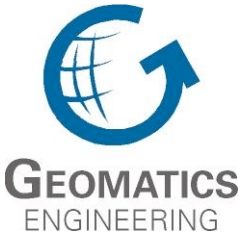

UCGE Reports
Number 20380

Department of Geomatics Engineering

Floor Plan Based Indoor Vision Navigation Using Smart Device

(URL: http://www.geomatics.ucalgary.ca/graduatetheses)

by

Bei Huang

July 2013

UNIVERSITY OF CALGARY

Floor Plan Based Indoor Vision Navigation Using Smart Device

by

Bei Huang

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

GEOMATICS ENGINEERING

CALGARY, ALBERTA

JULY, 2013

Abstract

The Global Positioning System (GPS) nowadays is sized down to a chip sensor and built into almost every smart phone and tablet. Therefore, navigation using those intelligent gadgets becomes a must-have function. GPS has been widely employed for outdoor navigation, while its performance suffers from severe degradation in challenging scenarios such as urban canyon and indoor. Due to the overwhelming signal noise, building reflection and blockage, indoor navigation using GPS frequently encounters poor accuracy or even signal outage. In order to improve the service availability and navigation accuracy, inertial measurement units (IMU) are integrated with GPS, which continuously measures the user acceleration and rotation rate. Integrating these relative motion measurements derives the position, velocity and orientation, therefore it bridges the gap during GPS outage. However, IMU raw measurements are contaminated by sensor bias and drift, and for low-cost sensors on smart devices, the bias and drift are extremely severe and unstable. The navigation solution derived from these poor quality sensors results in significant accumulative errors, which will destroy the system reliability very soon. Furthermore, most smart devices embrace cellular and Wi-Fi network positioning to improve service availability, time-to-first-fix, accuracy and reliability in indoor scenarios. Unfortunately, network based positioning performance highly depends on the signal reception, and quality of the database of Wi-Fi access points (APs) and cellular towers. Based on our experiments, network based positioning performance can merely achieve tens of meters position accuracy on average. For indoor navigation application, however, users' expectation is room-level, turn-by-turn navigation guidance.

In this thesis, a vision navigation system is developed for pedestrian indoor navigation using smart device. In order to derive the three-dimensional camera position from the monocular camera vision, a geo-reference database is needed. Floor plan is a ubiquitous geo-reference database that every building refers to it during construction and facility maintenance. Comparing with other popular geo-reference database such as geo-tagged photos, the generation, update and maintenance of floor plan database does not require costly and time consuming survey tasks. In the proposed system, user is asked to take a picture of the surrounding indoor scenario, and a robust feature matching method is designed to match the indoor features contained in the camera image to those in the floor plan database. Given the image-to-floor plan feature correspondences, a navigation algorithm is developed to integrate the monocular vision with the floor plan geo-reference information and derive the camera position and orientation. The vision navigation system is realized on an iOS App and tested with iPad in various indoor scenarios. The test results show that, comparing with Wi-Fi positioning, the proposed system has improved the position accuracy from tens of meters to 5 m on average.

Acknowledgements

As a master student in Dr. Yang Gao's group, I am the one with least background or experience in the field of positioning and navigation. I still remember the first class of GNSS theory in which I encountered big difficulty when understanding the most basic knowledge about GNSS related technologies. Fortunately, with Dr. Yang Gao's support, encouragement and patience, I started to gradually find my confidence in this department, in this domain and in this snowy country.

I am also grateful to the members of my examining committee, Dr. Ruisheng Wang, Dr. Steve Liang and Dr. Abraham Olatunji Fapojuwo, for their efforts in reading through this thesis.

There are three important people I would like to thank, Mr. Shuang Du, Mr. Yihe Li and Dr. Junbo Shi. Working with them is the most remarkable experience for me, and the critics, encouragements, questions and appreciations from them stimulate me to keep working hard.

The most precious person to thank is my boyfriend, Mr. Hongfu Sun. He has given me so many professional suggestions on software development and inspirations of the most popular technologies. It is really handy to have such a geek boyfriend assist my research.

It is impossible to forget the mental and financial supports from my parents. I am blessed to have them be my dear mom and dad.

Dedication

To my dear parents.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Definition
AP	Access Point
FAST	Feature from Accelerated Segment Test
GPS	Global Positioning System
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
LBS	Location Based Services
LOS	Line of Sight
MAC	Media Access Control
MEMS	Microelectromechanical System
RF	Radio Frequency
RMS	Root Mean Square
RANSAC	Random Sample Consensus
SIFT	Scale-invariant Feature Transform
SLAM	Simultaneously Localization and Mapping
SSID	Signal Strength Identification
STD	Standard Deviation
SURF	Speeded Up Robust Features
TTFF	Time To First Fix
UAV	Unmanned Aerial Vehicle
ZUPT	Zero Velocity Update

List of Symbols

Symbol	Definition
u, v	pixel location of image feature
f	camera focal length
u_0, v_0	pixel location of camera perspective center
P	image feature
Q^{cam}	image feature position in the camera frame
$Q^{floorplan}$	floor plan feature position in the floor plan frame
d	length constraint
α, β	coplanar constraint
δ	residual of length constraint
ε	residual of coplanar constraint
$\left\|\bullet\right\|_2$	norm of vector
J	Jacobian matrix
$R_{cam}^{floorplan}$	camera orientation matrix
pos	camera position
$\text{rank}(\bullet)$	rank of matrix
$(\bullet)^T$	transpose of matrix

Chapter One: **Introduction**

This chapter describes current status of indoor positioning technologies and several pedestrian indoor navigation strategies. In these systems, a large variety of low cost built-in sensors are utilized, including GPS, accelerometer, gyro, magnetic compass and Wi-Fi. However, they suffer from various problems in indoor scenarios such as service outage and poor accuracy. Camera is also a built-in sensor on smart device, and a few vision navigation systems can potentially solve the problems when using other sensors. In order to develop a ubiquitous vision navigation system for pedestrian indoor applications, the advantages and disadvantages of several popular vision navigation methodologies are reviewed.

1.1 Radio Frequency Signal Based Indoor Positioning Systems

The expansion of personal mobile devices such as smart phone and tablet has boosted the popularity of location-based services (LBS). Previously, LBS was a basic concept proposed in the Enhanced 911 (E-911) program in North America, which is designed to promote the efficiency of the traditional emergency calls by accurately linking users and staff to appropriate places and resources. Nowadays, thousands of LBS software are available in the App stores focussing on serving users' daily activities other than dealing with the emergency needs, especially connecting customers with public commercials such as airport and shopping mall. In order to assist costumers finding correct gate and store or wanted item in supermarket, the position and navigation requirement for those scenarios should achieve room-level, or even aisle-level accuracy and turn-by-turn guidance. In outdoor scenarios where the visibility of GPS satellites is good, achieving these performance requirements is not difficult. However, for indoor environments, users always suffer from partial or entire GPS outage due to severe signal noise,

multipath and blockage. Downloading the GPS almanac takes up to 12.5 minutes to acquire satellites, and the acquired satellites should be stably tracked for up to 30 seconds to download the ephemeris data (Lachapelle, 2010). However, GPS signal is badly fragmentary in indoor scenarios, which results in extremely long time-to-first-fix (TTFF) or even tracking failure. Therefore, most of indoor location solutions rely on the hybrid location of GPS, cellular and Wi-Fi network, and are using the initial position obtained from cellular and Wi-Fi network to improve the GPS start-up performance. However, the improvement from the assisted GPS by using wireless networks is rather limited in deep indoor applications where GPS signal is totally unavailable.

In deep indoor scenarios, most LBS applications have to solely rely on cellular and Wi-Fi network based positioning solution. Both cellular and Wi-Fi positioning methods need the reference to a very comprehensive database containing the geolocations of cellular towers and Wi-Fi APs covering the world wide area. Comparing with GPS positioning, cellular and Wi-Fi positioning systems are much more power-economic, so they are particularly attractive for smart devices whose battery life is a concern. Taking advantage of the rapid growth in the early 21st century of the Wi-Fi APs in urban areas, Wi-Fi positioning has now become the most popular indoor positioning technology. Many commercial service providers including Google, Navizon and Skyhook have taken efforts to improve the availability, accuracy and reliability of Wi-Fi positioning system. Most Wi-Fi positioning technologies are based on measuring the received signal strength and their positioning methods can be further categorized into two different types. The first type of method uses the signal strength as an indicator of range. Several Wi-Fi APs with most intensive signal strengths are selected, and their geolocations are requested from the

database server of the service provider. And then multilateration is applied on range measurements between user and the selected Wi-Fi APs to calculate user positions. The second type of method uses the pattern of all received signal strengths as a fingerprint. Since the user's motion will cause minor variation of the fingerprint, this allows the derivation of the user position. Obviously, these two approaches rely on a comprehensive Wi-Fi AP database, which contains the geolocations of APs and their beacon signal strengths.

However, the accuracy of Wi-Fi positioning is not consistent everywhere and it highly depends on the quality of Wi-Fi AP database and the signal reception. On the one hand, the density of the APs in the server database varies dramatically in different scenarios. When user uploads the identifications (usually contain two kinds of identifications, namely Media Access Control (MAC) address and service set identification (SSID)) of all scanned Wi-Fi APs, their geolocations and beacon signal strengths are retrieved from the database. However, if the current area is covered by very sparse APs, the resultant positioning accuracy will be poor. On the other hand, Wi-Fi signal suffers from reflection and blockage by constructions, and disturbance from nearby electrical appliances, which result in rather weak signal reception (Mohammadi, 2011). As for multilateration based Wi-Fi positioning, due to the poor signal quality, the signal propagation model describing the transformation from received signal strength to range becomes inaccurate. As for the fingerprinting based Wi-Fi positioning, the pattern of attenuated signals does not agree well with the fingerprint in database, which results in poor positioning accuracy. According to our indoor tests in various scenarios, Wi-Fi positioning accuracy on average is at tens of meters. With this accuracy level, Wi-Fi positioning can hardly satisfy the required performance of indoor navigation.

1.2 Inertial Navigation System (INS) Based Indoor Positioning Systems

In order to improve the position accuracy and service availability, inertial sensors are widely employed. Current smart devices are equipped with various inertial sensors including accelerometer, gyroscope and magnetic compass. Based on the INS mechanization equations, with given initial position and orientation, the acceleration and rotation rate measurements are integrated over time to derive the user position, velocity and orientation, which can bridge the gap during GPS outage. However, this dead-reckoning system suffers from severe accumulative error due to sensor bias and drift. Especially, the IMU on smart devices is manufactured with the Micro-Electromechanical system (MEMS) technology to reduce the size, weight, power consumption and price. The manufacture process of MEMS sensors has introduced many complex error sources, which results in extremely poor error characteristics such as large bias and drift, and poor run-to-run stability (El-Sheimy, 2003). Basically, using these poor quality measurements during GPS outage will result in tens of meters position error in less than five minutes run. Petovello (2003) and Li (2010) have implemented ultra-tight scheme to integrate high-end IMU to assist GPS signal tracking loop, hence to improve the signal availability in indoor scenario. This approach requires at least tactical grade IMU, e.g. HG1700 by Honeywell, while the signal tracking performance improvement is not prominent in deep indoor scenario.

With such poor error characteristics of low-cost IMU, additional constraints are needed to improve the accuracy of the dead-reckoning algorithm. As for land vehicle application, nonholonomic constraints are frequently used to limit the orientation error. Wang (2006) has implemented fuzzy control to adjust the constraints according to the real dynamics hence to

improve the reliability of navigation solution. As for pedestrian navigation, the foot-mounted IMU assisted with the Zero Velocity Update (ZUPT) is widely implemented in sports applications. When pedestrian steps forward, the foot on ground is static. During this static period, the ZUPT is applied to limit the velocity error growth. Although the foot-mounted IMU is suitable for sports application, as for pedestrian navigation, however, it is not a practical way for pedestrian using smart devices. An improved method was developed by Susie (2012), namely pedestrian gait estimation. In her research, the smart device can automatically detect several common user contexts such as dangling in hand, staying in pocket, reading mode and staying in back-bag. Specifically, the frequency characteristics of IMU measurements vary dramatically in different contexts, and the dominant frequency of measurement indicates the user walking speed and stride length. An empirical gait model based on large amount of sample data is used to describe the relationship between the frequency characteristics of measurement and the stride length. Nevertheless, the gait model should vary from people with different height, age, gender and other aspects, and this is out of the research field of positioning and navigation domain. Moreover, both the ZUPT and pedestrian gait estimation method are dead-reckoning methods, but no absolute positions are available to avoid the accumulative errors.

1.3 Vision Navigation Systems

Vision navigation system is proposed in the computer vision domain to provide reliable navigation guidance for robotic vehicle. Due to its intelligence and reliability, vision navigation gains a lot of attentions from the field of survey and mapping applications. Because the platform of intelligent robotic vehicle is very similar with that of smart devices, which include GPS, IMU and camera, it means vision navigation methodologies can potentially be applied on smart

devices as well. Recent researches about the Augmented Reality (AR) systems also make vision navigation a hot topic, in which the navigation solution is accurate enough to be merged to the real scene in camera view, hence to improve users' navigation experience. Vision navigation system usually consists of three parts, navigation, positioning and vision aiding: first, the navigation part is in charge of continuously tracking robotic motion, and it implements IMU to collect motion measurements; second, the positioning part employs GPS to calibrate the accumulative sensor error, which guarantees the navigation performance in long term use; third, the vision aiding part adopts vision measurements collected from monocular or stereo camera(s) to act like the robotics eyes and arms to "see" and "touch" the surrounding scenarios.

Vision aiding methods can be categorized into two types: stereovision and monocular vision. For both methods, the first step is to apply feature detection methods on camera image flow to extract reliable image features those can be easily recognized and repeatedly detected in upcoming images. After feature detection, the second step is using feature matching techniques to match the features detected in sequential images, and guarantee the camera is tracking the same features in image sequence. Examples of using stereovision and monocular vision in navigation application are introduced in the following sections.

1.3.1 Stereovision Navigation

A typical example of the stereovision based navigation system is the visual Simultaneously Localization and Mapping (SLAM) robotics. SLAM was originally designed for robotic vehicle, and classic SLAM robotics employ laser scanner for mapping. Soloviev (2008) and Trawny et al. (2007) have developed laser scanner based SLAM robotics for land vehicle navigation and

aircraft landing respectively. Comparing with the costly laser scanner, visual SLAM employs two or more cameras to form the stereovision, which can substitute the measurements of laser scanners, hence largely reduce the system costs. Novak and Bossler (1995) have developed a on-vehicle visual SLAM system for traffic mapping, which has largely reduced the system costs in traditional laser scanner based traffic mapping system. Visual SLAM is also adopted in high-end application such as aircraft landing, and Chu et al. (2011) have compared two different integration schemes for stereovision based SLAM robotic vehicle. Given the two-dimensional image features in the images captured from different perspectives, calculating their three-dimensional positions in the world frame is called triangulation. Once these features with known positions are revisited in the upcoming images, calculating the camera position and orientation is called resection. The goal of triangulation is to generate a map of surrounding landmarks (or features), and the goal of resection is to estimate robotics ego-motion such as position, velocity and orientation. Furthermore, the triangulation and resection are two complimentary tasks in the visual SLAM, and they are fulfilled concurrently. Specifically, the mapping of feature positions is accurate only if the robotics ego-motion is accurately known, while the robotics ego-motion estimation becomes more reliable if a precise map of surrounding is available.

Computer vision domain has dedicated in visual SLAM robotics systems and algorithms for decades, with their specialized focus and experts in image processing, pattern recognition and object reconstruction, which enable the visual SLAM robotic has the intelligence to understand the scene it has seen. Se et al. (2005) has developed a visual mobile robot implementing stereovision based SLAM algorithm. Efforts are dedicated to improve the image feature detection and matching quality, though the heavy computation burden only allows the proposed

system to be implemented in post mission. Although the feasibility in real-time mission is limited, the image processing methods implemented in their approach are inspiring for our research. The image processing methods consist image feature detection and robust feature matching. As for image feature detection, Se et al. (2002) have implemented the Scaled-invariant Feature Transform (SIFT) to accurately detect image features in sequential images no matter of the camera view. As for robust feature matching, the Random Sample Consensus (RANSAC) matching proposed by Fischler and Bolles (1981) has been well recognized as the most popular feature matching method. When using SIFT along with RANSAC, the computation burden is extremely heavy. A more light-weight image feature detection method, the Features from Accelerated Segment Test (FAST) corner detection proposed by Rosten and Drummond (2006) is demonstrated for its feasibility in high-speed feature detection.

The accuracy and reliability of mapping and ego-motion estimation highly depends on the triangulation and resection geometry, which is determined by the distribution of features and the alignment of stereo cameras. The stereo cameras with short baseline typically result in poor geometry, because the perspective variation is insignificant and the stereovision is actually highly correlated. Given this fact, the stereovision based SLAM system is not applicable on the platform of smart device. Although a few smart phone manufactures have already released their binocular camera phones for enhanced camera vision, the binocular cameras are separated with only several centimeters as limited by the size of the smart phone. An example of binocular phone HTC EVO is shown in Fig. 1. With such short baseline, triangulation using the resultant stereovision will cause large uncertainty when deriving feature positions, and even destroy the reliability of the visual SLAM system (Lichti, 2011).

Figure 1: HTC three-dimensional phone equipped with binocular cameras (picture downloaded from http://dvice.com/archives/2011/03/htc-brings-3d-t.php)

1.3.2 Monocular Vision Navigation

Given the lack of qualified stereovision on smart device, monocular vision based navigation systems are reviewed. Unlike stereovision, only two-dimensional measurements are available from monocular vision, where the imaging scale is unknown. This unknown imaging scale is easily solved in the field of aerial photogrammetry, because the aerial vehicle height to ground is accurately measured by barometer or laser scanner. An example of using a ground-facing camera as visual odometer for unmanned aerial vehicle (UAV) is presented by Ding et al. (2010). According to the flight tests, their research is demonstrated to achieve meter-level position accuracy during GPS outages. With the inspiration of aerial photogrammetry, Hide et al. (2010) has realized a similar system on pedestrian handheld camera. In their proposed system, the hand held system consists of a ground facing camera and a low-cost MEMS IMU, and the features on the ground are continuously tracked in camera image sequence. Since no barometer or laser scanner is used in their system, the camera height to ground is set to an empirical value, and it is assumed to be stable and constant during the test. The derived trajectory has shown great

accuracy improvement comparing with the IMU-only result. A similar system is developed by Huang et al. (2011), and they have demonstrated that this system is very limited in practice due to two drawbacks: first, the ground-facing pose is not a practical way for pedestrian using smart device; second, unlike forward looking camera pose, there are not many recognizable features in the view of ground-facing camera. These sparse features will disappear from the camera view very soon, and not sufficient observations are available to allow the Kalman filter for feature position and camera ego-motion estimation to converge.

Monocular camera based navigation system is still considered but with forward looking pose, although the problem of unknown imaging scale remains unsolved. Ignoring the lack of imaging scale, Ruotsalainen (2012) has developed a visual compass system on a Nokia camera phone to improve the heading accuracy of pedestrian indoor navigation. The vanishing point tracking method is employed to derive heading, which is used to calibrate the errors of the z-axis gyroscope in smart phone. According to their indoor tests, the heading error is successfully reduced to 2 degree, but no significant improvement is achieved on the position accuracy. Similar approach were proposed by Chu et al. (2012), Prahl and Veth (2011), which are designed for ground vehicle navigation. Field tests were conducted in highly constructed downtown area with straight streets. The vanishing point tracking has shown great improvement on heading accuracy.

