



HAL
open science

Multi-sources fusion based vehicle localization in urban environments under a loosely coupled probabilistic framework

Lijun Wei

► **To cite this version:**

Lijun Wei. Multi-sources fusion based vehicle localization in urban environments under a loosely coupled probabilistic framework. Other. Université de Technologie de Belfort-Montbéliard, 2013. English. NNT : 2013BELF0204 . tel-01004660

HAL Id: tel-01004660

<https://theses.hal.science/tel-01004660>

Submitted on 11 Jun 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SPIM

Thèse de Doctorat



école doctorale sciences pour l'ingénieur et microtechniques

UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

Multi-sources fusion based vehicle
localization in urban environments
under a loosely coupled
probabilistic framework

■ LIJUN WEI

SPIM

Thèse de Doctorat



école doctorale sciences pour l'ingénieur et microtechniques
UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

N° 2 | 0 | 5

THÈSE présentée par

LIJUN WEI

pour obtenir le

Grade de Docteur de

l'Université de Technologie de Belfort-Montbéliard

Spécialité : **Informatique**

Multi-sources fusion based vehicle localization in urban environments under a loosely coupled probabilistic framework

Unité de Recherche :
IRTES - SeT

Soutenue le 17 July 2013 devant le Jury :

ROLAND CHAPUIS	Président	Professeur au LASMEA, Polytech Clermont-Ferrand
PHILIPPE BONNIFAIT	Rapporteur	Professeur à l'Heudiasyc, UTC
JEAN-CHARLES NOYER	Rapporteur	Professeur au LISIC, ULCO
DAVID BÉTAILLE	Examinateur	Chargé de Recherche à l'IFSTTAR
MAAN EL BADAOUI EL NAJJAR	Examinateur	Maître de conférences HDR au LAGIS, USTL
VALÉRIE GOUET-BRUNET	Examinatrice	Directeur de Recherche à l'IGN
YASSINE RUICHEK	Directeur de thèse	Professeur à l'IRTES-SET, UTBM
CINDY CAPPELLE	Co-encadrante	Maître de Conférences à l'IRTES-SET, UTBM

ACKNOWLEDGMENTS

At first, I would like to acknowledge my thesis supervisor Prof. Yassine Ruichek and co-supervisor Dr. Cindy Cappelle. I still remember the first day when I arrived at Belfort, and every discussion with them. During the past four years, they have given me valuable guidance and a lot of encouragement. They patiently taught me and helped me to learn how to do research. They have provided me enough freedom to understand the research subject and propose new ideas, and also gave me many academic opportunities to broaden my research horizon. Those discussions with them not only enlarged my mind of research, their serious working style and academic attitudes also influenced me a lot.

I would like to acknowledge my thesis committee, Philippe Bonnifait, Jean-Charles Noyer, Valérie Gouet-Brunet, Roland Chapuis, Maan El Badaoui El Najjar and David Bétaille, for accepting and evaluating my thesis work. It's my great honor to discuss with them and get their pertinent suggestions for completing my thesis manuscript.

I would like to acknowledge my colleague Frédérick Zann, who helped me solving countless technical problems, from synchronizing sensors to acquiring experimental data, from debugging a piece of codes to update my Ubuntu system, etc. I would like to acknowledge my colleague You Li, for those discussions about research life and for his encouragement. My kind thanks also to my colleagues Dhouha Attia, Madeleine El Zaher and Safaa Moqqaddem, it's those daily talks with them made me more confident and relaxed to practice speaking French or even Arabic in my daily life. My thanks also to other group members, Cyril Meurie, Houssam Salmane, Youssef El MEerabet, Julian Murgia, Julien Moreau, Baudouin Dafflon, Saeid Moosavi and Yong Fang. They've helped me in numerous ways and given me a comfortable and friendly working atmosphere. I thank all the friends I've made in Belfort for their support and encouragement. They have made my life with more laughter during the past four years.

I would like to acknowledge Chinese Scholarship Council and UTBM for their financial support during my study in France. And finally, my best and sincere thanks to my families for their understanding and supports; my thanks to Cheng, who has always been there for sharing my happiness, complaints, confusion, who has been tirelessly supportive of me throughout my thesis work.

SOMMAIRE

1	Introduction	11
1.1	Background	11
1.2	Problem statement	12
1.3	Contribution	13
1.4	Structure of the manuscript	14
2	Related Work of Vehicle Localization	15
2.1	Overview	15
2.2	Relative localization approaches	16
2.2.1	Wheel encoder based odometry	16
2.2.2	Inertial navigation	17
2.2.3	Vision sensors based relative localization	19
2.2.3.1	Types of vision systems	20
2.2.3.2	Vision based relative localization methods	21
2.2.4	Range sensors based relative localization	23
2.3	Absolute localization approaches	25
2.3.1	Localization by triangulation	25
2.3.2	Localization by trilateration	26
2.3.3	Global navigation satellite system (GNSS)	26
2.3.3.1	Global positioning system (GPS)	27
2.3.3.2	DGPS	30
2.3.3.3	RTK-GPS	31
2.3.3.4	Other GNSS systems	32
2.3.4	Visual landmarks based localization system	32
2.3.4.1	Exteroceptive sensors with 2D/3D model	33
2.3.4.2	Exteroceptive sensors with visual feature model	36
2.3.5	Cellular localization	37
2.3.6	Ad-hoc localization (or VANET)	38
2.4	Multi-sensor fusion based localization	38

2.4.1	Loosely-coupled and tightly-coupled fusion	39
2.4.2	Different fusion strategies for vehicle localization	39
2.5	Conclusion	40
3	Stereovision/Laser Range Finder based Vehicle Localization	43
3.1	Overview	43
3.2	Visual odometry based vehicle motion estimation	43
3.2.1	Camera system modeling and calibration	44
3.2.1.1	Perspective camera modeling and calibration	44
3.2.1.2	Stereoscopic system modeling and calibration	48
3.2.2	Stereovision based visual odometry	51
3.2.2.1	Feature extraction	52
3.2.2.2	Feature matching	53
3.2.2.3	3D landmark reconstruction	55
3.2.2.4	2D feature tracking and reference pair updating	57
3.2.2.5	Moving feature elimination and rigid motion estimation	58
3.2.3	Error modeling of stereo visual odometry	62
3.2.3.1	Overview of uncertainty modeling and propagation	62
3.2.3.2	Uncertainty of stereo reconstruction	63
3.2.3.3	Uncertainty of stereo visual odometry	64
3.2.4	Conclusion	66
3.3	Laser range finder based vehicle motion estimation	66
3.3.1	Laser scan alignment methods	67
3.3.1.1	Classic Iterative Closest Point (ICP) Method	67
3.3.1.2	Outlier-rejection-ICP (OR-ICP)	68
3.3.1.3	Image-aided scan alignment	69
3.3.2	Error modeling of ICP process	73
3.3.2.1	Uncertainty of laser points	73
3.3.2.2	Uncertainty of OR-ICP estimation	75
3.3.3	Conclusion	76
3.4	Implementation and experimental results	76
3.4.1	Experimental platform	76
3.4.2	Experimental results of stereovision odometry and LRF odometry	78
3.4.3	Comparison of different LRF based ICP methods	79
3.5	Conclusion	83

4	GPS-Stereovision-LRF based Data Fusion for Vehicle Localization	87
4.1	Overview	87
4.2	Data fusion approaches	88
4.2.1	Kalman filter	88
4.2.2	Extended Kalman filter	89
4.2.3	Unscented Kalman filter	90
4.2.4	Information filter based sensor fusion	91
4.2.4.1	Information filter for linear process model	92
4.2.4.2	Unscented information filter for nonlinear process model	92
4.2.5	Other fusion methods for vehicle localization	93
4.2.6	Conclusion	94
4.3	Sensor coherence validation by extended NIS	94
4.3.1	Normalized innovation squared (NIS)	94
4.3.2	Extended NIS for multiple measurements validation	95
4.4	Integration of GPS-Stereovision-LRF for vehicle localization	96
4.4.1	Coordinate frames for vehicle localization	97
4.4.2	Vehicle motion prediction	98
4.4.3	Subsystem estimations	99
4.4.4	Validation of different sensor measurements	100
4.5	Implementation and experimental results	102
4.5.1	Experimental results of the industrial park sequence	102
4.5.2	Experimental results of the old town center sequence	104
4.5.2.1	With simulated GPS masks	105
4.5.2.2	With simulated GPS jumps	107
4.6	Conclusion	109
5	Horizontal/Vertical LRFs and GIS Maps Aided Vehicle Localization	111
5.1	Overview	111
5.1.1	Proposed method	113
5.1.2	Geographical Information System (GIS)	114
5.2	Vehicle pose estimation with GIS road map	116
5.2.1	Observation from road map-matching method	116
5.2.1.1	Road map-matching observation	117
5.2.1.2	Uncertainty of map-matching position	119
5.2.2	Pose update and covariance estimation with UIF fusion	120

5.3	Vehicle pose refinement with LRFs and original GIS building map	122
5.3.1	Environment analysis with horizontal and vertical LRFs	122
5.3.1.1	Line extraction from LRF scans	122
5.3.1.2	Environment analysis using detected features	127
5.3.2	Horizontal LRF based vehicle pose correction	129
5.3.2.1	Candidate building facades extraction	130
5.3.2.2	Association of LRF-H and map observations	130
5.3.2.3	Longitudinal/lateral position and orientation correction	133
5.3.2.4	Vehicle orientation correction	135
5.3.3	Vertical LRF based vehicle lateral position correction	137
5.4	New map layer generating with LRFs and GIS	138
5.4.1	New map layer of building landmarks	139
5.4.1.1	Uncertainty of LRF-V observation	139
5.4.1.2	Association with the original building map	141
5.4.1.3	Organization of new map objects	141
5.4.2	Localization with new building facades map	143
5.5	Implementation and experimental results	144
5.5.1	Experimental platform	144
5.5.2	Experimental results of the old town center sequence	146
5.5.3	Experimental results of the industrial park sequence	148
5.6	Conclusion and perspectives	157
6	Conclusions and Future Works	159
6.1	Conclusions	159
6.2	Perspectives	160
A	Transformation of GPS coordinates	181
A.1	Coordinate systems	181
A.2	Transformation from WGS84 to Extended Lambert II	182

INTRODUCTION

1.1/ BACKGROUND

Over the past three decades, intelligent vehicle systems or advanced driver assistance systems (ADAS) have continued to be an important research topic in transportation area. They can promise to reduce road accidents and eliminate traffic congestions. Intelligent vehicle systems are composed of two parts : onboard sensors are used to observe the environments around ; and then, the captured sensor data can be processed to interpret the environments and to make appropriate driving decisions. Some sensors have already been widely used in existing driver-assistance systems, e.g., camera systems are used in lane-keeping systems to recognize lanes on road ; radars (Radio Detection And Ranging) are used in adaptive cruise systems to measure distance to the vehicle ahead such that a safe distance can be guaranteed ; lidar (Light Detection And Ranging) sensors are used in the autonomous emergency braking system *Volvo City Safety*¹, to detect other vehicles or pedestrians in the vehicle's path to avoid collision ; accelerometers are used to measure vehicle speed changes which are especially useful for airbags ; wheel encoder sensors are used to measure wheel rotations in vehicle anti-lock brake system ; and GPS sensors are embedded on vehicles to provide the vehicle's global positions for path navigation.

In order to promote the development of autonomous vehicles, American Department of Defense has organized an off-road autonomous vehicle competition respectively in 2004² and 2005³, called **DARPA (Defense Advanced Research Projects Agency) Grand Challenge**. In this competition, vehicles have to pass through three narrow tunnels and navigate in more than 100 sharp left and right turns. No vehicle finished the whole trajectory in 2004. Then in 2005, Stanley vehicle from the Stanford Racing Team won the prize. After that, **DARPA Urban Challenge**⁴ was held in 2007. This competition requires teams to build autonomous vehicles capable of self-driving in real urban traffic. The vehicles have to perform complex maneuvers such as merging into the traffic flow, overtaking other vehicles, parking and negotiating without human intervention. BOSS vehicle from the Tartan Racing team⁵ of Carnegie Mellon University was the winner of this competition.

China has also organized this kind of competition for autonomous vehicles since 2009

-
1. http://www.euroncap.com/rewards/volvo_city_safety.aspx
 2. <http://archive.darpa.mil/grandchallenge04/>
 3. <http://archive.darpa.mil/grandchallenge05/>
 4. <http://archive.darpa.mil/grandchallenge/>
 5. <http://www.tartanracing.org/>

- **Future Challenge.** In 2011, the competition area was first time changed from closed road to real urban road. In November 2012, this competition was first held on both real urban and rural roads. In order to test the “4S” characteristics of vehicles, i.e., Safety, Smartness, Smoothness and Speed, all the self-driving vehicles have to pass through nearly 7km urban road and 16km rural road with tasks like traffic sign recognition, obstacle avoidance (static/moving vehicles and pedestrians), traffic merging, U-style turning, and decision making ability in area with simulated foggy weather, etc. 14 teams from different Chinese universities and research institutes have participated in this event.

Recently, companies like Google already made driverless cars onto the road. The Nevada Department of Motor Vehicles issued the first license for a Google self-driven car in May 2012. Up to September 2012, three U.S. states have passed laws permitting driverless cars : Nevada, Florida and California. Besides Google, a lot of traditional auto companies are developing their own self-driving vehicles. In 2011, a BMW vehicle drove itself from Munich to Ingolstadt in Germany, an Audi autonomous vehicle went up to Pikes Peak. GM's Alan Taub optimistically predicts that self-driving cars will be on road by the end of this decade⁶.

