to EKF, particle filter, another kind of Bayesian filter, is adopted for sensor fusion (Moreira, et al., 2007; Vadakkepat & Jing, 2006), which has ability to deal with the multi-modal distribution. The series of Bayesian filters belongs to the statistical fusion techniques, and other fusion methodologies based on uncertainty inference are also applicable such as Dempster-Shafer rules (Drocourt, et al., 2002) and fuzzy logic theory (Cohen & Edan, 2005; Matía & Jiménez, 1998) are employed to combine the different sensor measurements.

Most methods represented above required a *priori* knowledge, for example the prior probability in the Bayesian fusion method, for implementation of sensor fusion. However, sometimes it is difficult to obtain this information. To address this problem, a general fusion approach based on information theory was suggested (Manyika & Durrant-Whyte, 1994). In that work, the fusion algorithm was a distributed information filter and the measures of the fusion were posterior entropy, prior entropy and mutual information. The sensor management method made use of information-based utility functions. With Shannon's entropy, an Entropy Fusion Model (Fassinut-Mombot & Choquel, 2004) was defined the

purpose of which was to reduce the combination space by explicitly resending the notions of source redundancy and source complementarity in the form of entropy measures. Unfortunately, this model has been mainly used for object recognition and not applied to SLAM. Following the concept of information theory and borrowing the idea of decision analysis, this chapter presents and develops a direct sensor fusion management based on information entropy weight, which fuses the parameters of the features obtained from various sensor measurements and related variance to reduce the uncertainty. Furthermore, the fused features are consolidated into the EKF-SLAM framework and the uncertainty of the robot pose evidently decreased in comparison of the results without considering the sensor fusion. In addition, as for determination of associative features extracted from different sensors, we also suggest an algorithm based on information theory which focuses on computing the error entropy and confirm the relevant features through this error entropy. The advantages of this sensor fusion strategy are no extra postulated conditions, generality and simple implementation.

The rest of this chapter is organized as follows: the segment parameter extraction from rangefinders is presented in Section 6.2. Section 6.3 describes the acquisition of Plücker coordinates of lines obtained from monocular and stereo vision systems. The proposed direct sensor fusion management is presented in section 6.4. Elaborate experimental studies are implemented in Section 6.5. The discussion is expressed in Section 6.6.

6.2 Feature Extraction from Rangefinder Sensors

Following the procedure presented in Chapter 4, we apply the robust regression model to extract the segments for the Hessian line model from the data of laser rangefinder and extend this model to ultrasonic sonar data. In contrast to earlier work (Ip & Rad, 2004), in this study the sonar data are categorized by linear group algorithm (LGA) (Van Aelst, Wang, Zamar, & Zhu, 2006), in which the number

of classes is determined by gap statistic (Tibshirani, Walther, & Hastie, 2001; Van Aelst, et al., 2006). After that, for these clustered sonar data, we employ the robust regression algorithm to estimate the parameters of segments. As the segments extraction from laser readings have been explained in Chapter 4, only the algorithm of feature representation for sonar measurements is depicted here.

For feature extraction from ultrasonic sonar, we applied the LGA algorithm to group the sensor readings after determining the number of clusters via gap statistic. To make this chapter self-contained, the concepts and formulas on LGA and gap statistic are to be briefly introduced. Interested reader can refer to the related material (Tibshirani, et al., 2001; Van Aelst, et al., 2006) for more detail.

LGA uses orthogonal regression to identify the linear relationships and iterative optimization similar to K-means to converge to a local minimum. Suppose a data set of size n in d dimensions and k is the number of groups which is estimated via gap statistic, the aim is to minimize the within sum of squares that is the aggregated sum of the squared orthogonal residuals within each group. This can be written as

$$\min_{I_1, \dots, I_k} \sum_{j=1}^k \sum_{i \in I_j} r_i^2 \quad (6.1)$$

where the minimization is over all size k partitions I_1, \dots, I_k of $\{1, \dots, n\}$. For each group I_j the corresponding residuals r_i of the observations in that group are the orthogonal residuals, i.e. the distance between the observation and the orthogonal regression hyperplane through the points in the group. LGA tries to find the optimum of (6.1) following 5 steps (Van Aelst, et al., 2006).

To estimate the number of cluster, Tibshirani et al. (2001) proposed the gap statistic which uses the output of any clustering algorithm, comparing the change in within-cluster dispersion with that expected under an appropriate reference null

distribution. Applied for LGA, the gap statistic is

$$GAP(k) = \frac{1}{B} \sum_{b=1}^B \log(SSR_k(b)) - \log(SSR_k) \quad (6.2)$$

where SSR_k is the aggregated sum of squared orthogonal distance for the original data set separated into k clusters, and $SSR_k(b)$, $b=1, \dots, B$ is the aggregated sum of squared orthogonal distance for a data set generated from the reference distribution and separated into k clusters. The optimal number of cluster is the smallest k such that $GAP(k) \geq GAP(k+1) - s_{k+1}$, where $s_{k+1} = std_{k+1} \sqrt{1+1/B}$, std_{k+1} is the standard deviation of the $SSR_k(b)$.

In our case, the robot platform is the Pioneer 2DX and 3DX mobile robots equipped with a 16-sonar array. We store 20 robot sampling steps (each robot sampling time is 250ms) of sonar data into an overlapping sliding window sonar buffer using first-in-first-out strategy different from the work of Ip & Rad (2004) where the first-in-last-out was employed. As is shown in Figure 6.2(a), we firstly filter out the sonar data which are over the sonar range cut-off limit (1500mm), and after that use gap statistic to decide the number of groups, then cluster these preprocessed sonar data through LGA. Finally, the robust regression model is adopted to extract the parameters of categorized segments. This parameter computation procedure is a one-step implementation not iteration routine because the cluster has been determined, which is different from the case in Chapter 4. It should be noted that we select 5 as the maximum possible number of clusters k . This is because there are only 16 sets of sonar data which are not as dense as laser rangefinder especially after measurements filtering, and the available number of segment features is probably not more than five. Figure 6.1 illustrates the flow chart for feature extraction from the sonar data, and Figure 6.2 exhibits an example for some blocks of Figure 6.1. We can catch from Figure 6.2(b) that the gap statistic computation stopped at $k=3$ even though the maximum cluster number was assigned as the input argument. It is obvious that the gap statistic is the

maximum when $k=2$. Figure 6.2(c) is the results of LGA with $k=2$.

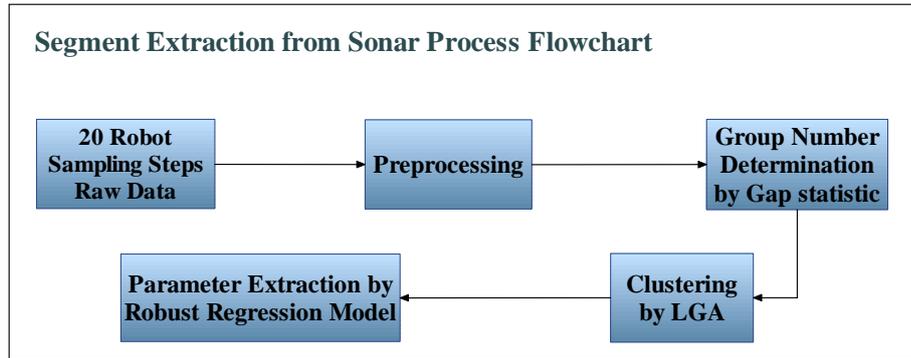
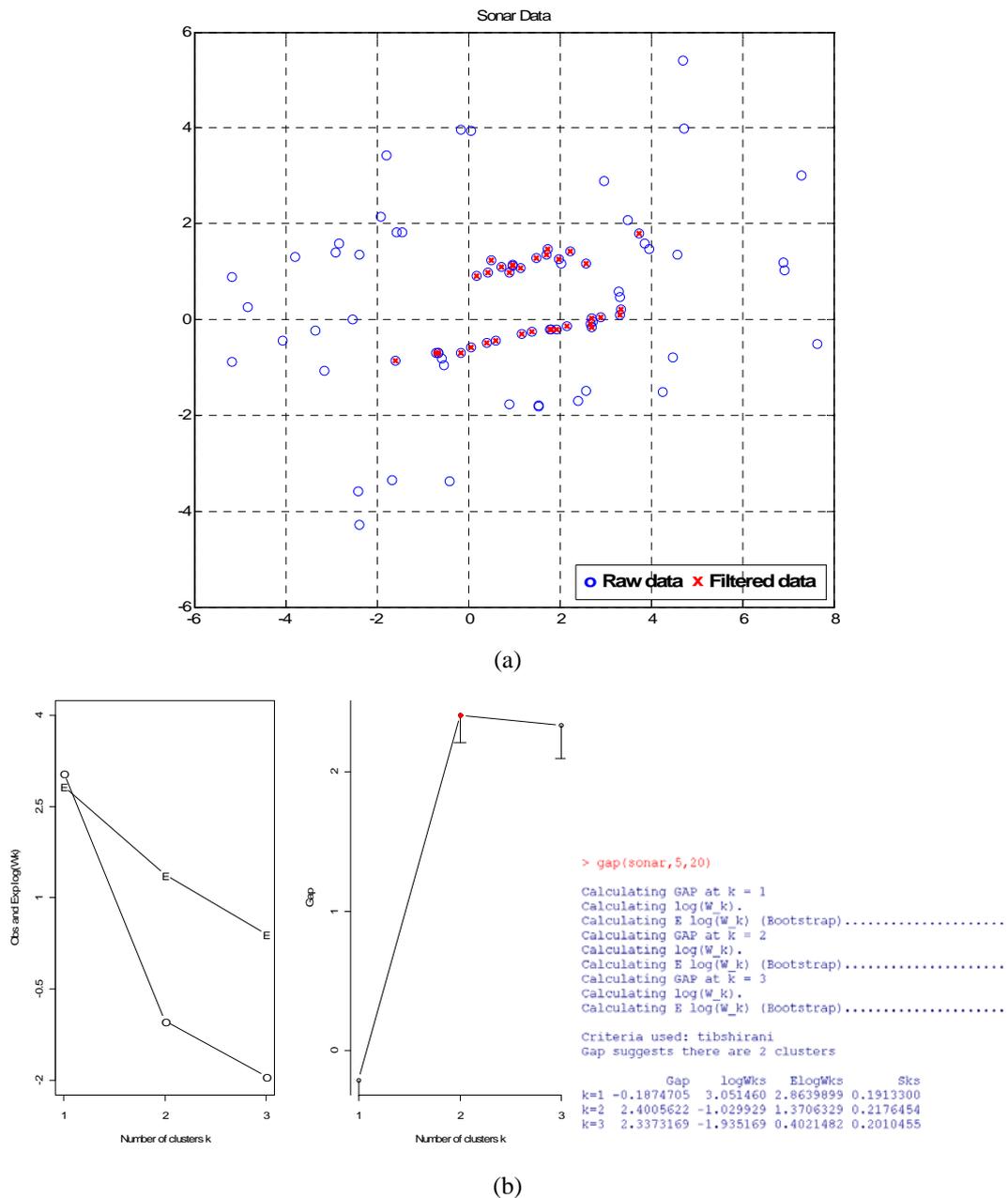
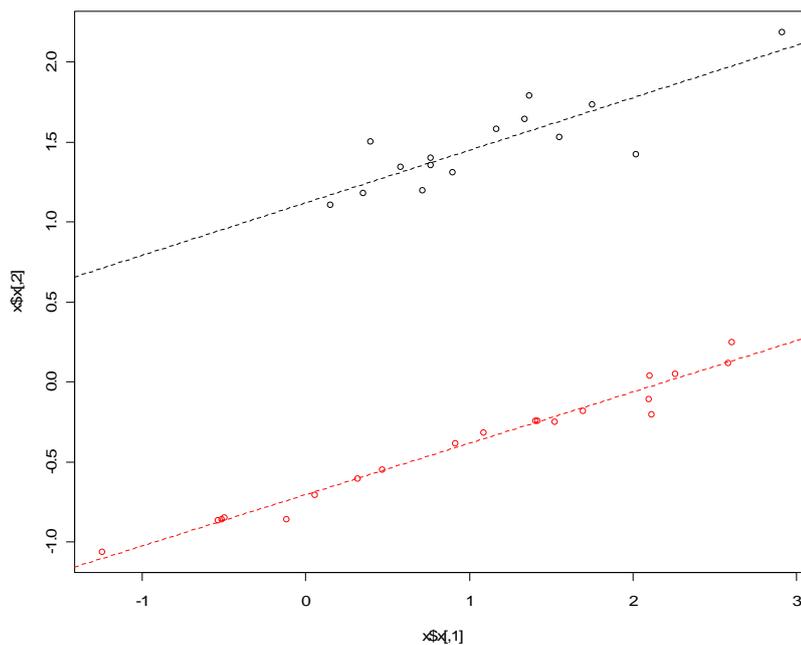


Figure 6.1 Segment extraction process for sonar data





(c)

Figure 6.2 An example of segment extraction from sonar data. (a) Raw and filtered sonar data; (b) Gap statistic computation; (c) The results of LGA

6.3 Feature Extraction from Vision Systems

For the line reflected in vision system, the minimal representation is with four parameters (e.g. Denavit-Hartenberg line coordinates) in 3D Euclidean space but it may be ineffective in some robotic research topics. There are several non-minimal representation for the 3D line, such as ends of the line (P. Smith, et al., 2006) which has been used in Chapter 5, center and unit direction vector of the line (Ethan Eade & Tom Drummond, 2006), two ends plus unit direction vectors (Gee & Mayol-Cuevas, 2006), and a powerful tool in vision and graphics referred to as Plücker coordinates (Lemaire & Lacroix, 2007). In this study, we also describe the lines in a non-minimal representation by using Plücker coordinates because the advantages of it are this presentation is homogenous and suitable for the projection through a pinhole camera.

6.3.1 Representation for Lines Obtained from Monocular Camera

A monocular camera has few limitations one of which is depth extraction. This makes the calculation of Plücker coordinates a bit more difficult. Fortunately, the depth can be approximately retrieved by a sequence of images. Saxena, Chung and Ng (2005) applied the supervised learning to predict the depth-map with a function of the image by integrating multi-scale local and global image features. Murphey et al. (2000) designed a DepthFinder to detect the distances of the objects through at least two images, which acts like a stereo vision system. The errors of this model, however, are serious when the camera motion is unparallel, that is, the ambiguity of image points affects the accuracy. To eliminate variety of ambiguity, Martin (2006) introduced several powerful domain specific constraints and presented an evolving visual sonar, but some of these constraints may generate the improper bound and confuse the robots to perceive the pseudo obstacles. These disadvantages may also affect the Plücker coordinates computation.

Concerning these problems, for the segments that locate in the ROI (defined in Chapter 5) we suggested a virtual rangefinder model for extracting the Plücker measurements of the interested lines, which is to be explained more in the following paragraphs. This model relaxes the constraint that the motion of the optical axis. In comparison with the work of Lemaire and Lacroix (2007) where they treated Plücker coordinates a Gaussian sum approximation process for the feature initial state, our suggested model is an un-delayed method to recover the depth information and is the simple geometric calculation without another Kalman filter that serves for the features initialization via considering constraints associated with the Plücker representation during the update step.

The overview of geometric relationship among four main coordinate references is displayed in Figure 6.3(a). The subscript **L** refers to robot or local frame and the related plane is Π_L . Similarly, **C** denotes camera reference, **I** refers to the image

frame and \mathbf{W} expresses world or global coordinate system. For convenience, as shown in Figure 6.3(b) we assume that the origins of local and camera coordinate frame (\mathbf{O}_L and \mathbf{C}) are identical. Given a world point \mathbf{P}_1 and \mathbf{P}_2 that are the ends of one segment on the ground plane (to make a simplified introduction we only use end \mathbf{P}_1 marked as \mathbf{P} for the following presentation), \mathbf{P}' is the projection of \mathbf{P} on line \mathbf{GE} that is the projection of optical axis on the ground. Here \mathbf{E} is the intersection of the optical axis with the ground plane. The images of \mathbf{P} and \mathbf{P}' in Π_I are \mathbf{p} and \mathbf{p}' . The height g of the optical center from the ground plane and the tilt angle δ of the optical axis from the \mathbf{X}_L axis are hand-measured (cf. Figure 6.3(c)). We can recover the range l which is the distance from the robot to the end \mathbf{P} following the equations (6.3)~(6.6).

$$\beta = \arctan \frac{v_p - v_o}{fk_v}, \varpi = \arctan \frac{u_p - u_o}{(v_p - v_o) / \cos \beta} \quad (6.3)$$

$$P'C = g / \sin(\delta + \beta), \quad (\text{cf. Figure 6.3c}) \quad (6.4)$$

$$PC = P'C / \cos \varpi, \quad (\text{cf. Figure 6.3b}) \quad (6.5)$$

$$l = \sqrt{PC^2 - g^2} = \frac{g}{\sin(\delta + \beta)} \sqrt{1 - \sin^2(\delta + \beta) \cos^2 \varpi} \quad (6.6)$$

where fk_u and fk_v are the focal length at the direction of U and V axis respectively; u_p and v_p are the pixel coordinates of \mathbf{p} ; u_o and v_o are those of principal point. With the same technique, in Figure 6.3(c) we can also retrieve the distance d_{pj} from the robot to the projection point \mathbf{P}' according to

$$d_{pj} = \frac{g}{\tan(\delta + \beta)} \quad (6.7)$$

Therefore, the bearing of \mathbf{P} from the \mathbf{X}_L axis is

$$\begin{aligned} \varphi &= -\text{sgn}(u_p - u_o) \arccos \frac{d_{pj}}{l} \\ &= -\text{sgn}(u_p - u_o) \arccos \frac{\cos(\delta + \beta) \cos \varpi}{\sqrt{1 - \sin^2(\delta + \beta) \cos^2 \varpi}} \end{aligned} \quad (6.8)$$

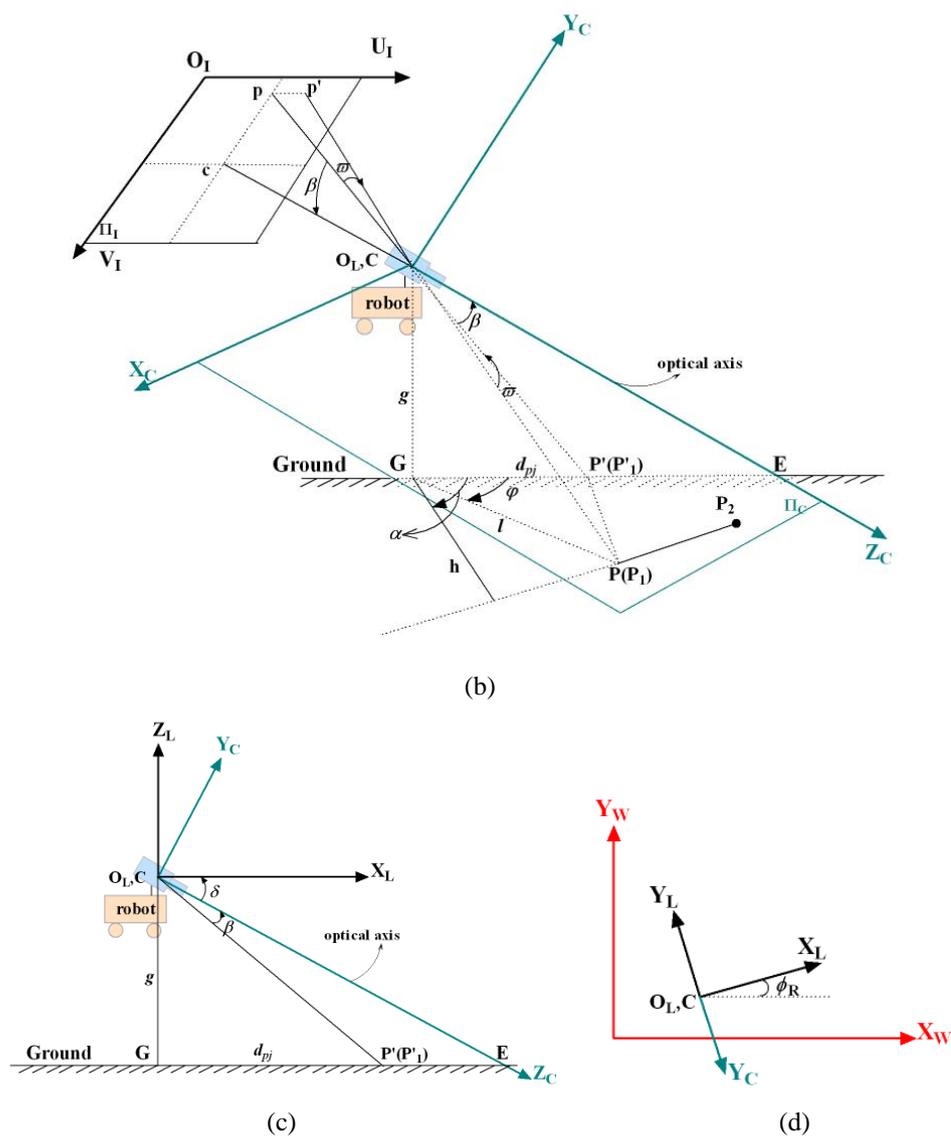


Figure 6.3 Geometric relationship on various coordinate references. (a) The overview of the coordinate systems; (b) The projection between the world points and image points; (c) The front view of the camera and local references; (d) The top view of the local and global references.