Furthermore, the vision navigation methods developed by Hide et al. (2010) and Ruotsalainen (2012) have two drawbacks which limited their feasibility in practice: first, tracking features in continuous camera images requires the camera keep turned-on for a long time to allow the

Kalman filter for the navigation states estimation to converge. But this working mode is not a practical way of using a smart device. It will also cause severe power consumption which will reduce its battery life; second, the feature detection and matching for each frame in the camera image flow is very time-consuming and with computational burden. As for feature detection, some delicate feature detection algorithms are employed in order to repeatedly detect reliable image features, such as the SURF and the Scale-invariant Feature Transform (SIFT), whose processing speed is extremely slow. As for feature matching, identifying the individual correspondences of the thousands of detected features in sequential images makes the computation speed even worse. Hide et al. (2010) have discussed the problem of computation speed, and their tests indicate that the camera sample interval is around 30 ms, but the feature detection and matching process costs 600 ms for each frame on average.

1.3.3 Monocular Vision Navigation with Geo-Reference Database

Considering the abovementioned limitations, some innovative vision navigation methods designed for tourism have attracted our attention, namely “landmark recognition” system. The landmark recognition system aims to solve the problem that, given one photo containing remarkable features such as a theater, museum or other landmarks, the camera position and orientation will be feedback to user. The key technique of retrieving camera position and orientation is matching the camera image with a powerful database containing large amount of photos tagged with geolocations. This geo-tagged photo database is usually collected by survey vehicles, and a practical outdoor application of geo-tagged photo database is Google Street View. Google dispatch survey vehicles equipped with GPS, high-end IMU and cameras periodically running through millions of streets in the world, collecting up-to-date panoramic

images of surrounding buildings and constructions, and concurrently tagging images with street-level accurate GPS positions. For indoor scenarios where GPS is unusable, collection of geo-tagged database is accomplished by the SLAM robotic vehicle with stereo cameras or even laser scanner. An example of indoor landmark recognition system is presented in Yuan et al. (2011). In their system, a SLAM robotic vehicle is sent out to travel around the indoor paths in the area of interest, capture the photo database of indoor scenes, and tag photos with robotic ego-motion. Later on, user revisits this area, uses the monocular camera on smart device, and takes one picture containing recognizable features. The features are detected and matched with geo-tagged photos in the database to find the most similar indoor scene. The geo-tag of the most similar photo in database is used to derive the camera position and orientation. By using their system, the battery life issue and the computation speed problem are no longer challenging because the feature detection and matching are only applied on single camera shot. However, the SLAM robotic vehicle requires the investment of high-end IMU, camera or even laser scanner, which increases the system costs. Moreover, the survey tasks of collecting the geo-tagged photos in a large area of interest are very time-consuming. Comparing with outdoor scenes, indoor scenarios change very often, and it means survey tasks should be conducted frequently to update the database. Although publications about landmark recognition systems have attracted lots of attentions from indoor navigation domain in recent years, due to the database limitations, there is no practical application yet.

1.4 Research Objectives and Contributions

Considering the limitations of the abovementioned systems, this thesis is devoted to the development of a ubiquitous monocular vision navigation system for pedestrian indoor

navigation using smart device. The development of other types of geo-reference database usually requires the use of expensive sensors and survey equipment, and it is time-consuming to collect measurements covering a large area of interest. Therefore, a major focus in this thesis is to apply floor plan database to replace other types of database, hence to reduce the costs of survey tasks and improve the database availability and coverage. Further, considering the severe accumulative error when using dead-reckoning algorithm with low-cost MEMS IMU, a navigation algorithm will be developed for the proposed system to provide absolute positions. Efforts will also be made to improve the accuracy and reliability of the navigation solution when comparing with the current popular indoor positioning systems like Wi-Fi positioning.

The major contribution of this thesis to the field of low-cost pedestrian indoor navigation system can be summarized as follows:

1) A method of generating accurate geo-reference information with floor plan pictures and outdoor maps is developed. The process is easy to implement in practice to produce customized floor plan database for indoor navigation. Comparing with the database collection methods those require high-end equipment and time-consuming survey tasks, the proposed method of using floor plan database has significantly reduced the database costs and improved the database availability and coverage.

2) Development of a navigation algorithm to derive absolute positions from monocular vision and floor plan database. In order to avoid the accumulative errors in the dead-reckoning navigation systems, the navigation algorithm in this thesis will derive absolute camera position and orientation. Two existing methods are modified to improve the

reliability of the navigation algorithm, which are the passive ranging method proposed by Hung et al. (1985) and the camera position and orientation derivation method proposed by Horn et al. (1988). The detailed mathematical model of the navigation algorithm is derived in this thesis.

3) Implementation of a robust feature matching method. Given the detected image features and the floor plan features in the area of interests, the robust feature matching method will automatically identify the image-to-floor plan feature correspondences. Instead of using robust least square to exclude mismatches, the Random Sample Consensus (RANSAC) method is employed for the robust feature matching, and it is demonstrated to be more effective to avoid mismatches.

4) Development of iOS App. Smart phone and tablet are the target devices of the proposed floor plan based monocular vision navigation system. An iOS App is developed to realize the proposed navigation system and algorithm. The reliability, accuracy and computation speed of the iOS App are evaluated with 500 indoor tests in various indoor scenarios. The derived positions are compared with Wi-Fi positioning, and the accuracy of using the proposed navigation system is demonstrated which can improve Wi-Fi position accuracy from tens of meters to 5 m on average. Further, the computation speed is proved suitable for real-time applications.

1.5 Thesis Outlines

Chapter one contains the literature review of the indoor positioning systems using GPS, Wi-Fi network, motion sensors and camera. The vision navigation systems using stereo and monocular camera(s) are our focus. With the discussion of their advantages and disadvantages, the landmark recognition system is of great interest. Considering the limitation of the geo-reference database used in landmark recognition system, the objectives of our system design are introduced to solve the problem encountered by current indoor navigation systems. Moreover, the contributions of this thesis are highlighted in this chapter.

Chapter two introduces the coverage, accessibility and availability of floor plan database. A simple example of generating customized floor plan database is illustrated to demonstrate the ubiquitous of the floor plan database. And then, the limitation when using two-dimensional vision measurements to derive three-dimensional navigation solution is mathematically analyzed. Further, detailed mathematics demonstrates the minimum requirement of geo-reference information to solve this limitation.

Chapter three illustrates the flowchart of the floor plan based vision navigation system structure and explains the function of each component and their relationship. The most important content in this chapter is the derivation of the navigation algorithm, which consists of two steps, the passive ranging method and camera position and orientation derivation. The mathematical model in the passive ranging method is derived, which can reconstruct the three-dimensional feature positions from their two-dimensional image features. Using the results of the passive ranging method, a close-form solution is derived for the camera position and orientation derivation.

Chapter four compares the robust least square based matching with the Random Sample Consensus (RANSAC) method. With several feature matching example tests under control, the RANSAC method is demonstrated to be more effective and reliable for feature matching problem.

Chapter five introduces the software development to realize the proposed system on an iOS App. The screenshots of an example have shown the functions realized in this App. A comprehensive indoor test plan is introduced to evaluate the system performance with the concerns of service availability, navigation algorithm precision and accuracy, performance consistency in various indoor scenarios and computation speed. 500 indoor tests are conducted in The University of Calgary, and the results of the vision navigation system are compared with the Wi-Fi positioning solutions.

Chapter Two: **Floor Plan, Indoor Hallway Features and Vision Measurements**

This chapter introduces a method of generating floor plan database with geo-reference information. The proposed vision navigation system is on the basis of matching floor plan with camera image, but only indoor hallway features are considered as targets in matching process. The definition and example of indoor hallway features are presented in this chapter. The most important part in this chapter is the discussion of the characteristics of monocular vision measurement. Due to the unknown imaging scales of monocular image, it is impossible to reconstruct three-dimensional position from two-dimensional image. However, with the knowledge of geo-reference information about image features, the reconstruction is possible. In this chapter, the minimum requirement of geo-reference information for reconstruction is demonstrated, and detailed mathematics of reconstruction is derived.

2.1 Floor Plan Database

Floor plan is a scaled drawing depicting the indoor arrangement of rooms, hallways and other indoor objects. The scale of floor plan comes from real world measurements of lengths, angles and geodetic coordinates collected by survey equipment and the accuracy of measurements is typically at decimeter level. As for construction and facility maintenance, every building has stored floor plan in online server. Therefore, comparing with other geo-reference database such as geo-tagged photos, using floor plan database saves the labour, time and cost of survey tasks since it can be generated from existing resources. Furthermore, wireless connection through Wi-Fi and 3G networks enables user download floor plan as an indoor map to supplement the insufficiency of traditional outdoor maps like Google Map. Thousands of LBS Apps dedicated in pedestrian indoor navigation have already made good use of floor plan as indoor maps. An

example LBS App, Point Inside, has dedicated in delivering reliable and accurate indoor navigation solutions for years with worldwide coverage of floor plans in major shopping malls and airport as shown in Fig. 2. Enrolled business partners are required to upload their floor plans to make their building supported by Point Inside. With this successful commercialized example of indoor navigation, building a floor plan database to support considerable area of interest has great feasibility in practice.

Figure 2: Floor plan database worldwide coverage of the LBS App, Point Inside (picture downloaded from http://www.pointinside.com/solutions/mapped-locations/)

Borrowing the inspiration from Point Inside, the proposed indoor vision navigation system in this thesis aims to provide innovative and reliable indoor positioning and navigation services for LBS application in major commercial and public places to facilitate their costumers. Any enrolled commercials who want to get their business supported by the indoor vision navigation service are expected to provide us a floor plan. This floor plan will be processed to generate all necessary geo-reference information to be added to our database. In the system developed in

Huang and Gao (2012), the floor plan database is supposed to be already available, while the database generation methods in practice is not introduced. Although lots of commercials provide floor plan for customers as indoor map, these floor plans are not survey-level floor plans and do not contain much geo-reference information, so most of these maps merely serve as pictures of indoor structure, which cannot be used to improve the indoor positioning accuracy. In order to develop a ubiquitous system no matter the commercials have survey-level floor plan database or not, the procedures of generating geo-reference information for newly added floor plan and updating old floor plan database should be easily implemented in practice. The example of the Google Floor Plan project launched by Google Maps team has given us the standard paradigm of generating floor plan database. Google Floor Plan project is a milestone for traditional outdoor Google Maps, which aims to realize its outstanding outdoor map and services in indoor floor plans. The procedure of generating customized floor plan and geo-reference information includes five steps:

1) Search the target building on Google Maps and select the floor layer to add floor plan;

2) Upload floor plan picture, which is widely available for download in most commercials such as shopping mall, airport, library, hospital et al.;

3) Fix several landmarks on the floor plan picture, and these landmarks should be visible on the framework of the building, for example, the wall corners;

4) Stretch the landmarks to coincide with the common points on outdoor Google Maps, hence to perfectly align the floor plan with outdoor map as shown in Fig. 3;

5) Image processing is applied to improve the alignment between the floor plan picture and outdoor Google Maps, and a customized floor plan with geo-reference information is generated.

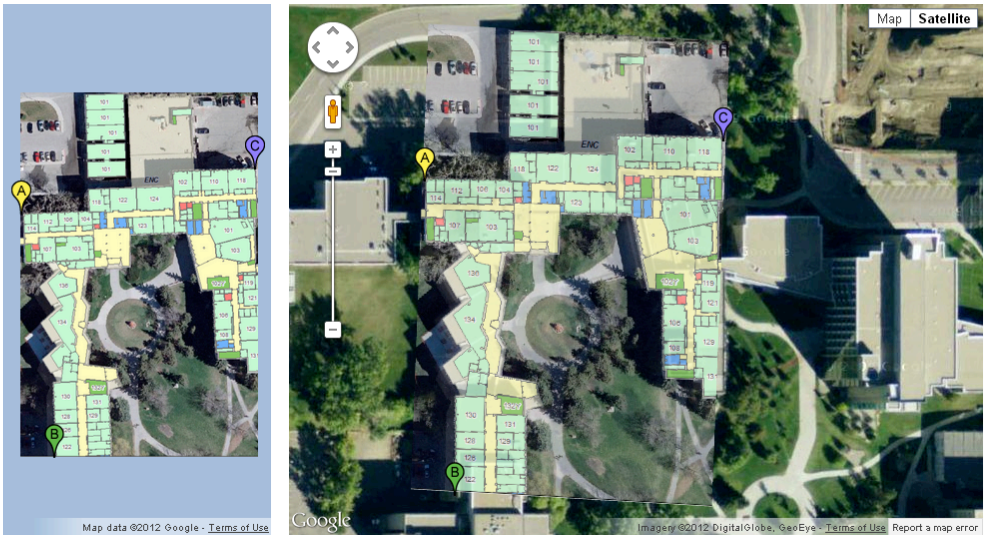

Figure 3: Google Maps Floor Plan program example to add floor plan to traditional outdoor Google Maps

2.2 Geo-reference Information of Floor Plan and Indoor Hallway Features

From the abovementioned paradigm of how Google Maps add floor plans to outdoor maps, the procedures suggest that some important geo-reference information is available in the generated floor plan database. Using the geo-reference information summarized as following, a floor plan frame is constructed, as shown in Fig. 4:

1) *Real scale of floor plan:* Fixing three landmarks on floor plan picture to coincide with corresponding points on Google Maps produces the geodetic positions of these landmarks. Stretching the floor plan picture to align with Google Maps gives the range of

latitude and longitude in the region covered by floor plan. Transforming the range of spherical geodetic positions to the East-North-Up frame, the real scale of the floor plan picture is obtained, and the accuracy of the real scale of floor plan is typically good at decimeter level. As shown in Fig. 4, the floor plan frame is constructed, and the scale of its axes determined by the real scale of floor plan;

2) *Geodetic position and heading of floor plan:* An origin is selected on floor plan, shown as the red dot in Fig. 4, and its geodetic position is available by referring to the Google Maps. Furthermore, once the axes of the floor plan frame are determined, the heading of the floor plan is also available. In the example shown in Fig. 4, the heading angle is the angle between the x-axis and the true North, which is 90 degree. With the local origin and heading, the coordinates transformation between the floor plan frame and the geodetic coordinates is determined.

3) *Floor plan features:* the indoor features and their positions in the floor plan frame are added to the floor plan database, e.g. rooms, paths, turnings and gates, shown as the green dots in Fig. 4. These floor plan features can be extracted by applying imaging processing methods on the floor plan pictures. However, in this thesis, these features are manually selected. Referring to the real scale of floor plan, pixel locations of indoor features are transformed to their positions in the floor plan frame.

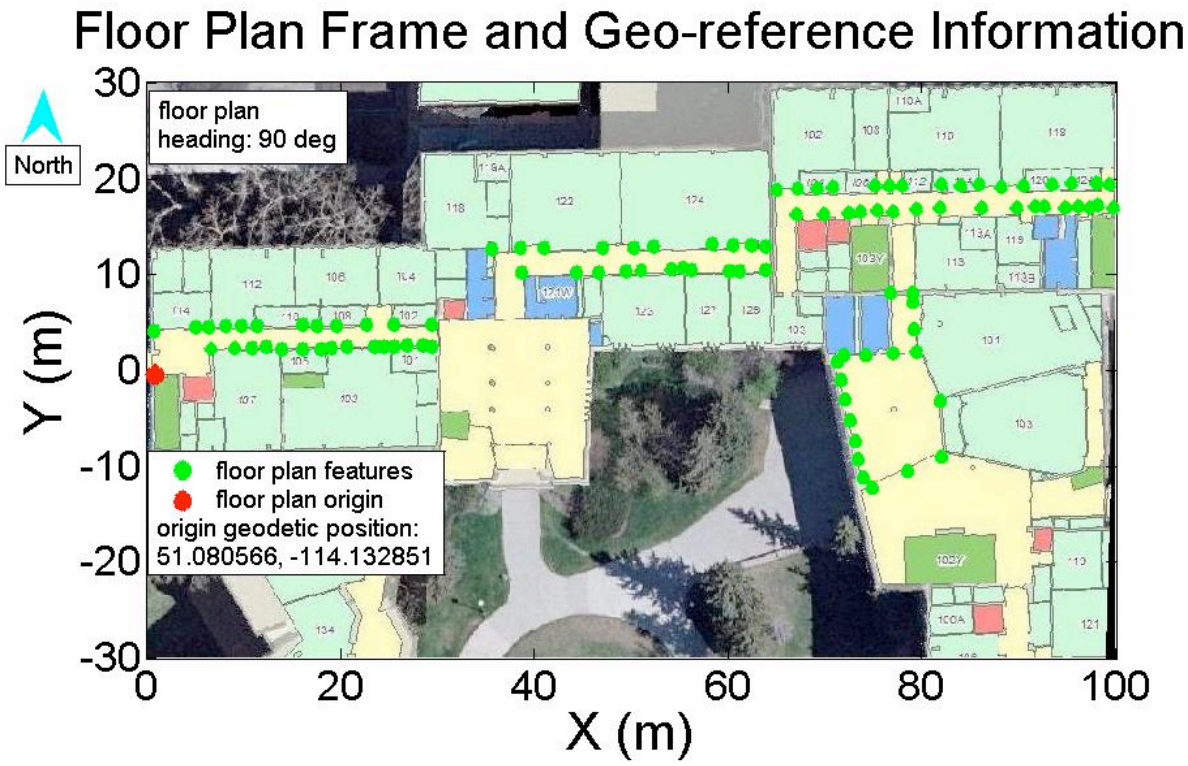

Figure 4: Floor plan frame and geo-reference information

In this thesis, with the inspiration from the landmark recognition system, the proposed system integrates the geo-reference information in the floor plan database with camera image to derive the camera position and orientation. The key technique of integration is to find the most similar indoor scene in the floor plan database to match with the scene in camera image. Fig. 5 shows a picture of a hallway in The University of Calgary, which is a very common indoor scene. A few indoor objects are noticeable such as door, wall, furniture and lights. When choosing these indoor objects as features, we have two concerns: first, when user visit the same indoor scene at different time, the indoor structure and arrangement in the camera scene should not change. Therefore, only fixed indoor objects are chosen as features, instead of furniture, trash bin, et al., since those objects may probably be moved frequently; second, the floor plan picture merely

depict the framework of the indoor structure such as room, doorway and wall, but it ignores details of furniture, lights and other indoor appliances. In order to guarantee that image features can find correspondences in the floor plan database, only the features existing in floor plan are considered. Above all, the corners of doorway and wall, namely indoor hallway features, are selected as targets during feature detection and matching process. As shown in Fig. 5, the red dots illustrate the example of indoor hallway features, and green lines connects pairs of image-to-floor plan feature correspondences.