1.2/ PROBLEM STATEMENT

In order to improve the accuracy of assisted navigation systems or autonomous vehicles so as to guarantee the driving security on road, it is essential to know absolute or relative vehicle positions and orientations at all times. From map provider's point of view, localization is also one of the key points in geo-referencing process for mobile mapping systems.

Different types of sensors have been used to ensure vehicle localization accuracy. Global sensors like GPS (Global Positioning System) receiver can provide absolute position of the vehicle. It has been considered as a basic sensor for vehicle localization in outdoor environments for its advantage of high localization precision in long term. However, GPS signals are affected by atmospheric conditions, satellites distribution, radio signal noises, etc., localization accuracy of GPS receivers in short term is only to a few meters. In some specific locations of urban environments (e.g., streets with tall buildings around, tunnels), information provided by the GPS receivers might not be accurate or even unavailable due to signal reflection or poor satellite visibility. In the case of GPS reflection, pseudo-ranges provided by the received reflected signals would be longer than the real ranges. If these contaminated ranges are used for position estimation, localization results would be erroneous. Another type of sensors is dead-reckoning sensor (DR), like gyro, inertial navigation system (INS), wheel encoder odometer, which can estimate the relative motion of a vehicle. The vehicle motion estimated by the dead-reckoning sensors is mostly accurate in short term, though vehicle trajectory might drift in long term due to error accumulation.

Therefore, the global GPS measurements are usually integrated with other information sources to provide accurate, reliable and continuous localization services in urban environments, which is also the main problem to be addressed in this thesis. This thesis is part of project CPER “Intelligence du Véhicule Terrestre” (Intelligence of ground vehicle), developed within Systems and Transports Laboratory (SET) of Institute for Transportation Research, Energy and Society (IRTES), UTBM, France. Three main objects of the thesis

6. http://www.wired.com/magazine/2012/01/ff_autonomoucars/

are explained as follows :

- The first object is to explore the use of multi-sensor fusion for vehicle localization, using GPS sensor to provide absolute location, and stereoscopic system and laser range finder (LRF) to provide vehicle motion information as dead-reckoning sensors. Coherence between different observations is also taken into account, and only the validated sensor measurements are integrated for vehicle pose estimation.
- Since dead-reckoning methods are suitable only in short period, if GPS receiver cannot provide vehicle position for long time, especially within urban areas where satellite signals might be blocked or reflected, vehicle trajectory would gradually drift and the localization error cannot be bounded. In order to compensate for the problem of GPS outages in long term, the second object of this thesis is to explore the use of horizontal and vertical LRFs with a road network map and a building footprint map from a GIS database for correcting the predicted vehicle pose.
- The third object is to explore the use of vertical LRF for updating building map layer information. Since the environment information from different data sources might be inconsistent with each other, a new map layer of building facades can be generated from the vertical LRF perceptions and be reused for vehicle pose correction. If a vehicle is supposed to be driven in the same area, the error between LRF perception and the initial GIS map (which are produced with aerial image, or digital maps transformed from paper maps, etc.) can also be handled.

1.3/ CONTRIBUTION

In this work, two methods are proposed to improve the accuracy of vehicle localization process. The main contributions of this work are in twofold :

- **1) Implementation of a vehicle localization system with a GPS receiver, an on-board stereoscopic system (visual odometry), and a laser range finder (scan alignment) considering their coherence with each other.**

At first, stereoscopic system based visual odometry method is used to predict vehicle displacement and orientation change. Stereovision based visual odometry is not limited to flat ground assumption, which means that it can provide vehicle pose with 6DOF (degree of freedom). This method is then complemented by adding a LRF sensor. Vehicle displacement and orientation change can be estimated by scan alignment between two consecutive LRF scans. Instead of directly using all sensor observations for vehicle pose correction, a sensor selection step is applied before the fusion step to validate the coherence of different observations. Then, information provided by the validated sensors are combined under a loosely coupled probabilistic framework with an information filter.

Since visual odometry and laser scan alignment are independent research subjects, there are a lot of existing problems and research results in each subject, such as detection of moving objects, strategy of poses estimation, solution of lateral drift and longitudinal scale problems. Classic algorithms are carefully chosen from those existing methods in the robotic research area, plus several modifications to improve the pose estimation accuracy. The proposed method in this work is tested with real data and evaluated by RTK-GPS data.

– 2) **A geographical information system (GIS) aided vehicle localization method is used to bound the localization error of dead-reckoning methods if GPS receiver fails for long time.**

If GPS receiver cannot provide vehicle positions for long time, vehicle trajectory based on only relative localization methods might gradually drift and the error cannot be bounded. In order to compensate for this problem, *a priori* environmental information provided by digital maps is added into the localization system. Two GIS map layers have been used : the road networks map and building footprints map. The road network map is used to estimate vehicle position by road map-matching method. Then, the vehicle pose is refined by registration between real observation from the two onboard LRFs (one horizontal and one vertical) and *a priori* observation provided by the building footprint map.

Experiments are implemented with two real data sequences : one in an industrial area and another one in the old town center of Belfort. Experimental results show that GIS maps can help to bound the localization error effectively : the road map can help to obtain an approximate estimation of vehicle position (by projecting vehicle position onto the corresponding road segment), while the integration of building information can help to refine this pose estimation.

1.4/ STRUCTURE OF THE MANUSCRIPT

In **Chapter 2**, existing approaches for vehicle localization are reviewed, including relative localization methods, absolute localization methods, and different sensor fusion strategies for multi-sensor based localization.

In **Chapter 3**, stereovision based visual odometry and horizontal LRF based scan alignment methods are respectively presented. The proposed methods are tested with real data and evaluated by using RTK-GPS data as ground truth.

In **Chapter 4**, a localization method is presented by integrating a stereoscopic system (odometry), a horizontal LRF (scan alignment), and a GPS receiver. For taking advantages of their complementarity and redundancy, coherence checking between the different measurements is performed before pose update.

In **Chapter 5**, a geographical information system (GIS) aided vehicle localization method is proposed to bound localization error when GPS receivers encounter long term outages. Two GIS layers are used : the road network map layer and the building map layer. This proposed method has been tested with two real data sequences.

Finally, conclusions and some research perspectives for this thesis are presented in **Chapter 6**.

RELATED WORK OF VEHICLE LOCALIZATION

2.1/ OVERVIEW

During the past years, ADAS (Advanced Driver Assistance Systems) and autonomous vehicle navigation (e.g., localization, path planning, obstacle avoidance, etc.) have become important research areas for constructing innovative transportation methods. An autonomous vehicle consists of a vehicle (moving robot), several sensing and processing devices, and mechanical actuators to implement commands like braking or steering. By perceiving environment around the vehicle (e.g., road lane, traffic signs, road crossing, dynamic and static objects in the driving area) with vision sensors, together with the vehicle's pose information (position and orientation), the vehicle can autonomously make local and global path planning, and send commands to drive itself to a desired destination.

To ensure the above function, it is essential to accurately localize the vehicle at all times. A lot of methods have been proposed in the literature, as presented in Fig. 2.1. These methods will be explained in the following sections.

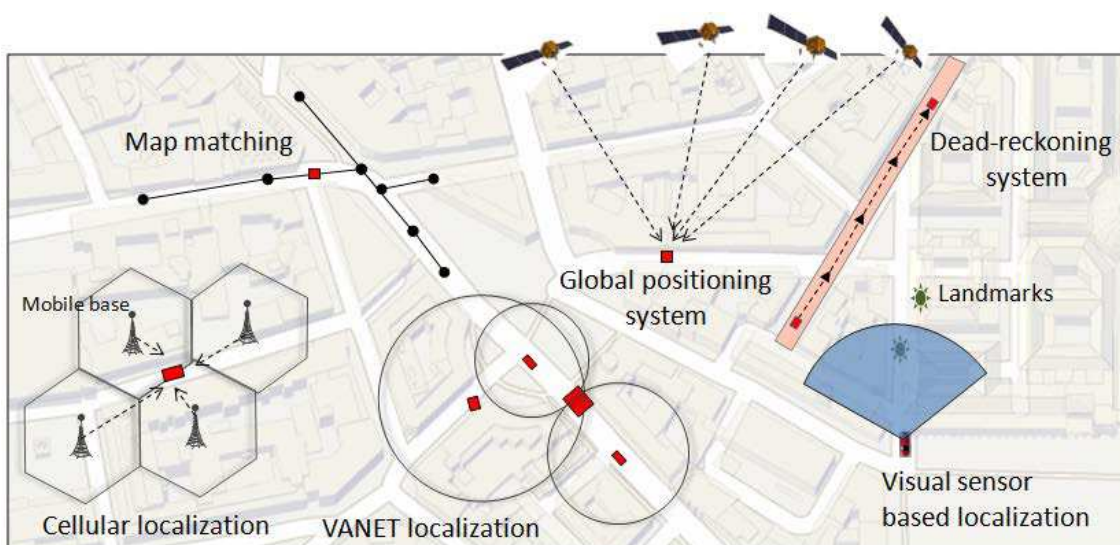


FIGURE 2.1 – Different localization methods based on the figure in [26]

In this chapter, we make a review of existing vehicle localization approaches. According

to measurement types, these methods are classified as relative localization approaches (dead-reckoning methods, DR) and global localization methods (global navigation satellite system, landmarks based localization, map matching, cellular localization and VANET localization). These two categories of localization methods are respectively presented in sections 2.2 and 2.3. Then, several multi-sensor fusion based localization methods are presented in section 2.4.

2.2/ RELATIVE LOCALIZATION APPROACHES

Relative localization approaches with proprioceptive sensors like gyro, inertial navigation system, wheel encoder odometer or with exteroceptive sensors like camera or laser range finder, are based on the estimation of vehicle motion in a period of δt . When initial state of a vehicle is known (initial position and orientation in the global reference system), its current position and orientation can be estimated by integrating the relative translation and rotation with the previous vehicle state. Proprioceptive sensors based relative localization methods are self-contained and no external information is needed.

2.2.1/ WHEEL ENCODER BASED ODOMETRY

Wheel encoder is a device which allows to measure the distance traveled by a wheel in a period of time. It is mounted on vehicle wheel to measure the elementary rotation of the wheel. When the wheel radius is known, the traveling distance can be deduced from the elementary rotation¹.

Optical and digital encoders are now widely used on mobile robots for their high frequency and low cost. As seen in Fig. 2.2(a), an optical encoder consists of a light source, a light detecting sensor, and a rotating disk coded with opaque and transparent patterns. When the disk rotates with the shaft, the patterns on the disk will interrupt the light emitted into the light sensor, and generate a pulse signal output. By counting the number of output pulses, the angular motion of the wheel can be measured.

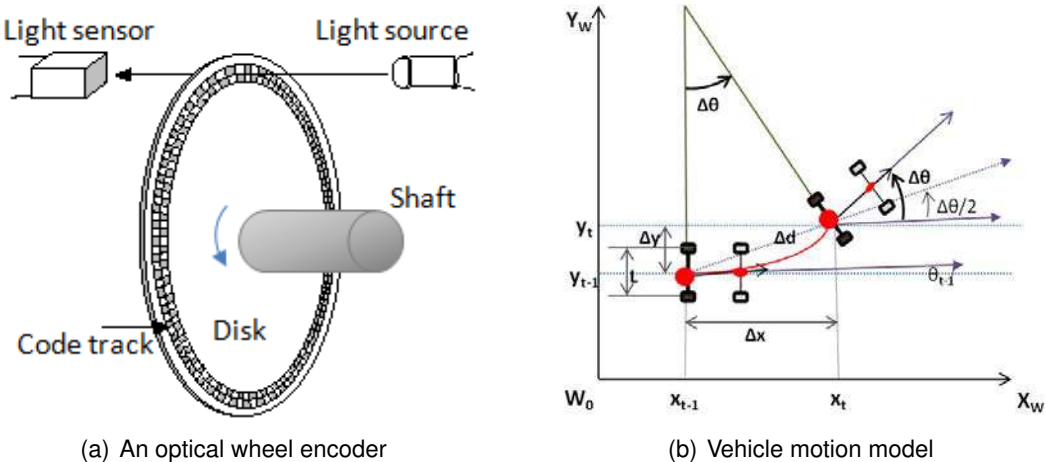


FIGURE 2.2 – Wheel encoder and vehicle motion model

1. http://geology.heroy.smu.edu/~dpa-www/robo/Encoder/imu_odo/index.IE.htm#sec2

As shown in Fig. 2.2 (b), a vehicle can be represented by a kinematic model with two rear wheels, and a single point in the center of the front wheel axle. Vehicle center is represented by a point in the center of the rear wheel axle. If two wheel encoders are respectively mounted on the two rear wheels with known wheel radius, the traveling distances of the left and right wheels during period δt can be respectively measured, as $\Delta d_{t,l}$ and $\Delta d_{t,r}$. Then, the translation Δd_t and rotation $\Delta \theta_t$ of the vehicle center can be estimated by the distance $\Delta d_{t,l}$ and $\Delta d_{t,r}$:

$$\begin{cases} \Delta d_t = \frac{\Delta d_{t,l} + \Delta d_{t,r}}{2} \\ \Delta \theta_t = \frac{\Delta d_{t,r} - \Delta d_{t,l}}{L} \end{cases} \quad (2.1)$$

where L is the axle length of the vehicle (see Fig. 2.2 (b)), i.e., the distance between the left and right rear wheels.