6.3.2 Representation for Lines Obtained from Stereo Camera

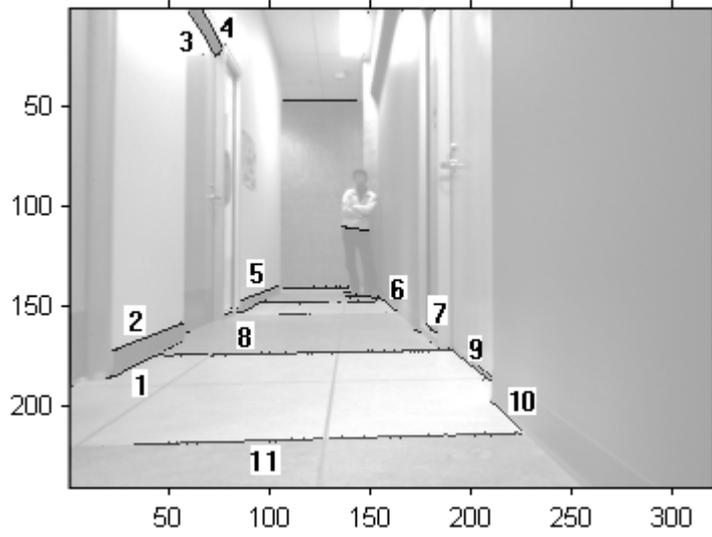
In comparison with the monocular camera, it is much easier to access to the Plücker parameters because the depth can be calculated from the stereo camera. The procedure of 3D line extraction and the computation of Plücker coordinates are specified as follow.

After the rectification is processed for the raw images acquired from the stereo camera, as the standard method Sobel operator combined with thresholding technique are adopted on the reference image (i.e. the one obtained from the right camera) to detect the horizontal edges. Note that we do not limit the edges within ROI in this case. To withdraw the potential outliers corresponding to moving objects and noise, the appropriate morphological operations have to be carried out on these detected edges. This edge detection process is similar to that represented in Chapter 5. Figure 6.4(a) shows the results of edge detection and it can be seen that the extracted edges are almost corresponding to static objects. For each edge, two pixel ends $[p_{e1}, p_{e2}]^T$ can be obtained. According to the calculated disparity between the images of two cameras in stereo vision system, they are projected into the 3D Euclidean space and the relevant 3D ends are $[P_{e1}, P_{e2}]^T$ (As for the algorithms of disparity computation and transformation between disparity and depth, please refer to the Bumblebee2 stereo camera SDK manual). With this pair of 3D ends coordinates, the 3D line is determined and therefore the related Plücker representation is described as

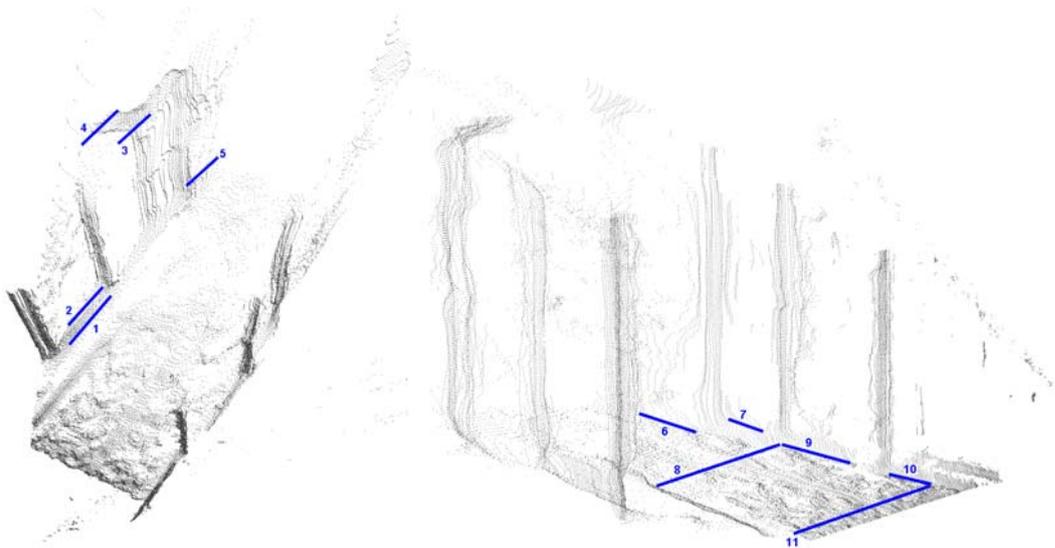
$$L_{pl} = \begin{pmatrix} \mathbf{u} \\ \mathbf{n} \end{pmatrix} = \begin{pmatrix} P_{e2} - P_{e1} \\ P_{e1} \times P_{e2} \end{pmatrix}, h = \frac{\|\mathbf{u} \times (-P_{e1})\|}{\|\mathbf{u}\|} \quad (6.11)$$

Note that before the computation of Plücker representation the 3D coordinates of the ends have to be converted into the local frame. Please refer to Appendix A on coordinate transformation of stereo camera for more detail.

Figure 6.4(b)&(c) display the extracted 3D segments from the 2D edges. After comparison of Figure 6.4(a) and Figure 6.4(b)&(c), it is evident that not all edges in images can be converted to a 3D lines. The reason is that the related disparity ranges between the left and right images exceed the predefined threshold. It is remarked that all 3D points have been transformed into the local coordinate system.

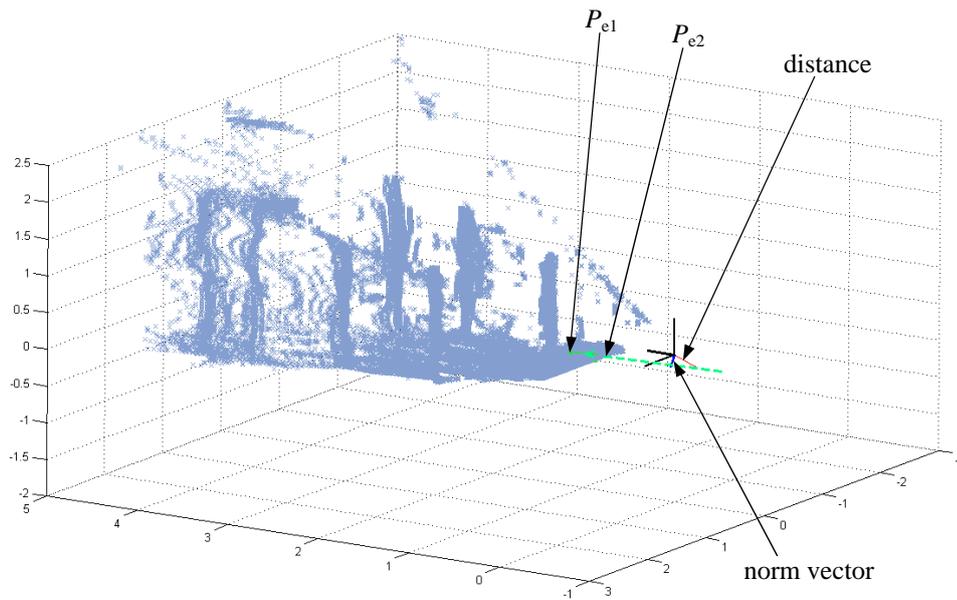


(a)



(b)

(c)



(d)

Figure 6.4 An example of segment extraction from stereo camera. (a) Results of edge detection by using Sobel operator and morphological operation; (b) and (c) Results of 3D points cloud and 3D lines corresponding to the edges in images; (d) Plücker coordinates representation for one of the 3D lines. Black: local coordinate reference. Green: solid is the direction (from P_{e2} to P_{e1}) of the line. Dash is the extension of the line. Red: distance from origin of reference to the line. Blue: normal vector.

6.4 Direct Sensor Fusion Management Based on Information Theory

Sensor fusion aims to integrate the measurements from distinct sensors loaded in one or distributed platform in order to decrease the feature uncertainty. The more a measurement from a certain sensor approximates to the true value, the more it has to be revealed more in other sensors. From this viewpoint, we borrow the idea of the decision analysis area (Ding & Shi, 2005; Zhou & Zhao, 2007), and develop a sensor fusion management based on information entropy weight, which fuses the sensor measurements, i.e. the parameters of Hessian line model and Plücker coordinates, and related variance of these parameters. The advantages of this strategy are no extra postulated conditions, generality and simple implementation.

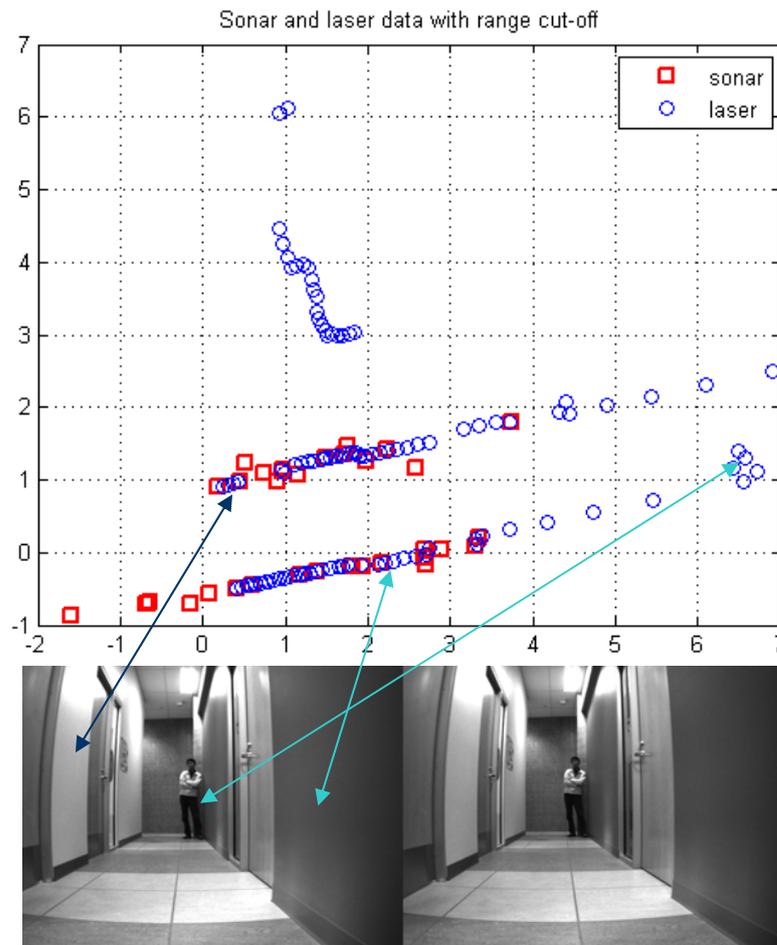
The reason we called the suggested fusion policy as *direct* is that the parameters of segments (obtained from rangefinder sensors) and lines (acquired from vision systems) straightforwardly contribute to compute the fused feature parameters, which is different from the strategy depicted in Chapter 5. We will use these fused features to serve for EKF-SLAM. For convenience, the fused feature parameters are marked as $[r_{fs}, \gamma_{fs}]^T$. As for the basic concepts of information entropy, please refer to Appendix C.

6.4.1 Information Entropy Based Associated Features Determination

Before implementation of proper sensor fusion, the associated segment features have to be determined. Because 20 robot sampling steps of sonar data have been stored into the sonar buffer, to make the laser measurements and camera images identical with the sonar data, it is necessary to carefully select the sampling time for getting the laser data and images. With reference to the data in sonar buffer, we select the sampling time for laser data and images acquisition as 2 seconds, i.e. 8 robot sampling times. As is shown in Figure 6.5(a), the various sensor measurements corresponding to the walls, doors and person are almost identical. Noted that the sonar data of 1st, 3rd~5th seconds in sonar buffer has to be transformed into the coordinate system where the robot pose is obtained at the 2nd second.

When sonar data update, in contrast to the earlier studies (Ip & Rad, 2004), we do not use feature tracking algorithm for consistent mapping but treat the buffered sonar data as a virtual laser rangefinder. As displayed in Figure 6.5(b), we move the sonar data located in the first three seconds out of the buffer and those in the latest three seconds into the buffer, which makes a new buffered data, and then we use this new buffered data to fuse with new sampled laser readings and the images. To make the explicit comparison, we also plotted the sonar data of previous

sampling time and those obtained by using two-out-two-in updated strategy. It can be seen that the sonar data updated by three-out-three-in updated scheme coincide with laser data better. From Figure 6.5, we can find our selected sampling time for laser and camera data almost maintains the data to coincide with sonar measurements, so does for sensor readings update strategy.



(a)

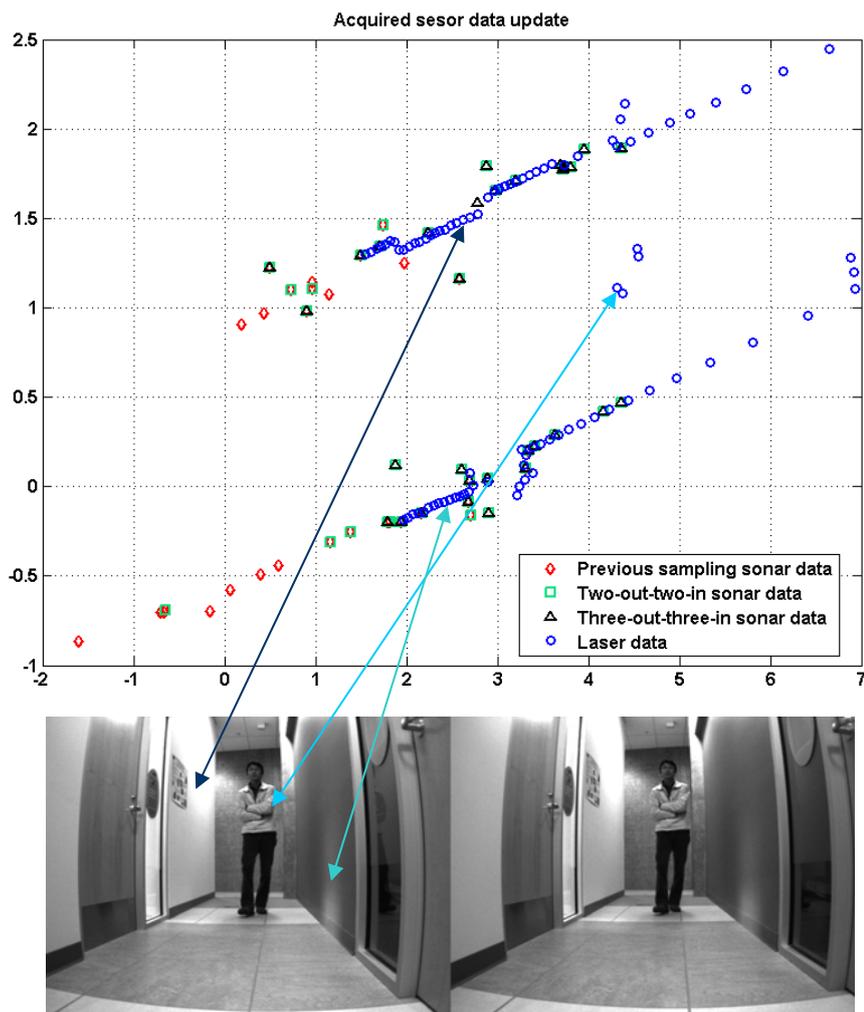


Figure 6.5 Identical sensor data in different sampling time. (a) 20 robot sampling sonar data and 8 robot sampling laser data; (b) updated sensor data in different sampling time. Red: previous sampling sonar; Green: Two-out-two-in sonar data of updated sonar buffer; Black: Three-out-three-in sonar data of updated sonar buffer; Blue: laser data.

After obtaining the sensor readings of the same moment, it is important to find the corresponding segments represented by sonar, laser and camera measurements so that the sensor fusion management can be implemented correctly. In this study, we design an associative segments determination algorithm based on information entropy, which can find the matched features from distinct sensors. The detailed algorithm is to be stated in the following paragraphs.