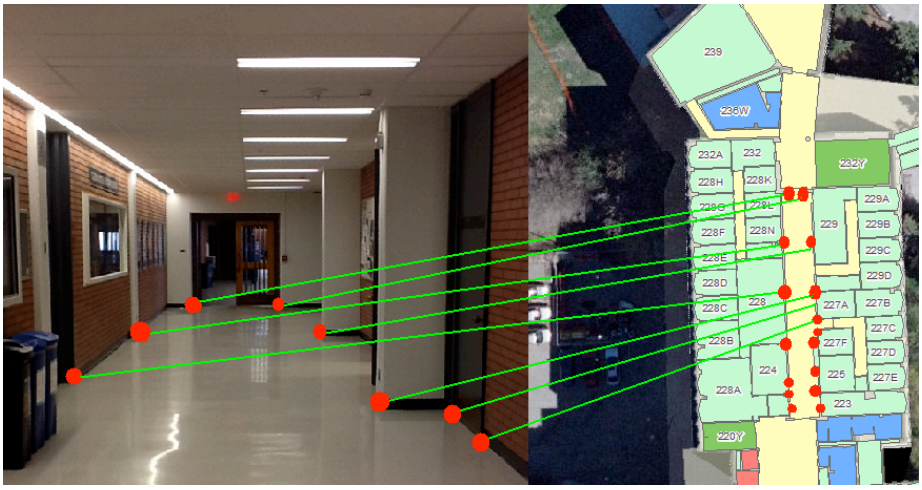

Figure 5: indoor hallway features in image and floor plan

2.3 Image Feature Detection

Feature detection is the first step of many vision tasks, and image features are defined as recognizable pattern of pixels that can be easily, stably and repeatedly detected in image sequence. Line and corner features are usually considered as targets in feature detection, but only corner features are considered in this thesis. Among the most popular corner detectors, the Feature from Accelerated Segment Test (FAST) corner detector is selected due to its simplicity and speed. The FAST corner detection is carried out on the basis of the segment test criterion by

considering a circle of pixels around the candidate pixel. The candidate pixel is classified as a corner if there is a set of n contiguous pixels in the circle which are all brighter than the gray value I_p of the candidate pixel plus a threshold t or darker than that minus threshold. However, when applying the FAST corner detection on the whole image, it is inevitable to extract not only indoor hallway features but also unwanted features such as corners of furniture, lights, bulletin boards et al. Further, sometimes those unwanted features appear to be more intensive than indoor hallway features, so the FAST corner detection probably will only detect the unwanted features but do not recognize the indoor hallway features.

More delicate image processing methods may solve this problem, for example, applying both line detection to find the intersection points as corner features. Since image processing is not a focus in this thesis, this problem is solved with assistance of user interaction. In order to accurately detect the indoor hallway features, the FAST corner detection is not applied on the whole image, but user interaction is needed to touch on the screen of smart device and point out wanted indoor hallway features. User's touches specify the search region to apply the FAST corner detection, but due to the simplicity of the segment test criterion, the FAST detector always detects several corner features even in such small search regions, and one indoor hallway feature is unexpectedly detected as multiple features. Therefore, after applying the FAST corner detection in each user touched region, only the most intensive corner is selected as feature from all detected FAST corners.

2.4 Monocular Vision Measurements

Most of smart phone and tablet are only equipped with monocular camera. Given some novel three-dimensional camera phone has binocular cameras, their stereovision should still be treated as monocular, because the short camera baseline results in highly correlated visions, which is not secured for triangulation. In this section, the geometry of monocular vision measurements employs the pinhole camera model, which is consisted with several elements: a feature object Q, its two-dimensional projection P on image and the position of the camera perspective center O. When the line of sight (LOS) connecting the object point Q and the camera point O pierces through the camera imaging plane, the image point P is formed. So all the three points should be collinear, and their relationship can be mathematically described as Eq. 1 and Eq. 2, which are well known as collinearity equations. Collinearity equations take the inputs of the position of the object point Q and the camera parameters, where the camera parameters are categorized in to two types:

1) Interior parameters including the camera focal length, and the pixel location of the image principal center, which is the projection point on image of the camera perspective center.

2) Exterior parameters including camera position and camera orientation matrix. In our case of deriving navigation solutions, the exterior parameters are unknowns.

Due to the art of manufacture, the image principal center does not necessarily locate in the middle of image. Camera calibration is expected to be finished before practical use to get interior parameters and they are treated as constant. For high-precision applications, camera lens

distortion should also be included into interior parameters, and they can also be obtained through camera calibration procedure. But in our case, Eq. 1 and Eq. 2 are the most basic model without consideration of lens distortion.

$$\begin{bmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \end{bmatrix} = R_{world}^{cam} \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} u_0 + x_{cam} \cdot f / z_{cam} \\ v_0 + y_{cam} \cdot f / z_{cam} \end{bmatrix} \tag{2}$$

where, $\begin{bmatrix} x_0 & y_0 & z_0 \end{bmatrix}^T$ is the position of the camera perspective center O in the world frame; $\begin{bmatrix} x & y & z \end{bmatrix}^T$ is the position of the object Q in the world frame; $\begin{bmatrix} x_{cam} & y_{cam} & z_{cam} \end{bmatrix}^T$ is the transformed position of the object Q in the camera frame; R_{world}^{cam} is the camera orientation matrix to transfer position from the world frame to the camera frame; $\begin{bmatrix} u & v \end{bmatrix}^T$ is the pixel location of the point P in the image frame; $\begin{bmatrix} u_0 & v_0 \end{bmatrix}^T$ is the pixel location of the camera perspective center O on image; f is the camera focal length in unit of pixel.

If the exterior parameters of camera position and orientation are given, each object position is uniquely mapped to a pixel location. But it is impossible to reverse this mapping because one pixel location does not uniquely correspond to an object position. This fact brings difficulty when using monocular vision measurement, because given a set of image features, each of them does not uniquely correspond to an object in the real world. To better understand this fact, homography coordinates are introduced as shown in Eq. 3 to expand two-dimensional pixel

location of P to three-dimensional. With homography coordinates, the collinearity equations in Eq. 1 and Eq. 2 can be rewritten with Cartesian coordinates and matrix multiplication as show in Eq. 3. Homography coordinates also mathematically explains the fact that the pixel location P of image feature does not uniquely correspond to one object position Q in the world frame, because in Eq. 3 multiplying Q with an unknown scale k still result in the same pixel location.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f/z_{cam} & 0 & u_0 \\ 0 & f/z_{cam} & v_0 \\ 0 & 0 & 1/z_{cam} \end{bmatrix} \cdot \begin{bmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \end{bmatrix} = \begin{bmatrix} f/k \cdot z_{cam} & 0 & u_0 \\ 0 & f/k \cdot z_{cam} & v_0 \\ 0 & 0 & 1/k \cdot z_{cam} \end{bmatrix} \cdot \begin{bmatrix} k \cdot x_{cam} \\ k \cdot y_{cam} \\ k \cdot z_{cam} \end{bmatrix} \tag{3}$$

2.5 Reconstructing Imaging Scale of Monocular Vision Using Passive Ranging

In order to reconstruct the unknown imaging scale of monocular vision, the passive ranging method proposed by Hung et al. (1985) is implemented. According to their demonstration, given an image of features and the relative spatial locations of features, the ranges from the camera to individual feature can be derived. A question arises that how many features are needed to form the basic geometry for passive ranging. To find the answer, Fischler and Bolles (1981) have demonstrated that, given three image features, their corresponding object positions in the world frame form a triangle pattern, and if the side lengths of triangle are known, this dataset is the minimum set leading to finite solutions of imaging scales. Specifically, there are four possible solutions, and each one should be verified to determine the unique solution of imaging scales. Hung et al. (1985) have proved that, given four image features whose corresponding positions are coplanar and form a quadrangle pattern with at least one known length, it is able to further

narrow down four possible solutions to a unique one. In the following section, the unique solution of imaging scales from four coplanar features is derived.

As shown in Fig. 6, when four features P_1, P_2, P_3 and P_4 are detected in camera image, their pixel locations are expressed in homography coordinate as shown in Eq. 4. The three-dimensional feature positions Q_1, Q_2, Q_3 and Q_4 can be determined with the pixel locations and the imaging scales as shown in Eq. 5.

$$P_1 = \begin{bmatrix} u_1 & v_1 & 1 \end{bmatrix}^T \quad P_2 = \begin{bmatrix} u_2 & v_2 & 1 \end{bmatrix}^T \quad P_3 = \begin{bmatrix} u_3 & v_3 & 1 \end{bmatrix}^T \quad P_4 = \begin{bmatrix} u_4 & v_4 & 1 \end{bmatrix}^T \tag{4}$$

$$Q_1 = k_1 P_1 \qquad Q_2 = k_2 P_2 \qquad Q_3 = k_3 P_3 \qquad Q_4 = k_4 P_4 \tag{5}$$

where, P is the image feature with pixel location of $\begin{bmatrix} u & v \end{bmatrix}^T$; k is the unknown imaging scale; Q is the feature position in the world frame.

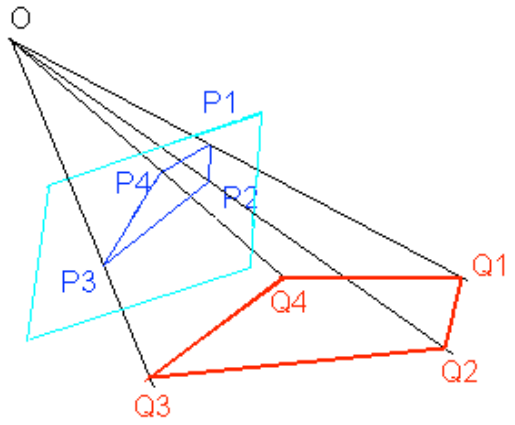

Figure 6: illustration of solving feature ranges with four image features

Assume these four features are coplanar, so their coordinates must satisfy the coplanar equation as shown in Eq. 6. α and β are nonzero constant factors, and given these two factors, the shape of the quadrangle formed by the four features is fixed, shown as the red quadrangle in Fig. 6.

$$Q_4 = (1-\alpha-\beta)Q_1 + \alpha Q_2 + \beta Q_3 \tag{6}$$

where, α and β are known factors, which describe the coplanar relationship of four feature points.

However, since multiplying scales on both sides of Eq. 6 will also make the equation satisfied, α and β only narrow down the possible imaging scales with a series of similar quadrangles but cannot lead to a unique solution of imaging scales. Therefore, instead of solving the imaging scales, only the ratios of scales as shown in Eq. 7 are solved as shown in Eq. 8.

$$P_4 = (1-\alpha-\beta)P_1 k_1/k_4 + \alpha P_2 k_1/k_4 + \beta P_3 k_1/k_4 \tag{7}$$

$$\begin{bmatrix} k_1/k_4 \\ k_2/k_4 \\ k_3/k_4 \end{bmatrix} = \begin{bmatrix} (1-\alpha-\beta)P_1 & \alpha P_2 & \beta P_3 \end{bmatrix}^{-1} P_4 \tag{8}$$

Moreover, at least one length should be given as shown in Eq. 9 to finally fix the real scale of the quadrangle, hence lead to the unique solutions of imaging scales. Finally, by taking the derived scales into Eq. 5, the three-dimensional feature coordinates in the camera frame are also calculated.

$$\left\| Q_1 - Q_4 \right\|_2 = \sqrt{\left(k_1 P_1 - k_4 P_4\right)^{\mathrm{T}} \left(k_1 P_1 - k_4 P_4\right)} \tag{9}$$

where, $\|\cdot\|_2$ calculates the Euclidean distance between two feature positions.

Chapter Three: **Navigation System and Algorithm**

In this chapter, all frames used in the proposed navigation system are defined. The system structure is explained with different function blocks and flowchart. The navigation algorithm is introduced and the mathematical models are derived for the passive ranging method and the derivation of camera position and orientation.

3.1 Definition of Frames

All coordinates involved in this thesis are expressed in three types of frames, the floor plan frame, the camera frame and the image frame, and all of them are illustrated in Fig. 7 and defined as following:

1) Floor plan frame: a three-dimensional world frame with origin selected on the local floor plan, x-axis pointing along the hallway, z-axis pointing up and y-axis orthogonal with both, shown as the green axes in Fig. 7;

2) Camera frame: a three-dimensional frame with origin at the camera perspective center, x-axis pointing right, y-axis pointing up, z-axis orthogonal to the camera imaging plane, shown as the red axes in Fig. 7;

3) Image frame: a two-dimensional frame of the camera imaging plane. It departs from the camera perspective center with the distance of focal length, shown as the blue axes in Fig. 7.

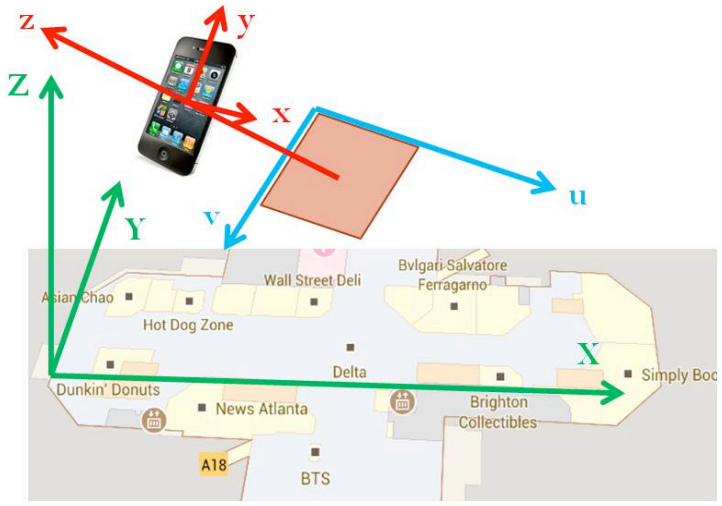

Figure 7: Definition of frames

According to the geo-reference information available in the floor plan frame introduced in the Chapter two, the scale of the floor plan frame is determined by the real scale of the floor plan; the geodetic position of the selected local origin and the heading of floor plan are all available in the floor plan database. In the navigation algorithm, the derived camera position is expressed in the floor plan frame. But using the geodetic position and the heading of floor plan, the camera position can be transformed from the floor plan frame to geodetic frame.

3.2 System Flowchart

The system flowchart consists of six components as shown in Fig. 8, and their individual functions are described as following:

1) *Initial position and accuracy:* The first component is in charge of collecting initial indoor position. The default indoor positioning method on smart device platform is GPS/Cellular/Wi-Fi hybrid location, but in most deep indoor scenarios, GPS signal is

totally unavailable. The indoor positioning technology on smart device relies on Wi-Fi network, and the accuracy of initial indoor position in this thesis is most of time at tens of meters;

2) *Floor plan database:* The second component is in charge of downloading the floor plan geo-reference information from the database, and this is completed with the geo-fencing function. Geo-fencing is an important technique adopted in LBS software, and it is defined as a technique to detect users' location when they are entering or leaving an area containing the points of interest and send notifications to their smart device with the information about the points of interest. In this thesis, the geo-fencing function is also needed when user has entered the area where the floor plan database is available. When user sending request message to server including the initial position and its accuracy, the areas of interest are determined. Then the feedback message is sent to user device which includes the floor plan pictures and geo-reference data of the indoor hallway features in the areas of interest;

3) *Photo of indoor scenario:* The third component requires user to take a picture of the indoor scenario containing as many indoor hallway features as possible. Usually 15 features are sufficient to derive reliable camera position and orientation. A suggested way to take picture is standing at the ends of hallways with camera looking forward. Then user interaction is needed to touch on screen and specify indoor hallway features;

4) *Image feature detection:* Instead of processing the whole image, the user-touched areas in the third component specify the search regions to apply the image feature detection. Specifically, the FAST corner detection is applied on user-touched regions to extract the exact pixel locations of indoor hallway features. Usually there are several features being detected in each region, but only the most intensive feature is selected.

5) *Robust feature matching:* The fifth component is in charge of the robust matching task. The feedback from server contains a few floor plan features and user also specifies several image features, but their corresponding relationship is not figured out. In order to let the software automatically identify the image-to-floor plan correspondences, the RANSAC method is applied. The RANSAC method is in fact based on a series of iterative random guesses, where the navigation algorithm nests inside the RANSAC iterations. Details about the RANSAC method are introduced in the Chapter four;

6) *Navigation algorithm:* The sixth component takes the image-to-floor plan correspondences as inputs to the navigation algorithm. The navigation algorithm contains two steps, the passive ranging method to derive the feature three-dimensional positions in the camera frame, and the derivation of camera position and orientation. The camera position is demonstrated in the Chapter five to have improved accuracy and reliability comparing with the initial position.

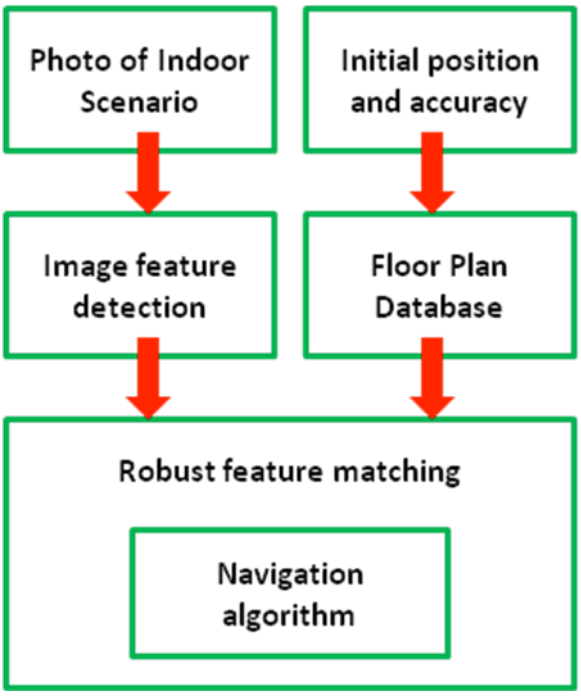

Figure 8: System flowchart

3.3 Navigation Algorithm

In this section, the navigation algorithm is developed, and Fig. 9 shows the structure of the navigation algorithm, which contains two steps, namely the passive ranging method and the camera position and orientation derivation. In the first step, the inputs for the passive ranging include the two-dimensional pixel locations of the image features and the three-dimensional positions of the floor plan features in the floor plan frame. The output of the passive ranging is the three-dimensional positions of the image features in the camera frame. In the second step, the inputs for the camera position and orientation derivation method include two sets of three-dimensional coordinates, which are image feature positions in the camera frame and the corresponding floor plan feature positions in the floor plan frame.

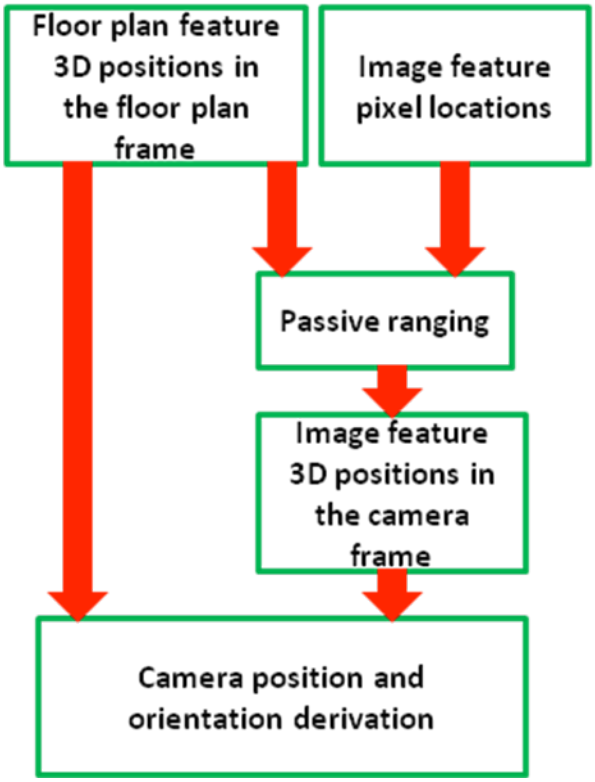

Figure 9: Navigation algorithm structure

3.3.1 The Passive Ranging Method

The first step of the navigation algorithm is to reconstruct the three-dimensional feature position in the camera frame from its two-dimensional pixel location. On the basis of the derivation in the Chapter two, given four image features, if their corresponding feature positions in three-dimensional space are coplanar and form a quadrangle pattern, then with the knowledge of the quadrangle shape and at least one known length, the unique set of imaging scales are derived, and the feature three-dimensional positions can be reconstructed. However, the indoor scene of camera image always contain more than four image features, so the mathematical model of the passive ranging method is derived in this chapter to deal with general case.