Assume that the ground is flat, let $X_t = [x_t, y_t, \theta_t]$ denote the vehicle pose vector at time t , where (x_t, y_t) and θ_t are vehicle position and orientation (yaw angle) in the global navigation system. The current vehicle pose at time t can be predicted with previous vehicle state $[x_{t-1}, y_{t-1}, \theta_{t-1}]$ and current motion vector $u_t = (\Delta d_t, \Delta \theta_t)$. The transition vehicle model at time t is $X_t = f(X_{t-1}, u_t, \delta t) + \alpha_t$, written as :

$$\begin{cases} x_t = x_{t-1} + \Delta d_t \cos(\theta_{t-1} + \Delta \theta_t / 2) + \alpha_{1t} \\ y_t = y_{t-1} + \Delta d_t \sin(\theta_{t-1} + \Delta \theta_t / 2) + \alpha_{2t} \\ \theta_t = \theta_{t-1} + \Delta \theta_t + \alpha_{3t} \end{cases} \quad (2.2)$$

where $\alpha_t = [\alpha_{1t}, \alpha_{2t}, \alpha_{3t}]^T$ is the process noise.

Due to bad estimation of the wheel radius or wheel deformation in bad soil conditions, like rock or muddy areas, the estimated vehicle motion from wheel encoders might be erroneous. The localization error will accumulate gradually in long term due to motion integration.

2.2.2/ INERTIAL NAVIGATION

Inertia is the resistance of a physical object body to maintain constant translation/rotational velocities if any force is applied on it. Inertial navigation system (INS), also named as inertial measurement unit (IMU), is composed of three orthogonal rate-gyros and three orthogonal accelerometers, respectively measuring the 3D angular rates (yaw, pitch and roll, see Fig. 2.3) and 3D linear accelerations (in direction X_M , Y_M and Z_M) of the object on which the system is mounted.

INS has high sampling frequency and it is able to accurately measure the rapid changes of angular rotation rates and linear accelerations in short term. If the system is well initialized, we can estimate the current vehicle orientation and position by integration (Fig. 2.4). Vehicle orientations can be derived by integrating the vehicle angular velocity over time. And vehicle linear velocity and position can be respectively derived by single and double integration of the acceleration in a period of time. INS systems have been used on ground vehicles, spacecrafts, ships, and submarine vehicles [168], etc.

There are two types of inertial systems : stable platform systems and strap-down systems.

- **Stable platform system (mechanical system)** : stable platform is an inertial platform which uses gyros to maintain the accelerometers in a fixed attitude. Inertial sensors are mounted on the platform which is aligned with the global frame. The gyros mounted on

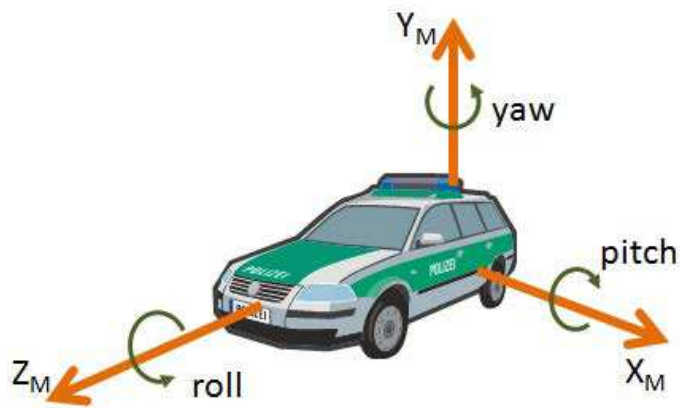


FIGURE 2.3 – 3D orientations of vehicle movement : yaw, pitch, roll

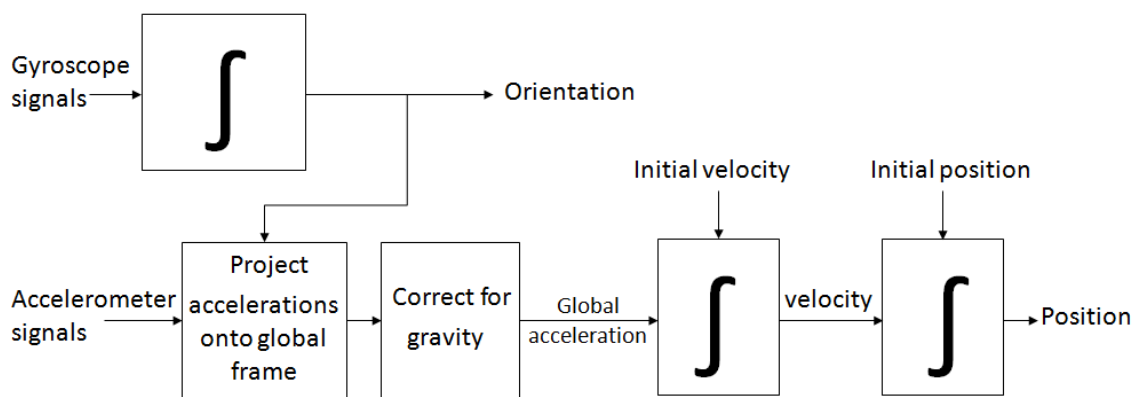


FIGURE 2.4 – Localization algorithm of INS system

the platform can detect any platform rotations. Then, these rotation signals are sent back to torque motors to rotate the gimbals (frames) of the platform in all three axes, in order to remove the rotations and keep the platform aligned with the global frame.

- **Strap-down system (electrical system)** : there is no gimbals or inertial platform in a strap-down system. gyros and accelerometers are rigidly mounted on a device (e.g., vehicle). The inertial sensors move with the device, thus movements of the device are measured in the sensors' body frame instead of the global frame. Therefore, the measured acceleration should be transformed into the global frame for navigation application.

Gyrometer : gyrometer (gyro) measures the angular velocity of a system in one direction. The three gyros of an INS can measure angular velocities of the system in three directions of the inertial reference frame, then 3D orientations can be derived (see Fig. 2.3). There are various types of gyros, like mechanical gyro, fibre optic gyro (FOG), and light-weight MEMS (Micro-machined electromechanical systems).

Accelerometer : accelerometer measures the linear acceleration of a system in one direction. The three accelerometers of an INS can measure the linear accelerations of a system in three directions (left - right, up - down and forward - back) with respect to the moving system. The outputs of an accelerometer should minus the gravitational acceleration : $a = f - g$, where f is the vehicle acceleration with respect to the inertial frame (in m/s^2), g is the acceleration produced by gravity, and a is the acceleration produced without gravitational forces.

Though effects of gyro drift and accelerometer bias are relatively small in short term, as the angular and linear velocities must be integrated once and twice to provide the orientation and position, small errors in short term might result in unbounded error of integrated measurements. Furthermore, if the sensor navigation frame and body frame is not well calibrated, the localization results might be not accurate.

2.2.3/ VISION SENSORS BASED RELATIVE LOCALIZATION

During recent years, computer vision based odometry (visual odometry) was proposed as a kind of relative localization method. Without any prior knowledge of the environment nor a predefined motion model of the vehicle, visual odometry can estimate the path of a camera-equipped vehicle by calculating the ego-motion between consecutive images in a video flow. Structure from motion (SFM) methods also use camera system to provide both the environment structure and the camera motion information. The principles of the two methods are almost the same except that the former focuses more on camera motion estimation, while the latter focuses more on structure reconstruction.

Visual odometry has been studied and used in a lot of indoor and outdoor localization projects [87] [133] [72] [167] [41], even on Mars [37] (see Fig. 2.5).

Compared with wheel encoder based odometry, the estimation from visual odometry are not influenced by wheel deformation or slippage caused by bad soil condition, but much more related to the environment around the camera and the illumination condition.

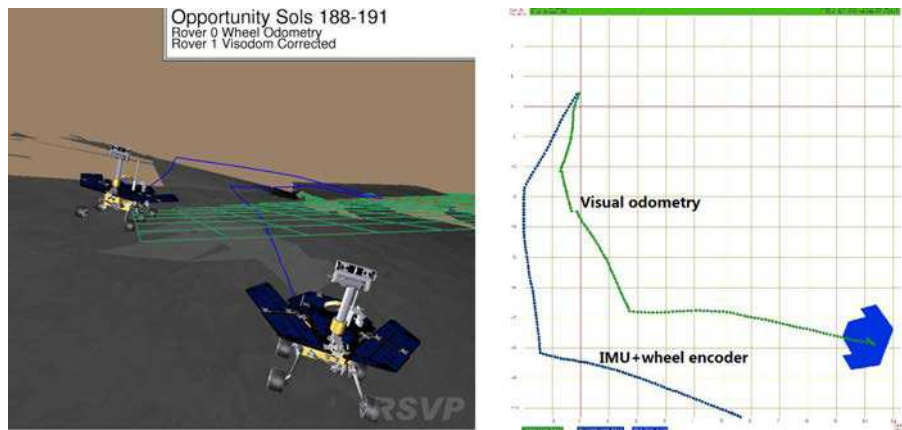


FIGURE 2.5 – (Left) NASA’s Mars Exploration Rovers (MER). (Right) Estimated trajectories of the rover : green trajectory shows the rover locations estimated by the visual odometry, while the blue trajectory shows the path estimated by the IMU and wheel encoders [37]

2.2.3.1/ TYPES OF VISION SYSTEMS

Different types of cameras have been used for vision based localization, such as monocular perspective cameras used in [115] [137], perspective stereoscopic systems used in [119] [79] [107], and omnidirectional cameras used in [152] [140]. The most often used camera systems are perspective camera systems (see Fig. 2.6(a)² and Fig. 2.6(b)³) for their simple geometric configuration and low-cost. Recently, omnidirectional camera systems have also been used for their larger fields of view (FOV), e.g., the 360° spherical camera system with multi-camera in Fig. 2.6(c)⁴ and the omnidirectional camera with mirror in Fig. 2.6(d)⁵. Omnidirectional camera systems have the advantage that the same object can stay longer in the camera’s FOV which is important for object tracking, but the geometric configurations of these camera systems are more complex than the perspective camera systems.



(a) Perspective camera (b) Stereoscopic system (c) Multi-camera system (d) Omnidirectional camera

FIGURE 2.6 – Some camera systems used for vision based localization

After choosing a camera type, there are different system configurations : monocular system with only one camera, like in Fig. 2.6 (a, d) ; or multi-camera system, like in Fig. 2.6 (b,

2. www.adept.net.au
 3. www.ptgrey.com
 4. www.ptgrey.com/products/ladybug5
 5. www.itr-store.com

c). For a multi-camera system, since the relative pose (relative position and orientation) between all the cameras can be known by calibration before the experiments, the system can directly reconstruct the corresponding image pixels into 3D Euclidean space.

2.2.3.2/ VISION BASED RELATIVE LOCALIZATION METHODS

After obtaining an image sequence from a camera system, corresponding image pixels (dense method for all the pixels in the image, or sparse method for only image features) are searched in consecutive image frames through matching or tracking methods. With corresponding pixels, the rigid transformation of the camera system between time t and time $t + 1$ can be estimated using 2D/2D [115], 3D/2D [115] or 3D/3D [52] methods. These motion estimation methods are summarized and concluded as follows.

in Tab 2.1 to Tab 2.3, Q^i is a 3D point in reference system, (q_1^i, q_2^i, q_3^i) are its corresponding image points in three consecutive image frames, $(q_{1,l}^i, q_{1,r}^i)$ are corresponding image points viewed by left and right cameras at time t , $(q_{2,l}^i, q_{2,r}^i)$ are corresponding image points viewed by left and right cameras at time $t + 1$.

1. 2D/2D method by minimizing 2D position error between corresponding image pixels.

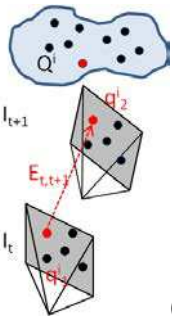
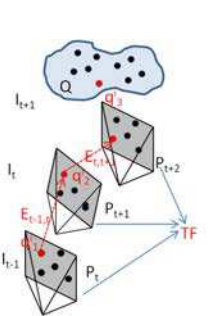
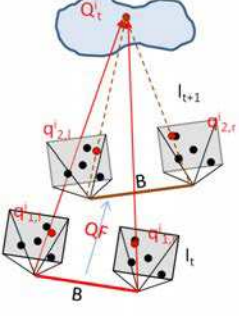
Method	2D-2D position error		
Camera type	Monocular	Monocular	Stereo
Constraint	Epipolar constraint	Trifocal tensor	Quadrifocal tensor
			
Cost function	$\arg \min \sum_{i=1}^N d(q_1^i \leftrightarrow q_2^i, F)^2$	$(b) : \arg \min \sum_{i=1}^N d(q_1^i \leftrightarrow q_2^i \leftrightarrow q_3^i, TF)^2$ $(c) : \arg \min \sum_{i=1}^N d(q_{1,l}^i \leftrightarrow q_{1,r}^i \leftrightarrow q_{2,l}^i \leftrightarrow q_{2,r}^i, QF)^2$	
Possible solutions	5 or 8 points algorithms	RANSAC or M-estimator with 6 points	
	For monocular method, a scale factor should be provided from other sensor like GPS or object with known size		

TABLE 2.1 – Motion estimation by 2D/2D method

• **Epipolar geometry** : as shown in Tab 2.1(a), camera motion parameters are estimated with two corresponding image point sets $\{q_1^i\}$ and $\{q_2^i\}$, $i = 1, \dots, n$ between two consecutive image frames at time t and $t + 1$. The calculation is based on epipolar geometry constraint : point $\{q_1^i\}$ can be transformed to $\{q_2^i\}$ on image frame I_{t+1} by the fundamental matrix F ,

which describes the geometric relation between two camera poses (see section 3.2.1.2 for more details). By minimizing the position error between $\{q_1^i\}$ and $\{q_2^i\}$, the fundamental matrix can be solved with Nistér's 5-point algorithm [114], or Longuet's 8-point algorithm [93], etc. Then, the translation vector and rotation matrix between two poses can be decomposed from the fundamental matrix F .