The calculation of the parameters for the segments extracted from different sensors has been depicted in Chapter 4 and previous sections of this chapter. We assume that these parameters have been converted in a proper form in the world coordinate reference and define them as P_s . Subscript s stands for sensor type, the available values of which in current study are sn (sonar), ls (laser) and ca (camera). We compute a set of distances of the parameters between one segment extracted from one sensor and all segments obtained from other two sensors. For example, we randomly select a segment extracted from sonar and we calculate two sets of distances. One is from this selected sonar segment to all segments extracted from laser rangefinder; the other is from this selected sonar segment to all from camera. Note that these two sets of distances are those of feature parameters. We name this type of distance as parameter error, and the set of errors for a certain sensor is

$$e_s^i := \{e_s^{ij} = |P_s^i - P_{s'}^j|, s \neq s'; j = 1, \dots, m_l\}, i = 1, \dots, n_l \quad (6.12)$$

where n_l is the number of features extracted from one sensor, and m_l is those from another sensor. The probability for each distance in e_s^i is estimated by Parzen window non-parametric estimator (Duda, 2001; Erdogmus & Principe, 2002) for 2-dimensional random variable. That is

$$p(e_s^i) = \frac{1}{N} \sum_{j=1}^N k(e_s^i - e_s^{ij}) \quad (6.13)$$

N is the number of elements in set e_s^i , $k(\cdot)$ is the bivariate kernel function, usually the bivariate Gaussian kernel function is selected i.e.

$$k(x, y) = \frac{1}{2\pi h_x h_y \sqrt{1-\rho^2}} \exp \left\{ -\frac{1}{2\sqrt{1-\rho^2}} \left(\frac{x^2}{h_x^2} - \frac{2\rho xy}{h_x h_y} + \frac{y^2}{h_y^2} \right) \right\} \quad (6.14)$$

where h_x, h_y are optimal bandwidths for the widths of Parzen window which are selected by the algorithm of Botev and Kroese (2008), and ρ is the correlation. When getting the probabilities sets, we can obtain the entropy for these errors as

$$H(e_s^i) = -\sum_{j=1}^N p(e_s^{ij}) \ln p(e_s^{ij}) \quad (6.15)$$

It is clear that removing an element e_s^{ij} randomly from the error set e_s^i will make the probability estimated from the rest of error elements definitely different. Following this idea, calculating the new probabilities with the same procedure for those remaining errors after removal of any one error, we can get a new entropy $H_k(e_s^i)$. Therefore, the variation of the entropy is

$$\Delta H_k(e_s^i) = |H(e_s^i) - H_k(e_s^i)| \quad (6.16)$$

Repetition of computation for variation of entropy until all of elements in e_s^i have been removed once in turn, we collect a set of variation of entropy $\Delta H = \{\Delta H_1, \dots, \Delta H_k\}$. The matched features are ones that make the following inequality satisfied

$$\Delta H > H_{th} \quad (6.17)$$

H_{th} is a threshold. This error entropy explains that the most approximate features in different sensors possess the most information on similarity.

6.4.2 Entropy Weight Fusion Management for Sensor Measurements and Variance

Suppose each of n sensors provides m measurements, and expressed as $S_i = [s_{i1}, \dots, s_{im}]^T$, $i=1, \dots, n$. Assume that the representations of these measurements have same form, i.e. Hessian line model parameter: r and γ and Plücker coordinates: h and u . We establish the fuzzy relation matrix according to min-max rule for these $n \times m$ sensor readings as

$$R = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nm} \end{bmatrix} = (r_{ij})_{n \times n}, \quad \text{where } r_{ij} = \frac{\sum_{k=1}^m (s_{ik} \wedge s_{jk})}{\sum_{k=1}^m (s_{ik} \vee s_{jk})} \quad (6.18)$$

\wedge and \vee mean minimum and maximum operation respectively. It is clear that each element of R falls into $[0,1]$ and R is a normalized matrix. Following the definition (Qiu, 2002), we calculate another type of entropy for i th sensor as

$$H_i = -k \sum_{j=1}^n f_{ij} \ln f_{ij}, \quad i = 1, 2, \dots, n \quad (6.19)$$

where

$$f_{ij} = \frac{r_{ij}}{\sum_{j=1}^n r_{ij}}, \quad k = \frac{1}{\ln n}$$

and assume that when $f_{ij} = 0$, $f_{ij} \ln f_{ij} = 0$. The difference between entropy (6.15) and (6.19) is on the probability calculation. The probability in equation (6.15) is estimated through Parzen window while that in equation (6.19) is computed by f_{ij} which is a ratio or frequency. These two formats of entropies are the distinct variants of the entropy definition (Cover, 2006). Furthermore, we obtain the entropy weight of i th sensor via

$$\omega'_i = \frac{1 - H_i}{\left| m - \sum_{j=1}^m H_j \right|}, \quad i = 1, \dots, n \quad (6.20)$$

and standardize (6.20)

$$\omega_i = \frac{\omega'_i}{\sum_{j=1}^n \omega'_j} \quad (6.21)$$

Equation (6.21) is the entropy weight for the measurements of each sensor.

Concerning the precision of the measurements for each sensor, we design a *subjective* index \mathcal{A}' which related to the variance of sensor measurements and has a larger value when the variance is smaller. To eliminate the influence induced by the difference of various quantity grade, we first normalize those variance and project them into $[\tau_2, \tau_1]$, $0 < \tau_1 < \tau_2 < 1$ before computing \mathcal{A}' . It should be noted that the projective space $[\tau_2, \tau_1]$ is designed for cost type index because the variance represents the accuracy of the measurements. After that \mathcal{A}' is calculated as the solution of

$$\max_{\substack{\mathbf{A}' \in E^n \\ \|\mathbf{A}'\|_2=1}} \sum_{i=1}^n ((\mathbf{A}')^T \boldsymbol{\zeta}_i)^2 = \sum_{i=1}^n (\boldsymbol{\zeta}_*^T \boldsymbol{\zeta}_i)^2 = \rho_{max} \quad (6.22)$$

where

$$\boldsymbol{\zeta} = \begin{bmatrix} \zeta_{11} & \cdots & \zeta_{1n} \\ \vdots & \ddots & \vdots \\ \zeta_{m1} & \cdots & \zeta_{mn} \end{bmatrix} \quad (6.23)$$

$\boldsymbol{\zeta}$ is the normalized variance matrix, and ρ_{max} is the maximum single positive eigenvalue of matrix $\boldsymbol{\zeta}^T \boldsymbol{\zeta}$; $\boldsymbol{\zeta}_*$ is the positive eigenvector corresponding to ρ_{max} . The equation (6.23) is the object function of Group Eigenvalue Method (Qiu, 2002), and it interprets that the expected \mathbf{A}' is equivalent to the optimal $\boldsymbol{\zeta}_*$.

We combine this index with the entropy weight of sensor measurements ω_i and derive a synthetic entropy weight for sensor measurements as

$$\lambda_i = \frac{\lambda'_i \omega_i}{\sum_{i=1}^n \lambda'_i \omega_i} \quad (6.24)$$

Therefore, the hybrid entropy weight vector for sensor measurements is $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_n]^T$, and the fused sensor measurements are

$$\mathbf{S}_f = \mathbf{S}_{(m \times n)} \boldsymbol{\lambda} \quad (6.25)$$

In accordance with equation (6.21) and the principal theory of information entropy, the smaller the entropy of a sensor, the larger the entropy weight is, and it claims that the sensor contributes more important and useful information for sensor fusion. The reason why the effect of the variance for sensor readings is incorporated with entropy weight of sensor measurements (cf. equation (6.24)) is that the variance is taken as a kind of evaluation index and reflects the quality of the sensor information. This strategy also seems like the integration of a subjective expert index with an objective index in the view of decision analysis.

As for the fusion of measurements variance, we develop a modified covariance

intersection (CI) on the basis of Julier and Uhlmann's work (Julier & Uhlmann, 2001), which draws the entropy weight λ_i into the basic CI. The relevant optimal problem becomes

$$\begin{aligned} \min f(w) &= \frac{1}{\det(\sum_{i=1}^n w_i \lambda_i C_i^{-1})} \\ \text{s.t. } \sum_{i=1}^n w_i &= 1 \\ 0 &\leq w_i \leq 1 \end{aligned} \quad (6.26)$$

Hence, the new variance after fusion is

$$C = \left(\sum_{i=1}^n w_i \lambda_i C_i^{-1} \right)^{-1} \quad (6.27)$$

Similarly, the modified CI management for variance fusion simultaneously considers the measurement contribution and the quality of the sensor information and in a sense avoids the loss of information in each local sensor. Additionally, the proposed sensor fusion management prevents the case of sensor failure, for example when detecting a glass object the laser and camera may lost the data but the sonar works as well as normal situation. In this case, the synthetical entropy weights corresponding to invalid sensors, such as laser or camera when detecting glass objects, approximate to zero which is induced by the objective or subjective index. Therefore, the measurements from these invalid sensors would not be considered into the fusion process. We will explain more about this idea in the following experiments.

6.4.3 Practical considerations

It is common that the 3D lines extracted from stereo camera do not locate within the same plane as the segments extracted from ultrasonic sonar and laser rangefinder. To make a reasonable sensor fusion, it is necessary to convert the parameters of 3D lines to those described in the plane which the sonar and laser segments lie in. This conversion is a simple geometrical transformation problem,

and if the 3D line exactly falls in the same plane as sonar and laser segments the converted parameters respectively are h and one sub-vector of direction vector \mathbf{u} .

The computation of the covariance of parameters for the 3D stereo vision lines is a elaborate process because each 3D point $[x_P, y_P, z_P]^T$ has its own variance which is calculated as

$$\Delta x = \Delta y = \frac{pz}{f}, \quad \Delta z = \frac{fBas}{d-m} - z, \quad d = \frac{fBas}{z} \quad (6.28)$$

where p is the calibration accuracy, m is the matching accuracy, f is the focal length in pixels, Bas is the base line in meters of the stereo camera and d is the disparity in pixels. After obtaining these variances for each coordinate, we use Taylor series expansion to calculate the covariance of the Plücker coordinates and related converted parameters, which is a little bit complex computation but tractable. The similar procedure of the covariance computation is applied for the parameters of the line obtained from monocular camera. For another covariance computation of laser segments, we firstly filter out the data of which the range is over 8 meters, then extract the Hessian line model parameters, and derive the corresponding covariance. This is because the variance for laser reading is measured under the condition that range is less and equal to 8 meters.

There exists a similar situation as stated in Chapter 5, i.e. it is inevitable to extract the pseudo segments from laser readings or buffered sonar data. Fortunately, in accordance with the proposed algorithm of associative features determination in Section 6.4.1, the plausible segments may be excluded. This is because the error entropy produced by those features may not surpass the configured threshold. In current study, only the features existing concurrently in three sensors are applied for subsequent sensor fusion.

6.5 Experimental Studies

We divided the experimental studies into two scenarios. Firstly the virtual rangefinder model is tested and then the sensor fusion management for SLAM in both simulation and real environments is validated.

6.5.1 Validation on Virtual Rangefinder Model

The intrinsic parameters of the Canon VCC4 monocular camera were listed in Table 5.1. With these calibration parameters and equations (6.6) & (6.8), we extracted the ranges and bearings of some randomly selected points in the control lab of Department of Electrical Engineering. Figure 6.6 shows the positions of the numbered points. Table 6.1 enlists the results compared with the hand-measured data. It can be seen from Table 6.1 that the errors of the range are within 5% wherever these points located. However, the deviation on bearings is slightly higher, around 10%. This is because the bearing is not directly detected but computed by measuring the two sides of the angle. This error will not severely affect the Plücker coordinates calculation. To manifest this idea, we did another experiment for computing the Plücker coordinates from the range and bearing data. The place is the corridor outside the control lab and four segments were extracted from the image, which is displayed in Figure 6.7. The range and bearing of each segment end were computed according to the virtual rangefinder model and the relevant parameters for computing Plücker coordinates are listed in Table 6.2. Compared with the hand-measured data, the error of the distance h is around 3.5% which closes to the error of virtual ranges and the maximal error for the angle α approximates to 5% that is smaller than that of the virtual bearing.



Figure 6.6 Positions of measured points in control lab

Table 6.1 Range and Bearing Comparison

Point	Measured		Calculated		Deviation	
	$l(m)$	$\varphi(rad)$	$l(m)$	$\varphi(rad)$	$l(\%)$	$\varphi(\%)$
1	1.29	0.330	1.28	0.314	0.7	4.8
2	1.91	0.261	1.82	0.236	4.7	9.6
3	2.13	0.266	2.03	0.240	4.7	9.7
4	1.20	0	1.17	-0.02	2.5	≈ 0
5	1.52	-0.305	1.54	-0.330	1.3	8

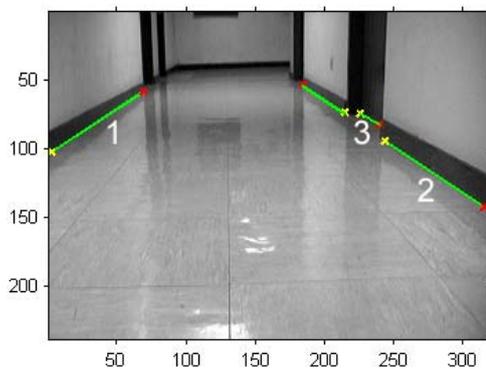


Figure 6.7 Obtained lines from the monocular camera outside the control lab

Table 6. 2 Lines parameters comparison

Line	Measured		Calculated		Deviation	
	$h(m)$	$\alpha(rad)$	$h(m)$	$\alpha(rad)$	$h(\%)$	$\alpha(\%)$
1	0.74	$\pi/2$	0.7642	1.6407	3.3	4.5
2	0.74	$-\pi/2$	0.7616	-1.6206	2.9	3.1
3	0.86	$-\pi/2$	0.8878	-1.6228	3.2	3.3

6.5.2 Validation on Direct Sensor Fusion Management

The robot platform applied in the following experiments is different from the previous chapters. It is the Pioneer 3DX mounted with a Bumblebee2 stereo camera, a SICK LMS200 laser rangefinder and a 16-sonar array. The mobile robot was wandering in the environments with an average speed of 300mm/s. SICK LMS200 has a maximum measurement range of 80 m, a range resolution of 10 mm and a statistical error standard deviation of 5 mm at normal reflectivity condition (for one sigma value). For technical specification detail, please refer to the *Technical Information LMS200/291. SICK, Inc.* In our experiments, we use a maximum scan range of 8.0 m, an angular resolution of 1° . For the sonar, the range deviation is on the order of 25cm (for one sigma value), but can vary with temperature, so it is best to give a larger value. Generally this deviation is set to be about 70 cm. The SLAM algorithm is still EKF and the relevant time update and measurement update equations have been explained in Chapter 4. It has to be noted that the state variables and covariance of the measurement are the fused form, i.e. equation (6.25) and (6.27). Other parameters for experimental studies are listed in Table 6.3.

Table 6.3 Parameters in experimental studies

Item	Value
Robot sampling time	250ms
Sonar range cut-off limit	1.5m
Sonar buffer size	20 robot sampling time (5s)
laser range cut-off limit	8m
laser & camera sampling time	8 robot sampling time (2s)
p	0.04
m	0.05
f	245.714
Bas	0.12m

Case 1 Validation on sensor fusion management in simulated environment

We tested our sensor fusion method only for sonar and laser readings in this

simulation. The simulated environment is a simple office with width 5m, length 7m and the origin (1,1), in which there is only one separator with length 1.5m and origin (3.15,5) and no other moving object except the mobile robot, which is illustrated in Figure 6.8(a). The robot simulator is the MobileSim provided by MobileRobots Co. and ran one loop (cf. Figure 6.8(b)). The origin of the world coordinate is (0,0). There are 5 segments in this simulated office and as is shown in Figure 6.8(a) they are marked in numbers. The feature parameters and variance before and after sensor fusion are listed in Table 6.4, where the fused information is calculated by equation (6.25) and (6.27). All the segment parameters have been transformed into the world reference. It can be seen that more information of laser data are reflected in the fused parameters and variance. This is because that the precise laser data have larger entropy weight according to our proposed fusion management. In comparison with the true parameters of segments, it is obvious that the fused parameters close to the true value and almost have smaller variance than that of each sensor especially better than the results obtained from sonar data. For segment 4 and 5, the measured angle is around $-\pi$ which is different from the true value. Actually, it is just a transformation of angle range, and we maintain the original results here for this explanation.

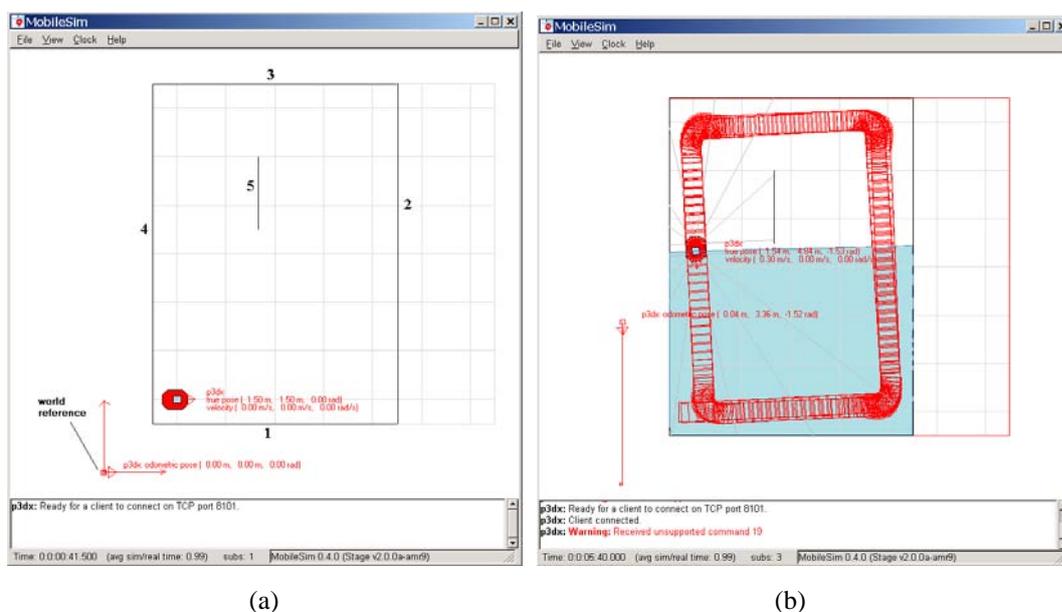


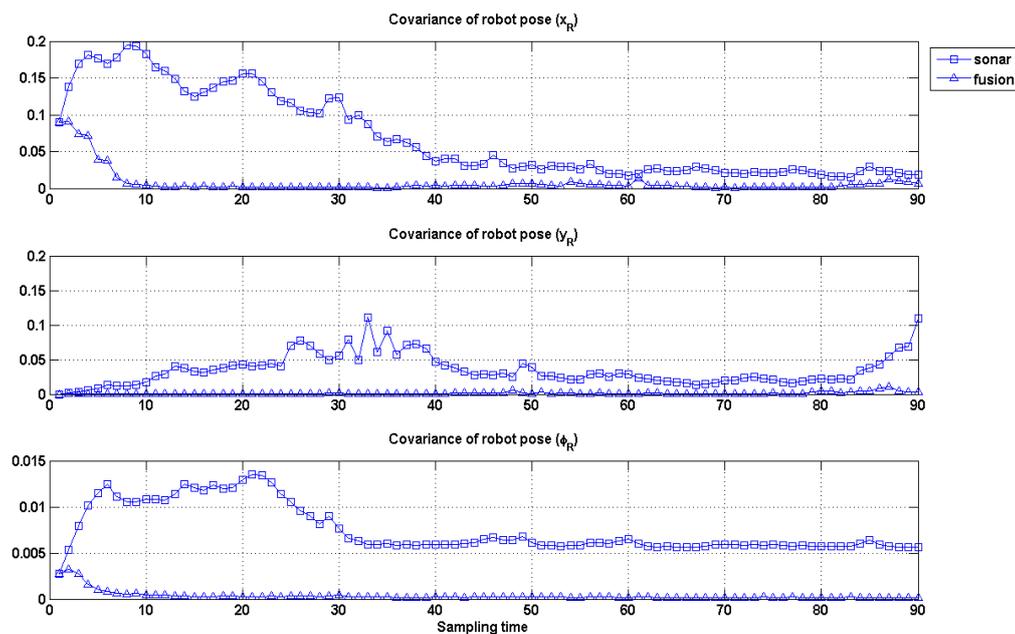
Figure 6.8 EKF SLAM based on proposed sensor fusion strategy in simulated environment.