Similar with derivation in the Chapter two, the image feature position in the camera frame is determined by the pixel location of image feature and an unknown imaging scale, as shown in Eq. 10. All imaging scales consist the vector of unknowns as shown in Eq. 11.

$$Q_i^{cam} = k_i \cdot \begin{bmatrix} u_i & v_i & 1 \end{bmatrix}^T \tag{10}$$

$$K = \begin{bmatrix} k_1 & k_2 & \cdots & k_n \end{bmatrix} \tag{11}$$

where, Q^{cam} is the feature position in the camera frame; u and v are pixel location of image feature; K is the unknown vector of feature scales;

Eq. 12 and Eq. 13 have mathematically described the coplanar relationship and Euclidian distance as functions of the feature positions in the camera frame. According to the discussion in the Chapter two, the real distance between features and the two constants of coplanar relationship should be obtained from the floor plan database, namely length constraint and coplanar constraint respectively.

$$\left\| Q_i^{cam} - Q_j^{cam} \right\|_2 = d_{ij} + \delta_{ij} \tag{12}$$

$$Q_{i_1}^{cam} - (1 - \alpha - \beta) Q_{j_1}^{cam} - \alpha Q_{i_2}^{cam} - \beta Q_{j_2}^{cam} = \varepsilon_{i_1 j_1}^{i_2 j_2} \tag{13}$$

where, d_{ij} is the real distance, while α and β are constant factors for coplanar relationship; δ and ε are residuals of distance and coplanar relationship respectively;

The key of integration between image and floor plan is to find out the length and coplanar constraints by referring the image features to their corresponding floor plan features.

Specifically, the length constraints are quite straightforward to calculate from the feature positions in the floor plan frame as shown in Eq. 14, and the calculation of the two constant factors of the coplanar constraints are shown in Eq. 14-19.

$$d_{ij} = \left\| Q_i^{floorplan} - Q_j^{floorplan} \right\|_2 \tag{14}$$

$$Q_{i_1}^{floorplan} - (1-\alpha-\beta) Q_{j_1}^{floorplan} - \alpha Q_{i_2}^{floorplan} - \beta Q_{j_2}^{floorplan} = 0 \tag{15}$$

$$\cos\theta_1 = \left(Q_{i_2}^{floorplan} - Q_{i_1}^{floorplan} \right)^T \left(Q_{j_1}^{floorplan} - Q_{i_1}^{floorplan} \right) \Big/ \left(\left\| Q_{i_2}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 \cdot \left\| Q_{j_1}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 \right) \tag{14}$$

$$\cos\theta_2 = \left(Q_{j_2}^{floorplan} - Q_{i_1}^{floorplan} \right)^T \left(Q_{j_1}^{floorplan} - Q_{i_1}^{floorplan} \right) \Big/ \left(\left\| Q_{j_2}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 \cdot \left\| Q_{j_1}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 \right) \tag{15}$$

$$\theta_3 = \pi - \theta_1 - \theta_2 \tag{16}$$

$$\sin\theta_3 \Big/ \left\| Q_{j_1}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 = \sin\theta_1 \Big/ \beta \left\| Q_{j_2}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 = \sin\theta_2 \Big/ \alpha \left\| Q_{i_2}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 \tag{17}$$

$$\alpha = \sin\theta_2 \left\| Q_{j_1}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 \Big/ \sin\theta_3 \left\| Q_{i_2}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 \tag{18}$$

$$\beta = \sin\theta_1 \left\| Q_{j_1}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 \Big/ \sin\theta_3 \left\| Q_{j_2}^{floorplan} - Q_{i_1}^{floorplan} \right\|_2 \tag{19}$$

where, $Q^{floorplan}$ is the feature position in the floor plan frame.

In the passive ranging method, any four pairs of image-to-floor plan correspondences contribute an equation of coplanar relationship as shown in Eq. 13, and each pair of correspondences contributes an equation of distance as shown in Eq. 12. Therefore, when more than four image features are detected and matched with floor plan, the resultant observations are highly redundant. Least square method is applied in the passive ranging to derive the unknown imaging scales, which can make the derived feature positions in the camera frame best-fit with the length and coplanar constraints. Apparently, the more image-to-floor plan correspondences are found,

the more reliable and accurate camera position is derived. The mathematical model is linearized and the Jacobian matrix is calculated as shown in Eq. 20-22, and the normal equations are shown as Eq. 23.

$$J = \begin{bmatrix} J_{coplanar} \\ J_{length} \end{bmatrix} \tag{20}$$

$$J_{coplanar} = \frac{\partial\left(Q_{i_1}^{cam} - (1-\alpha-\beta)Q_{j_1}^{cam} - \alpha Q_{i_2}^{cam} - \beta Q_{j_2}^{cam}\right)}{\partial K} \tag{21}$$

$$J_{length} = \frac{\partial\left\|Q_i^{cam} - Q_j^{cam}\right\|_2}{\partial K} \tag{22}$$

$$J^T J \cdot \partial K = J^T \begin{bmatrix} \varepsilon \\ \delta \end{bmatrix} \tag{23}$$

where, $J_{coplanar}$ and J_{length} are Jacobian matrix for coplanar relationship equation and distance equation respectively.

By taking the calculated imaging scales back to Eq. 10, each two-dimensional pixel location of image feature is reconstructed to three-dimensional position in the camera frame. Moreover, according to the derivation in the Chapter two, the four coplanar features are required to form quadrangle pattern, which means if three of them are collinear features, the geometry is not sufficient to derive unique solution of imaging scales. However, in practice, structures of indoor scenarios are very unpredictable and modern architectures especially have very irregular and complex indoor structures. When user touches on the screen to specify the indoor hallway features, it is possible that the geometry formed by the image features' correspondences does not

sufficiently lead to a unique solution of imaging scales. Therefore, it is necessary to examine whether or not the design matrix in Eq. 23 is near to singular matrix.

3.3.2 Camera Position and Orientation Derivation

Following passive ranging, the second step of the navigation algorithm is to derive camera position and orientation. After passive ranging, the feature positions in the camera frame are obtained, hence their ranges to camera are also known. With their corresponding floor plan features positions, multilateration method was implemented on the feature ranges to solve camera position. However, the geometry of indoor hallway features often results in failure of using multilateration in practice. Specifically, all indoor hallway features locate in front of user, and sometimes their mutual distances are quite close while their ranges to user are distant. As a result, the LOS observations of those features are always highly correlated. This poor geometry will either cause large uncertainty when deriving the camera position using the multilateration method.

Given the failure of multilateration, the method of deriving camera position is developed on the basis of the method proposed in Horn et al. (1988), in which the camera orientation is also derived. With the inspiration of their contributions, our problem is interpreted as following: after applying the passive ranging method, the three-dimensional feature positions in the camera frame are reconstructed. Referring to their correspondences in floor plan, the floor plan feature positions in the floor plan frame are paired with the reconstructed feature positions. These two sets of coordinates should be mutually transformable via a translation and a rotation as shown in

Eq. 24. The translation is equivalent with the camera position, and the rotation is equivalent with the camera orientation matrix.

$$R_{cam}^{floorplan} Q_i^{cam} = Q_i^{floorplan} - pos \tag{24}$$

where, pos is the unknown camera position; $R_{cam}^{floorplan}$ is the unknown camera orientation matrix; Q_i^{cam} is the i^{th} feature coordinates in the camera frame and $Q_i^{floorplan}$ is its corresponding floor plan feature position in the floor plan frame;

Furthermore, two concerns are taken care in this thesis to guarantee the accuracy and reliability of the camera position and orientation derivation:

1) *Best fits of two sets of coordinates:* Three pairs of coordinates are sufficient to derive a rotation matrix. In our case, however, in order to implement the passive ranging method, there are usually more than four features available, which result in redundancy when deriving the camera orientation matrix. Therefore, it is necessary to calculate a least square based solution to make the transformed feature positions in the camera frame to best fit with their correspondences in the floor plan frame. This least square problem is described in Eq. 25

$$\min \sum_{i=1}^{n} \left(R_{cam}^{floorplan} Q_i^{cam} - \left(Q_i^{floorplan} - pos \right) \right) \tag{25}$$

where, n is the total number of image-to-floor plan correspondences.

2) *Orthogonality:* An accurate camera orientation matrix is the premise of the derivation of camera position, and the most important property of orientation matrix is orthogonality as

shown in Eq. 26. It is necessary to preserve the orthogonal matrix property when deriving the orientation matrix. Specifically, when the orthogonal orientation matrix is derived to be the best-fit rotation from the camera frame to the floor plan frame, its transpose matrix directly gives the best-fit rotation from the floor plan frame back to the camera frame. In another word, preserving the orthogonality of the camera orientation matrix will ensure the finally derived transformation to be the best solution to make two sets of features in the camera frame and in the floor plan frame mutually transformable.

$$R_{floorplan}^{cam} = \left(R_{cam}^{floorplan}\right)^{-1} = \left(R_{cam}^{floorplan}\right)^{T} \tag{26}$$

With the abovementioned two concerns, the close-form solution derived in Horn et al. (1988) is modified in this thesis. At first, the least square based transformation is derived, and this problem is equivalent with finding the camera position and orientation parameters to solve the minimization problem as shown in Eq. 25. But with the unknown camera position, it is difficult to directly solve the problem. Instead, the sum of squares in the minimization problems shown in Eq. 25 is re-parameterized to that in Eq. 28. In order to eliminate the effect of unknown camera position, we expect the second and third terms containing camera position equal to zero.

$$R_{cam}^{floorplan} Q_i^{cam} - Q_i^{floorplan} + pos = R_{cam}^{floorplan} Q_i^{'cam} - Q_i^{'floorplan} + pos' \tag{27}$$

$$\begin{aligned} &\sum_{i=1}^{n}\left(R_{cam}^{floorplan} Q_i^{cam} - Q_i^{floorplan} + pos\right)^2 = \sum_{i=1}^{n}\left(R_{cam}^{floorplan} Q_i^{'cam} - Q_i^{'floorplan} + pos'\right)^2 \\ &= \sum_{i=1}^{n}\left(R_{cam}^{floorplan} Q_i^{'cam} - Q_i^{'floorplan}\right)^2 + 2pos'\sum_{i=1}^{n}\left(R_{cam}^{floorplan} Q_i^{'cam} - Q_i^{'floorplan}\right) + n \cdot pos'^2 \end{aligned} \tag{28}$$

As for making the second term in Eq. 28 equal to zero, the feature positions in both of camera frame and floor plan frame are centered to origin by deducting with their mean, as shown in Eq. 29-30. Then the sum of the new feature positions is zero, and only the third term is left in Eq. 28; as for making the third term in Eq. 26 equal to zero, the Eq. 31 is obtained, and it provide a clue that the camera position can be derived after the camera orientation is calculated.

$$\bar{Q}_{cam} = \sum_{i=1}^{n} Q_i^{cam} \Big/ n \qquad \bar{Q}_{floorplan} = \sum_{i=1}^{n} Q_i^{floorplan} \Big/ n \tag{29}$$

$$Q_i^{'cam} = Q_i^{cam} - \bar{Q}_{cam} \qquad Q_i^{'floorplan} = Q_i^{floorplan} - \bar{Q}_{floorplan} \tag{30}$$

$$pos' = pos - R_{cam}^{floorplan}\bar{Q}_{cam} - \bar{Q}_{floorplan} = 0 \tag{31}$$

where, $\bar{Q}_{cam}$ is the mean vector of all feature coordinates in the camera frame; $\bar{Q}_{floorplan}$ is the mean vector of all feature coordinates in the floor plan frame;

Through the previous steps, only the first term is left in the minimization problem, as shown in Eq. 32. And then, Eq. 32 is expended as shown in Eq. 33, where the first and third terms are irrelevant to camera orientation. Now, the minimization problem is solved only if the camera orientation is found to maximize the second term, as shown in Eq. 34.

$$\min \sum_{i=1}^{n} \left(R_{cam}^{floorplan} Q_i^{cam} - Q_i^{floorplan} + pos \right)^2 = \min \sum_{i=1}^{n} \left(R_{cam}^{floorplan} Q_i^{'cam} - Q_i^{'floorplan} \right)^2 \tag{32}$$

$$\begin{aligned} &\sum_{i=1}^{n} \left(R_{cam}^{floorplan} Q_i^{'cam} - Q_i^{'floorplan} \right)^2 \\ &= \sum_{i=1}^{n} Q_i^{'camT} Q_i^{'cam} - 2\sum_{i=1}^{n} Q_i^{'floorplanT} R_{cam}^{floorplan} Q_i^{'cam} + \sum_{i=1}^{n} Q_i^{'floorplanT} Q_i^{'floorplan} \end{aligned} \tag{33}$$

$$\max \sum_{i=1}^{n} Q_i^{'floorplanT} R_{cam}^{floorplan} Q_i^{'cam} = \max \left(R_{cam}^{floorplan} \right)^T \sum_{i=1}^{n} Q_i^{'floorplan} Q_i^{'camT} \tag{34}$$

Let the production of the two sets of feature coordinates to be represented by the square matrix M. The trick is to decompose M to the product of an orthogonal matrix U and a symmetric matrix S as shown in Eq. 35. If the matrix M is nonsingular, it is easy to conduct such decomposition as shown in Eq. 36. Obviously, the matrix U is orthogonal and the matrix S is symmetric. In the following derivation, the trick of decomposing the matrix M is the key to guarantee the derived camera orientation matrix is orthogonal.

$$M = \sum_{i=1}^{n} Q_i^{'floorplan} Q_i^{'camT} = US \tag{35}$$

$$U = M\left(M^T M\right)^{-1/2},\ S = \left(M^T M\right)^{1/2} \tag{36}$$

where, U is orthogonal matrix which holds the property $UU^T = I$; S is symmetric matrix which holds the property $S = S^T$.

Therefore, the maximization problem in Eq. 34 is now rewritten in Eq. 37. The symmetric matrix S is at first considered. If M is nonsingular, the symmetric matrix S is positive definite which means all eigenvalues of S are positive. Therefore, the matrix S is decomposed to its eigenvalues and corresponding eigenvectors as shown in Eq. 38.

$$\max\left(R_{cam}^{floorplan}\right)^T \sum_{i=1}^{n} Q_i^{'floorplan} Q_i^{'camT} = \max\left(R_{cam}^{floorplan}\right)^T US \tag{37}$$

$$S = \sqrt{\lambda_1}\, u_1 u_1^T + \sqrt{\lambda_2}\, u_2 u_2^T + \sqrt{\lambda_3}\, u_3 u_3^T \tag{38}$$

where, λ_1, λ_2 and λ_3 three eigenvalues of S; u_1, u_2 and u_3 are their corresponding eigenvectors.

However, the matrix M calculated from Eq. 35 is singular in our case, because all indoor hallway features are coplanar, which makes the matrix S has zero eigenvalue. Although both the feature coordinates in the camera frame $Q_i^{'cam}$ and those in the floor plan frame $Q_i^{'floorplan}$ are coplanar, two normal vectors of these two sets of coordinates are calculated. In order to make matrix M nonsingular, these two normal vectors are appended to the matrix M as shown in Eq. 39.

$$M = \sum_{i=1}^{n} Q_i^{'floorplan} Q_i^{'camT} + v_{floorplan} v_{cam}^{T}, \ \mathrm{rank}\left(M\right) = 3 \tag{39}$$

where, $\mathrm{rank}\left(\cdot\right)$ is the rank of matrix; v_{cam} and $v_{floorplan}$ are the normal vectors orthogonal to the feature coordinates in the camera frame and in the floor plan frame respectively;

Back to the maximization problem, now it is rewritten as Eq. 40. It is easy to prove the matrix U shown in Eq. 36 is orthogonal. So the camera orientation matrix $R_{cam}^{floorplan}$ should also be orthogonal.

$$\begin{aligned} &\max\left(R_{cam}^{floorplan}\right)^{T} US = \max\left(R_{cam}^{floorplan}\right)^{T} US \\ &= \sqrt{\lambda_1}\max\left(R_{cam}^{floorplan}u_1\right)^{T} Uu_1 + \sqrt{\lambda_2}\max\left(R_{cam}^{floorplan}u_2\right)^{T} Uu_2 + \sqrt{\lambda_3}\max\left(R_{cam}^{floorplan}u_3\right)^{T} Uu_3 \end{aligned} \tag{40}$$

Since the eigenvectors in Eq. 39 are unit vectors, we have Eq. 41. The equality is satisfied only when the camera orientation matrix $R_{cam}^{floorplan}$ equals to the matrix U. Therefore, the least square based camera orientation is obtained as shown in Eq. 42 while its orthogonality is perfectly

persisted as well. By taking the derived camera orientation back to Eq. 31, the camera position is finally calculated.

$$\max\left(R_{cam}^{floorplan}u_i\right)^T Uu_i = \max\left(u_i^T\left(R_{cam}^{floorplan}\right)^T Uu_i\right) \le \left\|u_i\right\|_2^2 = 1 \tag{41}$$

$$R_{cam}^{floorplan} = U = M\left(u_1u_1^T/\sqrt{\lambda_1} + u_2u_2^T/\sqrt{\lambda_2} + u_2u_2^T/\sqrt{\lambda_2}\right) \tag{42}$$

Chapter Four: **Robust Matching Methods**

The navigation algorithm derives camera position from the correspondences of the indoor hallway features in camera image and the floor plan database. Therefore, the robust matching problem should be solved to find out reliable image-to-floor plan correspondences. This chapter compares two matching methods, the robust least square method and the Random Sample Consensus (RANSAC) method. The RANSAC method is demonstrated to be more effective when excluding mismatches.

4.1 Robust Least Square Based Matching

Robust least square method concerns that many assumptions commonly made in classical statistics are at most approximations to reality, such as student distribution and normal distribution, while deviations from the probability distribution in assumption are results of outliers (Gao, 2009). Specifically, when image features are correctly matched to the corresponding floor plan features, the derived camera position and orientation should allow floor plan features to be projected to the image frame and agree well with image features. However, there are several error sources causing residuals between image features and floor plan feature projections, such as image feature detection errors and inaccurate floor plan geo-reference, and these error sources can be treated as Gaussian noise. Therefore, the pixel location residuals of correct image-to-floor plan matches should be normally distributed. In another word, when some image features are incorrectly matched with floor plan features, the derived camera position and orientation are inaccurate. The error in the derived camera position and orientation will cause deviation between the image features and floor plan feature projections, and the robust least square method expects that the distribution of the residuals should deviate from normality.

In order to exclude the residuals caused by mismatches, the statistical hypothesis test is frequently adopted for robust estimation to detect the outliers deviating from the normal distribution. As shown in Table. 1, hypothesis test includes two situations that the residual is or is not caused by mismatch. Four types of behaviors are resultant, which are right decision of accepting correct match, right decision of declining mismatch, type I wrong decision of declining correct match and type II wrong decision of accepting mismatch. A confidence level α is selected whose complement $1-\alpha$ equals to the probability of type I error. This confidence level α in the meantime sets an error tolerance level in terms of the standard deviation (STD). The residuals exceeding the error tolerance level are classified as mismatches. At each time only one outlier is excluded, and the mean value and STD of the residuals should be recalculated for the next outlier detection (Peterson and McFarlane, 1991).