- **Trifocal tensor** : trifocal tensor TF in Tab 2.1(b) plays the same role in three views as the fundamental matrix in two views. It is a $3 \times 3 \times 3$ matrix which incorporates geometric relations between three views. The corresponding image point sets $\{q_1^i, q_2^i, q_3^i\}$ can be found across three views, between three consecutive frames I_{t-1}, I_t, I_{t+1} as shown in Tab 2.1(b), or between two simultaneous frames $I_{t,l}, I_{t,r}$ plus one consecutive frame $I_{t+1,l}$ (or $I_{t+1,r}$) in the case of using a stereoscopic system as shown in Tab 2.1(c). Then, the trifocal tensor [70] can be estimated by minimizing the differences between the three point sets using RANSAC or M-estimator based linear/nonlinear minimization methods. With the estimated trifocal tensor, the transformation matrix between every two camera poses can be derived.

- **Quadrifocal tensor** : quadrifocal tensor QF in Tab 2.1(c) describes the geometric relations across four camera poses. For example, when a stereoscopic camera system is used [41], the quadrifocal tensor between four images (two consecutive image pairs) can be estimated by minimizing the differences between the four corresponding image point sets $\{q_{1,l}^i, q_{1,r}^i, q_{2,l}^i, q_{2,r}^i\}$ using linear or nonlinear minimization methods. Then, the transformation matrix between every two camera poses can be derived. Since the calculation of quadrifocal tensor is complicated, it is always replaced by trifocal tensor and fundamental matrices.

2. 3D/2D method by minimizing reprojection error of reconstructed 3D points and corresponding 2D image pixels.

Method	3D-2D re-projection error	
Camera type	Monocular	Stereo
	<p>(a)</p>	<p>(b)</p>
Cost function	$\arg \min \sum_{i=1}^N d(Q^i \leftrightarrow q_3^i, K[R_{t,t+1} T_{t,t+1}])^2$	
Solution	DLT (Direct linear transformation) Nonlinear minimization, e.g., Levenberg-marquardt	

TABLE 2.2 – Motion estimation by 3D/2D method

- **Monocular method** : as seen in Tab 2.2(a), corresponding image pixels q_1^i in I_{t-1} and q_2^i in I_t are reconstructed into a 3D point Q^i ; then, Q^i is reprojected onto another image frame I_{t+1} with camera matrix $K[R_{t,t+1}|T_{t,t+1}]$, which contains camera intrinsic parameters and camera motion from time t to $t + 1$ (see section 3.2.1.1 for more details about camera parameters). Camera motion parameters can then be estimated by minimizing the error between the reprojected point set $\{K[R_{t,t+1}|T_{t,t+1}]Q^i\}$ and the corresponding image point set $\{q_3^i\}$ on image frame I_{t+1} [115][137]. Since the baseline between two camera poses is unknown, the estimated translation vector is up to a scale factor which should be calculated with other sensors (like wheel encoder, GPS) or an object with known size.

- **Stereo method** : in a stereoscopic system, the relative pose between left and right camera centers is generally fixed and can be known before the localization process by calibration, therefore the scale problem of monocular method can be avoided [87]. As seen in Tab 2.2(b), the corresponding image points $\{q_{1,l}^i, q_{1,r}^i\}$ at time t are directly reconstructed into 3D point $\{Q^i\}$ in the 3D space. Then, the camera motion from time t to $t + 1$ can be estimated by reprojecting the reconstructed point $\{Q^i\}$ onto the left or right image frames at time $t + 1$, and minimizing the reprojected error like in the monocular method.

3. 3D/3D method by minimizing 3D position error between corresponding reconstructed 3D points.

This method is used with a stereoscopic system. As seen in Tab 2.3, image pixels are firstly matched between the left and right images at time t ; then, the corresponding image points of $q_{1,l}^i$ and $q_{1,r}^i$ on the current left and right images at time $t + 1$ are respectively searched by matching (or tracking) methods. The image point pairs $\{q_{1,l}^i, q_{1,r}^i\}$ at time t and pairs $\{q_{2,l}^i, q_{2,r}^i\}$ at time $t + 1$ are respectively reconstructed into 3D points $\{Q_t^i\}$ and $\{Q_{t+1}^i\}$. The camera pose is then estimated by minimizing the 3D position error between two sets of reconstructed 3D points at time t and time $t + 1$, with least square method or maximum likelihood estimation (MLE) with a weight factor w_i in the cost function (in Tab 2.3) [52] [3] [71] [160] [151].

As the above three methods are all based on corresponding image point sets, some efforts have been employed to reject outliers during the matching and tracking steps [133] [50].

2.2.4/ RANGE SENSORS BASED RELATIVE LOCALIZATION

Range finder (RF) is a device which uses an electromagnetic wave beam (e.g., laser, infrared light) or ultrasonic sound to determine the distance between signal emitter and objects. RF sensors have been largely used in transportation area for object detection and tracking, such as ultrasonic sensors are embedded in the rear bumper of vehicles to assist vehicle parking, radar systems are installed in urban environments to detect the speed of vehicles, etc.

A basic LIDAR (Light Detection And Ranging) system, or LRF (Laser Range Finder) system, consists of a laser emitter and a reflected rotating mirror. The laser emitter can obtain the distances between objects and the emitter by measuring the traveling time of emitted

Method	3D-3D position error
Cost function	$\arg \min \sum_{i=1}^N d(Q_t^i \leftrightarrow Q_{t+1}^i, [R_{t,t+1} T_{t,t+1}])^2$
	$\text{MLE} : \arg \min \sum_{i=1}^N w_i d(Q_t^i \leftrightarrow Q_{t+1}^i, [R_{t,t+1} T_{t,t+1}])^2$
Solution	SVD or nonlinear minimization (e.g., Levenberg-marquardt)

TABLE 2.3 – Motion estimation by 3D/3D method

pulse (time of flight (TOF)). The distance is calculated through :

$$d = c \cdot \frac{\Delta t}{2} \quad (2.3)$$

where d is the distance between an object and the emitter, c is the speed of light in vacuum $c = 299792458m/s$, Δt is the light pulse traveling time of round-trip between the object and the emitter.

With the rotating mirror, LRF systems can quickly scan around and gather range measurements with a specified angle interval. They have been installed on ground and aerial vehicles for environment mapping and survey⁶. Meanwhile, they can be used for relative self-localization by estimating the translation and rotation between two consecutive poses [54] [13] (see more details in section 3.3).

Compared to ultrasonic sensors, LRF systems have narrower beam width, higher resolution and response rate. The benefit of sonar sensors is that they have large FOV, which is interesting for obstacle detection [92], especially in poor visibility areas like underwater area. Infrared devices [88] can calculate the distance of an object by the angle of returned infrared radiation beam. However, their FOV is narrow and the maximum measured range is limited. In outdoor environments, the observation of infrared sensors might be affected by other lights like sunshine.

6. <http://www.lidarmap.org/ELMF/vehicles/Default.aspx>

2.3/ ABSOLUTE LOCALIZATION APPROACHES

Absolute localization systems are based on multiple beacons with precisely known positions in the environment. These beacons can be active beacons, which transmit their identity periodically using light (infrared, laser) or RFID (radio-frequency identification) tags [67]. The receiver mounted on the mobile robot can receive the signal and calculate its distance (or angle) to the corresponding beacon. The beacons can also be passive beacons to reflect the light.

Beacon based localization and navigation is common for navigation of ships and airplanes, and also for localization in indoor environments (e.g., inside buildings, warehouses, etc.). Estimation of the absolute 2D position of a vehicle (mobile robot) requires its distances (or angles) information from at least three beacons. When at least three angles are known, the vehicle position can be obtained by triangulation (see section 2.3.1) ; when the distance information is known, the vehicle position can be deduced by trilateration (see section 2.3.2).

2.3.1/ LOCALIZATION BY TRIANGULATION

Localization by triangulation is to determine a vehicle's 2D position (x_t, y_t) based on at least three angle measurements to position-known beacons.

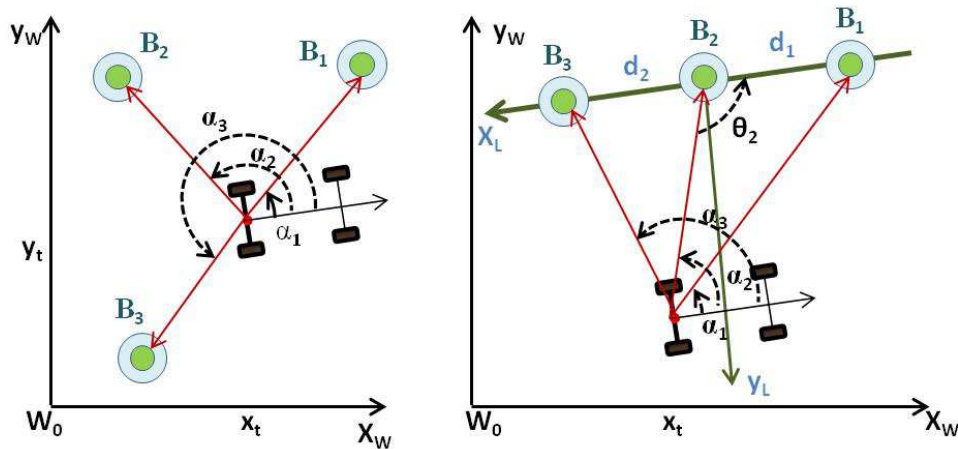


FIGURE 2.7 – Localization by triangulation with three beacons

As seen in Fig. 2.7 (left), if a sensor mounted on the vehicle can observe angles $\alpha_1, \alpha_2, \alpha_3$ between at least three beacons and the vehicle's longitudinal axis, the vehicle pose (x_t, y_t, θ_t) can be estimated with these three angles.

In Fig. 2.7 (right), we consider the case of three collinear beacons. Let beacon B_2 the origin of the local beacon frame L , and

$$\begin{cases} \beta_1 = \alpha_2 - \alpha_1 \\ \beta_2 = \alpha_3 - \alpha_2 \end{cases} \quad (2.4)$$

Then,

$$\tan(\theta_2) = \frac{(d_1 + d_2)\tan\beta_2\tan\beta_1}{d_1\tan\beta_2 - d_2\tan\beta_1} \quad (2.5)$$

where d_1, d_2 are respectively the distance from beacon B_1 and beacon B_3 to beacon B_2 . Then, the vehicle pose in the current beacon frame L is calculated by :

$$\begin{cases} x_L = \frac{d_1}{\tan\beta_1} \frac{\tan\theta_2 - \tan\beta_1}{1 + \tan^2\theta_2} \\ y_L = \frac{d_1}{\tan\beta_1} \frac{\tan\theta_2 - \tan\beta_1}{1 + \tan^2\theta_2} \tan\theta_2 \\ \theta_L = 2\pi - \alpha_2 - \theta_2 \end{cases} \quad (2.6)$$

2.3.2/ LOCALIZATION BY TRILATERATION

Trilateration is to determine a vehicle's position with distances to the beacons of known positions. Three beacons are required for determining a 2D position and 4 beacons are required for a 3D position.

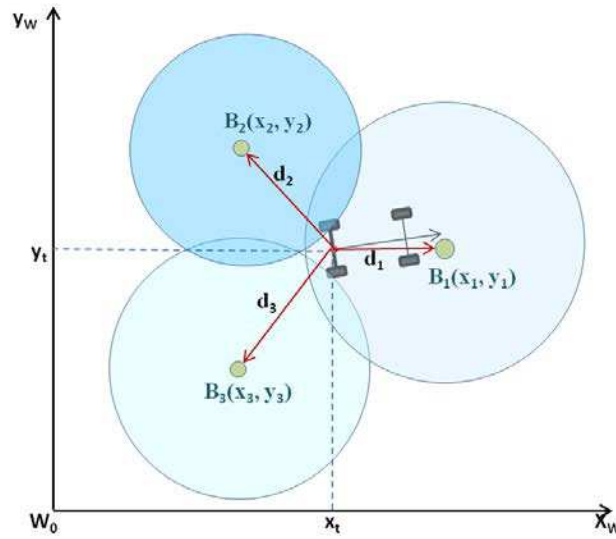


FIGURE 2.8 – 2D position estimation by trilateration with three beacons

As seen in Fig. 2.8, if the vehicle can observe three beacons $B_1(x_1, y_1), B_2(x_2, y_2), B_3(x_3, y_3)$ respectively with distances d_1, d_2 and d_3 , the vehicle position (x_t, y_t) can be calculated by solving the simultaneous equations in Eq. 2.7 :

$$\begin{cases} \sqrt{(x_t - x_1)^2 + (y_t - y_1)^2} = d_1 \\ \sqrt{(x_t - x_2)^2 + (y_t - y_2)^2} = d_2 \\ \sqrt{(x_t - x_3)^2 + (y_t - y_3)^2} = d_3 \end{cases} \quad (2.7)$$

When more beacons are observed, the vehicle location can be estimated by solving the overdetermined system with linear least squares method or nonlinear minimization method to reduce the ambiguity. Global navigation satellite system (GNSS) localization method is also one kind of beacons based global localization methods, as seen in Fig. 2.9.