(a) Simulated environment with numbered segments; (b) One closed loop trajectory.

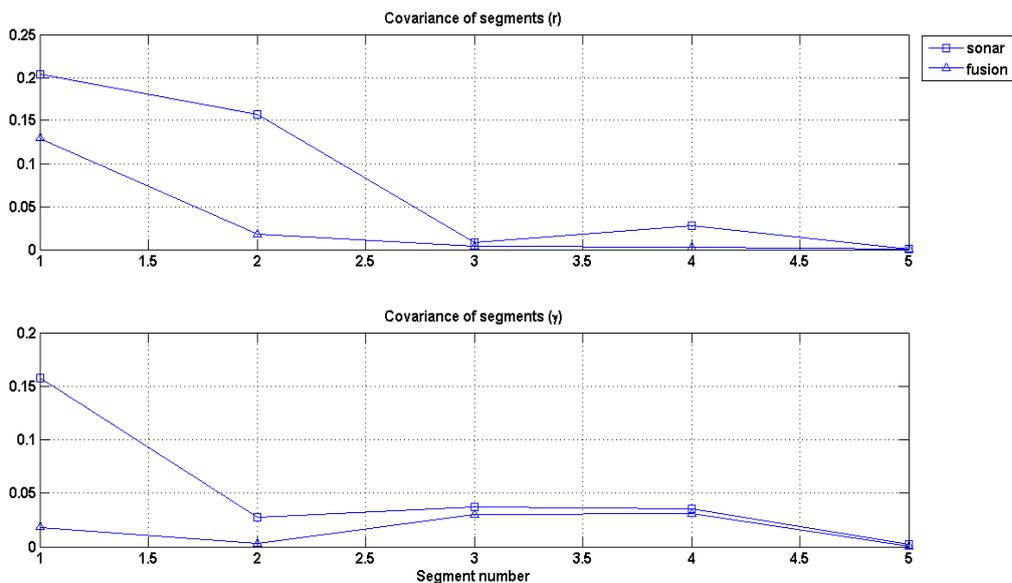
Table 6.4 Results before and after sensor fusion

segment NO.	true value (m, rad)	sonar $[r, \gamma]^T$		laser $[r, \gamma]^T$		fused $[r_{fs}, \gamma_{fs}]^T$	
		para	var	para	var	para	var
1	1	0.9394	0.2597	0.9421	0.0016	0.9415	1.59E-4
	$\pi/2$	1.6234	0.0454	1.6236	0.0002	1.6236	1.996E-5
2	6	6.0230	0.3091	6.0436	0.0005	6.0395	0.4789E-3
	0	0.0572	0.0797	0.0530	0.00003	0.0538	0.0303E-3
3	8	8.5759	0.2789	8.5929	0.0033	8.5895	0.0033
	$\pi/2$	1.4558	0.1462	1.4503	0.0003	1.4514	0.0003
4	1	1.5194	0.7315	1.1874	0.0117	1.2538	0.01165
	0	-3.0212	0.3035	-3.0225	0.0011	-3.0222	0.01089
5	4.15	4.1694	0.1774	4.1485	0.0003	4.1527	0.2824E-3
	0	-3.0240	0.125	-3.0229	2e-5	-3.0231	0.0289E-3

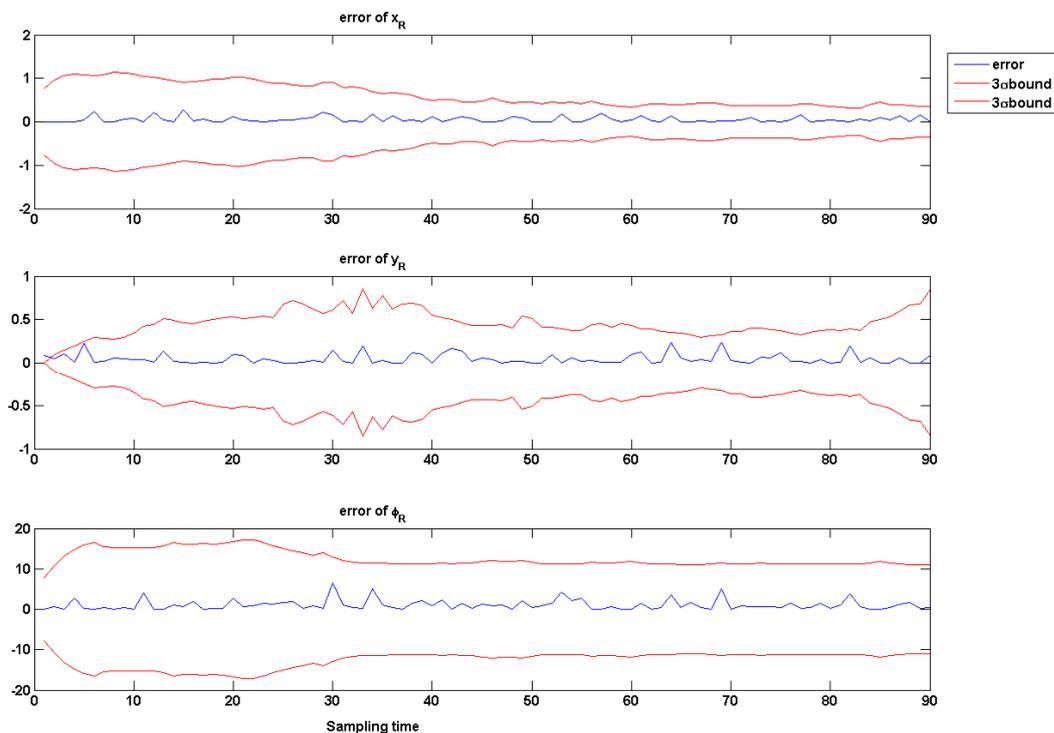
We incorporated the sensor fusion results into the EKF-SLAM procedure and compare the results with only one sensor. To make the comparative results clearly, here we applied sonar array as the perception system for SLAM. Figure 6.9(a)&(b) shows the results of the covariance for robot pose and features without and with sensor fusion strategy, which obviously demonstrates the covariance decreases after implementation of sensor fusion. Figure 6.9(c) also indicates that the consistency of robot pose is almost maintained when the sensor fusion is employed.



(a)



(b)



(c)

Figure 6.9 Comparison on the covariance of robot pose and feature parameters, and consistent validation for robot pose when using proposed sensor fusion management. (a) Comparison results on covariance of robot pose; (b) Comparison results on covariance of feature parameters; (c) Consistent validation on robot pose.

We used the laser readings of the 54th laser sampling time and related sonar

measurements to interpret our associative feature determination strategy. Figure 6.10 displays the raw readings of two sensors in the local reference framework at the 54th laser sampling time, in which there exist segments 1, 3, 4 and 5. Sonar data depict segments 3, 4 and 5, and laser data have segments 1, 3, 4 and 5. As an example, we selected segment 5 in sonar reading and attempted to find the matched segment in laser data. After the feature parameters are transformed into world framework, the errors between segment 5 in sonar data and all segments in laser data are listed in Table 6.5, and the total entropy and other entropy without any one of errors are computed following equation (6.15) and shown in Figure 6.11. It can be seen from Figure 6.11 that the change of entropy is [0.2915, 0.6071, 2.995, 0.4587]. If the threshold for the entropy variance is set to be 1, ΔH_3 is selected. That means without e_{sn}^{55} the entropy changes most seriously, which claims that the sonar segment 5 is associated with laser segment 5.

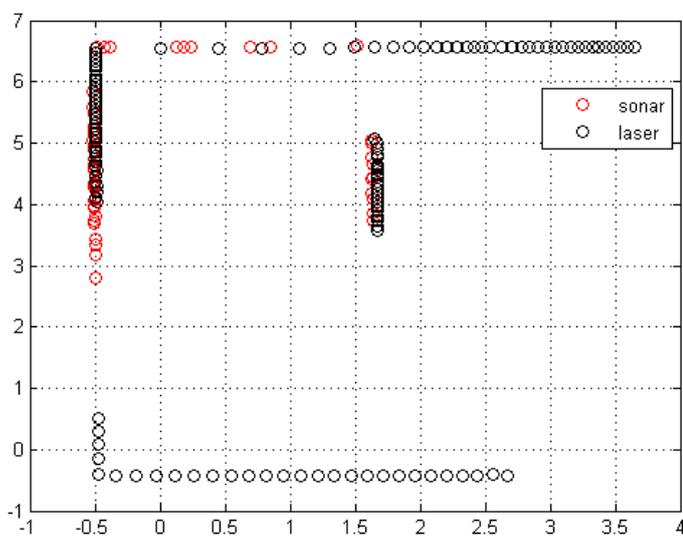


Figure 6.10 The sonar and laser readings at the 54th laser sampling time.

Table 6.5 Errors between sonar segment 5 and all laser segments

$para \backslash error$	e_{sn}^{53}	e_{sn}^{55}	e_{sn}^{54}	e_{sn}^{51}
r	3.9236	0.0209	2.1820	3.0602
γ	4.4743	0.0012	0.0015	4.4757

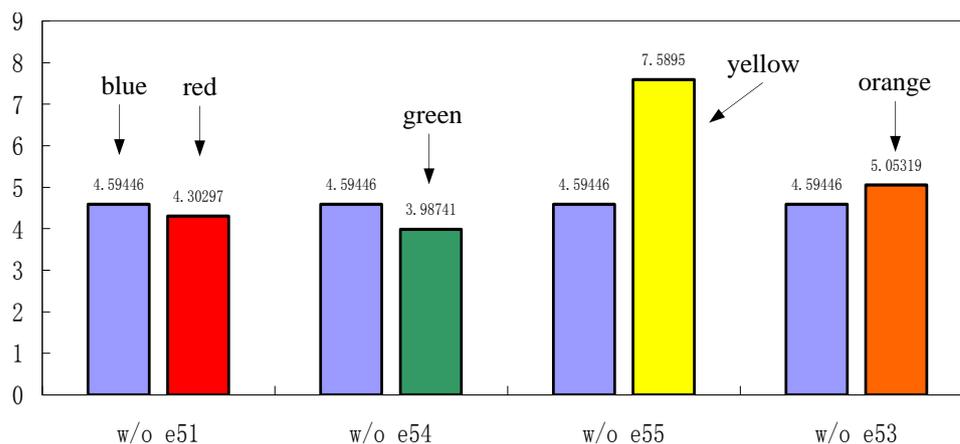


Figure 6.11 The entropy comparison bar. Blue: total entropy; Red: the entropy without e_{sn}^{51} ; Green: the entropy without e_{sn}^{54} ; Yellow: the entropy without e_{sn}^{55} ; Orange: the entropy without e_{sn}^{53}

Case 2 Validation on sensor fusion management in real environment

It reveals from the simulation results in *Case 1* that even for the sensors whose measurements have large errors the suggested sensor fusion management can still provide a reasonable results. It is feasible to believe that the sensor fusion policy may efficiently incorporate the feature parameters acquired from the monocular camera with those obtained from other sensors. Therefore, in the following experimental validation, we will mainly focus on the integration of stereo vision system with other rangefinder sensors because obtaining the depth is much easier. We tested our sensor fusion management in a narrow corridor outside Robotics and Automation Lab, and a wide corridor with several glass walls on 4th floor in Simon Fraser University Surrey Campus. Several persons walked in the corridors at average speed when the robot was moving. The mobile robot also wandered a closed loop. After collecting and preprocessing the various sensor data, we validated our method in MATLAB platform.

Similar to the simulation experiments, we also test the sensor fusion management for sonar, laser and camera readings. As an example, we selected the laser data, images and related sonar data which are obtained in narrow corridor at the 2nd laser sampling time for fusion. Figure 6.12 illustrates the labeled segments extracted from sonar and laser measurements. The lines extracted from images

have been numbered in Figure 6.4(b)&(c). The parameters for these features are listed in Table 6.6 and they are represented in local framework. Note that some laser segments have been merged following the criteria represented in the work of Ip et al. (2002). Figure 6.13 illustrates the schematic diagram for matching feature procedures according to our proposed criteria. It can be seen that some segments represented in images are merged firstly and then these merged features are considered as the proper ones for feature matching. We can also find that some features reflected in laser and camera measurements are not matched. This is because some of them are false features such as in laser data segments 3, 5 and 6 representing the objects behind the glass door, which can not be used for SLAM. Another reason is that the complementary information between distinct sensors is not applied for example the segment 4 and 7 reflected in images. In current study, we applied a plain way to handle those features in images, i.e. treat them as available features for SLAM. The topic of processing complementary information will be further studied in the future work. Table 6.7 catalogues the results of sensor fusion.

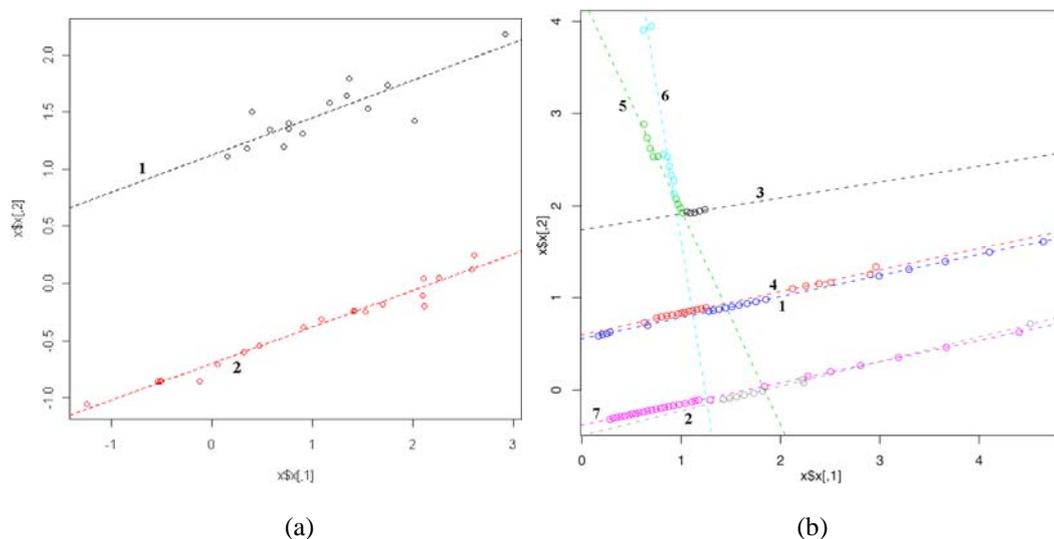


Figure 6.12 Segments obtained from sonar and laser. (a) Sonar segments; (b) Laser segments

Table 6.6 Parameters for the segments from different sensors

Segment NO.	sonar $[r, \gamma]^T$		Segment NO.	laser $[r, \gamma]^T$		Segment NO.	camera $[h, u]^T$	
	para	var		para	var		para	var
1	0.8372	0.1824	1	0.8379	8.89E-04	1	0.8026	1.63E-07
	1.8175	0.0521		1.8041	7.23E-05		1.5534	1.20E-06
2	0.5617	0.0455	2	0.5702	4.13E-04	2	0.7624	3.67E-06
	-1.7769	0.0127		-1.525	7.13E-05		1.4339	3.62E-05
			3	2.6625	0.5588	3	0.8618	1.40E-05
				1.7471	0.1151		1.6729	9.57E-06
			4	0.9106	0.003	4	0.4223	1.64E-05
				1.8063	0.0006		1.3469	1.16E-05
			5	2.5389	0.0296	5	0.7691	1.04E-05
				0.3947	0.0034		1.4729	4.17E-05
			6	1.9314	0.0198	6	0.4925	4.40E-05
				0.1693	0.0013		-1.4297	4.95E-05
			7	0.7338	0.0033	7	0.554	6.90E-06
				-1.8416	0.0003		-1.3329	2.33E-04
					8	3.2015	2.81E-05	
					9	-0.1789	3.59E-05	
					10	0.6611	1.39E-06	
					11	-1.5807	3.24E-05	
						0.5436	1.00E-07	
						-1.4374	1.08E-04	
						0.5796	1.18E-05	
						-0.2063	1.53E-05	

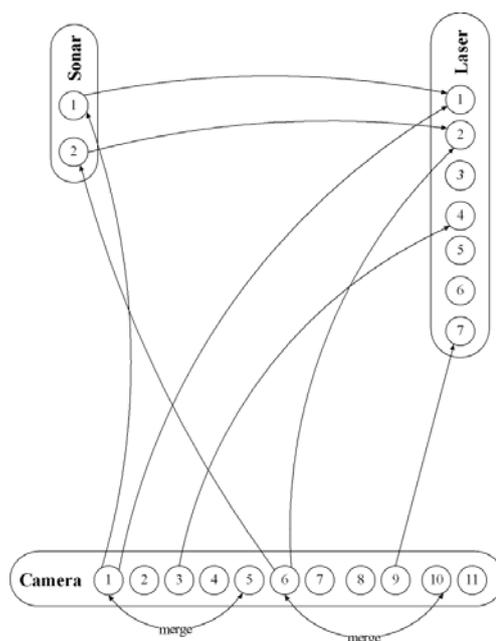


Figure 6.13 Associative features

Table 6.7 Fused parameters

Segment NO.	1 ($1_{sn}, 1_{ls}, 1_{ca}$)	2 ($2_{sn}, 2_{ls}, 6_{ca}$)	3 ($4_{ls}, 3_{ca}$)	4 ($7_{ls}, 9_{ca}$)
para	0.8191	0.5522	0.8813	0.6756
	1.6725	-1.4998	1.7263	-1.6329
var	0.0326e-5	0.8107e-4	0.1426e-4	0.014e-4
	0.2369e-5	0.6362e-4	0.0948e-4	0.3238e-4

Another case usually occurs in the indoor environments. As is shown in Figure 6.14(a), there is a glass wall in front of the robot when it was moving in the wide corridor of the 4th floor of SFU Surrey campus. Several edges can be detected from the reference image (cf. Figure 6.14(b)), but the camera can not detect any useful depth information (cf. Figure 6.14(c)). This is because there is hardly any disparity corresponding to the detected edges. In addition, all laser readings are zeros or default maximum as the light passes through the glass. In our current case, the laser readings are zeros. However, the sonar measurements survive and can still percept the glass objects which is shown in Figure 6.14(d). This is also a kind of complementary measurements. To use the entropy weight fusion method correctly, we assign the N_{sn} weights of measurements from camera and laser as 0 where N_{sn} is the number of features extracted from sonar data, and the weight for the sonar readings in this case is 1. That means the features extracted from sonar are incorporated into SLAM directly. From this viewpoint, the proposed sensor fusion management can dispose the situation in which the sensor is invalid.