Table 1: Statistical hypothesis test for robust least square

	H0 is correct match	H1 is mismatch
Accept H0	Right decision	Wrong decision Type II Error
Reject H1	Wrong decision Type I Error	Right decision

The feature matching process aims to automatically find correspondences of image features from floor plan features in the area of interest. Referring to the current Wi-Fi indoor position accuracy, the floor plan area of interest always covers an entire hallway, so usually there are more than 30

indoor hallway features in the floor plan database should be considered. However, since the camera view is limited, the image features are always less than the floor plan features, which means there are lots of matching possibilities. Exhaustive test of each possible matching is not an efficient method. Instead, iterative random tests can achieve great probability to find out correct matching. Therefore, an iterative robust least square method is designed for feature matching, and the paradigm is described as following:

1) The matching starts with an initial guess to randomly select equal number of floor plan features to match with all image features. This initial guess of matches is named M_1. Then the navigation algorithm is applied on this unverified initial guess M_1, and the camera position and orientation are derived, namely P_1.

2) Statistical test is conducted to verify if the initial guess M_1 contains mismatches. By using the derived camera position and orientation P_1, the floor plan features are projected to the image frame and compare with image features. The mean value and STD of their pixel location residuals are calculated. When the confidence level α is determined, the pixel error tolerance in terms of STD is referred in statistical test. Table. 2 tabulates the several frequently confidence level and corresponding error tolerance for statistical test.

Table 2: Confidence level and corresponding error tolerance for statistical test

$n\sigma$	α
n=1	68.27%
n=1.28	80.00%

n=1.64	90.00%
n=1.96	95.00%
n=2	95.45%
n=3	99.73%

3) If the initial guess M_1 is free of mismatch, the pixel location residuals are normally distributed and they will successfully pass the statistical test. However, since the initial guess is totally random, it is always inevitable to contain mismatches. These mismatches are expected to result in residuals deviating from normal distribution. By referring to the error tolerance, the statistical test excludes the mismatches one by one, whose pixel errors exceed the error tolerance. After statistical test, the mismatches in the initial guess are excluded, and the remaining matches are named M_1^*.

4) A predefined threshold m is set to examine the number of remaining good matches in M_1^*. If too many mismatches are detected, the remaining matches in M_1^* are not reliable, and a new round of random guess M_2 is started. If the number of remaining matches in M_1^* exceeds the threshold m, the navigation algorithm is applied on M_1^* to recalculate the camera position and orientation P_1^*. A threshold N is set to limit the maximum number of iterations, and the iteration will be terminated when the threshold is reached.

4.2 Reliability Test of Robust Least Square Matching

In order to examine the reliability of the robust least square method, the following test is conducted under control. As shown in the right picture of Fig. 10, 12 floor plan features are selected from the floor plan picture, marked with magenta dots. In the left picture of Fig. 10, their correct image feature correspondences are manually selected from the camera image, marked with green dots. After applying the navigation algorithm on the manually selected correct matches, the camera position and orientation are derived. Using the correct camera pose, the floor plan features are projected to the image frame, marked with red dots. The residuals between the pixel locations of image features (green dots) and floor plan feature projections (red dots) are examined in statistical test. Specifically, given the fact that projections of close floor plan features have large pixel location residuals while distant features have small pixel location residuals, the pixel location residuals are weighted according to their feature ranges. The mean and STD of the weighted pixel locations errors are calculated. The confidence level is set to be 95%, hence the pixel error tolerance is set to be the mean value plus and minus 1.96 times of the STD. If a pixel location residual exceeds the boundary, this pair of image-to-floor plan matches are excluded as mismatch, and marked with stars. In the left picture in Fig. 10, since all of matches are manually selected correct matches, no mismatch is detected.

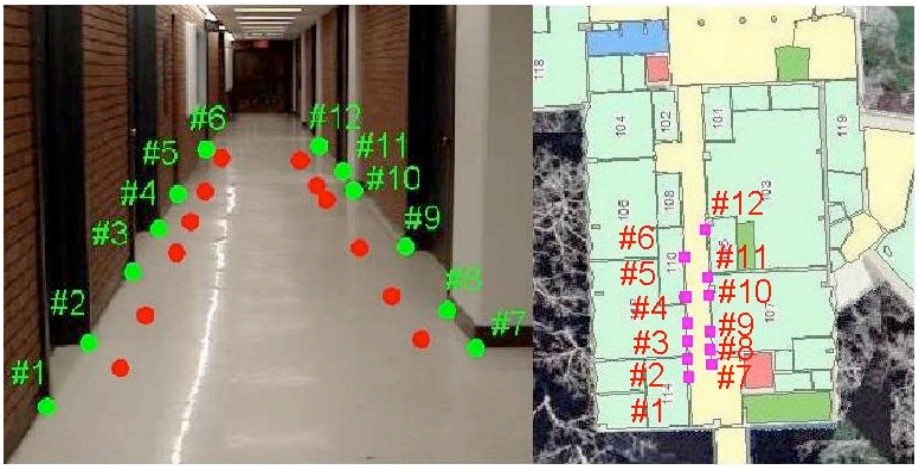

Figure 10: Example of correct matches which have all passed the statistical test

Fig. 10 demonstrates that, all correct matches can pass the statistical test. Further, it is necessary to test whether the statistical test can detect mismatches. As shown in the left picture of Fig. 11, the pixel location of the #6 image feature is changed to create one mismatch. The statistical test immediately detects that the residual of the #6 image feature has exceeded the error tolerance and marked the mismatch with stars. The test with one mismatch is continued in which the #12 image feature is changed, as shown in the right picture of Fig. 11, and the statistical test successfully detects the mismatch as well. The tests are conducted with two mismatches, three mismatches and four mismatches as shown in Fig. 12 – 14. Unfortunately, when more than two mismatches are created, there are always undetected mismatches, marked as crosses. Moreover, when four mismatches exist out shown in Fig. 14, no mismatch is detected.

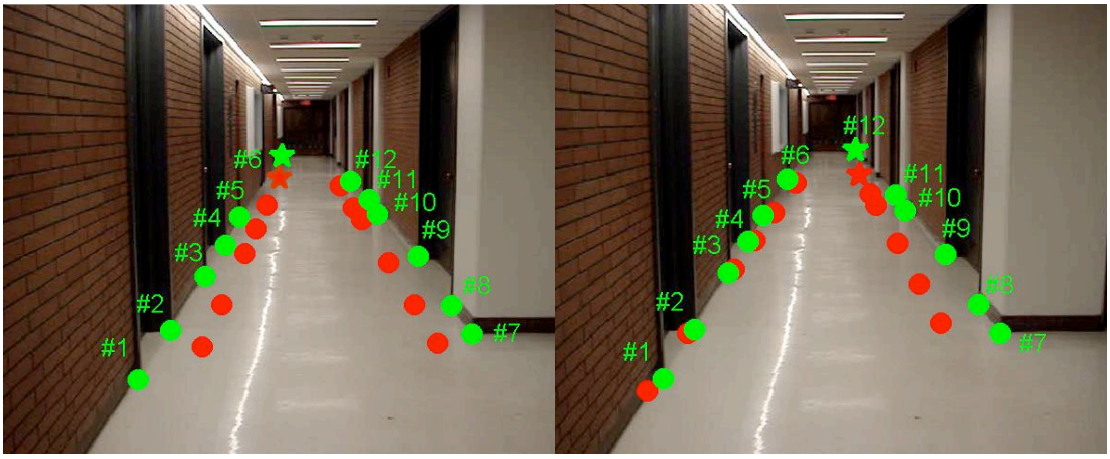

Figure 11: One mismatch is added and successfully detected

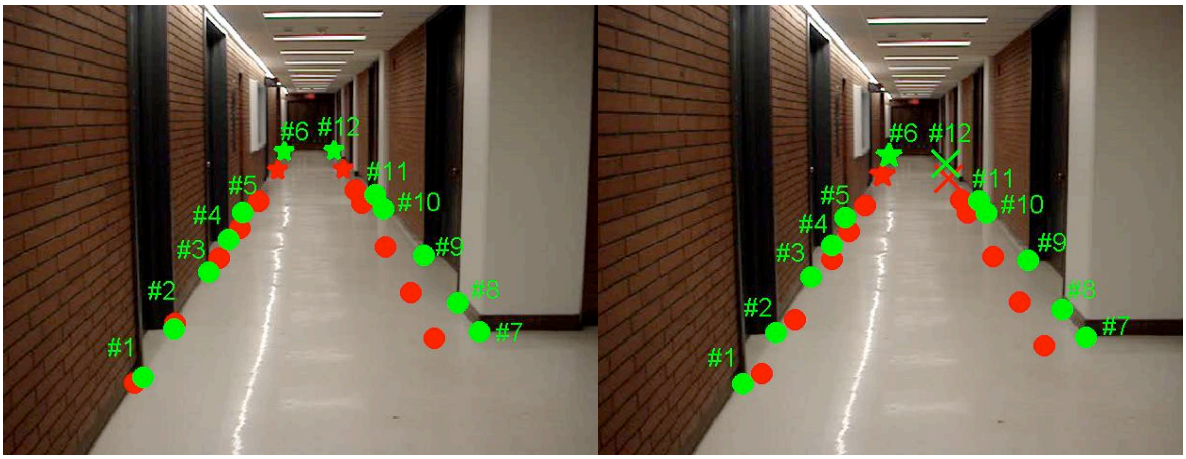

Figure 12: Two mismatches are added; both are detected in the left picture; only one mismatch is detected in the right picture.

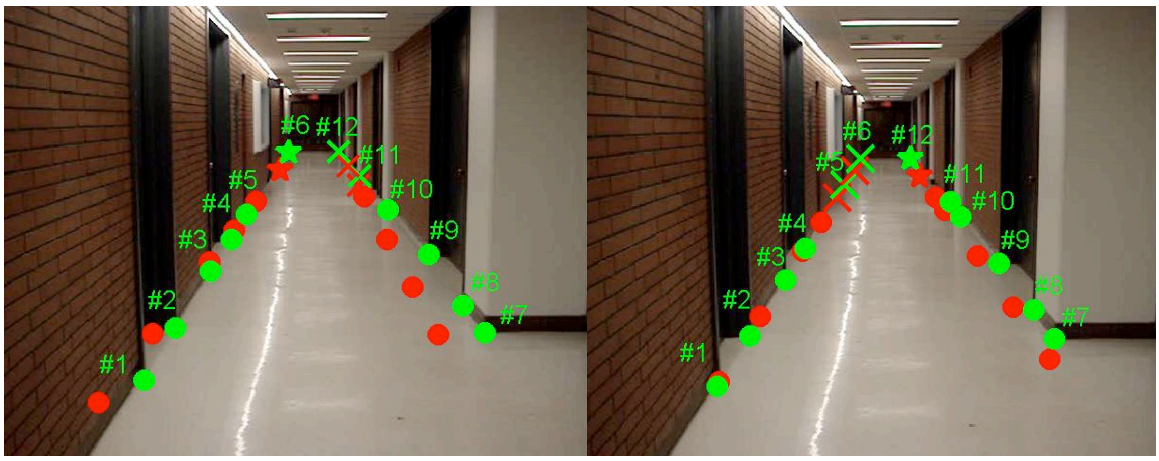

Figure 13: Three mismatches are added; only one mismatch is detected

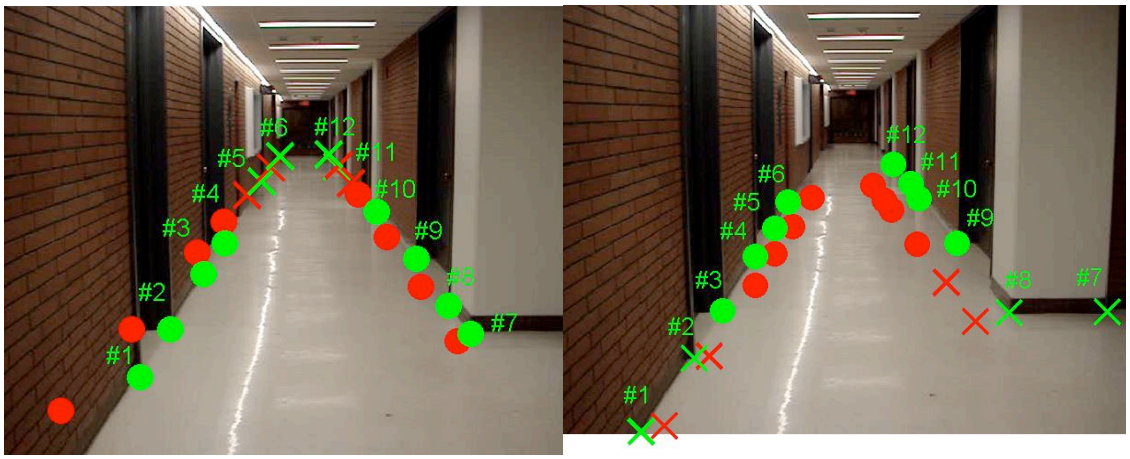

Figure 14: Four mismatches are added; all of them are not detected

Comparing Fig. 11 – 14 with Fig. 10, when mismatches are added, the floor plan feature projections do not significantly deviate from image features. As a result, it is difficult for the statistical test to distinguish the mismatches. The reason causing the ineffectiveness of the robust least square method is due to the assumption of normal distribution. Specifically, the robust least

square method matches all image features to selected floor plan features. However, when the floor plan features are randomly selected, there are probably large amount of mismatches. With these unverified random matches, least square method best fits all image features with floor plan features by minimize the residuals between them, no matter how many mismatches exist. When the random matches merely contain few mismatches, it is valid to assume that the distribution of residuals is approximate to normality, because the normality is not destroyed if majority of matches are guaranteed to be correct. However, the random matches are totally unpredictable and always contain certain number of mismatches, so the assumption of the normality does not represent the real distribution of residuals. As a result, the mean and STD of residuals are distorted by mismatches, and the outlier residuals are unexpectedly smoothed out, which makes the statistical test ineffective to distinguish mismatches.

4.3 Random Sample Consensus (RANSAC) Method

The RANSAC method is first proposed in Fischler and Bolles (1981) and becomes a popular paradigm applied in camera scene analysis and automated cartography. Comparing with the abovementioned robust least square based feature matching, the RANSAC method is also based on the iterative random guesses, but it is novel from two aspects: first, rather than using all image features to match with floor plan features, the RANSAC method uses a sample set as small as feasible to derive camera position and orientation. Specifically, four image features consist the minimum set to apply the navigation algorithm; second, robust least square based feature matching employs the statistical test to exclude mismatches one by one if their residuals exceed the error tolerance. Instead of excluding mismatches, the RANSAC method enlarges the initial

random guess with consensus matches when their residuals are under error tolerance. The consensus matches are added to the initial guess to form a consensus set. If the size of consensus set accounts for the majority of image features, they are accepted to be reliable matches. The paradigm for RANSAC method is stated as following and illustrated in Fig. 15:

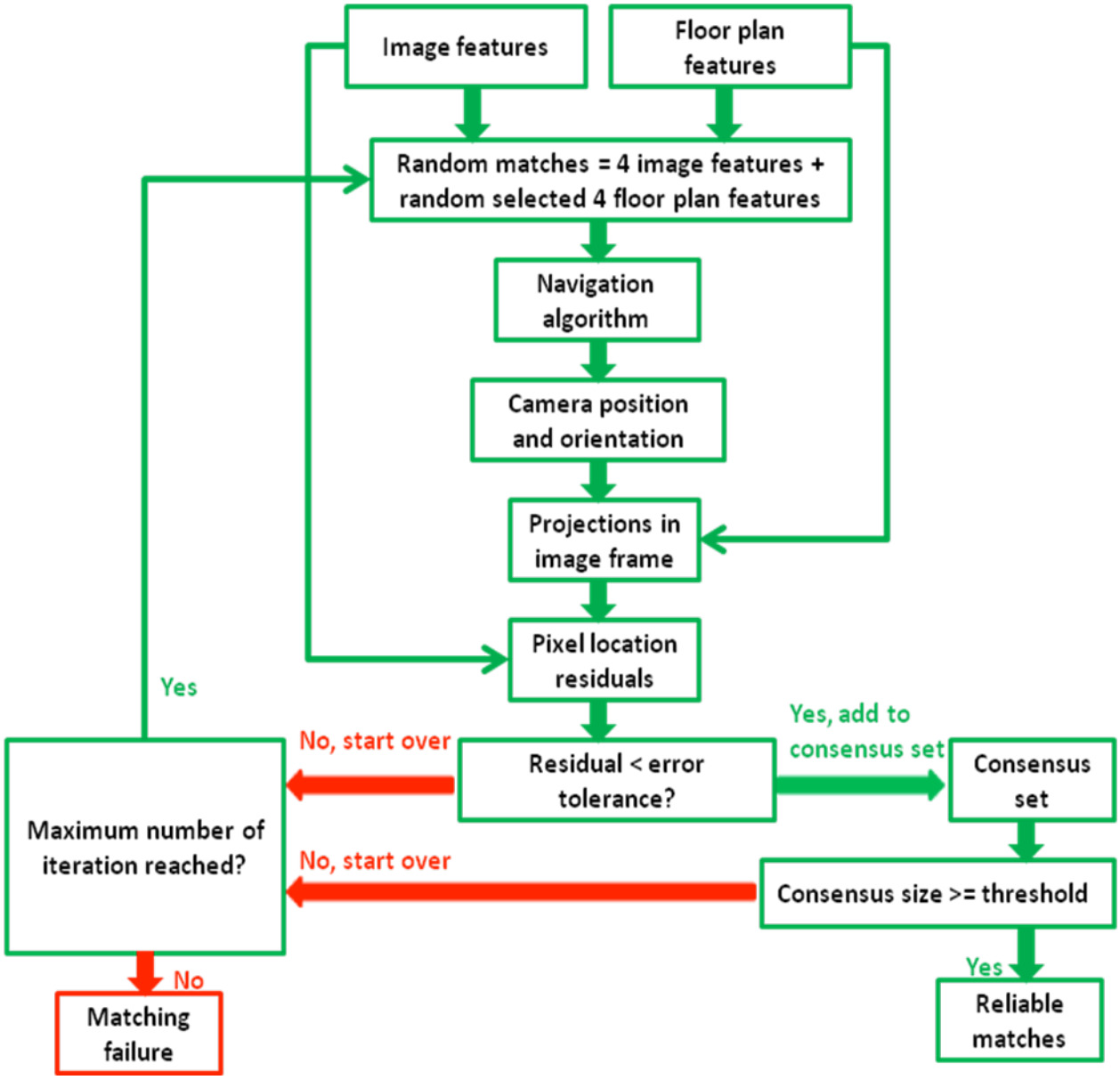

Figure 15: Flowchart of RANSAC routine

1) Randomly select four image features and floor plan features as the initial sample set S_1. Apply the navigation algorithm on S_1 to derive camera position and orientation, namely P_1;

2) Use P_1 to project the rest floor plan feature positions in the image frame, calculate the pixel location of their projections, and compare with image features. If the pixel location residual between a floor plan feature projection and an image feature is smaller than a predefined threshold t, this pair of features is identified as consensus match and added to the initial sample set S_1;

3) After adding all consensus matches to S_1, the consensus set S_1^* is obtained. If the number of correspondences in S_1^* is larger than threshold m, the navigation algorithm is applied on S_1^* to recalculate the camera position and orientation, namely P_1^*;

4) If the size of S_1^* fails to pass the threshold m, a new iteration starts with another random sample set S_2 until the maximum number of iteration N is reached.