2.3.3/ GLOBAL NAVIGATION SATELLITE SYSTEM (GNSS)

Global navigation satellite system (GNSS) is the most popular tool for vehicle global localization and navigation in outdoor environments. It is a kind of absolute localization

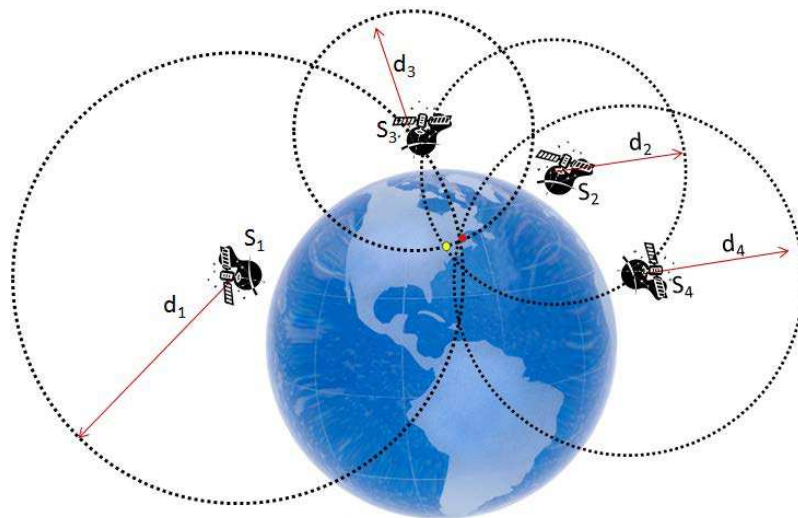


FIGURE 2.9 – Localization based on GPS trilateration

strategies, which calculates the traveling distances of satellite signals from at least four visible satellites to on-board receiver, then uses the trilateration method to compute the position of the receiver. This method can provide accurate positions in long term, but as the GNSS signals are affected by atmospheric conditions, satellite positions, radio signal noises, etc., the localization accuracy in short term is only to a few meters.

2.3.3.1/ GLOBAL POSITIONING SYSTEM (GPS)

Global Positioning System⁷ (GPS) [53] is a satellite navigation system owned by USA to provide positioning, navigation, and timing services. It was at first used only by American Defense Department for military purpose. It has been open to the public since 1985.

This system is composed of three parts : spatial part, control part, and user part.

- **Spatial part** : as seen in Fig. 2.10, GPS constellation consists of 24 solar-powered satellites equally-spaced on 6 earth orbit plans above the earth, with altitude about 20,000km. Each satellite circles the Earth twice a day. The 24 satellites can ensure that there are at least four satellites in view from any point on the earth planet. In June 2011, U.S. Air Force completed a GPS constellation expansion configuration. Now, the GPS constellation consists of 27-satellites with improved coverage in most area of the world. Each satellite transmits a periodic pseudo-random code to users on earth with two different frequencies (designated L1 and L2) in the internationally assigned navigational frequency band.

- **Control part** : GPS control part is a global network of ground control stations on earth. These stations continually track the GPS satellites positions and analyze their transmissions, then send commands and data to the constellation to adjust the errors of GPS ephemeris and satellite atomic clock. Current operational control part includes a master control station, an alternate master control station, 4 dedicated ground antennas, 12 command and control antennas, and 16 monitoring sites (six from the Air Force and 10 from

7. <http://www.gps.gov/>

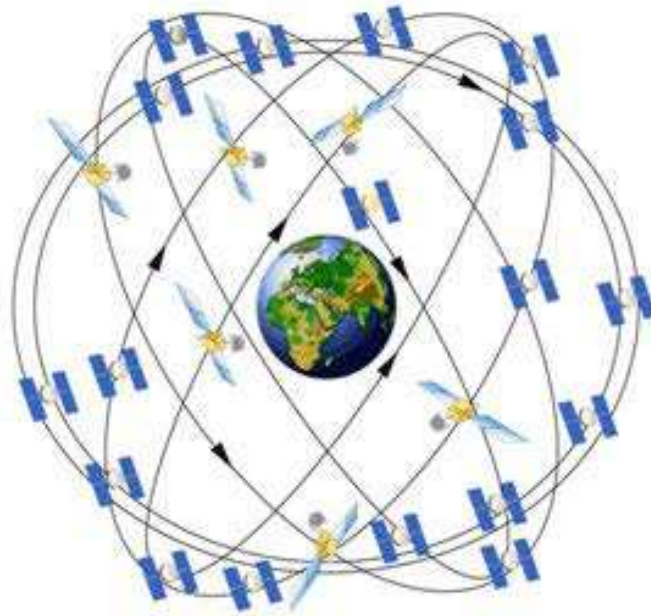


FIGURE 2.10 – GPS satellites in space

the National Geospatial-Intelligence Agency).

- **GPS user part** : user part of GPS is a set of civilian and military GPS receivers capable of receiving the GNSS signals and estimating the positions of themselves.

• **GPS localization procedure** : each GPS satellite can send long digital patterns called pseudo-random code. The transmit time of each signal is known since each GPS satellite is equipped with an accurate atomic clock. Meanwhile, as the GPS receiver is equipped with an ordinary quartz clock, the receive time of the satellite's signal arriving at the receiver can also be known.

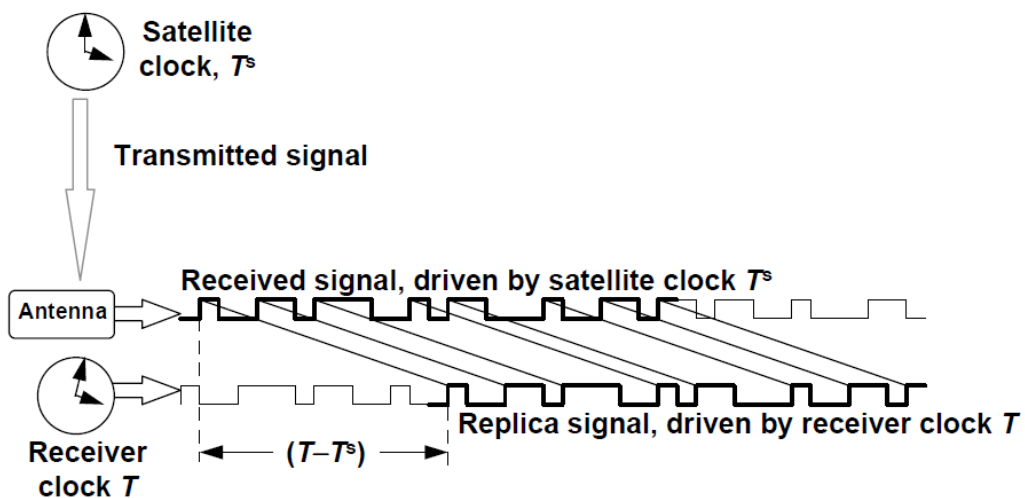


FIGURE 2.11 – GPS pseudo-range is related to the satellite and receiver clocks [17]

With the difference of transmit and receive times of the signal pattern (see Fig. 2.11), the

distance observation ρ_i to the i^{th} satellite can be calculated by multiplying the traveling time with the speed of light :

$$\rho_i = (T_i - T_i^S)c \quad (2.8)$$

where T_i is the known reading of the receiver clock when the signal is received, T_i^S is the reading of the satellite clock when the signal is transmitted, and c is the speed of light. If the clock time T_i equals the true receive time t_i plus a receiver clock bias τ , T_i^S equals the true transmit time t_i^S plus the satellite clock bias τ_i^S , as :

$$\begin{aligned} T_i &= t_i + \tau \\ T_i^S &= t_i^S + \tau_i^S \end{aligned} \quad (2.9)$$

Replacing Eq. 2.8 by Eq. 2.9, the range ρ_i from the GPS receiver (at receive time) to the i^{th} satellite (at transmit time) can be written as :

$$\rho_i = (t_i - t_i^S)c + (\tau - \tau_i^S)c = d_i + (\tau - \tau_i^S)c \quad (2.10)$$

where

$$d_i = \sqrt{(x - S_{i,x})^2 + (y - S_{i,y})^2 + (z - S_{i,z})^2} \quad (2.11)$$

thus, ρ_i is called pseudo-range and d_i is the true range. Since GPS messages allow knowing the i^{th} satellite 3D position $(S_{i,x}, S_{i,y}, S_{i,z})$ in space and the satellite clock bias τ_i^S , there are 4 unknowns in Eq 2.10 and Eq 2.11 that we need to solve : the receiver position (x, y, z) and the receiver clock bias τ . Therefore, at least four satellites are needed for estimating the 3D position of a GPS receiver.

• **Errors in GPS localization process.** Though GPS is reliable and precise for global localization, several errors exist in GPS localization procedure [53].

- **Satellite errors** : GPS satellites might send bad almanac data or report false positions of themselves. The ephemeris errors (or orbital error) and satellite clock error can be periodically corrected by GPS control stations.

- **Signal propagation errors** : as described above, a GPS receiver calculates its distance to the satellites by assuming that signals pass through the atmosphere at the speed of light in vacuum. However, the earth's atmosphere slows the propagation of signals, especially when they go through the ionosphere and troposphere layers. This delay will add errors into the distance measurements.

- **Geometry arrangements of the satellites seen by the GPS receiver** : as shown in Fig. 2.12 (left), if two satellites seen by the GPS receiver are far apart in the sky, localization uncertainty area will be small ; as shown in Fig. 2.12 (right), if two satellites are close to each other, the size of uncertainty area will be large. The ideal geometry arrangement of the satellites requires that the satellites in the receiver's FOV are separated with large relative angles.

Influence of the satellite relative geometry to the accuracy of the GPS receiver position is measured by a parameter called PDOP (position dilution of precision) [39] :

$$PDOP = HDOP^2 + VDOP^2 \quad (2.12)$$

where $HDOP$ is the Horizontal Dilution of Precision on horizontal plane, and $VDOP$ is the Vertical Dilution of Precision for altitude. A higher PDOP represents a poor satellite configuration.

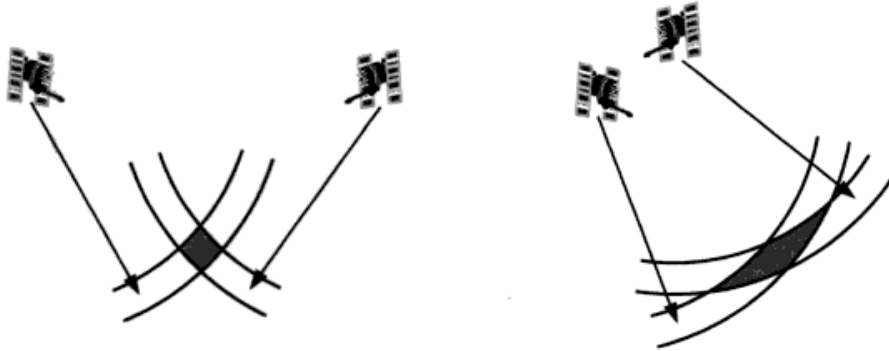


FIGURE 2.12 – Uncertainty areas of GPS localization under different satellites distributions

- **Multi-path error** : when a GPS receiver-equipped vehicle is driven along a street with tall buildings around, GPS signals might be reflected by objects around the antenna before being received by the GPS receiver, as shown in Fig. 2.13. The distance between the satellite and the receiver measured by the GPS receiver might be longer than the actual distance. If these contaminated distances are used for position estimation, the localization result would be erroneous.

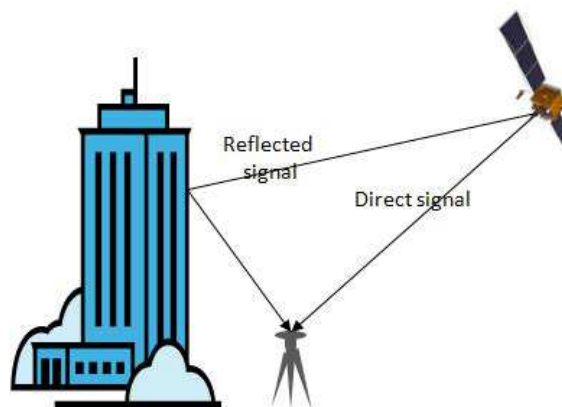


FIGURE 2.13 – Multi-path problem of GPS signal

2.3.3.2/ DGPS

Differential GPS (DGPS) can help to correct GPS errors during signal propagation process. The principle is to measure the range inaccuracy with a fixed GPS reference station (see in Fig. 2.14).

Since position of the reference station is known before experiments, we can easily deduce the difference between the measured satellite pseudo-ranges and the actual pseudo-ranges to the reference station. As the GPS satellites are far away from the earth, when a second GPS receiver can observe the same satellite like the reference station, we can assume that this receiver have the same pseudo-range error as the reference station.

The reference station then broadcasts the range correction information to all DGPS-equipped GPS receivers by radio signals. The DGPS receivers then correct their pseudo-

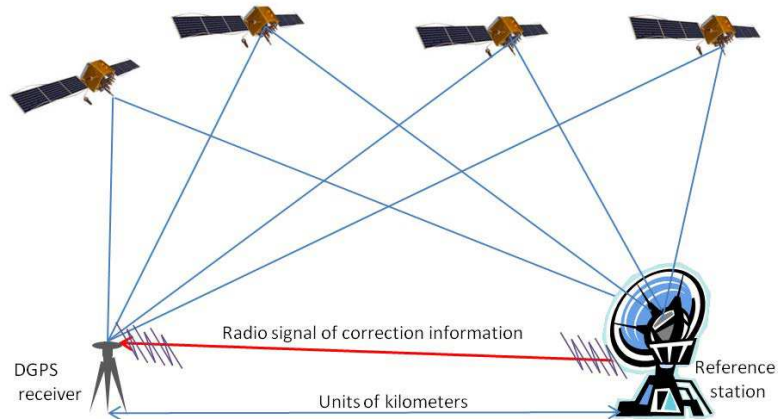


FIGURE 2.14 – Differential GPS : reference station is tens of kilometers away from a GPS receiver

ranges with these signal correction information in real-time or by post-processing. The accuracy of commercial DGPS receivers could be around 10 meters.