The complemented maps are displayed in Figure 6.15 where the glass walls can be detected by sonars and they make the maps completeness. We also compared the covariance of robot pose before and after the implementation of sensor fusion. We carried out the EKF-SLAM by only using laser data, and it is obvious that the covariance of robot pose increase when the laser detected the glass objects which is shown around at the 40th, 100th and 140th sampling time in Figure 6.16(a), at the 100th-150th and 220th-260th sampling time in Figure 6.16(b). While by applying the sensor fusion management, the covariance avoids growing. This is because the

sonar can correctly measure the glass objects.

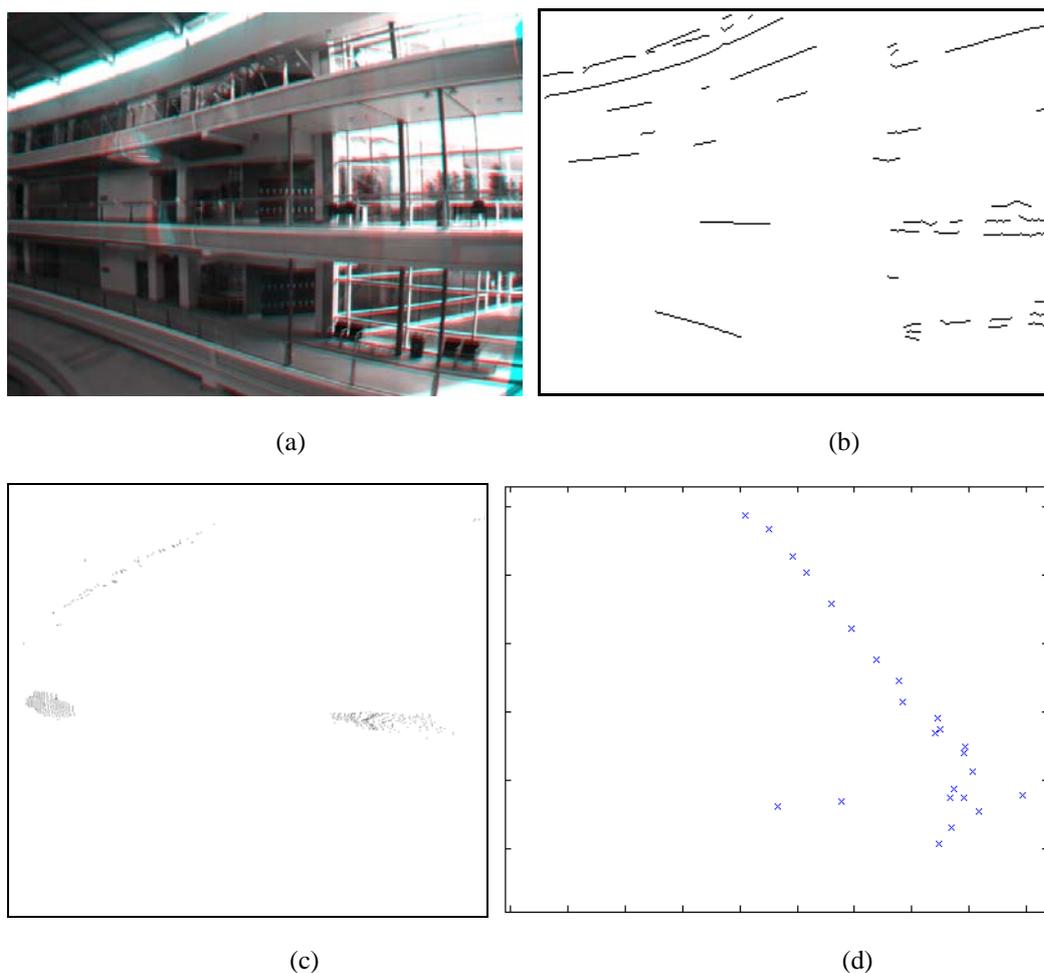


Figure 6.14 A wide glass wall in front of the robot in the wide corridor, and can not be detected by camera but can be done by sonar. (a) The raw image of glass wall; (b) The detected edges; (c) The depth information for the glass wall; (d) The measurement in sonar buffer.

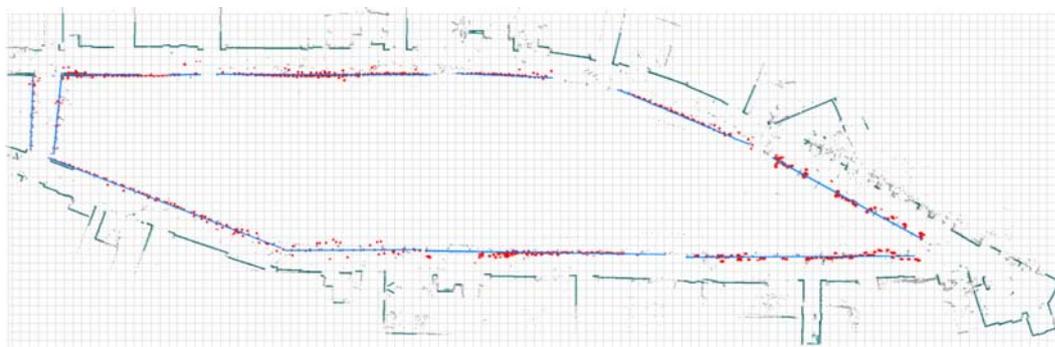
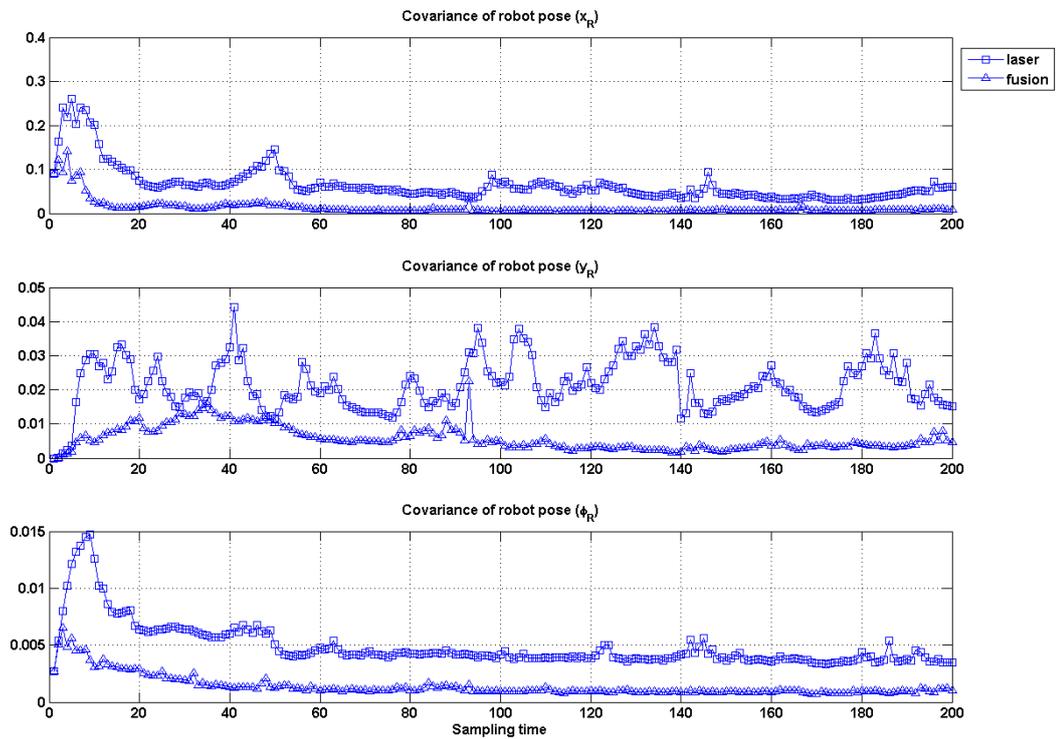
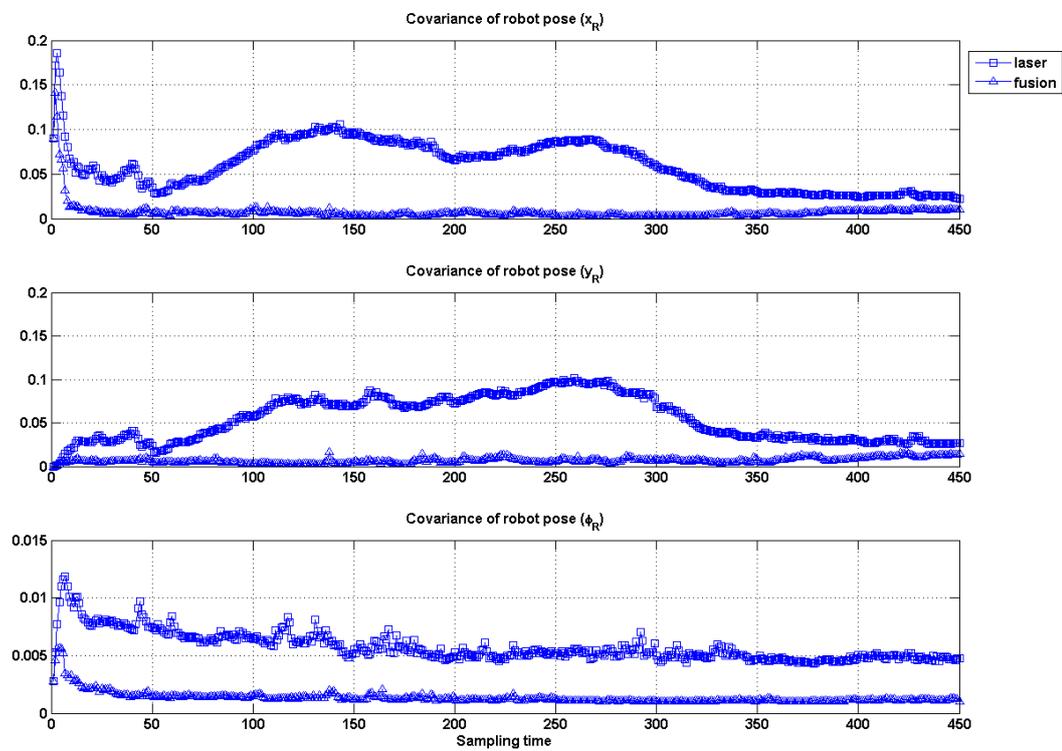


Figure 6.15 The final map after combining the measurements of sonar, laser and camera. To make the results readable, we folded two layers for map representation. The bottom layer is the raw laser and sonar data in light gray, the upper layer is the segments incorporating sonar, laser and camera readings (green lines) and the complementary sonar segments related to the glass walls (blue lines,

red points are raw sonar data).



(a)



(b)

Figure 6.16 Comparison on the covariance of robot pose. (a) In the narrow corridor;

(b) In the wide corridor.

6.6 Discussion

A direct sensor fusion management is formulated on the basis of information theory. We borrow the idea of the entropy weight from the decision analysis area and design a simple and general entropy weight method which considers the measurements and measurement variances to integrate the sensor information from various devices. It directly combines the feature parameters represented in the similar form i.e. Hessian line model for segments extracted from rangefinders and Plücker coordinates for the lines from vision systems. Similarly, the entropy weight is introduced into the covariance intersection. A modified CI technique is proposed for fusing measurements covariance matrices of different sensors. These fused features and relevant covariance matrices are incorporated into EKF-SLAM procedure to reduce the uncertainty of the features furthermore improve the efficiency of the SLAM especially for the case that some of sensors are invalid. A possible disadvantage of this fusion management is the ingenuous strategy of complementary information processing. In this study, this information reflected in camera is simply added into the SLAM procedure as the available features, which may import the potential invalid features. The topic of how to apply the complementary information appropriately will be studied in future. We also exploited the information entropy to design an associative feature determination strategy via calculating the error entropy to find the matched features. However, this method has enormous computation when the number of features to be matched is significant. The future works will focus on releasing the computational complex or developing another proper information entropy index.

When the objects are constructed by glass, the camera and laser can not detect anything but only sonar can reflect the useful measurements. That is why the combination has to include the sonar data. The sensor fusion algorithm in this chapter is the improvement of that presented in Chapter 5. These improvements are: All laser measurements are used for sensor fusion without considering the

limit of FOV; the representation for the features extracted from the camera is a compact form; when the implementation of feature fusion, no extra prior knowledge such as the prior probability in the Bayesian fusion method is required; the computational complexity of modified MPEF is relaxed. In Chapter 5, the modified MPEF has to implement two individual EKF-SLAM procedure and concurrently several previous state variables and covariance should be saved for fusion, and the fused values have to be stored and propagated backward to each individual EKF-SLAM. Additionally, for laser SLAM before measurement update, the laser segments have to be updated by feature fusion to remove the pseudo segments. Those procedures above are a little bit laborious. However the sensor fusion method described in this chapter releases the computational complexity. It directly fuses the feature parameters and adopts only one EKF-SLAM procedure which is similar to the presentation in Chapter 4. Although we have not done the experiments with the same sensory system of Chapter 5 separately, through other experiments in this chapter and the improvements above the direct sensor fusion method based on information theory probably provide a better performance.

As for Plücker coordinates computation for the lines extracted from the monocular camera, a virtual rangefinder model is proposed. This model alleviates the deviation problem caused by unparallel motion of the optical axis. However, the limitation of the proposed model is that the features should be on the flat-floor and the external knowledge on the camera installation has to be required. In future research, we will relax these constraints by employing such as active vision mechanism in computer vision community.

Chapter 7 Conclusions

The preceding chapters reported the author's contribution in the field of autonomous mobile robots. The main objective of this research was to investigate the problem of simultaneous localization and mapping (SLAM) in dynamic environments. The author has proposed viable solutions to this problem by borrowing concepts from robust regression, graph theory, computer vision, sensor fusion, and information entropy methodologies. The proposed algorithms eliminate the unexpected effects caused by dynamic objects and improve the performance of the EKF based SLAM. The first stage of the research identified an appropriate perception system, formulated the model of the environmental representation, and arrived at a proper mapping methodology by comparing existing methods. The robust regression model has been exploited to remove most of the features corresponding to moving objects. As for removal of those pseudo features related to temporary stationary dynamic elements, the indirect sensor fusion strategy has been employed. Meanwhile, the direct sensor fusion management enhanced the efficiency of the indirect strategy by integrating the feature parameters straightforwardly based on information entropy. In addition, the mathematical proof of optimal data association approach that also enhances the accuracy of SLAM in dynamic environments is provided.

In this chapter, we reflect on the findings and provide a brief summary of this research work along with major contributions. The author will conclude the chapter by providing some suggestions on possible extensions and future developments of this research.

7.1 Major Contributions

The author provided an applicable solution to SLAM in dynamic environments

through robust technique and sensor fusion viewpoint. Major contributions that deserve consideration in this research work are outlined as:

- A robust regression model for mapping to manage most dynamic objects was proposed in Chapter 4. This model applied the MM-estimate that combines a high breakdown point with good efficiency to address the sensor noise and dynamic items concurrently without requiring any special functions for noise and dynamic properties. This robust regression model alleviates the difficulty of thresholds selection for clustering and requires less time to extract the feature parameters.

- An indirect sensor fusion strategy including feature fusion and a modified Multi-sensor Point Estimation Fusion (MPEF) was synthesized in Chapter 5. Feature fusion policy incorporates the static line features extracted from monocular camera with the segments represented by robust regression model from laser sensor, the purpose of which is to delete the potential pseudo segments corresponding to moving objects in laser measurements. The parameters describing the features obtained from distinct sensors were imported into the fusion center to determine misidentified laser segments through Bayesian fusion rule. After removal of those pseudo laser features, the modified MPEF was employed to integrate two individual SLAM procedures — monocular and laser SLAM — and feedback the fused state estimates to each SLAM. The aim was to reduce the covariance of the state variables and improve the accuracy of localization.

- The author also proposed a direct sensor fusion management based on information entropy weight in Chapter 6. Different from the indirect strategy, the proposed technique combines the parameters of the features as well as related parameters variances obtained from various sensor measurements in a straightforward fashion in order to reduce the feature uncertainty. The fused

features are consolidated into the EKF-SLAM framework and the uncertainty of the robot pose evidently decreased in comparison of the results without considering the sensor fusion. As a part of the direct sensor fusion management, the method for the determination of associative features extracted from different sensors is also on the basis of information theory. It concentrates on computing the error entropy and confirms the relevant features through this error entropy. The advantages of this sensor fusion strategy are no extra postulated conditions, generality and simple implementation.

- The data association problem was revisited by presenting an optimal graph approach in Chapter 4. It is first formulated as a 0-1 integer programming (IP) problem, and then is mathematically proved that the IP problem is equivalent to a minimum weight bipartite perfect matching problem which can be optimally solved. Hence obtaining the optimal solution of the minimum weight bipartite perfect matching means optimally solving the data association problem. As for the data association in monocular SLAM, in Chapter 5 we suggested a technology using the homography transformation matrix (HTDA) estimated by the matched points in two images. These matched points are determined by SIFT descriptors. This method is generated from the idea that each 3D point has only one projection in the image plane, and to confirm the matched points in 3D space is equivalent to find the identical 2D image points. HTDA relaxes the repetitious process induced by the computation pixel by pixel in the predefined region.

- A virtual rangefinder model was introduced in Chapter 6 for extracted lines from a monocular camera. This model alleviates the deviation problem caused by unparallel motion of the optical axis, as well its simple geometric calculation makes it as an un-delayed method to recover the depth information.

7.2 Suggestions for Future Research

It is certainly impossible to address all the problems associated with the SLAM in dynamic environments within the limited scope of this research study. The research findings, however, has opened new path to follow. Here is some direction to future research in this field:

- The clustering algorithm for robust regression model can be studied further. One of the proposed clustering methods applied the variance ratio of slope (VROS) to detect the generation of a new cluster. This could produce the incorrect class when the sensor data are scrambled which is the case that numerous dynamic objects such as a crowd movement in the environment. Additionally, the inherent disadvantage of robust regression model is that the breakdown point can not exceed 50%, which makes this model may not dispose the mapping for considerably complex dynamic environments. Therefore the incorporation of robust regression model with other more robust techniques can be explored.

- The improved approach on solving the minimum weighted bipartite matching for graph theory based data association algorithm can be further studied. The way employed in this thesis is the Johnson's algorithm whose running time is $O(V^2 \log V + VE)$, where the V and E mean the numbers of vertices and edges. When the capacity of the vertex and edge becomes enormous, Johnson's method has low efficiency. To address this problem, the fast algorithm will be investigated. Furthermore, to reveal the performance of the proposed data association algorithm, more methods will be compared with the graph theoretic approach.

- A sensible module on processing the complementary information that exists in different sensor devices has to be developed and incorporated into the

proposed information theory based sensor fusion management. This makes the fusion results much more practical. Meanwhile, the suggested sensor fusion strategy based on information entropy could be extended to other area of autonomous robots such as exploration, navigation, distributed sensor fusion of multiple mobile robots, etc.

- The computer vision techniques will be considered to serve for the 3D information recovery from the monocular vision system. In this thesis, we assumed that the features have to locate at the flat floor with prior known external parameters of the camera when we extracted the lines from the monocular camera. In practice, these assumptions lead to lost more information reflected in the images, and could not always maintain. To considerably make use of abundant image information, the methodology in computer vision community such as active vision and structure from motion could be applied for implementing the autonomous tasks of mobile robots only with monocular vision system.