Obviously, there are three important parameters to be determined including the pixel error tolerance t, the maximum number of iterations N and the threshold m on the size of consensus set:

1) The pixel error tolerance t describes the deviation between the image feature pixel location and the projection of its corresponding floor plan features. The error tolerance is a function of the error sources including the feature detection error, inaccurate camera interior parameters during camera calibration process and the inaccurate geo-reference information in the floor plan database. However, the function is always not straightforward expressed mathematically in practice. Taking the feature detection error as an example, when the luminosity in indoor scenario is too bright or too dark, the image features become not clearly recognizable by feature detection method, and large feature detection error occurs. In this thesis, the pixel error tolerance is determined experimentally. Specifically, several images are taken at various indoor scenarios, and selected image features are manually matched with their correct floor plan correspondences. The correct image-to-floor plan correspondences are perturbed by adding Gaussian white noise, and the error tolerance is derived by the residuals of the perturbed matches.

2) The maximum number of iterations N is the expected trials of random guesses until sufficient consensuses are selected. Exhaustive test of each possible matching is not efficient, because from the statistics point of view, randomly sampling a portion of possibilities can achieve considerable probability of finding the correct matching. Specifically, given several image features, there are always thousands of combinations when matching with the large amount of floor plan features in the area of interest. Instead of exhaustive test, random sampling is the key to achieve great probability of finding the

correct matching. Obviously, the more random guesses are tested, the larger probability is realized. But we wish to find out the minimum number of iterations to meet an expected probability. Fischler and Bolles (1981) have introduced detailed mathematical derivation of the relationship between the number of iterations and the probability of successfully finding correct matching. In this thesis, computation speed is a major concern in practice, therefore when there are numerous floor plan features in the area of interest, instead of setting the maximum number of iterations, a threshold of computation time is set. When the computation time is reached, the RANSAC method will be terminated mandatorily even if the correct matching is not found.

3) The threshold m on the size of consensus set answers the question that how many consensuses being found implies a sufficiently large consensus set to terminate the RANSAC iterations. In our case, since the number of all detected image features may vary from time to time, the ratio of the consensus set over the total number of image features is determined. This ratio must be large enough to prove the initial random guess is correct, that not only the features in the random guess but also a large number of remaining features all agree with it. However, unrealistically large ratio will result in matching failure. According to the experiments in this thesis, the threshold of the consensus ratio is found being closely related to the pixel error tolerance t. Specifically, the higher the error tolerance is, the fewer consensus matches can be found, and vice versa. The previous discussion shows that, the pixel error tolerance may vary in different conditions, and it has to be determined experimentally. The determination of the

consensus ratio also encounters this problem. Adjustability is an important aspect to guarantee the system reliability in practice (Yang, 2006). Especially, when the environment is uncertain, some error sources are impossible to be modeled accurately, parameters used in system model should subject to the variation of environment (Xie and Sol, 1993). Therefore, in this thesis, in order to make the RANSAC method adjustable to different scenarios, the RANSAC matching is conducted for three rounds. If the first round of RANSAC matching method fails, the threshold for consensus ratio decreases in the second round and the third round.

4.4 Reliability Test of RANSAC Matching

Four floor plan features are selected from the 12 features as the sample set, which are #1, #6, #7 and #12. The reliability of the RANSAC method is tested with control:

In Fig. 16, 12 images features are marked in red color. The corresponding image features of the #1, #6, #7 and #12 floor plan features are manually selected, marked as squares, therefore the sample set in Fig. 16 is free of mismatch. When the navigation algorithm is applied to the sample set, the derived camera position and orientation is correct. Using the derived camera pose, the floor plan features are projected to the image frame, marked with green color. Comparing image features with floor plan feature projections, if a pair of consensus match is found, they are marked with stars. In Fig. 16, since the accuracy of the georeference information in floor plan database is at decimeter level, it is possible that the camera image cannot be 100% matched with floor plan. As a result, #8 feature is detected as mismatch while it is actually correct match.

Nevertheless, remaining features are all successfully added to the initial sample set, which shows the matches are in great consensus.

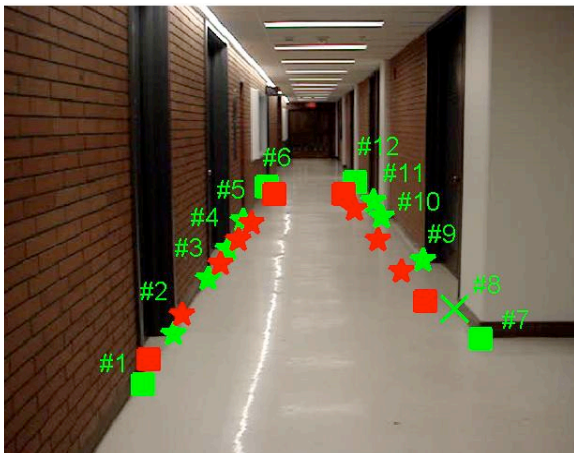

Figure 16: No mismatch in sample set; this matching is accepted by RANSAC

In Fig. 17, the pixel location of #6 image feature is changed to create one mismatch. Since the sample set is not redundant, with one mismatch and three correct matches, the floor plan feature projections are significantly deviated. As a result, there are five image features fail to find consensus matches. At this moment, a new round of random matching should be started.

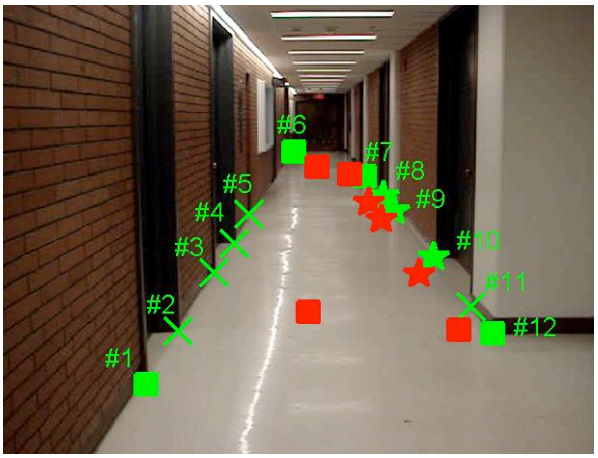

Figure 17: One mismatch in sample set; this matching is rejected by RANSAC

In Fig. 18, the pixel locations of #1, #6, #7 and #12 image features are all changed to make the sample set entirely mismatches. As a result, four image features fail to find consensus matches, and another round of random matching should be continued.

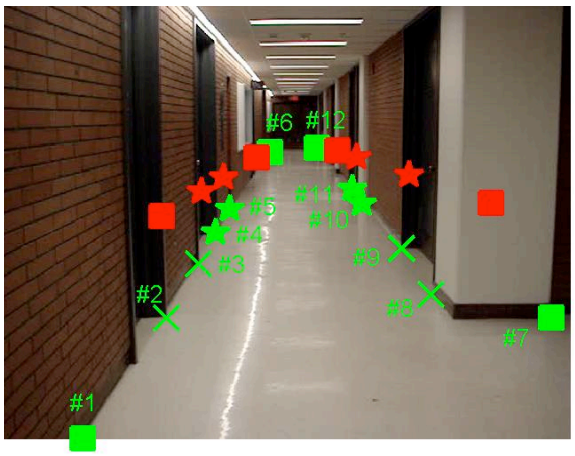

Figure 18: Four mismatches in sample set; this matching is rejected by RANSAC

Comparing Fig. 17 – 18 to Fig. 16, when mismatches exist in the sample set, the projections of the floor plan features are significantly deviated, so it is very easy to distinguish whether or not the sample set contains mismatches. In contrast, the robust least square method matches all image features to form a redundant set, but the residuals contaminated by mismatches have been averaged. Obviously, comparing with the robust least square method, the RANSAC method is more effective to obtain reliable image-to-floor plan matches.

However, the disadvantage of the RANSAC method comparing with the robust least square is the slow computation speed. Since the number of floor plan features in the area of interest is fixed, the robust least square matches all image features to floor plan features, while the

RANSAC method only matches four features in each iteration. It means the RANSAC method has much more matching possibilities than the robust least square. Therefore, in order to achieve the same probability of finding correct matches, the RANSAC method always need much more iterations than the robust least square. The computation speed will be analyzed in the Chapter five.

4.5 Unsolved Limitations

Although the RANSAC method is demonstrated to be more effective than robust least square method when detecting mismatches, success of robust matching is guaranteed only if the arrangement of indoor hallway features is sufficiently distinguishable. However, in most indoor scenarios, rooms and hallways are arranged very similarly from block to block. When the area of interest is too large, numerous indoor hallway features in floor plan database should be considered during matching. This problem is attributed to a common limitation suffered by all indoor navigation applications, which is the initial position determination. To sum up, poor accuracy of indoor position will causes limitations as follows:

1) If the initial position is totally biased that user is located in a wrong building, incorrect floor plan will be downloaded. Further, when position accuracy is too poor, it is possible that user location is out of any building contained in the floor plan database, and no floor plan will be downloaded.

2) In order to download the correct layer of floor plan, additional sensor should be employed to get height measurements, and the accuracy is expected to be better than 3 m.

Further, the floor plan database should add the height of floor plan to its geo-reference information.

3) The initial position accuracy should limit the area of interest where the arrangement of indoor hallway features forms a unique pattern. In practice, sometimes the arrangements of the indoor hallway features in different hallways are very similar. The initial position accuracy is expected to only cover one hallway.

4) When the area of interest contains large amount of indoor hallway features, the robust matching will be challenged and results in slow matching or even failure. In this case, more accurate initial position is needed to further narrow down the area of interest.

Barometer is available on some smart devices, and using barometer can provide meter level accurate height measurements. But the horizontal accuracy of initial position is still challenging. Current initial position only relies on the GPS/Cellular/Wi-Fi hybrid location, but for deep indoor environment, positioning only relies on Wi-Fi. According to our experiments, the accuracy is often worse than 60 m, which means the uncertainty area covers two or three building blocks. In this thesis, position accuracy was mandatorily set to 30 m during the indoor tests, and it is accurate enough to limit the area of interest in one hallway. In future works, integrating inertial sensors with pedestrian dead-reckoning algorithm is believed to bring significant improvement of the initial position accuracy.

Chapter Five: **Experiments, Software Development and Results Analysis**

This chapter introduces the development of iOS App, the indoor test plan, performance evaluation methods and test results. Screenshots are presented to illustrate a paradigm of using this developed App. 500 indoor tests were conducted to evaluate the system performance from the aspects of accuracy, repeatability, computation speed et al. In various indoor test scenarios in The University of Calgary, the system performance is demonstrated to be significantly improved comparing with Wi-Fi positioning.

5.1 Experiment Methodology

The University of Calgary online interactive map is used, and the geo-reference floor plan database is manually collected online. In order to verify the accuracy of the geo-reference information, the distances of the feature positions in the floor plan frame are compared with lengths measurements collected in the Engineering building. The accuracy of the floor plan feature positions is demonstrated to be good at decimeter level. This database currently only covers the first level of Engineering building A, B, C, D and E Block. An iPad without 3G cellular module was adopted for indoor tests in Engineering Complex, where GPS signal is totally unavailable. Therefore, the initial positions during the indoor tests were only Wi-Fi network based result and the accuracy was on average at tens of meters.

Structure of indoor environment can be extremely irregular and complex, and it is necessary to demonstrate the proposed system can work in general scenarios. In order to verify the performance in various indoor environments, indoor tests were conducted in the Engineering

Complex, where the test areas are categorized into three scenarios as following and illustrated with example pictures shown in Fig. 20:

Figure 19: Different scenarios of indoor tests

1) END, ENC and ENB are D, C and B blocks in the Engineering Complex respectively. These three blocks have parallel hallways with recognizable indoor features, as shown in the left picture in Fig. 20;

2) ENE is the ground level E block and it represents the indoor scenario with irregular hallways, as shown in the middle picture in Fig. 20;

3) ENA in A block is an open area that no hallways are available, as shown in the right picture in Fig. 20.

Figure 20: Left: standard hallway picture of END block; Middle: irregular hallway of ENE block; Right: open area of ENA block

In the abovementioned test areas, 10 landmarks are selected in each area, and in total there are 50 landmarks. For mean and STD positioning error analysis, the App was run 10 times at each landmark, and total number of indoor tests is 500 times. The performance analysis in this chapter is arranged to evaluate the following aspects:

1) *The RANSAC matching reliability:* in the RANSAC method, only if the consensus correspondences account for the majority of all detected features, the RANSAC matching is considered to be reliable. The ratio of consensus matches over all detected features, namely matching rate, is examined to evaluate the RANSAC matching reliability.

2) *Repeatability:* when using Wi-Fi positioning at the same landmark, the Wi-Fi positions vary from switch-on to switch-on. Given this repeatability problem, since the RANSAC matching is based on random guesses, it is necessary to verify if the proposed system can produce more repeatable results than Wi-Fi positioning. At each landmark, the App was run 10 times, and the STD position errors of the proposed system and the Wi-Fi positions are compared.

3) *Position accuracy:* by using the proposed system, the final derived indoor position should have better accuracy than Wi-Fi positioning. Indoor tests were conducted at landmarks, and the reference positions of these landmarks can be obtained from The University of Calgary online interactive map, whose accuracy is at decimeter level. Referring to the landmark reference positions, the mean and RMS position errors of the proposed system and the Wi-Fi positions are compared.

4) *Success rate:* the indoor test is considered to be a successful test only if the proposed system has improved the RMS position error comparing with Wi-Fi positioning. The ratio of the successful tests over the total 500 tests is examined, namely success rate.

5) *Computation speed:* the RANSAC matching process takes numerous iterations to find the correct matching. When the area of interest contains large number of indoor hallway features, the RANSAC matching speed may become very slow. In order to verify the

ability of delivering near real-time navigation results, the computation time of the 500 indoor tests is analyzed.

The computation speed highly depends on the hardware specification and the operating system. The device used in the experiments is a fourth generation iPad, which is equipped with Apple A6 processor running with the iOS 6 operating system. The A6 processor is featured with 1GB Elpida LP DDR2 SDRAM, dual ARM cores and three PowerVR graphics chips. With this specification, the computation speed, especially the graphic processing capability is very prominent. Similar specification can also be found in other high-end smart phone and tablet manufactured by Samsung, Sony, Motorola et al.

5.2 Development of iOS App

An iOS App is developed to realize the system design on the iPhone and iPad platforms. The software structure and the objective-C frameworks being used in software development are illustrated in Fig. 21: the initial indoor location is collected by using the CoreLocation framework, which outputs user's current latitude, longitude, and accuracy in unit of meter. The CoreLocation framework output is by default the combined results from GPS, cellular network and Wi-Fi fingerprinting; downloading floor plan data requires the communication with server through Wi-Fi connection, and the remote server is simulated with the Application Programming Interface (API) of a cloud storage Dropbox. The floor plan database is stored in Dropbox, and all necessary communication methods like sending request to server and receiving feedback to user are supported by API functions; once the user takes a picture of the indoor scenario, the feature detection implements the image processing library OpenCV, which provides the FAST corner

detection function; the navigation algorithm and RANSAC matching involve lots of matrix manipulations such as eigenvalue decomposition, and a linear algebra library LAPACK is employed.

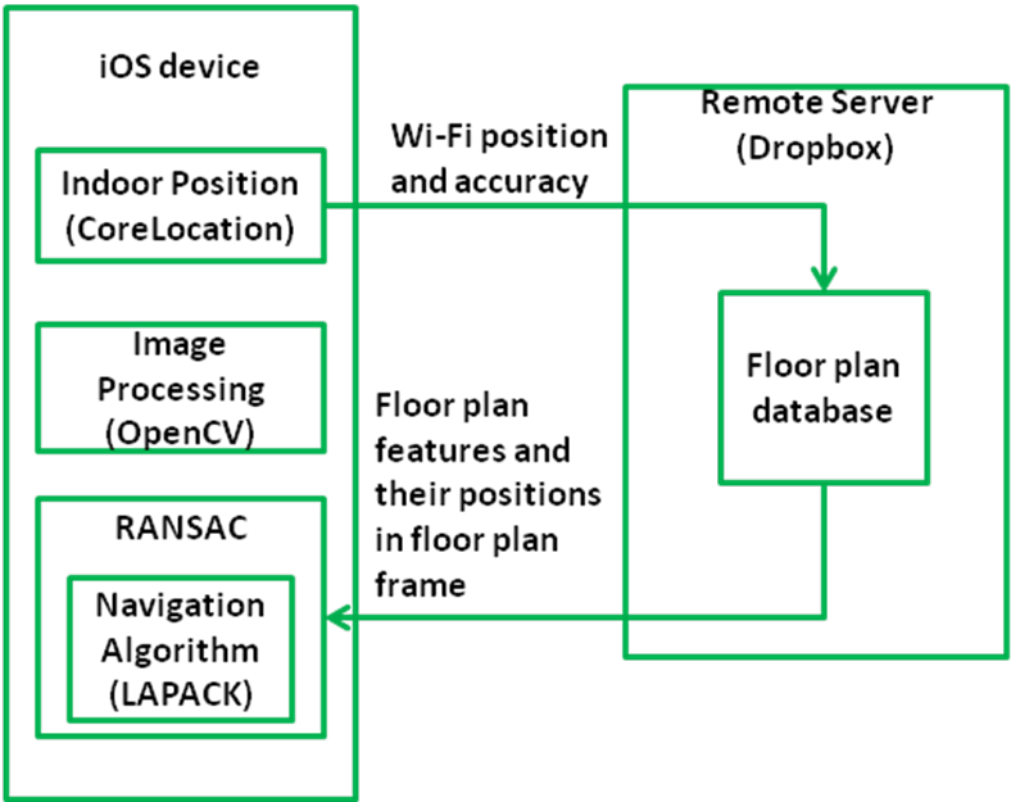

Figure 21: System and software structure

Fig. 22 illustrates the paradigm of how to use this App:

1) At the moment of the screen shot as shown in the first picture of Fig. 22, the initial indoor location was obtained with an accuracy of 74 m. In the meantime, the area of interest is determined. “Link to server” button is shown at the bottom of the screen.

2) After connecting to the remote server, the floor plan picture and geo-reference data are downloaded. The floor plan picture is centered around the initial location and overlaid on the map, as shown in the second picture of Fig. 22.

3) User takes a photo of the hallway and touch on indoor hallway features. User touched regions are marked with red dots in the third picture of Fig. 22. The FAST corner detection is applied on those dots to extract the image features. If user can specify more than 10 image features, the App can usually achieve reliable and accurate results.

4) After execution of the RANSAC matching and the navigation algorithm, the user position is derived and pinned on the floor plan, shown as the red dot in the fourth picture of Fig. 22. Meanwhile, user can type in a destination room in the search bar.

5) The navigation algorithm not only derives the camera position but also the orientation, which enables the floor plan picture and a navigation arrow to be transformed to user's perspective and overlaid on the camera view. iPhone and iPad are equipped with magnetic compass, and its orientation measurements are more reliable than the derived camera orientation. Finally, the augmented navigation reality is realized with the camera position derived by the proposed system and the camera orientation measured by magnetic compass. If an up-to-date database of room inventory is available, event in the destination room can also be displayed, as shown in the fifth picture of Fig. 22.