2.3.3.3/ RTK-GPS

Real time kinematic (RTK) satellite navigation is a GPS augmenting technique using both the DGPS reference station corrections and carrier phase measurements from GPS, GLONASS or Galileo systems. Compared with the DGPS station, a RTK-GPS reference base station can be installed by the user in a place much closer to the GPS rovers, mostly about hundreds of meters away (see in Fig. 2.15). The correction information can be transmitted from the base station to GPS rover by radio signals and used for position correction.

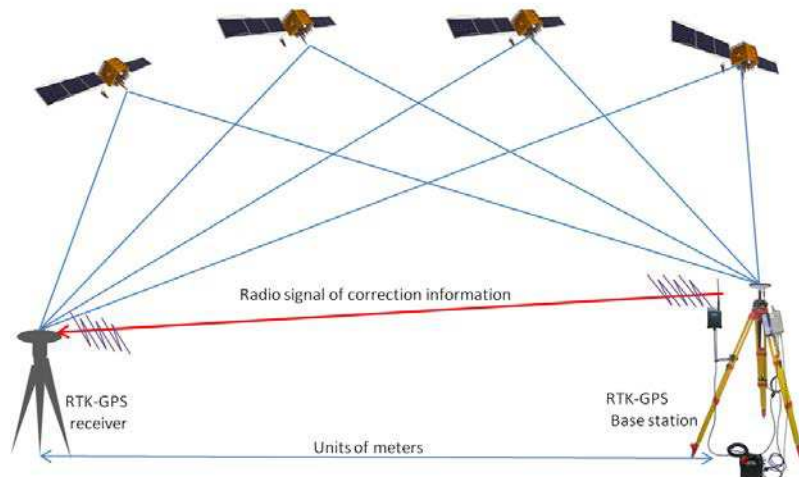


FIGURE 2.15 – Real time kinematic (RTK) satellite navigation : reference station is hundreds of meters away from a GPS receiver

Although RTK-GPS can achieve centimeter localization accuracy, it is challenged in some particular dense urban environments (e.g., urban canyons), as the satellite signals might be blocked or reflected by tall buildings around the receiver. The multi-path problem cannot be avoided by RTK-GPS.

2.3.3.4/ OTHER GNSS SYSTEMS

Besides GPS systems, there are some other global navigation satellite systems already launched in the world, including Global Orbiting Navigation Satellite System (**GLONASS**)⁸ of Russia, **Galileo** navigation system jointly funded by members of European Union⁹, and **Compass (Beidou)** navigation system¹⁰ of China. Japan's **Quasi-Zenith Satellite System**¹¹ and India's **IRNSS**¹² system are also being constructed.

2.3.4/ VISUAL LANDMARKS BASED LOCALIZATION SYSTEM

Landmarks are distinguishable features that can be repeatedly and reliably recognized from sensory data, like geometric shapes (e.g., points, lines, circles). The landmarks should be invariant to changes of orientation, scales and resolution. In outdoor environments, a visual landmarks-based positioning systems are generally composed of three parts [19] :

1. Onboard vision sensors (like camera, laser) for reliably recognizing 2D or 3D landmarks in data sequences when the vehicle moves [130]. Before using landmarks for navigation, the characteristics of different landmarks should be known. There are two kinds of landmarks : artificial landmarks and natural landmarks.

- i Localization with artificial landmarks : artificial landmarks are special objects placed at known positions in the environment. These landmarks are designed to be easily and repeatedly detected by visual sensors [5], e.g., known visual patterns or other distinctive characteristics (e.g., unique retro-reflective bar-codes). However, the artificial landmarks require modifications of the environment and landmarks maintenance in long term.
- ii Localization with natural landmarks : instead of modifying the environment, natural landmarks based localization method takes use of natural landmarks [154] existing in the environment. For example, corners, doors or walls in indoor environment. This method becomes more and more popular in outdoor environments since natural landmarks are abundant, such as roof of buildings, edges of windows, skylines, sidewalks, lines connecting two building facades, road signs, tree trunks, traffic signs [153], ground planes ; or invariant geometric features like reliable visual points, curvature extrema of laser range scan [99], etc. But the data association problem (matching) of natural landmarks are more complicated than the artificial landmarks.

8. <http://www.glonass-ianc.rsa.ru/en/GLONASS/>

9. <http://www.esa.int/esaNA/galileo.html>

10. <http://www.beidou.gov.cn>

11. http://qzss.jaxa.jp/index_e.html

12. <http://www.navipedia.net/index.php/IRNSS>

2. **Method for matching or associating the observed features from sensors with world model/map of position-known landmarks** [9]. Regarding sources of world model (or map), mobile robot/vehicle navigation can be divided into two approaches [42] :

- i the first approach is based on exteroceptive sensors with *a priori* world model, e.g., cartography map : 2D map, DEM (Digital elevation map), aerial image provided map, or CAD model (Computer Aided Design), etc.
- ii the second one is based on exteroceptive sensors and a visual feature model from the same type of sensor [153]. The feature model is typically constructed with the same type of sensor system during learning stage before the localization process, or during the localization stage, known as simultaneous localization and map building (SLAM).

For vision sensors, the photometric information of landmarks are usually used for landmarks matching ; for range sensors, the data association problem is more challenging, methods like nearest-neighbor filter, joint-compatibility filter have been proposed to reduce the matching ambiguities. 3. **Method for computing vehicle location and uncertainty from the corresponding features** to improve both longitudinal and lateral positioning precisions. Localization accuracy depends on the accuracy of landmarks detection, and accuracy of relative position of the landmarks to the vehicle.

The methods using exteroceptive sensors with 2D model, 3D model, or visual feature model are presented in the following sections.

2.3.4.1/ EXTEROCEPTIVE SENSORS WITH 2D/3D MODEL

A priori information of the environment can be given in the form of a two-dimensional map, three-dimensional model of the environment structure, or digital elevation map (DEM). The visual features are first extracted from the sensor observation, then matched with the model under a set of constraints to estimate the vehicle poses. The main problem of this method is that sensor observation and world model are in different forms since they are generally from different data sources. In order to use the map information for vehicle localization, visual observations from the on-board exteroceptive sensors should be features that can be matched with the pre-defined map or model.

Different features and map/model matching methods have been proposed in the literature as follows.

Feature - 2D map matching

DEM (Digital Elevation Models) have been used for aircraft localization and navigation for long time, as a complementary system to INS navigation for Unmanned Aerial Vehicles (UAV) when GPS was not yet available [31]. The digital elevation recovered from real-time data of LIDAR is matched with the reference DEM to determine the position of the sensor platform.

For ground vehicle localization, 2D digital maps can provide global environment information, such as trees and street lamps (point-style landmarks), shapes of urban roads (line-style landmarks), building footprints (polygon-style landmarks), as well as attributes of these objects (e.g., width of a road, height of a building). The map information can also contribute to constrain the vehicle positioning error with different features.

- **Point-style landmarks.** Point-style landmarks (such as trees and street lamps) in urban environments can be detected by perception sensors (e.g., laser, camera), then associated with independent objects in GIS map [141]. The measurement model can be the distance and orientation between the landmarks and vehicle.

- **Line-style landmarks.** In urban environments, there are a lot of line-style landmarks that can be detected by perception sensors, for example, lane marking on the ground. The current vehicle position on map can be determined by matching the detected vehicle lane marking features in front of the vehicle [159] with a geo-referenced road lane map. The geo-referenced road lane map can be built in advance with a geographically referenced aerial/satellite images [123], or represented by *a priori* GIS road network [127] [126], or a road ITN layer (Integrated Transportation Network) consisting of a set of connected arcs [7] (see Fig. 2.16). Mueller et al. [110] proposed to detect crossroad features from an equipped LIDAR system, then matched these observation with crossroad features extracted from a raw GIS road network map. The crossroad hypotheses from LIDAR data are then utilized within a particle filter to estimate robot's position within the extracted road network.

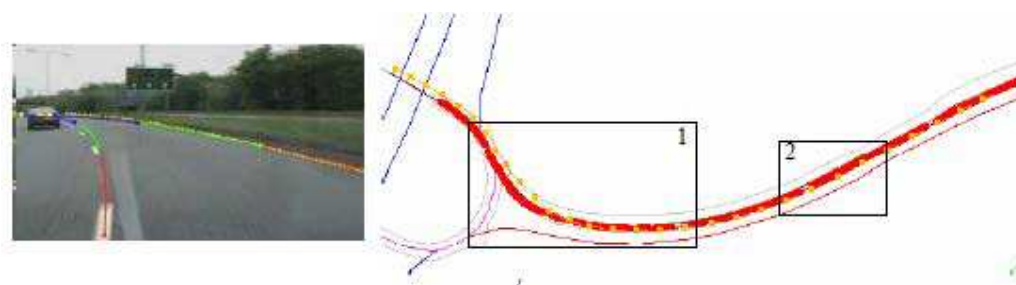


FIGURE 2.16 – Road geometry features respectively extracted from image and road network [7]

- **Vertical plane landmarks.** Vertical planes in the environment are also used as landmarks and compared with *a priori* building outline map. Bioret et al. [15] used on-board monocular camera to extract the vertical building facades, and applied a global pose searching strategy (see Fig. 2.17) for vehicle localization by comparing the building facade angle and width ratio with a building footprint map [16]. Cham et al. [34] also proposed to identify the vertical corner edges and neighboring plane normals of buildings from omnidirectional image, then associate the planes with a 2D building outline map without using any appearance data.

Feature - 3D GIS/CAD model matching

In addition to representing the environment with 2D digital maps, three-dimensional models like 3D CAD model or 3D virtual model can also be used. A 3D virtual city model (also called geographical 3D model, textured geo-referenced 3D database or 3D map) is a database of geographical and textured 3D data managed by a 3D Geographical Information System. 3D model of city scenes can be automatically generated from aerial images, 2D digital map, human surveys, or from camera data, laser data, etc., and tagged with global locations from high-precision GPS/INS systems.

Cappelle et al. [28][29][30] proposed to estimate the absolute position of a vehicle by

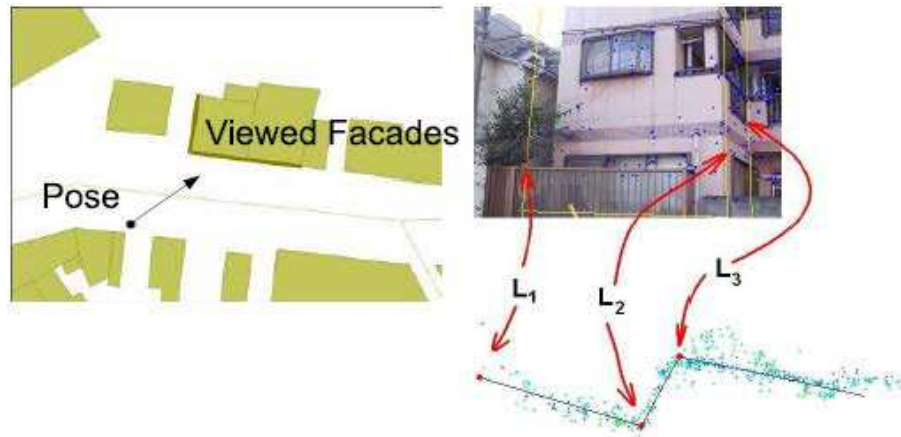


FIGURE 2.17 – Vertical planes extracted from an image with a building outline map [16]

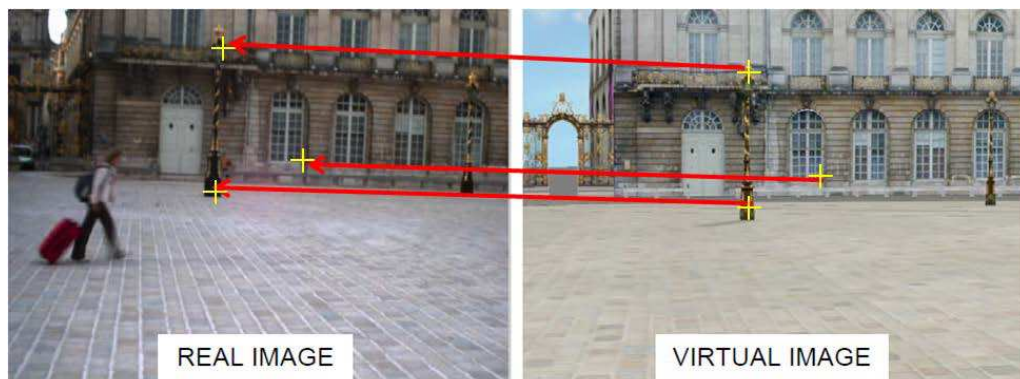


FIGURE 2.18 – Localization by a monocular camera and a 3D virtual model [28]

matching the acquired 2D image with a 3D virtual city model with Harris corners [69] (see Fig 2.18). Zhang et al. [171] proposed to extract SURF features on building facades from the acquired image and estimate the homography between the query view and the closest reference views in a 3D image database. P. Lothe et al. [94][95] proposed to use a 3D CAD model to align a deformed map (reconstructed 3D points and vehicle positions) reconstructed from the monocular SLAM.