7.3 Closing Remarks

As a central ability of autonomous machines, SLAM plays an important role for autonomous robots to implement various advanced tasks in unstructured dynamic environments without continuous human supervision. The thesis has attempted to add value to the research on this open topic, and the author hopes that his modest efforts in the design of the method for SLAM in dynamic environments could be a contribution towards achieving autonomous tasks. An autonomous robot has a high degree of self-sufficiency and is able to explore and seek information about its environment; work for an extended period without human intervention; move either all or part of itself throughout its operating environment without human assistance; avoid situations that are harmful to people, property, or itself unless those are part of its design specifications. It may also learn or gain new capabilities

like adjusting strategies for accomplishing its task(s) or adapting to changing surroundings. Maybe in future, autonomous machines might even have emotions, could speak fluently and learn to live alongside humans. The movie "*I, Robot*" echoed concerns and fear of a world where man and intelligent humanoids co-exist. However, as man is writing the script, let us hope that autonomous machines are going to be used for the benefit of mankind.

Appendix

A. Coordination Reference Systems and Transformation

Mobile robots, by definition, move any place of their environments and acquire the knowledge about the world via their perception systems. This naturally leads to a need for representing the positions and orientation of the robots, of map features and of the motion itself. To define and manipulate mathematical quantities which describe the position and orientation, we have to elucidate the coordinate systems and develop the transformation among these distinct reference frameworks. These coordinate systems consist of the global/world, the local/robot, the rangefinder including sonar and laser, and the monocular and stereo cameras.

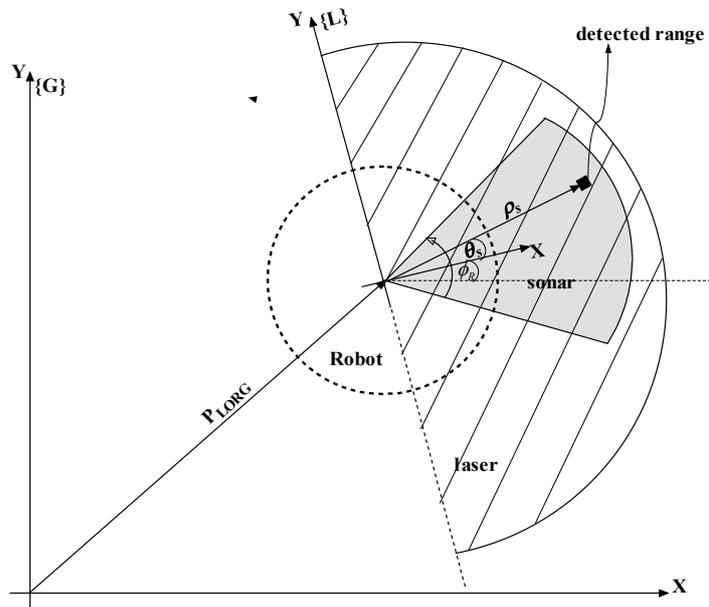


Figure A.1 The coordinate systems of global, robot and rangefinders.

Figure A.1 plots the coordinate references of the global, robot and rangefinders. Practically, different from the robot coordinate system, each sensor has its own reference and the data are also measured within this individual framework. To integrate features into the SLAM process, the measurements have to be

transformed into the robot coordinate reference. For convenience, we assume that the ultrasonic sonar and laser rangefinder have the same coordinate system as the robot and name it as local framework $\{L\}$, where X-axis is the heading of the robot and Y-axis is perpendicular to X-axis and points to left. This is because all the sensors are almost fixed on the center of robot console. Similarly, $\{G\}$ means the global frame. The gray sector denotes the sensing range for one ultrasonic sonar device, and the shadow area is for the laser. We define the robot pose as $[x_R, y_R, \phi_R]^T$, where x_R, y_R are the robot position and ϕ_R is the orientation with respect to the X-axis of the global framework. The measurement of rangefinders is $[\rho_s, \theta_s]^T$ where ρ_s is the detected range of the map features from the robot and θ_s is the orientation with respect to the X-axis of the local reference. The measurement in Cartesian format is computed as,

$${}^L P_m = \begin{bmatrix} {}^L x_m \\ {}^L y_m \end{bmatrix} = \begin{bmatrix} \rho_s \cos \theta_s \\ \rho_s \sin \theta_s \end{bmatrix} \quad (\text{A.1})$$

and converted into the global coordinate through

$${}^G P_m = {}^G R {}^L P_m + {}^G P_{LORG} \Rightarrow \begin{bmatrix} {}^G x_m \\ {}^G y_m \end{bmatrix} = {}^G R \begin{bmatrix} {}^L x_m \\ {}^L y_m \\ 1 \end{bmatrix} + {}^G P_{LORG} \quad (\text{A.2})$$

where

$${}^G R = \begin{bmatrix} \cos \phi_R & -\sin \phi_R & 0 \\ \sin \phi_R & \cos \phi_R & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad {}^G P_{LORG} = [x_R \quad y_R \quad 0]^T$$

${}^G R$ is the rotation matrix describing local frame relative to global frame, and ${}^G P_{LORG}$ is the translation vector that locates the origin of the local frame. It is noted that the format of the local measurement in (A.2) is the homogenous representation.

The model for describing the monocular and stereo cameras is the ideal pinhole camera model. The geometry related to the mapping of a pinhole camera is

illustrated in Figure A.2. A 3D orthogonal coordinate system with its origin at C is the camera aperture labeled as $\{C\}$. The three axes of the coordinate system are referred to as X , Y and Z . Axis Z is pointing in the viewing direction of the camera and is referred to as the optical axis. The 3D plane which intersects with axes X and Y is the principal plane. An image plane remarked as $\{I\}$ where the 3D world is projected through the aperture of the camera is parallel to axes X and Y and is located at distance f from the origin C in the negative direction of the Z axis. A practical implementation of a pinhole camera implies that the image plane is located such that it intersects the Z axis at coordinate $-f$ where $f > 0$. f is referred to as the focal length of the pinhole camera. A point R at the intersection of the optical axis and the image plane is referred to as the principal point or image center. The relationship between the image point Q and the world point P is

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{f}{z} \begin{pmatrix} x \\ y \end{pmatrix} \quad (\text{A.3})$$

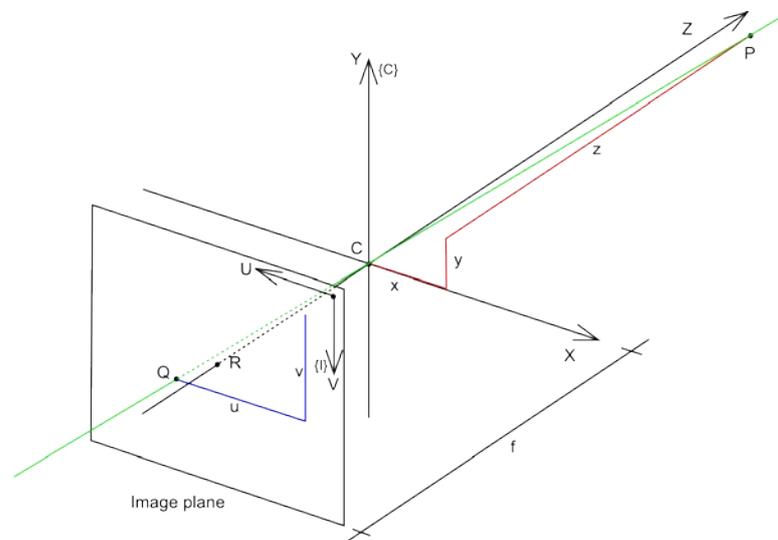


Figure A.2 Ideal pinhole camera model.

If the camera is un-calibrated for example the Canon VCC4 monocular camera load on Pioneer 2DX, we do not know the intrinsic parameters of the camera such focal length and distortion ratio. It is impossible to obtain an analytical transformation between image frame and robot frame, i.e. the image pixel coordinate can not be described in terms of metric coordinate according to (A.3). We can only obtain the

approximate pixel transformation. And that is one reason why we employ fuzzy logic in our previous work (G. Q. Huang, et al., 2006) to fuse sonar and vision information. However, even through the calibration parameters are obtained, it is also complex for monocular camera to extract the 3D information for a map feature. The common method is to make a parallel or similar motion for the camera to generate a parallax so that the depth information can be recovered approximately. This strategy looks like deliberately constructing a stereo vision system.

For the stereo camera, it is easy to extract the 3D information. There is a little bit difference on the coordinate system between the stereo camera applied in this thesis and the ideal pinhole camera. As shown in Figure A.3, the origin of the system is the optical center of the lens of the reference camera, X axis points to the right of the camera (from the camera's point of view), Y axis points towards the ground, and Z axis points forward from the camera. Please refer to the green reference framework in Figure A.3. The pink one is the local framework depicted in 3D format, where the origin is the center of emitting laser light, X axis is perpendicular to the paper and directs outside, Y axis points to the left of the laser and Z axis orientates to ceiling. When we get a pixel coordinate of a point, the relevant 3D position is

$$z = \frac{fBas}{d}, x = \frac{uz}{f}, y = \frac{vz}{f} \quad (\text{A.4})$$

where f is focal length in pixels, Bas is the baseline of the camera in meters and d is the disparity in pixels. It is noted that the f and other calibration parameters have been obtained from the pre-saved configuration of the camera.

As the coordinate system of stereo camera is different from the local framework, when the 3D position of one feature is obtained which is described in camera system as ${}^C[P_x, P_y, P_z]^T$, it is required to transform it to local reference with the following formulas:

$${}^L P = {}^L R {}^C P + {}^L P_{CORG} \Rightarrow {}^L \begin{bmatrix} P_x \\ P_y \end{bmatrix} = {}^L R {}^C \begin{bmatrix} P_x \\ P_y \end{bmatrix} + {}^L P_{CORG}, \quad {}^L P_z = {}^C P_z \quad (A.5)$$

where

$${}^L R = \begin{bmatrix} \cos \pi & -\sin \pi & 0 \\ \sin \pi & \cos \pi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad {}^L P_{CORG} = [-0.06 \quad 0.14 \quad 0]^T$$

L' means a virtual local coordinate reference which is generated through only rotating the camera reference by π (cf. Figure A.4). ${}^C P$ is the 3D position of a point measured by stereo camera and ${}^L P$ is the position in virtual frame. ${}^L P_{CORG}$ is the translation vector that locates the origin of the camera frame, and ${}^L R$ is the rotation matrix describing camera frame relative to virtual frame. This step is illustrated in process 1 of Figure A.4.

The second step of the transformation is to calculate the related coordinates in local reference from the virtual local framework. As is shown in process 2 of Figure A.4, this step is easy to implement, that is the local coordinates ${}^L P$ is obtained by exchanging the elements order of ${}^L P$, i.e.

$${}^L P = [{}^L P_z \quad {}^L P_x \quad {}^L P_y] \quad (A.6)$$

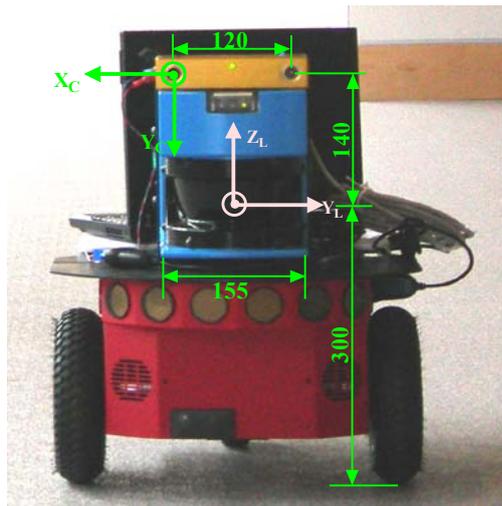


Figure A.3 Various coordinate systems for Pioneer 3DX

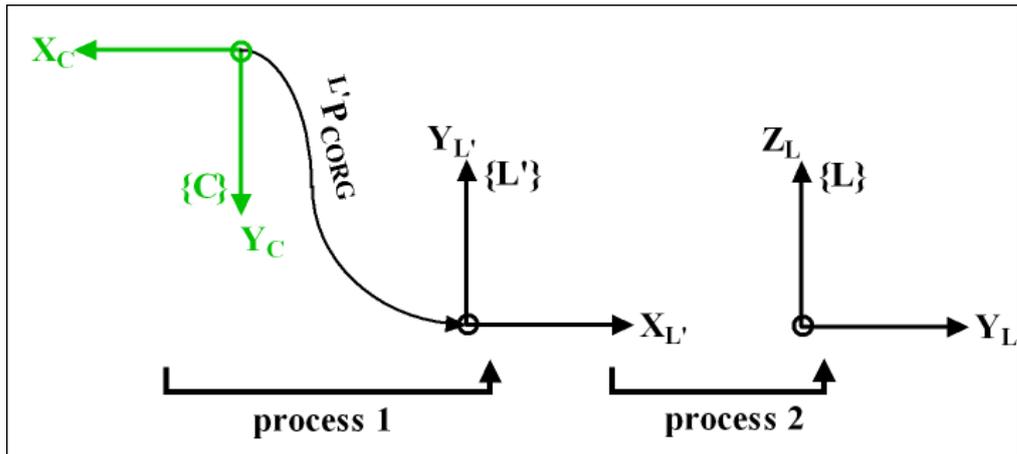


Figure A. 4 Transformation from camera frame to virtual local frame

B. The Derivation of Modified Multi-sensor Point Estimation Fusion

We can easily transform equation (5.1) into a 2D version because the robot only moves in the flat plane. Therefore the motion models for the two sensor distributed systems are identical and represented as

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k) + v_k; k = 0, 1, \dots, \quad (\text{B.1})$$

and the observation model for two different sensors is

$$\mathbf{y}_k^i = h^i(\mathbf{x}_k) + w_k^i; i = 1, 2; k = 0, 1, \dots, \quad (\text{B.2})$$

where the motion noise v_k and measurement noise w_k are both zero mean random variables independent of each other and are not cross correlated. Their covariance matrices are Q_k and R_k^i respectively. $i=1,2$ has the same meaning as stated in Chapter 5. For convenience, we temporarily scrape off the subscript C and L, and superscript f and b in the following equations. To compare performances between fused and distributed filtering, the stacked measurement equation is

$$\mathbf{y}_k = h(\mathbf{x}_k) + w_k; i = 1, 2; k = 0, 1, \dots, \quad (\text{B.3})$$

where

$$\mathbf{y}_k = [(\mathbf{y}_k^1)^T, (\mathbf{y}_k^2)^T]^T, h(\mathbf{x}_k) = \begin{bmatrix} h^1(\mathbf{x}_k) \\ h^2(\mathbf{x}_k) \end{bmatrix}, w_k = [(w_k^1)^T, (w_k^2)^T]^T$$

and the covariance of the noise w_k is given by

$$\text{Cov}(w_k) = \text{diag}(R_k^1, R_k^2) = R_k, \text{Cov}(w_k^i) = R_k^i$$

Each individual EKF SLAM is

$$\begin{aligned} \mathbf{x}_{k|k}^i &= \mathbf{x}_{k|k-1}^i + K_k^i [\mathbf{y}_k^i - h^i(\mathbf{x}_{k|k-1}^i)] \\ &= \mathbf{x}_{k|k-1}^i - K_k^i h^i(\mathbf{x}_{k|k-1}^i) + K_k^i \mathbf{y}_k^i \end{aligned} \quad (\text{B.4})$$

$$K_k^i = P_{k|k-1}^i (H_k^i)^T (S_k^i)^{-1}, S_k^i = H_k^i P_{k|k-1}^i (H_k^i)^T + R_k^i \quad (\text{B.5})$$

$$P_{k|k}^i = (I - K_k^i H_k^i) P_{k|k-1}^i, P_{k|k-1}^i = F_k^i P_{k-1|k-1}^i F_k^T + Q_k \quad (\text{B.6})$$

where F_k^i and H_k^i are the Jacobian matrix with respect to the state variable \mathbf{x}_k . From (B.5)-(B.6), we have

$$K_k^i = P_{k|k}^i (H_k^i)^T (R_k^i)^{-1} \quad (\text{B.7})$$

$$(P_{k|k}^i)^{-1} = (P_{k|k-1}^i)^{-1} + (H_k^i)^T (R_k^i)^{-1} H_k^i \quad (\text{B.8})$$

The fused EKF SLAM has similar formulas

$$\begin{aligned} \mathbf{x}_{k|k} &= \mathbf{x}_{k|k-1} + K_k [\mathbf{y}_k - h(\mathbf{x}_{k|k-1})] \\ &= \mathbf{x}_{k|k-1} - K_k h(\mathbf{x}_{k|k-1}) + K_k \mathbf{y}_k \end{aligned} \quad (\text{B.9})$$

$$K_k = P_{k|k} H_k^T R_k^{-1} \quad (\text{B.10})$$

where $H_k = [(H_k^1)^T, (H_k^2)^T]^T$. Also the covariance of fused filter is deduced as

$$\begin{aligned} P_{k|k}^{-1} &= P_{k|k-1}^{-1} + H_k^T R_k^{-1} H_k \\ &= P_{k|k-1}^{-1} + \sum_{i=1}^2 (H_k^i)^T (R_k^i)^{-1} H_k^i \end{aligned} \quad (\text{B.11})$$

According to (B.8) and (B.11), $P_{k|k}$ can be represented by $P_{k|k}^i$, i.e.

$$\begin{aligned} P_{k|k}^{-1} - P_{k|k-1}^{-1} &= \sum_{i=1}^2 (H_k^i)^T (R_k^i)^{-1} H_k^i \\ &= \sum_{i=1}^2 [(P_{k|k}^i)^{-1} - (P_{k|k-1}^i)^{-1}] \end{aligned} \quad (\text{B.12})$$

Multiplying \mathbf{y}_k at both side of (B.10), we have

$$K_k \mathbf{y}_k = P_{k|k} H_k^T R_k^{-1} \mathbf{y}_k = P_{k|k} \sum_{i=1}^2 (H_k^i)^T (R_k^i)^{-1} \mathbf{y}_k^i \quad (\text{B.13})$$

and by (B.4) we obtain

$$\begin{aligned} K_k^i \mathbf{y}_k^i &= \mathbf{x}_{k|k}^i - [\mathbf{x}_{k|k-1}^i - K_k^i h^i(\mathbf{x}_{k|k-1}^i)] \Rightarrow \\ P_{k|k}^i (H_k^i)^T (R_k^i)^{-1} \mathbf{y}_k^i &= \mathbf{x}_{k|k}^i - [\mathbf{x}_{k|k-1}^i - K_k^i h^i(\mathbf{x}_{k|k-1}^i)] \Rightarrow \\ (H_k^i)^T (R_k^i)^{-1} \mathbf{y}_k^i &= (P_{k|k}^i)^{-1} (\mathbf{x}_{k|k}^i - [\mathbf{x}_{k|k-1}^i - K_k^i h^i(\mathbf{x}_{k|k-1}^i)]) \end{aligned} \quad (\text{B.14})$$

Substituting (B.14) into (B.13), and then substituting (B.13) and (B.10) into (B.9), we find

$$P_{k|k}^{-1} \mathbf{x}_{k|k} = P_{k|k}^{-1} \mathbf{x}_{k|k-1} - H_k^T R_k^{-1} h(\mathbf{x}_{k|k-1}) + \sum_{i=1}^2 (P_{k|k}^i)^{-1} (\mathbf{x}_{k|k}^i - [\mathbf{x}_{k|k-1}^i - K_k^i h^i(\mathbf{x}_{k|k-1}^i)]) \quad (\text{B.15})$$

Actually in each iteration, $h(\mathbf{x}_{k|k-1})$ and $h^i(\mathbf{x}_{k|k-1}^i)$ are constant matrix calculated by $\mathbf{x}_{k|k-1}$, and therefore $H_k^T R_k^{-1} h(\mathbf{x}_{k|k-1})$ and $(H_k^i)^T (R_k^i)^{-1} h^i(\mathbf{x}_{k|k-1}^i)$ are linear items. With these properties of linearity, we may determine

$$H_k^T R_k^{-1} h(\mathbf{x}_{k|k-1}) = \sum_{i=1}^2 (H_k^i)^T (R_k^i)^{-1} h^i(\mathbf{x}_{k|k-1}^i) \quad (\text{B.16})$$

By (B.16) and (B.7), (B.15) is rewritten as

$$\mathbf{x}_{k|k} = \mathbf{x}_{k|k-1} + P_{k|k} \sum_{i=1}^2 (P_{k|k}^i)^{-1} (\mathbf{x}_{k|k}^i - \mathbf{x}_{k|k-1}^i) \quad (\text{B.17})$$

It is necessary to note that (B.12) and (B.17) manifest the relationship of the state variable vector between the fused and individual EKF SLAM as well as the covariance matrix. From (B.17), the weight matrix for each individual state variable vector can be determined. That is

$$\mathbf{W} = [I, P_{k|k} (P_{k|k}^1)^{-1}, -P_{k|k} (P_{k|k}^1)^{-1}, P_{k|k} (P_{k|k}^2)^{-1}, -P_{k|k} (P_{k|k}^2)^{-1}] \quad (\text{B.18})$$

If the latest fused state estimate $\mathbf{x}_{k|k}$ is broadcasted to every individual state estimate as the back propagation (we call it feedback), we can prove that the covariance of state variable is reduced with this feedback but the performance of the fused EKF SLAM is unchanged with and without the feedback.