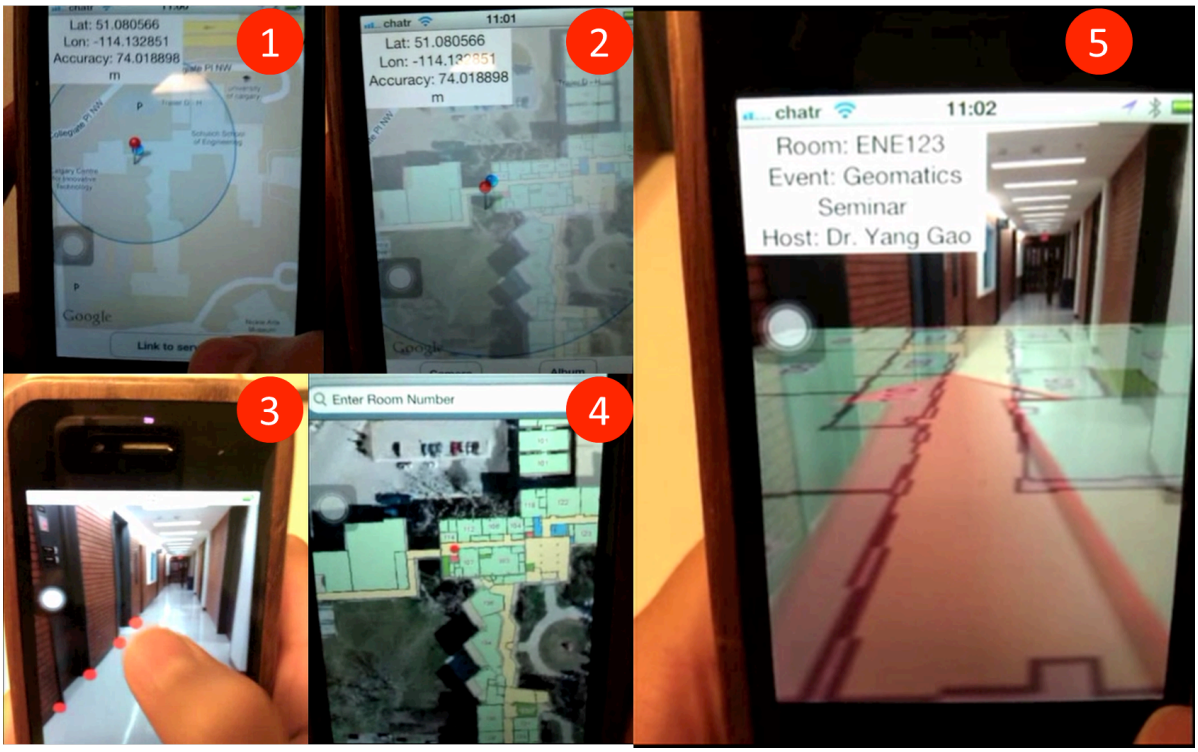

Figure 22: Screen shots of iOS App

5.3 The RANSAC Matching Reliability

The RANSAC matching is expected to run randomly and automatically identify image-to-floor plan correspondences. In this session, the RANSAC matching results are discussed, which demonstrate its matching reliability. Fig. 23 has illustrated an example of finding the consensus matches. With a random guess of four pairs of image-to-floor plan correspondences, the camera position and orientation are calculated by applying the navigation algorithm on the random guess. Using the derived camera pose, the remaining floor plan features are projected to the image frame and compare with image features. In Fig. 23, the red dots are detected image features. Each of them has a yellow circle, and their radiuses are determined by the error tolerance for the pixel location residual. If the pixel location of a floor plan feature projection

falls into the yellow circle of an image feature, this pair of image and floor plan features is identified as a consensus match. In the consensus match, the image feature is marked with yellow dot, and the floor plan feature is marked with green dot. Furthermore, due to the fact that close features have larger pixel residuals while distant features have smaller pixel residuals, the pixel location residuals are weighted according to the feature ranges. As shown in Fig. 23, the radius of yellow circle decreases as feature range increase.

When the random guess of four pairs of image-to-floor plan correspondences are correctly matched with each other, the derived camera pose is reliable, which will allow the remaining floor plan feature projections fall into the error tolerance circles. In contrast, if the random guess contains mismatches, the derived camera pose is distorted, which will only allow few projections fall into error tolerance circles. In another word, if more consensus correspondences are identified by the RANSAC method, the derived camera pose is typically more reliable. In Fig. 23(a), it shows an example being rejected by the RANSAC matching, because the number of identified consensus matches does not exceed the threshold. It means the initial random guess contains mismatches, and a new round of random guess is needed. In Fig. 23(b), it shows an example being accepted by the RANSAC matching, because a large amount of consensuses are found, and the ratio of consensuses over the total number of feature has exceeded the threshold. It indicates the random guess is correct matching, and the camera position and orientation derived from the random guess are more reliable than those derived in Fig. 23(a).

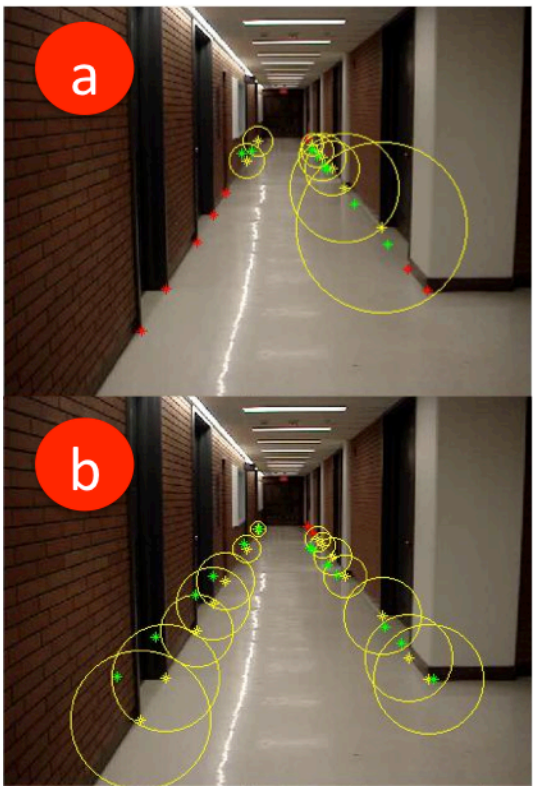

Figure 23: RANSAC matching example

With the consensus matches identified in Fig. 23 (b), it is necessary to recalculate the camera position and orientation using consensus matches. Because the random guess is the minimum set which only contains four pairs of features, the consensus set is redundant to apply the navigation algorithm, and the recalculated camera position and orientation will have better accuracy than those calculated from the random guess. What is more, the thresholds of the RANSAC matching method are determined experimentally in this thesis, and it is inevitable that the thresholds become too strict or too loose when the indoor scenario changes. As a result, few mismatches may not be excluded, and they will distort the camera position and orientation when applying the

navigation algorithm. Therefore, if large amount of consensus matches are found, errors due to few mismatches can be averaged since the majority of matches are correct.

The number of consensus matches determines the reliability of the RANSAC matching. Fig. 24 has shown the ratio of consensus matches over all detected features. The RANSAC matching reliability in areas of END, ENC and ENB is very identical, where more than 80% features can always be successfully matched. The ENE block with irregular hallway is more challenging to the RANSAC matching, but still over 75% features are successfully matched. In the open area of ENA, the RANSAC matching can only find over 60% consensus matches. This low matching rate may probably result in degraded accuracy in ENA block.

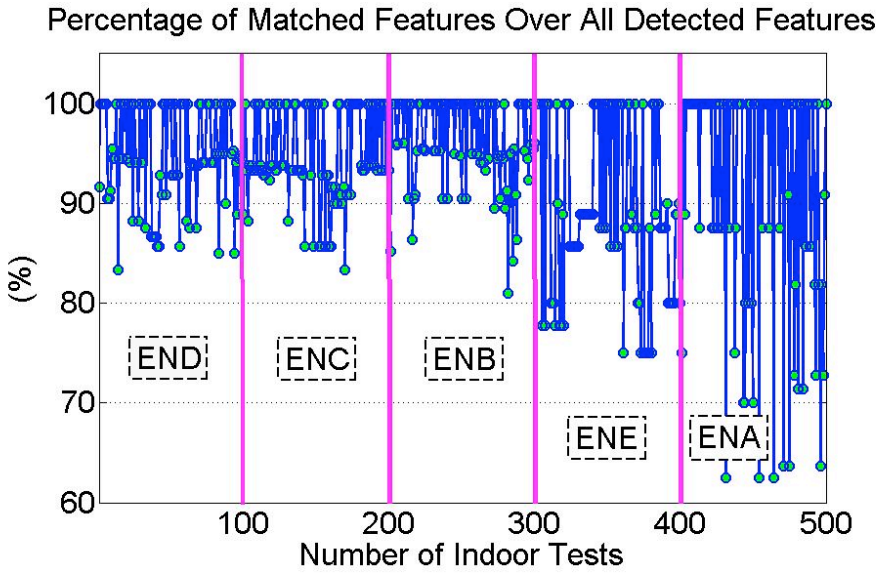

Figure 24: Percentage of consensus matches out of all detected features

5.4 Repeatability and Position Accuracy

In order to analyze the repeatability and position accuracy, both the Wi-Fi positions and the results derived by the proposed system are compared with the landmark reference positions. The 50 indoor landmarks are marked with colorful dots on the floor plan picture as shown in Fig. 25 and Fig. 26. From the 10 times of repeatability tests at each landmark, the mean and STD position error of both the proposed system and Wi-Fi positions are calculated. The radius of the dots indicate the mean position errors. The color scales indicate the STD position, from cold color representing small STD error to warm color representing large STD error.

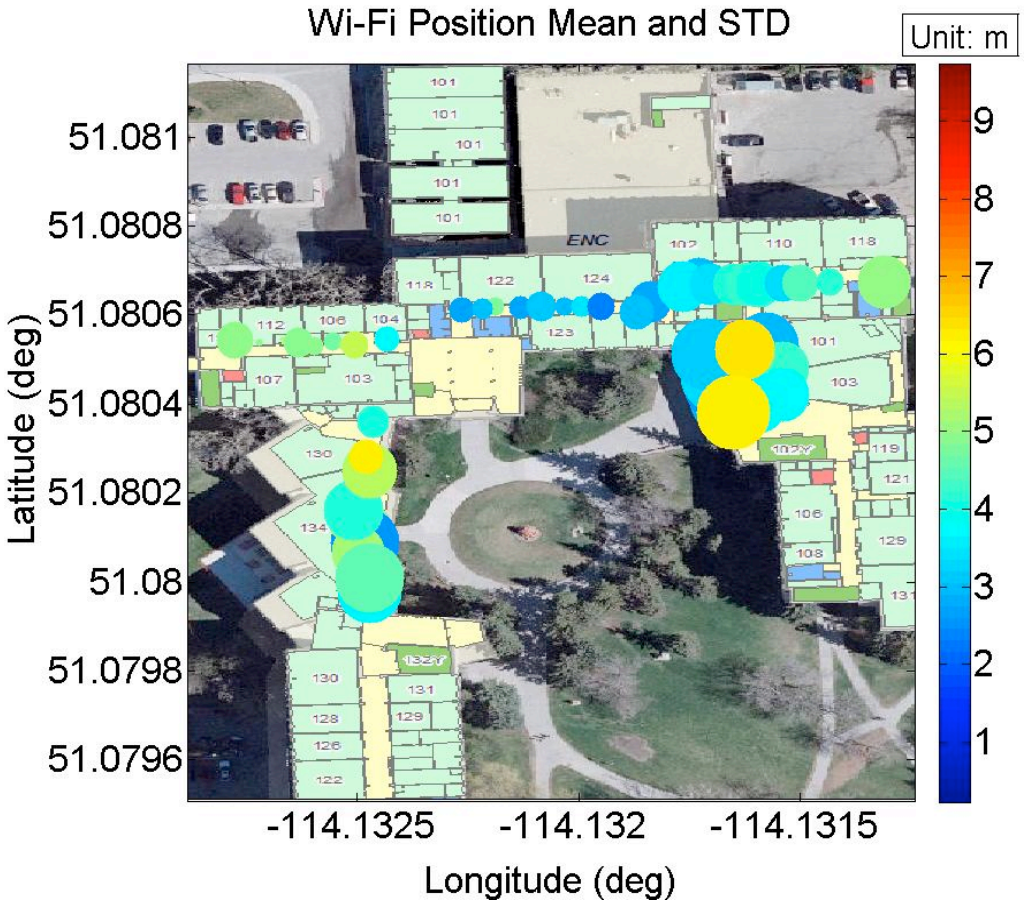

Figure 25: mean and STD position error of Wi-Fi positions

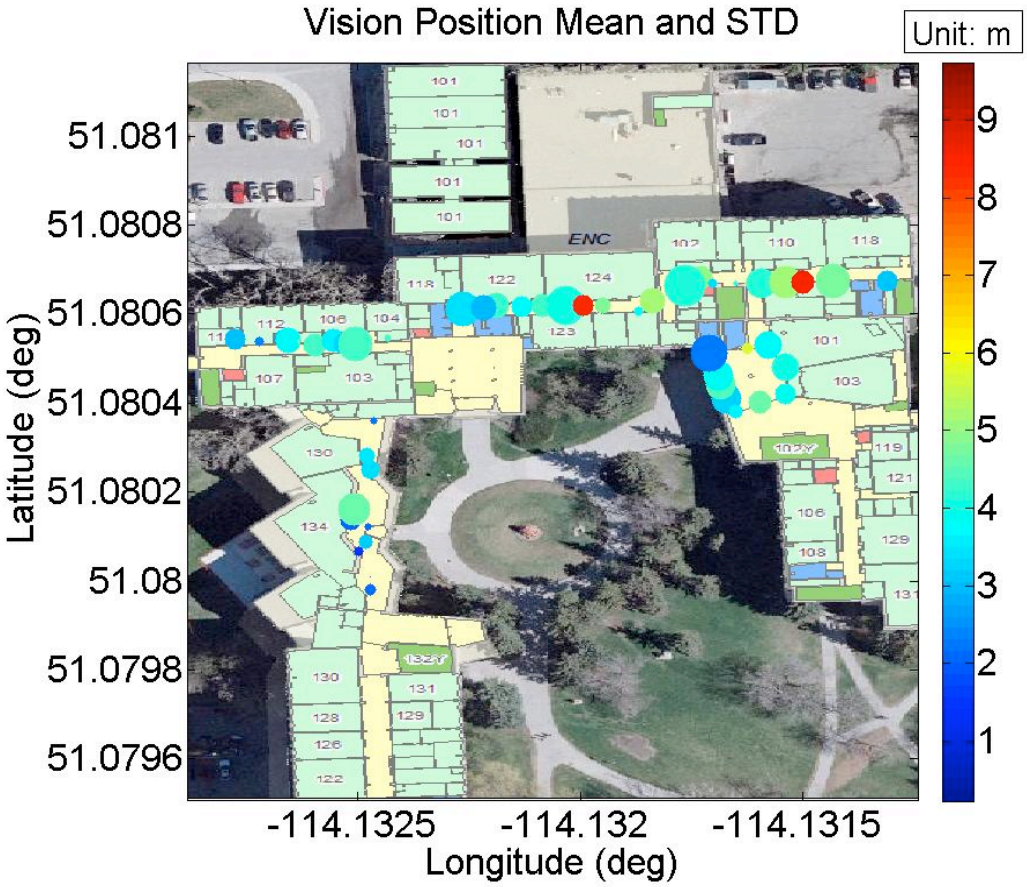

Figure 26: mean and STD position error of the floor plan based vision navigation system

5.4.1 Repeatability

The STD errors are compared in this section, which indicates the repeatability of positioning solutions.

1) *Wi-Fi position STD error*

Referring the colourful dots to the color bar in Fig. 26, the overall STD error varies from 1.28 m to 8.17 m, which is very typical Wi-Fi positioning performance. Nevertheless, it is obvious to find significant performance degradation in some areas. For example, the dots in END, ENE and ENA blocks are with relatively warm color, while majority of the

dots in ENB and ENC blocks are blue. The reasons causing the inconsistent STD position errors of Wi-Fi positions are attributed to two reasons, the quality of the Wi-Fi AP database and the received signal strengths of Wi-Fi APs. Specifically, the CoreLocation framework adopted in this thesis employs the fingerprinting technique for Wi-Fi positioning. As for the database quality, although there is no way to investigate the coverage and density of the Apple's Wi-Fi AP database, it is reasonable to conclude that indoor tests with large Wi-Fi positioning errors are with sparse Wi-Fi APs in database. The fingerprint formed by the sparse Wi-Fi APS does not provide sufficient statistical information about the signal distribution, which results in poor positioning reliability. As for the received signal strength, during the indoor tests, the Wi-Fi signal bar of iPad implies that the Wi-Fi reception in ENB and ENC is much stronger than other areas. Consequently, the weak signals are relatively fragile to disturbance by noise, reflection and attenuation by indoor structures and other disturbing RF signals. When the fingerprint formed by weak signals is compared with those in Wi-Fi AP database, the derived positions are also subject to various disturbances.

2) *STD error of the vision navigation results*

Comparing with Fig. 25, the dots in Fig. 26 have shown two significant improvements: first, the overall color of dots in Fig. 26 is cooler. Especially in the areas of ENA, END and ENE, where the Wi-Fi position repeatability is poor, by using the proposed system, the STD errors in these areas are significantly reduced; second, the consistency of the colors in Fig. 26 is much better than Fig. 25. Due to the signal reception and the quality of the Wi-Fi AP database, the repeatability of Wi-Fi positioning may suffer from severe

degradation in different scenarios. However, the repeatability of the position results is constantly good in various scenarios.

The indoor tests in various scenarios have also demonstrated that, the developed navigation algorithm can provide reliable results in the indoor scenarios with parallel hallways, irregular hallways and open area. In most modern architectures, the indoor scenarios are highly irregular. However, given the floor plan database and sufficient recognizable indoor hallway features, the proposed system is always capable to derive reliable positions.

The repeatability tests also imply the reliability of the RANSAC matching method. The RANSAC matching is based on the iterations of random guesses. Through the repeatability tests, the STD position errors are better than Wi-Fi positioning solutions at most landmarks. It indicates that although the RANSAC routine is random, the image-to-floor plan correspondences identified by the RANSAC are repeatable and reliable.

3) *Limitations causing large STD error in the vision navigation system*

However, in the middle of the hallways in ENB and ENC blocks, there are two red dots with STD errors as large as 9 m. Furthermore, in ENB, ENC, END and ENE blocks, the dots in the middle of hallway are relatively with warmer color than the dots at the ends of hallway. This fact unveils one limitation of the proposed system, that user is expected to take picture near the ends of hallway. Specifically, the camera image is expected to contain as many indoor hallway features as possible to provide sufficient details of indoor

scenarios for the RANSAC matching. When user is approaching the middle of hallway, more indoor hallway features are left behind the user but less features are visible in camera view. Taking the largest STD error in ENB block as example: at the moment of snapshot, there was only 10 visible indoor hallway features in the middle of ENB, while the floor plan database in ENB block contains more than 60 features. Therefore, with the limited number of indoor hallway features, the RANSAC matching merely has very abstract information to match with the crowd of numerous features in the database. Moreover, sparse indoor hallway features not only degrades the repeatability of the RANSAC matching, but also challenges the matching speed, or even results in matching failure. In practical use, user is suggested to stand near the end of hallway instead of in the middle, hence to contain as many indoor hallway features in camera view as possible. According to the indoor test experience, if user can capture snapshot containing around 50% of the indoor hallway features in the area of interest, the RANASAC matching is always fast and reliable. Assuming user takes indoor pictures as the suggested way, it is able to avoid the large STD errors at the red dots in ENB and ENC blocks as shown in Fig. 26. The STD errors at the remaining landmarks range from 0.22 m to 7.25 m.