2.3.4.2/ EXTEROCEPTIVE SENSORS WITH VISUAL FEATURE MODEL

In order to take use of the existing natural landmarks for localization, another approach is to use on-board exteroceptive sensor with a visual feature model, which is typically constructed with the same type of sensor system. Depending on whether the visual feature model is known or not before the localization process, there are two different methods : learning constructed map - localization method, and simultaneous localization and map building (SLAM) method.

Learning constructed map - Localization

This approach is composed of a learning stage and a localization stage. A predefined visual feature model is constructed by firstly manually driving the sensor-equipped vehicle (or mobile robot) along a desired path and recording a sequence of sensor data. Robust visual features are extracted from the data sequence to build an accurate feature model of the environment including landmarks and their corresponding sensor poses. After that, this model is used to locate and navigate the vehicle with sensor data in real-time.

Learning : such as the work of [36] and [136][137], a single forward-looking camera is used to capture image sequence, then features (e.g., Harris, SIFT, MSER) are extracted from the images and a feature model is built by structure from motion method. The associated camera poses and landmarks positions are stored in the model (as seen in Fig. 2.19 (left)). As discussed in section 2.2.3.2, the ambiguous scale factor problem during the monocular vision based reconstruction process can be solved by entering the path length of GPS trajectory as the last step. If more than one camera are provided, the scale and scene geometry of the environment can be recovered by triangulation of 3D points with stereovision methods [85][117][142] (see in Fig. 2.19(right)).

The reconstructed landmarks and camera poses can be then refined by hierarchical local [109] or global [135] bundle adjustment using Levenberg-Marquardt algorithm. The feature model can also be built by laser data with extracted landmarks like curvature extrema in a laser scan [99], sidewalk landmarks from a vertical laser scan [74] [75] [76], etc.

Localization : during the localization process, a coarse localization is firstly applied with inertial sensors and GPS data (if available) ; meanwhile, the vehicle observes its surroundings with on-board sensors and detects features in the captured data (one image, one image pair, or a laser scan) ; then, a set of landmarks which should be visible in the sensor's FOV are extracted from the learned map with the coarse pose, and the observed features are associated with the extracted landmarks from map ; finally, the vehicle pose can be refined with enough corresponding features by nonlinear optimization [136] [117]. During the localization step, if a coarse vehicle pose can not be available from other sensors, searching for the corresponding key frames in the global model might be

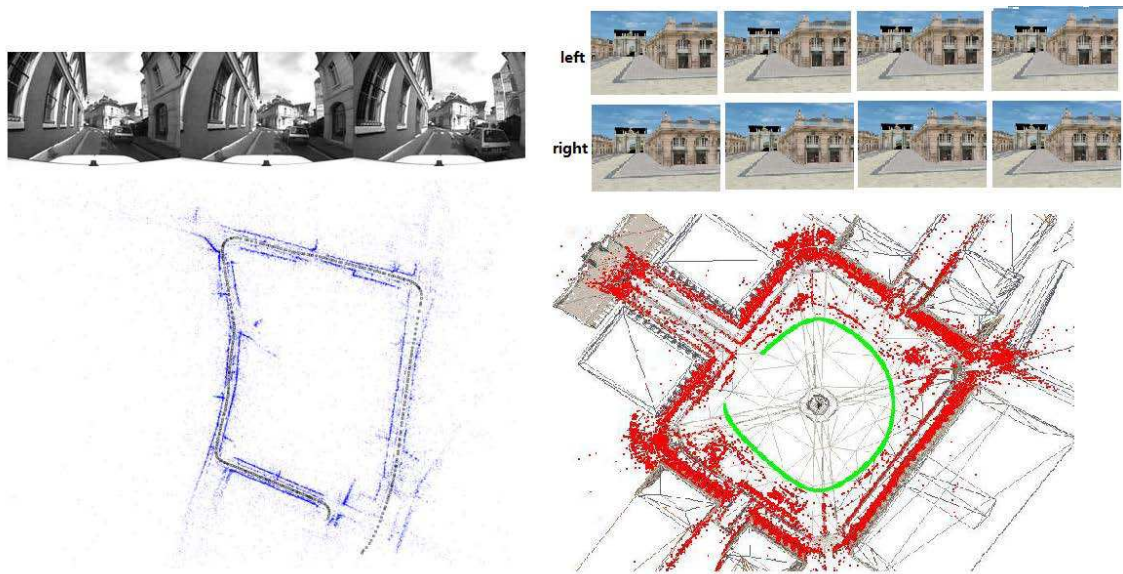


FIGURE 2.19 – Left : reconstructed model from a monocular city sequence [137]; right : reconstructed model from a simulated stereo image sequence [116]

time-consuming.

Simultaneous localization and mapping

Besides the “learning - localization” pipeline, another approach is the well known Simultaneous Localization and Mapping method (SLAM). It exploits duality between the localization and the mapping, and addresses both issues in the same process in order to build a new map or to update an existing map and localize the vehicle simultaneously [8][132].

- Classic SLAM is based on a vehicle motion model and probabilistic method with filters : the vehicle motion is firstly predicted by vehicle dynamic motion model, then the state vector (composed of vehicle pose and landmarks positions) can be updated by the observed natural or man-made landmarks [156] with range sensors (section 2.2.4) like laser range finder, sonars, etc.
- Vision based SLAM estimates the vehicle trajectory by matching features between a live map of the scene structure and the current image [60] [91] [167] using monocular, stereoscopic or trinocular camera systems [143].

2.3.5/ CELLULAR LOCALIZATION

Instead of using GNSS satellites, cellular localization [26] uses existing cellular communication infrastructure (distributed cellular base stations) to estimate the real-time position of a wireless mobile user.

When a mobile phone moves around in an area, the base stations that serve for this area can provide communications to the mobile phone. The distance between the base stations and the user can be measured by several characteristics of the signal : 1) Time of Arrival (TOA, or TOF - Time of Fly) of the signal traveled between the emission source and the destination ; 2) Time Difference of Arrival (TDOA) that uses synchronization of cellular

network of users to compute the time difference of arrivals between different users ; 3) Angle of Arrival (AOA) based on the angle of signal from cellular stations to the mobile user ; and 4) Received Signal Strength (RSS) based on mathematical models, which describe the path loss of signals as a function of the traveling distance [48].

When enough distances or angles to the base stations are known, the location of user can be calculated by trilateration or triangulation as presented in section 2.3.1 and section 2.3.2. In most cases, the average localization accuracy of cellular localization is between 90m and 250m [149]. Cellular localization is less precise than GPS, but the signals from cellular infrastructure are more available in urban environments than the signals from satellites, especially for indoor environments like parking lots or tunnels.

2.3.6/ AD-HOC LOCALIZATION (OR VANET)

Ad-hoc (or VANET) based localization method [139] was introduced for localizing vehicles or enhancing the localization estimation, especially for vehicles without GPS receivers, or vehicles whose GPS receivers cannot receive GPS signals in a specific location (e.g., urban canyon). Communication between the position unknown vehicle and other mobile nodes (position known) can be used to compute its relative locations with other vehicles and then calculate its global position [47].

2.4/ MULTI-SENSOR FUSION BASED LOCALIZATION

After respectively presenting relative and absolute localization methods in the previous sections, their characteristics (accuracy in long and short terms, localization robustness and frequency) are compared in the following Tab 2.4.

Methods	Outputs	Advantage	Shortcomings
Relative method	Relative motion : translation and rotation ($\Delta x, \Delta y, \Delta \theta$)	High precision in short term, high sampling frequency	Unbounded error due to error accumulation
Absolute Method	Global pose : (x, y, θ)	Accurate in long term	GPS : Multi-path, visibility of satellites, sampling frequency is low ; World model method : uncertainty in construction and landmarks association process

TABLE 2.4 – Comparison of relative and absolute localization methods

Relative localization methods can provide good accuracy in short term. However, as there is no global optimization for dead-reckoning methods, the predicted vehicle trajectory might drift in long term due to errors accumulation from point to point. Though error in global localization method does not accumulate like in relative methods, sampling rate of global methods is often slower than relative methods. For GNSS based approach, only if at least four satellites with good distribution geometry are visible in the sky by the GPS

receiver, an accurate positioning result might be provided. In addition to this, satellite signals might be blocked or reflected by tall buildings in urban environments. For world model based method, the localization accuracy is influenced by the noises of sensor observation and environment model.

Considering the advantages and shortcomings of different methods, information from multiple data sources can be combined together to take advantage of their redundancy and complementarity, to provide accurate and robust vehicle localization results. Relative localization methods can be used as backup for GPS-denied situations, such as in dense urban areas. Meanwhile, world map/model can be used as another global reference to adjust the vehicle pose drift aroused by error accumulation.

2.4.1/ LOOSELY-COUPLED AND TIGHTLY-COUPLED FUSION

For integrating information from multiple sensors, there are essentially two fusion approaches : loosely coupled and tightly coupled approaches.

- **Loosely-coupled multi-sensor fusion method** uses a decentralized filter with several sub-filters to independently process the information in different sub-systems of the localization system [35] [78]. Independent localization solutions from all the sub-systems are synchronously combined in an overall filter to provide an integrated navigation solution. If one of the sensors fails, a solution can still be given by other sensors. Loosely-coupled fusion methods have been widely used for its simplicity of integration. For example, Sukkarieh and al. [150] combine GPS position with INS estimation for localization, Grimes and al. [64] augment wheel odometry with visual orientation to yield better localization accuracy, Ignacio et al. [73] integrate visual odometry, map-matching and GPS methods for navigation assistance.

- **Tightly-coupled multi-sensor fusion method** uses a single main filter to process output of all sensors. Without using the processed solution of each sensor system, the raw sensor observations are used as measurements. For example, in a tightly-coupled GPS/INS integration system, GPS pseudo-ranges are directly fused with INS readings [164] or odometer readings [22] : if the available satellite signals are not sufficient for calculating a GPS position (< 4), the limited GPS pseudo-range and carrier phase measurements can still be possible to be fused with INS readouts or odometer/map information.

2.4.2/ DIFFERENT FUSION STRATEGIES FOR VEHICLE LOCALIZATION

By using different sensor combination or different coupling methods, several fusion strategies used for vehicle localization in the literature are summarized as follows :

- **1. GPS + Dead-reckoning solution (DR)** : a lot of solutions have been proposed to augment GPS localization with dead-reckoning solutions. Dead-reckoning sensors could be used to compensate GPS outages in dense urban environments by continuously estimating relative movements, then GPS positions are periodically used to reduce the accumulated error of dead-reckoning sensors when a GPS position is available, e.g. GPS/INS [150], GPS/INS/encoder odometry [113], GPS/visual odometry [163].

- **2. GPS/DR + Landmarks localization** : in order to overcome the drift of GPS/DR

method if GPS receivers encounter long term outages, another global information source is needed for vehicle localization :

- **GPS/DR + Digital road map based map matching** : a road network map captures the road topology with road links and road nodes. Map-matching method is to find a correspondence between a vehicle/personal trajectory position (e.g., from a GPS receiver or GPS/DR system) and a path in the road network (provided by a GIS map) for navigation assistance [63] [77]. There are a lot of research work concerning the map-matching ambiguity problem in changellend conditions like road intersections [165] [125]. Observations from the road map matching method can be used as a measurement of the vehicle pose, and integrated with GPS/DR method to restrict the vehicle pose on road [21] [128] [157].
- **GPS/DR + Vision + Landmarks model or world model** : landmarks based localization methods (in section 2.3.4) are on the basis of a coarse vehicle pose estimation from GPS/DR system, the landmarks stored in a model, and several onboard perception sensors [29][76]. Regions of Interest (ROI) are extracted from the world model with the coarse vehicle pose ; features detected in the current sensor observation are matched with landmarks in the ROI of the world model ; after that, the vehicle pose can be refined by the corresponding landmarks. Or, like in the work of Lanerit et al. [89], GPS/DR system can also be coupled with a vision algorithm (lane detection) and a precise numerical map for vehicle pose correction.

• **3. GPS/DR + Camera based satellite visibility detection** : as seen in Fig. 2.13, the accuracy of GNSS localization might be degraded by multi-path problem around the receiver antenna. Several methods have been proposed to detect the multi-path problem by placing a camera near the GPS antenna and detecting the invisible satellites in the camera's FOV. Then, the signals from satellites which are "seen" by the GPS receiver but without line of sight (LOS) are excluded, only the raw pseudo-ranges of satellites with LOS are used to estimate the vehicle position. For example, Meguro et al. [103] used an omnidirectional infrared (IR) camera on the vehicle roof to detect the borderline between the sky and surrounding buildings. For the same purpose, Attia et al. [4] used an upward fish-eye camera to detect the visible sky by autonomous image segmentation and classification. Instead of using real camera, Peyraud et al. [121] proposed to use a 3D urban model to predict the satellite LOS visibility in urban contexts.

2.5/ CONCLUSION

In this chapter, different localization approaches are presented and discussed, including relative localization methods like wheel encoder odometry, inertial navigation method, visual odometry and range scan alignment ; and absolute localization methods like GNSS, cellular localization, ad-hoc localization and world map/model based localization. Considering advantages and shortcomings of the different methods, a set of sensor sources are usually proposed to be combined together to provide more accurate and more robust localization results. In our work, stereovision based visual odometry or LRF based scan alignment method is used to provide the odometry information of vehicle (chapter 3). Then, the odometry informations are integrated with GPS measurements together with a sensor coherence validation step (chapter 4). In order to compensate for the accumulated localization error of DR-only method when GPS encounters long term outages, a 2D GIS road network map is used to provide a coarse pose estimation, and a GIS building

map layer is associated with the building facades detected by two onboard LRF systems (horizontal and vertical LRFs) to correct the vehicle pose error (chapter 5).