To maintain the identity with (Zhu, 2003), we apply the same symbols and assumptions. Concerning the feedback, the individual and fused one-step predictions are

$$\mathbf{x}_{k|k-1}^i = f(\hat{\mathbf{x}}_{k-1|k-1}^i, \mathbf{u}_{k-1}) = \mathbf{x}_{k|k-1}, \quad \hat{P}_{k|k-1}^i = P_{k|k-1} \quad (\text{B.19})$$

Rewriting (B.12) and (B.17) by using (B.19) as

$$\hat{\hat{P}}_{k|k}^{-1} = \sum_{i=1}^2 (P_{k|k}^i)^{-1} - (l-1)P_{k|k-1}^{-1}, l=2 \quad (\text{B.20})$$

$$\hat{\hat{P}}_{k|k}^{-1} \hat{\hat{\mathbf{x}}}_{k|k} = \sum_{i=1}^2 (P_{k|k}^i)^{-1} \hat{\mathbf{x}}_{k|k}^i - \left[\sum_{i=1}^2 (P_{k|k}^i)^{-1} - P_{k|k}^{-1} \right] \mathbf{x}_{k|k-1} \quad (\text{B.21})$$

Suppose that the initial values of state variable vector and covariance for fused and individual EKF SLAM are same, i.e.

$$\hat{\hat{\mathbf{x}}}_{0|0} = \mathbf{x}_{0|0} = \mathbf{x}_{0|0}^i = \mathbf{x}_{0|0}^i, \hat{\hat{P}}_{0|0} = P_{0|0} = P_{0|0}^i = P_{0|0}^i \quad (\text{B.22})$$

And we also employ the assumptions listed in (Zhu, 2003)

$$\begin{aligned} \hat{\hat{\mathbf{x}}}_{k-1|k-1} &= \mathbf{x}_{k-1|k-1}, \mathbf{x}_{k|k-1} = \mathbf{x}_{k|k-1} \\ \hat{\hat{P}}_{k-1|k-1} &= P_{k-1|k-1}, P_{k|k-1} = P_{k|k-1} \end{aligned} \quad (\text{B.23})$$

At step k , substituting (B.23) into (B.20), we have

$$\hat{\hat{P}}_{k|k}^{-1} = P_{k|k-1}^{-1} + \sum_{i=1}^2 \left[(P_{k|k}^i)^{-1} - P_{k|k-1}^{-1} \right] \quad (\text{B.24})$$

Similar to (B.8), we can get

$$\begin{aligned} (\hat{\hat{P}}_{k|k}^{-1})^{-1} &= (P_{k|k-1}^i)^{-1} + (H_k^i)^T (R_k^i)^{-1} H_k^i \\ &\stackrel{(31)}{=} P_{k|k-1}^{-1} + (H_k^i)^T (R_k^i)^{-1} H_k^i \end{aligned} \quad (\text{B.25})$$

Substituting (B.25) into (B.24), we obtain

$$\hat{\hat{P}}_{k|k}^{-1} = P_{k|k-1}^{-1} + \sum_{i=1}^2 (H_k^i)^T (R_k^i)^{-1} H_k^i \quad (\text{B.26})$$

In comparison with (B.12), we claim that

$$\hat{\hat{P}}_{k|k}^{-1} = P_{k|k}^{-1} \quad (\text{B.27})$$

On the other hand, we get the following equations by substituting (B.7) into (B.4)

,

$$\begin{aligned} (\hat{\hat{P}}_{k|k}^{-1})^{-1} \hat{\hat{\mathbf{x}}}_{k|k} &= (P_{k|k}^i)^{-1} \mathbf{x}_{k|k-1} + (H_k^i)^T (R_k^i)^{-1} \mathbf{y}_k^i \\ &\quad - (H_k^i)^T (R_k^i)^{-1} h^i(\mathbf{x}_{k|k-1}) \end{aligned} \quad (\text{B.28})$$

and (B.10) into (B.9),

$$\begin{aligned}
P_{k|k}^{-1} \hat{\mathbf{x}}_{k|k} &= P_{k|k}^{-1} \mathbf{x}_{k|k-1} + H_k^T R_k^{-1} \mathbf{y}_k - H_k^T R_k^{-1} h(\mathbf{x}_{k|k-1}) \\
&= P_{k|k}^{-1} \mathbf{x}_{k|k-1} + \sum_{i=1}^2 \left[(H_k^i)^T (R_k^i)^{-1} \mathbf{y}_k^i \right. \\
&\quad \left. - (H_k^i)^T (R_k^i)^{-1} h^i(\mathbf{x}_{k|k-1}) \right]
\end{aligned} \tag{B.29}$$

With replacing the related item in (B.21) by (B.28), and considering the conditions (B.23), the following derivation is obtained.

$$\begin{aligned}
\hat{P}_{k|k}^{-1} \mathbf{x}_{k|k} &= P_{k|k}^{-1} \mathbf{x}_{k|k-1} + \sum_{i=1}^2 \left[(H_k^i)^T (R_k^i)^{-1} \mathbf{y}_k^i \right. \\
&\quad \left. - (H_k^i)^T (R_k^i)^{-1} h^i(\mathbf{x}_{k|k-1}) \right] \\
\stackrel{\hat{P}_{k|k}^{-1} = P_{k|k}^{-1}}{\Rightarrow} P_{k|k}^{-1} \mathbf{x}_{k|k} &= P_{k|k}^{-1} \mathbf{x}_{k|k-1} + \sum_{i=1}^2 \left[(H_k^i)^T (R_k^i)^{-1} \mathbf{y}_k^i \right. \\
&\quad \left. - (H_k^i)^T (R_k^i)^{-1} h^i(\mathbf{x}_{k|k-1}) \right]
\end{aligned} \tag{B.30}$$

Comparing (B.29) and (B.30), we assert that

$$\hat{\mathbf{x}}_{k|k} = \mathbf{x}_{k|k} \tag{B.31}$$

It is obvious from (B.27) and (B.31) that the performance of fused EKF SLAM does not change in the presence or absence of feedback. However, when the feedback is allowed into the individual EKF SLAM, the fused covariance of the state vector is decreased. This result is verified as follow. By (B.8) and (B.26) we have the equation

$$\begin{aligned}
(P_{k|k}^i)^{-1} - (P_{k|k-1}^i)^{-1} &= (\hat{P}_{k|k}^i)^{-1} - P_{k|k-1}^{-1} \Rightarrow \\
(\hat{P}_{k|k}^i)^{-1} - (P_{k|k}^i)^{-1} &= P_{k|k-1}^{-1} - (P_{k|k-1}^i)^{-1} \\
&= (F_{k-1} P_{k|k-1} F_{k-1}^T + Q_k)^{-1} - (F_{k-1} P_{k|k-1}^i F_{k-1}^T + Q_k)^{-1}
\end{aligned} \tag{B.32}$$

It is easy to prove that (B.32) is equal and larger than zero because $P_{k-1|k-1} \leq P_{k-1|k-1}^i, k = 2, 3, \dots$, (cf. (Zhu, 2003)). Therefore, we have

$$(\hat{P}_{k|k}^i)^{-1} - (P_{k|k}^i)^{-1} = P_{k|k-1}^{-1} - (P_{k|k-1}^i)^{-1} \geq 0, \text{ that is}$$

$$\hat{P}_{k|k}^i \leq P_{k|k}^i, P_{k|k-1} \leq P_{k|k-1}^i \tag{B.33}$$

and also

$$P_{k|k} = \hat{P}_{k|k} \leq P_{k|k}^i \tag{B.34}$$

which derives from (B.8), (B.26) and (B.27) if and only if $\sum_{j \neq i} (H_r^j)^T (R_r^j)^{-1} H_r^j > 0$, for some $r \leq k-1$. Please refer to (Zhu, 2003) for this condition in detail.

It can be concluded that (B.33) and (B.34) suggest that under a certain constraint the fused covariance of the state variable is reduced with the feedback. And when we use this fused state variables in SLAM, it will reduce the error of the localization and map features without changing the performance of individual EKF SLAM.

C. Information Entropy

The concept of entropy term by itself usually refers to the Shannon's entropy, which is a measure of the uncertainty of a random variable. Let X be a discrete random variable with alphabet \mathcal{X} and probability mass function $p(x) = \Pr\{X = x\}$, $x \in \mathcal{X}$.

Definition The entropy $H(X)$ of a discrete random variable X is defined by

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x) \quad (\text{C.1})$$

The log is to the base 2 and entropy is expressed in bits. If the base of the logarithm is b , we denote the entropy as $H_b(X)$. If the base of the logarithm is e , the entropy is measured in nats. Note that entropy is a functional of the distribution of X . It does not depend on the actual values taken by the random variable X , but only on the probabilities. For more explanation on information entropy, please refer to the monograph (Cover, 2006).

Bibliography

- Ahn, SungHwan, Choi, Jinwoo, Doh, Nakju, & Chung, Wan (2008). A practical approach for EKF-SLAM in an indoor environment: fusing ultrasonic sensors and stereo camera. *Autonomous Robots*, 24(3), 315-335.
- Ahn, Sunghwan, Choi, Minyong, Choi, Jinwoo, & Chung, Wan Kyun (2006). *Data Association Using Visual Object Recognition for EKF-SLAM in Home Environment*. Paper presented at the Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on.
- Alempijevic, Alen, & Dissanayake, Gamini (2004). *An efficient algorithm for line extraction from laser scans*. Paper presented at the 2004 IEEE Conference on Robotics, Automation and Mechatronics, Singapore, Singapore.
- Andrade-Cetto, Juan, & Sanfeliu, Alberto (2002). CONCURRENT MAP BUILDING AND LOCALIZATION ON INDOOR DYNAMIC ENVIRONMENTS. *International Journal of Pattern Recognition and Artificial Intelligence*, 16(3), 361 - 374.
- Arras, Kai O. (2003). *Feature-Based Robot Navigation in Known and Unknown Environments*. Unpublished Dissertation/Thesis, Swiss Federal Institute of Technology Lausanne (EPFL), Theses No. 2765.
- Bailey, T., & Durrant-Whyte, H. (2006). Simultaneous localization and mapping (SLAM): part II. *Robotics & Automation Magazine, IEEE*, 13(3), 108-117.
- Bailey, T., Nebot, E. M., Rosenblatt, J. K., & Durrant-Whyte, H. F. (2000). *Data association for mobile robot navigation: a graph theoretic approach*. Paper presented at the Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on.
- Beckerman, M. (1992). *A Bayes-maximum entropy method for multi-sensor data fusion*. Paper presented at the Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on.
- Bengtsson, Ola, & Baerveldt, Albert-Jan (2003). Robot localization based on scan-matching--estimating the covariance matrix for the IDC algorithm. *Robotics and Autonomous Systems*, 44(1), 29-40.
- Biswas, R., Limketkai, B., Sanner, S., & Thrun, S. (2002). *Towards object mapping in non-stationary environments with mobile robots*. Paper presented at the Intelligent Robots and System, 2002. IEEE/RSJ International Conference on.
- Botev, Zdravko, & Kroese, Dirk (2008). Non-asymptotic Bandwidth Selection for Density Estimation of Discrete Data. *Methodology and Computing in Applied Probability*, 10(3), 435-451.
- Bowditch, Nathaniel (2002). Chapter 7 Dead Reckoning. In Imagery United States. National & Agency Mapping (Eds.), *The American practical navigator : an epitome of navigation* (2002 bicentennial ed.). Bethesda, Md. : National Imagery and Mapping Agency.
- Burgard, Wolfram, Cremers, Armin B., Fox, Dieter, Hefner, Dirk, Lakemeyer, Gerhard, Schulz, Dirk, et al. (1999). Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1-2), 3-55.
- Castellanos, J. A., Neira, J., & Tardos, J. D. (2001). Multisensor fusion for simultaneous localization and map building. *Robotics and Automation, IEEE Transactions on*, 17(6),

- 908-914.
- Chow, K. M., Rad, A. B., & Ip, Y. L. (2002). Enhancement of Probabilistic Grid-based Map for Mobile Robot Applications. *Journal of Intelligent and Robotic Systems*, 34(2), 155-174.
- Civera, J., Davison, A. J., & Montiel, J. (2008). Inverse Depth Parametrization for Monocular SLAM. *Robotics, IEEE Transactions on*, 24(5), 932-945.
- Civera, J., Davison, A. J., & Montiel, J. M. M. (2007a). *Inverse Depth to Depth Conversion for Monocular SLAM*. Paper presented at the Robotics and Automation, 2007 IEEE International Conference on.
- Civera, J., Davison, Andrew, & Montiel, J. (2007b). *Dimensionless Monocular SLAM*. Paper presented at the Iberian Conference of Pattern Recognition and Image Analysis.
- Cohen, O., & Edan, Y. (2005). *A Sensor Fusion Framework for On-Line Sensor and Algorithm Selection*. Paper presented at the Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on.
- Cover, T. M. (2006). *Elements of information theory* (2nd ed.). Hoboken, N.J. : Wiley-Interscience.
- Davison, Andrew J., Reid, Ian D., Molton, Nicholas D., & Stasse, Olivier (2007). MonoSLAM: Real-Time Single Camera SLAM. *Transactions on Pattern Analysis and Machine Intelligence*, 29(6), 1052-1067.
- Davison, Andrew. J. (2003, October). *Real-Time Simultaneous Localisation and Mapping with a Single Camera*. Paper presented at the Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2.
- Dellaert, F., Seitz, S. M., Thorpe, C. E., & Thrun, S. (2000). *Structure from motion without correspondence*. Paper presented at the Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on.
- Dezert, J., & Bar-Shalom, Y. (1993). Joint probabilistic data association for autonomous navigation. *Aerospace and Electronic Systems, IEEE Transactions on*, 29(4), 1275-1286.
- Ding, Shi-fei, & Shi, Zhong-zhi (2005). Studies on incidence pattern recognition based on information entropy. *Journal of Information Science*, 31(6), 497-502.
- Diosi, A., & Kleeman, L. (2004). *Advanced sonar and laser range finder fusion for simultaneous localization and mapping*. Paper presented at the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sendai, Japan.
- Diosi, A., & Kleeman, L. (2005). *Laser scan matching in polar coordinates with application to SLAM*. Paper presented at the Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on.
- Dissanayake, M. W. M. G., Newman, P., Clark, S., Durrant-Whyte, H. F., & Csorba, M. (2001). A solution to the simultaneous localization and map building (SLAM) problem. *Robotics and Automation, IEEE Transactions on*, 17(3), 229-241.
- Drocourt, C., Delahoche, L., Marhic, B., & Clerentin, A. (2002). *Simultaneous localization and map construction method using omnidirectional stereoscopic information*. Paper presented at the Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on.
- Duda, Richard O (2001). *Pattern classification* (Peter E. Hart & David G. Stork, Trans. 2nd ed.). New York Wiley.
- Durrant-Whyte, H., & Bailey, T. (2006). Simultaneous localization and mapping: part I. *Robotics & Automation Magazine, IEEE*, 13(2), 99-110.