5.4.2 Position Accuracy

The position accuracy with respect to the landmark positions is analyzed in this section. The mean position errors are examined by referring to the radiuses of the dots in Fig. 25 and Fig. 26.

1) *Mean error of Wi-Fi positions*

   In Fig. 25, the dots in ENC block are on average smaller than the dots in other blocks. It means the mean position errors in ENC block are smaller than other blocks. A reasonable

explanation to this phenomenon is attributed with the uneven density of Apple's Wi-Fi AP database, although it is not possible to investigate Apple's database. The database probably has stored highly dense APs in ENC area but relatively sparse APs in other blocks. When user is not in ENC block, the device has collected the fingerprint consisted by the nearby Wi-Fi APs and the APs in ENC block. However, when comparing the crowd-sourcing fingerprint with the database, the crowded Wi-Fi APs in ENC block are given more weight while the nearby APs are making less contribution. Therefore, the fingerprinting technique unexpectedly inclines to map the user position to ENC block, and the large mean errors are resultant.

2) *Mean error of the vision navigation results*

Comparing with Fig. 25, Fig. 26 has shown great improvement of the mean position errors. On the one hand, in ENC block where the Wi-Fi positioning accuracy is good, radiuses of dots in Fig. 26 are as small as those in Fig. 25. On the other hand, in the areas of ENB, END, ENE and ENA where the mean errors of Wi-Fi positioning are large, the radiuses of the dots are significantly reduced by using the proposed system.

Three conclusions can be drawn from the Fig. 26: first, extremely small dots indicating trivial mean errors always appear at the ends of the hallways. At the landmarks locate in the middle of hallways, the mean position errors are larger. This fact further confirms the limitation discussed previously, that in order to get reliable RANSAC matching result, the camera image should contain as many indoor hallway features as possible. Therefore, user is suggested to take indoor pictures at the ends of hallway; second, the proposed

system works well not only in the typical indoor scenarios with parallel hallways such as ENB, ENC and END blocks, but also perform well in irregular scenarios like ENE block. Decimeter level mean errors are found in ENB, ENC, END and ENE block, where the smallest mean position error is less than 0.5 m; third, the mean errors in the indoor scenario having open area like ENA block are larger than other scenarios. By looking back to Fig. 24, the performance degradation in ENA block is caused by the poor matching quality. Specifically, in Fig. 24, the open area of ENA block is the most challenging area for the RANSAC matching, and only 60% image features can be matched. With such poor matching quality, the image-to-floor plan correspondences identified by the RANSAC matching are relatively unreliable. Therefore, the derived camera positions are subject to more mismatches comparing with other test areas.

3) *RMS error comparison*

Fig. 27 shows the RMS position errors at the 50 landmarks, and the RMS error of Wi-Fi positioning ranges from 2.83 to 30.87 m, while the RMS error of the proposed system ranges from 0.58 to 10.22 m. By using the proposed system, the tests at 42 landmarks have improved Wi-Fi positioning accuracy. The greatest accuracy improvement occurs in ENE block in which the proposed system has improved the Wi-Fi position RMS error from 27 m to 2 m.

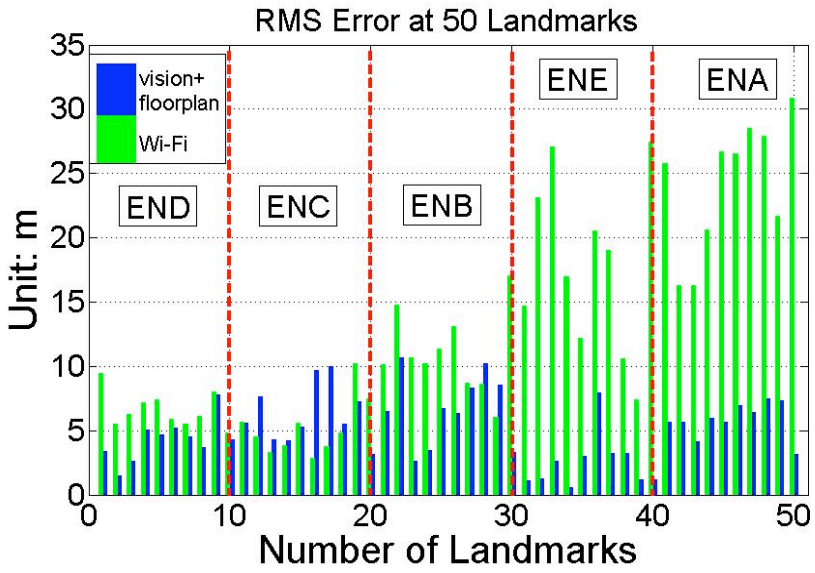

Figure 27: RMS error comparison of the floor plan based vision navigation and Wi-Fi positioning

Table. 3 has summarized the position RMS error in each test area. It further demonstrates three contributions of the proposed system: first, the proposed system can bring position accuracy improvement to Wi-Fi positions; second, in the area where Wi-Fi performance is good, comparable accuracy is also achieved by the proposed system; third, unlike the inconsistent performance of Wi-Fi positioning, the accuracy of the proposed system have shown great consistency in different blocks.

Table 3: Position RMS error in different test area

RMS (m)	END	ENC	ENB	ENE	ENA
Wi-Fi based initial position	6.74	5.60	11.47	19.02	24.60
Floor Plan based vision navigation	4.56	6.63	7.20	3.24	5.99

5.5 Success Rate

In order to answer the question that how many times the proposed system produces more accurate position than Wi-Fi, the success rate and failure rate is analyzed in this section. Success rate is defined as the percentage of indoor tests with improved RMS position error, and these successful indoor tests are marked with green color in Fig. 28. In contrast, the failure rate is defined as the percentage of indoor tests with degraded RMS position error, and these failed indoor tests are marked with red color in Fig. 28.

The success rate is 76%, and the more than 50% of indoor tests have brought more than 10 m accuracy improvement to Wi-Fi positioning. The failure rate is 24%, and among these tests, 51.7% of them have achieved comparable accuracy with Wi-Fi positioning, in which the accuracy degradation is less than three meters. However, in the 500 indoor tests, there are still 11.6% tests much worse than Wi-Fi positioning, and most of these failures happen in the middle of hallways in ENB, ENC and END blocks. This fact again emphasizes the necessary to take indoor pictures at the ends of hallways to contain as many as indoor hallway features as possible.

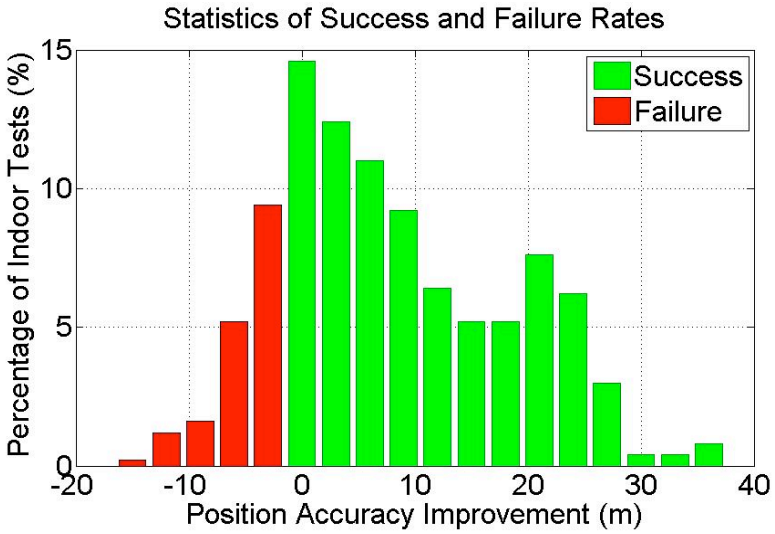

Figure 28: Success and failure rate

5.6 Computation Speed

The last aspect of performance evaluation focuses on the real-time capability of the proposed system. The feature detection process and the RANSAC matching account for the major reasons of slow computation speed. The statistics of the computation times of the 500 indoor tests are shown in Fig. 29. 59% tests can accomplish the feature detection, the RANSAC matching and the navigation algorithm in one second. There are still 13% tests spend more than three seconds. We have found these indoor tests with slow computation speed all locate in ENB block, and the reason is due to the crowded indoor hallway features in this area. Specifically, in this 40 m hallway of ENB block, there are more than 20 doorways and other indoor objects. In another word, more than 60 ENB block indoor hallway features are contained in the floor plan database, and the number is almost double of that in other blocks. These populated features in ENB block is challenging for the RANSAC matching, because it means the number of all matching possibilities is even more populous. According to the discussion in the Chapter four about

thresholds setting in the RANSAC method, in order to meet certain probability level, the maximum number of iterations should cover a portion of all matching possibilities. Apparently, the more indoor hallway features exist in the area of interest, the more possible matching should be tested, and the more iterations of random guesses are needed. Therefore, the indoor tests conducted in ENB block always need more iterations and longer time to wait the RANSAC matching finally identify reliable matches.

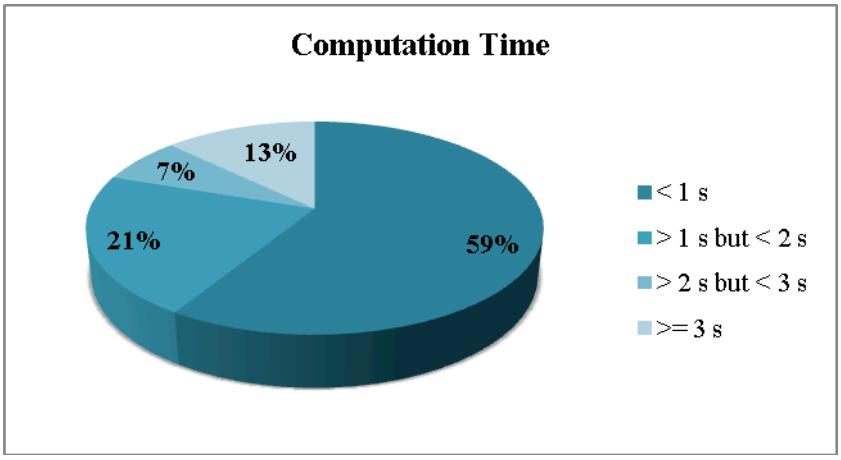

Figure 29: Computation speed

Chapter Six: **Conclusions and Future Works**

This thesis has demonstrated the advantage and effectiveness of using the floor plan based vision navigation system to improve the indoor positioning accuracy and reliability. This proposed system can provide satisfactory indoor positions in various scenarios with great performance consistency. The conclusions are drawn as following:

1) *Ubiquitous navigation system:* an innovative vision navigation system is developed for pedestrian indoor navigation. This system only relies on the camera on smart devices, and does not makes any unpractical assumption of camera pose, neither causes threatens to the device battery life. Furthermore, the ubiquitous floor plan geo-reference database is used for geo-referencing. Unlike the other popular geo-reference database such as geo-tagged photos, floor plan database does not require the investment on survey equipment, labour and time, which enables the floor plan database used in this thesis have outstanding availability and coverage. Therefore, the floor plan based vision navigation system is demonstrated to be easily implementable in practice for low-cost pedestrian indoor navigation.

2) *Robust feature matching:* both the robust least square method and the RANSAC method are introduced, and they have very different schemes to avoid mismatches. However, the robust least square is demonstrated to be not suitable, because its assumption of normality does not represent the real error probability distribution in the feature matching problem. Instead, the RANSAC matching is employed in this thesis due to its effectiveness when

excluding mismatches. Although the RANSAC matching needs numerous iterative tests, the average computation time is demonstrated to be as fast as 1 s.

3) *Reliable navigation algorithm:* two methods are employed in the navigation algorithm, which are the passive ranging and the derivation of camera position and orientation. These two methods are at first proposed by Hung et al. (1985) and Horn et al. (1988) respectively. On the basis of their derivation, the mathematical models are modified to interpret the proposed navigation system. Detailed derivation and modification are elaborated in this thesis, and they are demonstrated to enforce the reliability of the navigation solution.

4) *Repeatability and accuracy improvement:* 500 indoor tests are conducted in The University of Calgary to demonstrate the performance of the proposed system. Comparing with Wi-Fi positioning, the proposed system has significantly improved the mean and STD position errors. The success rate shows that 76% of the indoor tests have achieved more accurate positions than Wi-Fi positioning. Furthermore, the accuracy of Wi-Fi positioning varies from meter level to tens of meters, while the proposed system has much more consistent performance in different indoor scenarios.

Recommendations for future works are summarized as following to further improve the performance of the proposed system:

1) The initial position accuracy is very important to determine the area of interest. Integrating MEMS sensors such as accelerometer, gyroscope will significantly improve the current initial position accuracy by Wi-Fi, which is tens of meters. Accurate initial positions will not only accelerate the RANSAC matching speed but also improve the matching reliability.

2) In order to refer to the correct layer of floor plan database, barometer is needed in future to provide height measurements. The geo-reference information in floor plan database should also contain the height information of floor plan in future.

3) More delicate image processing methods are needed to automatically detect indoor hallway features in camera image. Currently, user is required to touch on screen to specify the search region, where the feature detection method is applied. In future, the line detection combined with corner detection can improve this process, and automatically extract the indoor hallway features.

References

Chu C., L. Lie, L. Lemay and G. Egziabher (2011), "Performance Comparison of Tight and Loose INS-Camera Integration", Proceedings of the Institute of Navigation GNSS 2011, Portland, Oregon, USA, September 2011, pp. 3516-3526.

Chu T., Guo N., Backen S., Akos D. (2012), "Monocular Camera/IMU/GNSS Integration for Ground Vehicle Navigation in Challenging GNSS Environments", Sensors, Vol. 12, No. 3, pp. 3162-3185.

Ding W., Wang J., Almagbile A. (2010), "Adaptive Filter Design for UAV Navigation with GPS/INS/Optic Flow Integration", Proceedings of the International Conference on Electrical and Control Engineering, 2010, pp.4623-4626.

El-Sheimy N. (2003), "ENGO 633 Inertial Techniques and INS/DGPS Integration", Lecture notes, Department of Geomatics Engineering, University of Calgary, Canada.

Fischler M. A., Bolles R. C. (1981), "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography", Communication of the ACM, Vol. 24, No. 6, pp. 381-395.

Gao Y. (2009), "ENGO 629 Advanced Estimation Methods and Analysis", Lecture notes, Department of Geomatics Engineering, University of Calgary, Canada.

Hide C., Botterill T., Andreotti M. (2010), "Vision-Aided IMU for Handheld Pedestrian Navigation", Proceedings of the Institute of Navigation GNSS 2010, Portland, Oregon, USA, September 2010, pp. 534-541.

Horn B. K. P., Hilden H. M., Negahdaripour S. (1988), "Closed-Form Solution of Absolute Orientation using Orthonormal Mareices", Journal of the Optical Society America, Vol. 5, No. 7, pp. 1127-1135.

Huang B., Du S., Gao Y. (2011), "An Integrated MEMS IMU/Camera System for Pedestrian Navigation in GPS-denied Environments", Proceedings of the Institute of Navigation GNSS 2011, Portland, Oregon, USA, September 2011, pp. 2373-2380.

Huang B., Gao Y. (2012), "Indoor Navigation with iPhone/iPad: Floor Plan Based Monocular Vision Navigation", Proceedings of the Institute of Navigation GNSS 2012, Nashville, Tennessee, USA, September 2012.

Hung Y., Yeh P., Harwood D. (1985), "Passive Ranging to Known Planar Point Sets", Proceedings of the IEEE International Conference of Robotics and Automation, St. Louis, Missouri, USA, March 25-28, 1985, pp. 80-85.

Lachapelle G. (2010), "ENGO 625 Advanced GNSS Theory and Applications", Lecture notes, Department of Geomatics Engineering, University of Calgary, Canada.

Li T. (2010), "Ultra-tightly Coupled GPS/Vehicle Sensor Integration for Land Vehicle Navigation", Ph.D. Thesis, Department of Geomatics Engineering, University of Calgary, Canada.

Lichti D. (2011), "ENGO 642 Optical Imaging Metrology", Lecture notes, Department of Geomatics Engineering, University of Calgary, Canada.

Mohammadi E. (2011), "Indoor Location Based Services", MSc Thesis, Department of Geomatics Engineering, University of Calgary, Canada.

Novak K., Bossler J. D. (1995), "Development and application of the highway mapping system of Ohio State University", The Photogrammetric Record, Vol: 15, Issue: 85, pp. 123-134.

Petersent I. R., McFarlane D. C. (1991), "Robust State Estimation for Uncertain Systems", Proceedings of the 30th IEEE Conference on Decision and Control, December 11-13, 1991, Vol. 3, pp. 2630-2631.

Petovello M. G. (2003), "Real-Time Integration of a Tactical-Grade IMU and GPS for High-Accuracy Positioning and Navigation", Ph.D. Thesis, Department of Geomatics Engineering, University of Calgary, Canada.

Prahl D., Veth M. J. (2011), “Coupling Vanishing Point Tracking with Inertial Navigation to Produce Drift-Free Attitude Estimates in a Structured Environment”, Proceedings of the Institute of Navigation GNSS 2011, Portland, Oregon, USA, September 2011, pp. 3571-3581.

Rosten E., Drummond T. (2006), “Machine learning for high speed corner detection”, Proceedings of the 9th Euproean Conference on Computer Vision, Vol. 1, 2006, pp. 430–443.

Ruotsalainen L. (2012), “Visual Gyroscope and Odometer for Pedestrian Indoor Navigation with a Smartphone”, Proceedings of the Institute of Navigation GNSS 2012, Nashville, Tennessee, USA, September 2012.

Se S., Lowe D. G., Little J. J. (2002), “Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks”, The International Journal of Robotics Research, Vol. 21, pp: 735-758, August 2002.

Se S., Lowe D. G., Little J. J. (2005), “Vision-Based Global Localization and Mapping for Mobile Robots”, IEEE Transaction On Robotics, Vol. 21, No. 3, pp: 364-375, June 2005.

Soloviev A. (2008), “Tight coupling of GPS, laser scanner, and inertial measurements for navigation in urban environments”, IEEE/ION PLANS Conference, Monterey, California, May 2008.

Susie M. (2012), “Gait Analysis for Pedestrain Navigation Using MEMS Handheld Devices”, MSc Thesis, Department of Geomatics Engineering, University of Calgary, Canada.

Trawny N., Mourikis A. I., Roumeliotis S. I., Johnson A. E. (2007), “Vision-aided inertial navigation for pin-point landing using observations of mapped landmarks”, Journal of Field Robotics-Special Issue on Space Robotics, Vol. 24, Issue. 5, pp. 357-378.

Wang J. H. (2006), “Intelligent MEMS INS/GPS Integration For Land Vehicle Navigation”, PhD Thesis, Department of Geomatics Engineering, University of Calgary, Canada.

Xie L., Soh Y. (1993), “Robust Kalman filtering for uncertain system”, Systems and Control Letters, Vol. 22, No. 2, pp. 123-129.

Yang Y. (2006), “Adaptive Navigation and Kinematic Positioning”, Beijing Survey and Mapping Press

Yuan Z., Li X., Wang J. L., Yuan Q., Xu D., Diao J. (2011), “Methods of 3D map storage based on geo-referenced image database”, Transactions of Nonferrous Metals Society of China, Vol. 21, No. 3, pp. 654-659.