- Eade, E., & Drummond, T. (2006). *Scalable Monocular SLAM*. Paper presented at the Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on.
- Eade, Ethan, & Drummond, Tom (2006, September). *Edge Landmarks in Monocular SLAM*. Paper presented at the Proc. 17th British Machine Vision Conference (BMVC).
- Eliazar, A. I., & Parr, R. (2004). *DP-SLAM 2.0*. Paper presented at the Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on.
- Erdogmus, D., & Principe, J. C. (2002). An error-entropy minimization algorithm for supervised training of nonlinear adaptive systems. *Signal Processing, IEEE Transactions on*, 50(7), 1780-1786.
- Everett, H. R. (1995). *Sensors for mobile robots : theory and application*. Wellesley, Mass.: A.K. Peters.
- Fassinut-Mombot, Bienvenu, & Choquel, Jean-Bernard (2004). A new probabilistic and entropy fusion approach for management of information sources. *Information Fusion*, 5(1), 35-47.
- Fichtner, M., & Grobmann, A. (2004). *A probabilistic visual sensor model for mobile robot localisation in structured environments*. Paper presented at the Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on.
- Folkesson, J., Jensfelt, P., & Christensen, H. I. (2005). *Vision SLAM in the Measurement Subspace*. Paper presented at the Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on.
- Fox, D., Burgard, W., & Thrun, S. (1999). Markov Localization for Mobile Robots in Dynamic Environments. *Journal of Artificial Intelligence Research*(11), 391-427.
- Fox, Dieter, Burgard, Wolfram, Dellaert, Frank, & Thrun, Sebastian (1999, July 18-22). *Monte Carlo Localization: Efficient Position Estimation for Mobile Robots*. Paper presented at the Proc. 16th National Conference on Artificial Intelligence, AAAI'99, Orlando, Florida, USA.
- Fox, Dieter, Burgard, Wolfram, & Thrun, Sebastian (1998). Active Markov localization for mobile robots. *Robotics and Autonomous Systems*, 25(3-4), 195-207.
- Garrido, S., Moreno, L., Abderrahim, M., & Martin, F. (2006). *Path Planning for Mobile Robot Navigation using Voronoi Diagram and Fast Marching*. Paper presented at the Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on.
- Garulli, Andrea, Giannitrapani, Antonio, Rossi, Andrea, & Vicino, Antonio (2005). *Mobile robot SLAM for line-based environment representation*. Paper presented at the 44th IEEE Conference on Decision and Control, and the European Control Conference, CDC-ECC '05, Seville, Spain.
- Gasparri, A., Panzieri, S., Pascucci, F., & Ulivi, G. (2007). *A Hybrid Active Global Localisation Algorithm for Mobile Robots*. Paper presented at the Robotics and Automation, 2007 IEEE International Conference on.
- Gee, A. P., & Mayol-Cuevas, W. (2006, November). *Real-time model-based SLAM using line segments*. Paper presented at the 2nd International Symposium on Visual Computing, Lake Tahoe, NV, USA.
- Gemeiner, Peter, Einramhof, Peter, & Vincze, Markus (2007). Simultaneous Motion and Structure Estimation by Fusion of Inertial and Vision Data. *The International Journal of Robotics Research*, 26(6), 591-605.
- Gil, A. , Reinoso, O. , Payá, L. , Ballesta, M. , & Pedrero, J.M. (April 2007). Managing Data

- Association in visual SLAM using SIFT features. *International Journal of Factory Automation, Robotics and Soft Computing*, 179-184.
- Gil, A., Reinoso, O., Martinez Mozos, O., Stachniss, C., & Burgard, W. (2006). *Improving Data Association in Vision-based SLAM*. Paper presented at the Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on.
- Gourley, C., & Trivedi, M. (1994). *Sensor based obstacle avoidance and mapping for fast mobile robots*. Paper presented at the Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on.
- Gutmann, J. S., & Schlegel, C. (1996). *AMOS: comparison of scan matching approaches for self-localization in indoor environments*. Paper presented at the Advanced Mobile Robot, 1996., Proceedings of the First Euromicro Workshop on.
- Hähnel, D., Schulz, D., & Burgard, W. (2003). Mobile robot mapping in populated environments. *Advanced Robotics*, 17(7), 579-597.
- Hähnel, D., Triebel, R., Burgard, W., & Thrun, S. (2003). *Map building with mobile robots in dynamic environments*. Paper presented at the Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on.
- Hartley, Richard, & Zisserman, Andrew (2003). *Multiple view geometry in computer vision* (2nd ed.). Cambridge [England] :: Cambridge University Press.
- Huang, G. Q., Rad, A. B., & Wong, Y. K. (2005). *Online SLAM in dynamic environments*. Paper presented at the Advanced Robotics, 2005. ICAR '05. Proceedings., 12th International Conference on.
- Huang, G. Q., Rad, A. B., & Wong, Y. K. (2006). A New Solution to Map Dynamic Indoor Environments. *International Journal of Advanced Robotic Systems*, 3(3), 199-210.
- Huang, G. Q., Rad, A. B., Wong, Y. K., & Ip, Y. L. (2004). *SLAM with MTT: theory and initial results [mobile robot localisation]*. Paper presented at the Robotics, Automation and Mechatronics, 2004 IEEE Conference on.
- Huang, Guoquan, Rad, Ahmad B., Wong, Yiu-Kwong, & Ip, Ying-Leung (2007). Heterogeneous multisensor fusion for mapping dynamic environments. *Advanced Robotics*, 21, 661-688.
- Ip, Y. L., & Rad, A. B. (2004). Incorporation of Feature Tracking into Simultaneous Localization and Map Building via Sonar Data. *Journal of Intelligent and Robotic Systems*, 39(2), 149-172.
- Ip, Y. L., Rad, A. B., Chow, K. M., & Wong, Y. K. (2002). Segment-Based Map Building Using Enhanced Adaptive Fuzzy Clustering Algorithm for Mobile Robot Applications. *Journal of Intelligent and Robotic Systems*, 35(3), 221-245.
- Ip, Y. L., Rad, A. B., & Wong, Y. K. (2004). Map Building and Localization for Autonomous Mobile Robots *Robotic Welding, Intelligence and Automation* (pp. 161-189).
- Jensfelt, P., & Kristensen, S. (2001). Active global localization for a mobile robot using multiple hypothesis tracking. *Robotics and Automation, IEEE Transactions on*, 17(5), 748-760.
- Jeong, Woo Yeon, & Lee, Kyoung Mu (2006). *Visual SLAM with Line and Corner Features*. Paper presented at the Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on.
- Johnson, David S., & McGeoch, Catherine C. (1993). *Network flows and matching : first DIMACS implementation challenge*. Providence, R.I. :: American Mathematical Society.
- Julier, Simon, & Uhlmann, Jeffrey K. (2001). General Decentralized Data Fusion with Covariance

- Intersection (CI). In David L. Hall & James Llinas (Eds.), *Handbook of multisensor data fusion* (pp. 12-11~12-24). Boca Raton, Fla.: CRC Press.
- Kapach, K., & Edan, Y. (2007). *Evaluation of grid-map sensor fusion mapping algorithms*. Paper presented at the Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on.
- Kim, C., Sakthivel, R., & Chung, W. K. (2008). Unscented FastSLAM: A Robust and Efficient Solution to the SLAM Problem. *Robotics, IEEE Transactions on*, 24(4), 808-820.
- Kim, Gon Woo, Kim, Ji Min, Kwak, Nosan, & Lee, Beom Hee (2007). Hierarchical sensor fusion for building a probabilistic local map using active sensor modules. *Robotica*, 26 (03), 307-322.
- Lee, Se Jin, Cho, Dong Woo, Chung, Wan Kyun, Lim, Jong Hwan, & Kang, Chul Ung (2005). *Feature based map building using sparse sonar data*. Paper presented at the Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on.
- Lemaire, T., & Lacroix, S. (2007). *Monocular-vision based SLAM using Line Segments*. Paper presented at the Robotics and Automation, 2007 IEEE International Conference on.
- Leonard, J. J., & Durrant-Whyte, H. F. (1991a). Mobile robot localization by tracking geometric beacons. *Robotics and Automation, IEEE Transactions on*, 7(3), 376-382.
- Leonard, J. J., & Durrant-Whyte, H. F. (1991b). *Simultaneous map building and localization for an autonomous mobile robot*. Paper presented at the Intelligent Robots and Systems '91. 'Intelligence for Mechanical Systems, Proceedings IROS '91. IEEE/RSJ International Workshop on.
- Leonard, John J., & Durrant-Whyte, Hugh F. (1992). *Directed sonar sensing for mobile robot navigation / John J. Leonard*, . Boston :: Kluwer Academic Publishers.
- Li, Xinliang, Luo, Zhi-Quan, Wong, K. M., & Bosse, E. (1999). An interior point linear programming approach to two-scan data association. *Aerospace and Electronic Systems, IEEE Transactions on*, 35(2), 474-490.
- Lizarralde, F., Nunes, E. V. L., Liu, Hsu, & Wen, J. T. (2003). *Mobile robot navigation using sensor fusion*. Paper presented at the Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on.
- Lowe, David G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2), 91-110.
- Lu, Feng, & Milios, Evangelos (1997). Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans. *Journal of Intelligent and Robotic Systems*, 18(3), 249-275.
- Manyika, J., & Durrant-Whyte, H. (1994). *Data fusion and sensor management : a decentralized information-theoretic approach* (Hugh F. Durrant-Whyte, Trans.). New York ; London:: Ellis Horwood.
- Maronna, Ricardo A., Martin, R. Douglas, & Yohai, Vaictor J. (2006). *Robust statistics : theory and methods*. Chichester, England: J. Wiley.
- Martin, A. Fischler, & Robert, C. Bolles (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6), 381-395.
- Martin, Martin C. (2006). Evolving visual sonar: Depth from monocular images. *Pattern Recognition Letters*, 27(11), 1174-1180.
- Marzorati, Daniele, Matteucci, Matteo, Migliore, Davide, & Sorrenti, Domenico G. (2007,

- September 19-21). *Integration of 3D Lines and Points in 6DoF Visual SLAM by Uncertain Projective Geometry*. Paper presented at the 3rd European Conference on Mobile Robots, Freiburg, Germany.
- Matía, F., & Jiménez, A. (1998). Multisensor Fusion: An Autonomous Mobile Robot. *Journal of Intelligent and Robotic Systems*, 22(2), 129-141.
- Milstein, Adam, Sánchez, Javier Nicolás, & Williamson, Evan Tang (2002). *Robust global localization using clustered particle filtering*. Paper presented at the Proceedings of the Eighteenth National Conference on Artificial Intelligence, Edmonton, Alberta.
- Miro, J. V., Dissanayake, G., & Weizhen, Zhou (2005). *Vision-based SLAM using natural features in indoor environments*. Paper presented at the Intelligent Sensors, Sensor Networks and Information Processing Conference, 2005. Proceedings of the 2005 International Conference on.
- Montemerlo, M., Thrun, S., Koller, D., & Wegbreit, B. (2003). *FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges*. Paper presented at the The International Joint Conference on Artificial Intelligence.
- Montemerlo, M., Thrun, S., Koller, D., & Wegbreit, B. (2002). *FastSLAM: A factored solution to the simultaneous localization and mapping problem*. Paper presented at the The 18th Proceedings of the AAAI National Conference on Artificial Intelligence, Edmonton, Alberta, Canada.
- Montemerlo, M., Thrun, S., & Whittaker, W. (2002). *Conditional particle filters for simultaneous mobile robot localization and people-tracking*. Paper presented at the Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on.
- Montiel, J.M.M, Civera, Javier, & Davison, Andrew (2006). *Unified Inverse Depth Parametrization for Monocular SLAM*. Paper presented at the Robotics: Science and Systems.
- Moreira, M. A. G., Machado, H. N., Mendonca, C. F. C., & Pereira, G. A. S. (2007). *Mobile robot outdoor localization using planar beacons and visual improved odometry*. Paper presented at the Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on.
- Muñoz-Salinas, Rafael, Aguirre, Eugenio, & García-Silvente, Miguel (2006). Detection of doors using a genetic visual fuzzy system for mobile robots. *Autonomous Robots*, 21(2), 123-141.
- Murphey, Y. L., Chen, J., Crossman, J., Zhang, J., Richardson, P., & Sieh, L. (2000). *DepthFinder, a real-time depth detection system for aided driving*. Paper presented at the Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE.
- Neira, J., & Tardos, J. D. (2001). Data association in stochastic mapping using the joint compatibility test. *Robotics and Automation, IEEE Transactions on*, 17(6), 890-897.
- Nguyen, V., Martinelli, A., Tomatis, N., & Siegwart, R. (2005). *A comparison of line extraction algorithms using 2D laser rangefinder for indoor mobile robotics*. Paper presented at the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Alta., Canada.
- Nieto, J., Guivant, J., Nebot, E., & Thrun, S. (2003). *Real time data association for FastSLAM*. Paper presented at the Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE

- International Conference on.
- O'Kane, J. M. (2006). *Global localization using odometry*. Paper presented at the Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on.
- Oriolo, G., Ulivi, G., & Vendittelli, M. (1997). Fuzzy maps: A new tool for mobile robot perception and planning. *Journal of Robotic Systems*, 14(3), 179-197.
- Oriolo, G., Ulivi, G., & Vendittelli, M. (1998). Real-time map building and navigation for autonomous robots in unknown environments. *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, 28(3), 316-333.
- Pfister, S. T., Roumeliotis, S. I., & Burdick, J. W. (2003). *Weighted line fitting algorithms for mobile robot map building and efficient data representation*. Paper presented at the 2003 IEEE Int. Conf. on Robotics and Automation.
- Porta, J. M., Verbeek, J. J., & Kröse, B. J. A. (2005). Active Appearance-Based Robot Localization Using Stereo Vision. *Autonomous Robots*, 18(1), 59-80.
- Prez Lorenzo, J. M., Vazquez-Martin, R., Nunez, P., Perez, E. J., & Sandoval, F. (2004). *A Hough-based method for concurrent mapping and localization in indoor environments*. Paper presented at the Robotics, Automation and Mechatronics, 2004 IEEE Conference on.
- Qiu, Wanhua (2002). *Guan li jue ce yu ying yong shang xue* (1st ed.). Beijing: Ji xie gong ye chu ban she.
- Rao, Malvika, Dudek, Gregory, & Whitesides, Sue (2007). Randomized Algorithms for Minimum Distance Localization. *The International Journal of Robotics Research*, 26(9), 917-933.
- Reid, D. (1979). An algorithm for tracking multiple targets. *Automatic Control, IEEE Transactions on*, 24(6), 843-854.
- Remolina, Emilio, & Kuipers, Benjamin (2004). Towards a general theory of topological maps. *Artificial Intelligence*, 152(1), 47-104.
- Roumeliotis, S. I., & Bekey, G. A. (2002). Distributed multirobot localization. *Robotics and Automation, IEEE Transactions on*, 18(5), 781-795.
- Rousseeuw, Peter J., & Leroy, Annick M. (1987). *Robust regression and outlier detection*. New York: Wiley.
- Saxena, A., Chung, S.H., & Ng, A.Y. (2005). *Learning Depth from Single Monocular Images*. Paper presented at the 19th Annual Conference on Neural Information Processing Systems.
- Sim, Robert, Elinas, Pantelis, Griffin, Matt, & Little, James J. (2005). *Vision-based SLAM using the Rao-Blackwellised Particle Filter*. Paper presented at the IJCAI Workshop on Reasoning with Uncertainty in Robotics (RUR).
- Smith, Paul, Reid, Ian, & Davison, Andrew J. (2006, September). *Real-Time Monocular SLAM with Straight Lines*. Paper presented at the Proc. 17th British Machine Vision Conference (BMVC).
- Smith, R., Self, M., & Cheeseman, P. (1990). Estimating uncertain spatial relationships in robotics *Autonomous robot vehicles* (pp. 167-193): Springer-Verlag New York, Inc.
- Smith, Randall C., & Cheeseman, Peter (1987). On the representation and estimation of spatial uncertainty. *International Journal of Robotics Research*, 5(4), 56-68.
- Smith, Randall, Self, Matthew, & Cheeseman, Peter (1988). *A stochastic map for uncertain spatial relationships*. Paper presented at the on The fourth international symposium, Univ. of

- California, Santa Clara, California, United States.
- Stephen, Se, Lowe, D., & Little, J. (2002). *Global localization using distinctive visual features*. Paper presented at the Intelligent Robots and System, 2002. IEEE/RSJ International Conference on.
- Thrun, Sebastian (1998a). Bayesian Landmark Learning for Mobile Robot Localization. *Machine Learning*, 33(1), 41-76.
- Thrun, Sebastian (1998b). Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1), 21-71.
- Thrun, Sebastian (2003). Robotic mapping: a survey *Exploring artificial intelligence in the new millennium* (pp. 1-35): Morgan Kaufmann Publishers Inc.
- Thrun, Sebastian, Burgard, Wolfram, & Fox, Dieter (2005). *Probabilistic robotics*. Cambridge, Mass.: MIT Press.
- Thrun, Sebastian, Liu, Yufeng, Koller, Daphne, Ng, Andrew Y., Ghahramani, Zoubin, & Durrant-Whyte, Hugh (2004). Simultaneous Localization and Mapping with Sparse Extended Information Filters. *The International Journal of Robotics Research*, 23(7-8), 693-716.
- Thrun, Sebastian, & Montemerlo, Michael (2006). The Graph SLAM Algorithm with Applications to Large-Scale Mapping of Urban Structures. *The International Journal of Robotics Research*, 25(5-6), 403-429.
- Tibshirani, Robert, Walther, Guenther, & Hastie, Trevor (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411-423.
- Tsai, Ching-Chih, Lin, Hung-Hsing, & Lai, Szu-Wei (2005). Multisensor 3D Posture Determination of a Mobile Robot Using Inertial and Ultrasonic Sensors. *Journal of Intelligent and Robotic Systems*, 42(4), 317-335.
- Vadakkepat, P., & Jing, L. (2006). Improved Particle Filter in Sensor Fusion for Tracking Randomly Moving Object. *Instrumentation and Measurement, IEEE Transactions on*, 55(5), 1823-1832.
- Van Aelst, Stefan, Wang, Xiaogang, Zamar, Ruben H., & Zhu, Rong (2006). Linear grouping using orthogonal regression. *Computational Statistics & Data Analysis*, 50(5), 1287-1312.
- Wang, Chieh-Chih (2004). *Simultaneous localization, mapping and moving object tracking*. Carnegie Mellon University, Pittsburgh, PA.
- Weingarten, Jan, & Siegart, Roland (2006). *3D SLAM using planar segments*. Paper presented at the Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on.
- West, Douglas Brent. (2001). *Introduction to graph theory* (2nd ed.). Upper Saddle River, N.J. :: Prentice Hall.
- Wijesoma, W. S., Perera, L. D. L., & Adams, M. D. (2006). Toward multidimensional assignment data association in robot localization and mapping. *Robotics, IEEE Transactions on [see also Robotics and Automation, IEEE Transactions on]*, 22(2), 350-365.
- Wolf, Denis F., & Sukhatme, Gaurav S. (2005). Mobile Robot Simultaneous Localization and Mapping in Dynamic Environments. *Autonomous Robots*, 19(1), 53-65.
- Wong, Y.K., Shi, K.L., & Chan, T.F. (1997). Object Detection by Hough Transform. *Advances in Modelling and Analysis B: Signal Processing and Pattern Recognition*, 38(1), 1-14.
- Xavier, Joao, Pacheco, Marco, Castro, Daniel, Ruanot, Antnio, & Nunes, Urbano (2005). *Fast line*,

-
- arc/circle and leg detection from laser scan data in a player driver*. Paper presented at the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain.
- Xu, Ze, Zhong, Liu, Ji, Lin, & Xiang, Zhi, Yu (2003). *Map building and localization using 2D range scanner*. Paper presented at the Computational Intelligence in Robotics and Automation, 2003. Proceedings. 2003 IEEE International Symposium on.
- Yee, Man Yin, & Vermaak, J. (2005). *A grid-based proposal for efficient global localisation of mobile robots*. Paper presented at the Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on.
- Yifeng, Zhou, & Leung, H. (1997). *Minimum entropy approach for multisensor data fusion*. Paper presented at the Higher-Order Statistics, 1997. Proceedings of the IEEE Signal Processing Workshop on.
- Yuen, D. C. K., & MacDonald, B. A. (2003). *An evaluation of the sequential Monte Carlo technique for simultaneous localisation and map-building*. Paper presented at the Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on, Taipei, Taiwan.
- Yuen, David C. K., & MacDonald, Bruce A. (2003). *Line-based SMC SLAM Method in Environments with Polygonal Obstacles*. Paper presented at the Proceedings of the Australasian Conference on Robotics and Automation, CSIRO, Brisbane, Australia.
- Zhang, Sen, Xie, Lihua, & Adams, Martin (2005). An Efficient Data Association Approach to Simultaneous Localization and Map Building. *The International Journal of Robotics Research*, 24(1), 49-60.
- Zhou, Wei, & Zhao, Rui (2007). *A Combination Evaluation Method Based on FAHP and Entropy Weigh Method*. Paper presented at the Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on.
- Zhu, Yunmin (2003). *Multisensor decision and estimation fusion*. Boston :: Kluwer Academic Publishers.