STEREOVISION/LASER RANGE FINDER BASED VEHICLE LOCALIZATION

3.1/ OVERVIEW

As we presented in section 2.2.3, onboard vision systems can be used to estimate motion parameters of a moving platform, especially in outdoor urban environments rich in visual landmarks.

In this chapter, stereovision based visual odometry method is presented. Compared with wheel encoder based odometry, visual odometry (VO) can provide the vehicle motion with six degree of freedom ($X, Y, Z, yaw, pitch, roll$). This estimation is not influenced by wheel deformation or slippage caused by bad soil condition, but more related to the environment around the camera and the illumination conditions. VO is then completed by a horizontal LRF sensor based scan alignment method. Different laser range finder based motion estimation methods are presented and compared in order to obtain an accurate and robust LRF based vehicle motion estimation. An outlier-rejection ICP (OR-ICP) and two image-aided ICP methods (ICP-Color, ICP-Descriptor) are proposed to reduce the matching ambiguities of scan alignment in outdoor environment. For the purpose to be integrated with other sensor measurements, the uncertainty of the two relative visual localization methods (stereovision based odometry, and LRF based scan alignment) are discussed. These methods are tested with real data and evaluated by using RTK-GPS data as ground truth.

This chapter is organized as follows : the procedure of stereovision based visual odometry is detailed in section 3.2 ; then, laser range finder based vehicle ego-motion estimation method is described in section 3.3 ; finally, some experimental results obtained with real data acquired by our experimental vehicle are given in section 3.4.

3.2/ VISUAL ODOMETRY BASED VEHICLE MOTION ESTIMATION

The word “visual odometry” was firstly used by Nistér [115] in 2004. Camera based visual odometry can estimate camera motion by matching (or tracking) corresponding image points between two consecutive frames. This approach has been studied and used in a lot of indoor and outdoor localization projects [41] [72] [87] [133] [167], even on Mars [37]. With image sequences provided by the camera system (such as monocular camera or

stereoscopic system), corresponding image pixels (dense method for all image pixels, or sparse method for only image features) are searched between consecutive image frames or between corresponding left and right images for stereoscopic system. Then, these corresponding pixels are used to estimate the camera motion (see section 2.2.3.2).

In this section, the basic concepts of stereoscopic camera system are introduced in section 3.2.1 ; then, the process of stereovision based visual odometry is presented in section 3.2.2 ; finally, error modeling of visual odometry based pose estimation is discussed in section 3.2.3.

3.2.1/ CAMERA SYSTEM MODELING AND CALIBRATION

A preliminary and prerequisite step of visual odometry approach is to calibrate the camera system to find its intrinsic and extrinsic parameters.

3.2.1.1/ PERSPECTIVE CAMERA MODELING AND CALIBRATION

Pinhole model

Pinhole model is the mostly used model to represent a camera projection process.

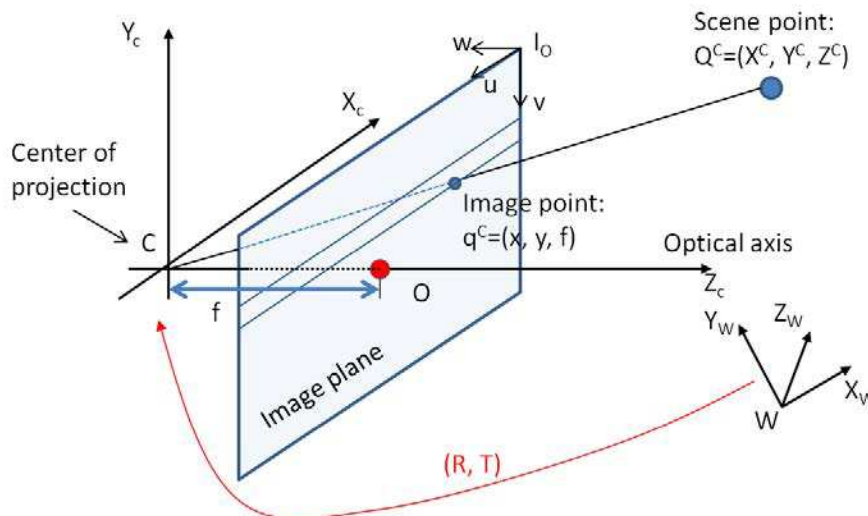


FIGURE 3.1 – A point Q with coordinates $Q^C = (X^C, Y^C, Z^C)$ in the camera frame is projected onto image plane by the ray passing through the projection center C ; the resulting point is an image point q with coordinates $q^c = (x, y, f)$

Three coordinate systems are considered (Fig. 3.1) :

- World system : world reference system is denoted as $R_W\{W, X_W, Y_W, Z_W\}$, where W is the system origin and X_W, Y_W, Z_W are the three orthogonal axes. Coordinates of a point Q in this system are written as : $Q^W = (X^W, Y^W, Z^W)$.
- Camera frame : the coordinate system attached to the camera is denoted as $R_C\{C, X_C, Y_C, Z_C\}$. The origin of the camera frame is camera center C , also called projection center. The $X_C Y_C$ plan is parallel to the image plane. The axis Z_C is pointing to the viewing direction, called optical axis. Coordinates of a point Q in this system is written as : $Q^C = (X^C, Y^C, Z^C)$.

- Image frame : image frame is denoted as : $R_I\{I_o, v, u, w\}$. The image plane is on $Z^C = f$ in the camera frame, where f is the focal length of the camera. The origin point I_o of the image frame is the upper left corner of the image. O is the intersection of the optical axis and the image plane, called principal point. In this system, coordinates of a point q are expressed by pixels, and written as : $q^I = (v, u, w)$.

Extrinsic and intrinsic parameters

A camera has intrinsic and extrinsic parameters [70]. The intrinsic parameters are related to its intrinsic characteristics, including focal length, position of the principle point on image plane, image pixel size, scaling factors of row and column pixels, skew factor, and lens distortion. The extrinsic parameters are related to its position and orientation with respect to a fixed world system. The process to map a point $Q(X^W, Y^W, Z^W)$ in the world system to a point q with coordinates $(v, u, 0)$ on the image plane is called a projective transform.

• **1) Extrinsic parameters.** With the extrinsic parameters of a camera, the coordinates of Q in the world frame $Q^W = (X^W, Y^W, Z^W)$ can be transformed to coordinates in the camera frame, as $Q^C = (X^C, Y^C, Z^C)$. The extrinsic parameters are represented by a translation vector T and a rotation matrix R (see in Fig. 3.1).

- **Rotation matrix** : orientation matrix of a camera in the world system is related to its rotation from the world frame to the camera frame. The rotation matrix R in 3-dimensional space can be decomposed into three rotation matrices : R_X with angle α around X_C axis, R_Y with angle β around Y_C axis, and R_Z with angle γ around Z_C axis. They are respectively written as :

$$R_X = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \quad (3.1)$$

$$R_Y = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \quad (3.2)$$

$$R_Z = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.3)$$

Then, the full rotation matrix R is given by the product of these three matrices, as :

$$R = R_Z R_Y R_X = \begin{bmatrix} \cos\beta\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \cos\beta\sin\gamma & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma \\ -\sin\beta & \sin\alpha\cos\beta & \cos\alpha\cos\beta \end{bmatrix} \quad (3.4)$$

- **Translation vector** : translation vector $T = (t_X, t_Y, t_Z)$ describes the position of camera center C in the world frame.

- **Full extrinsic model** : with the rotation matrix R and translation vector T , the world coordinates Q^W of point Q can be transformed to coordinates Q^C in the camera frame through :

$$Q^C = RQ^W + T \quad (3.5)$$

As the homogeneous coordinates of a n -d point in projective space can be expressed by a $(n + 1)$ dimensions vector, the homogeneous coordinates of $Q^C(X^C, Y^C, Z^C)$ are written as :

$$\begin{bmatrix} X^C \\ Y^C \\ Z^C \\ 1 \end{bmatrix} = M \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix} \quad (3.6)$$

where M is the camera extrinsic model containing the extrinsic parameters of the camera :

$$M = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \quad (3.7)$$

• **2) Intrinsic parameters.** In central projection, the point Q with coordinates $Q^C = (X^C, Y^C, Z^C)$ in the camera frame can be projected onto the image plane by a ray passing through both the point Q and the projection center C (Fig. 3.1). The resulting point on image is q with coordinates $q^C = (x^C, y^C, z^C)$ in the camera frame, as :

$$\begin{cases} x^C = f \frac{X^C}{Z^C} \\ y^C = f \frac{Y^C}{Z^C} \\ z^C = f \end{cases} \quad (3.8)$$

All the image points in the camera frame can be represented by 4–dimensional homogeneous coordinates and a constant s , as :

$$\begin{bmatrix} sx^C \\ sy^C \\ sz^C \\ s \end{bmatrix} = P \begin{bmatrix} X^C \\ Y^C \\ Z^C \\ 1 \end{bmatrix} \quad (3.9)$$

where P is the projection matrix :

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \quad (3.10)$$

- **Transformation from camera frame to image frame :** image points are measured by pixels in the image frame. Let $q^I = (u, v, w)$ the pixel coordinates of a point q in image frame. The transformation of its coordinates from camera frame to image frame is based on image principal point $O = (u_0, v_0)$ (pixels), horizontal scaling factor k_u and vertical scaling factor k_v , written as :

$$\begin{cases} u = -k_u x^C + u_0 \\ v = -k_v y^C + v_0 \\ w = 0 \end{cases} \quad (3.11)$$

The homogeneous coordinates of q in the image frame are represented by a 3-dimensional vector :

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K_0 \begin{bmatrix} x^C \\ y^C \\ z^C \\ 1 \end{bmatrix} \quad (3.12)$$

where :

$$K_0 = \begin{bmatrix} -k_u & 0 & 0 & u_0 \\ 0 & -k_v & 0 & v_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.13)$$

- **Full intrinsic model** : with Eq. 3.9 to Eq. 3.13, the full intrinsic model is used to transform the point coordinates Q^C in the camera frame to coordinates q^I in the image frame, written as :

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = K_0 P \begin{bmatrix} X^C \\ Y^C \\ Z^C \\ 1 \end{bmatrix} = \begin{bmatrix} -k_u & 0 & u_0/f & 0 \\ 0 & -k_v & v_0/f & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \begin{bmatrix} X^C \\ Y^C \\ Z^C \\ 1 \end{bmatrix} = K \begin{bmatrix} X^C \\ Y^C \\ Z^C \\ 1 \end{bmatrix} \quad (3.14)$$

where $K = K_0 P$ is the camera intrinsic model containing the intrinsic parameters of the camera. Since the point q is in homogeneous coordinates, we need to divide q by s to recover its image coordinates, as : ($u = su/s, v = sv/s$).

• **3) Full camera model.** With the extrinsic model M of the camera (Eq 3.7), the point coordinates Q^W in the world reference system can be transformed to point coordinates Q^C in the camera frame ; then, with the intrinsic model K of the camera (Eq 3.14), Q^C can be transformed to coordinates q^I in the image frame. The full camera model can then be written as :

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = KM \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix} = \begin{bmatrix} -k_u & 0 & u_0/f & 0 \\ 0 & -k_v & v_0/f & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix} \quad (3.15)$$

Calibration of a camera

Calibrating a camera is to know the value of its intrinsic/extrinsic parameters and distortion coefficients, especially for the intrinsic parameters since they are constant at different camera poses. Precision of the calibration process is important because the intrinsic parameters will be used during the visual odometry process and might affect the robustness of the whole procedure.

The principle of camera calibration is to target the camera on a known structure with a set of identifiable features. By viewing this structure from different poses, it is possible to compute the intrinsic parameters of the camera and its relative locations and orientations with respect to the calibration structure.

We use the planar calibration method proposed by Zhang [172] to calibrate our camera system. The calibration target used is a plane with printed chessboard pattern (Fig. 3.2). There are 23×16 squares ($30mm \times 30mm$) on the chessboard. The camera is calibrated through the following steps :

- i) At first, the chessboard is hold at different positions and orientations in the field of view of the camera, to provide a set of chessboard images ;
- ii) Then, internal corners of the chessboard are extracted from each image ;
- iii) Since the actual size of chessboard squares is already known, the third step is to calculate the intrinsic parameters of the camera, and the extrinsic parameters R and T with respect to each chessboard plane ;



FIGURE 3.2 – Chessboard images used for camera calibration

- iv) Finally, the initial estimated camera parameters are optimized with the global Levenberg - Marquardt optimization algorithm by minimizing the re-projection error.

There are many toolboxes for camera calibration, we have tested Matlab Toolbox¹ developed by Jean-Yves Bouguet [25], and open source library OpenCV². Both the two toolboxes implement the calibration method proposed by Zhang [172]. Bouguet's toolbox is used in our work to estimate the focal length on u and v directions in pixels, $f_u = k_u f$, $f_v = k_v f$, the coordinates of the principle point O , the angle of pixel axes, and a 5-dimensional vector containing the radial and tangential distortion parameters of camera [27].

3.2.1.2/ STEREOSCOPIC SYSTEM MODELING AND CALIBRATION

Geometric model of a stereoscopic system

Just as human binocular vision system, a stereoscopic system can perceive the 3D structure of an object in the environment. A binocular stereoscopic system is composed of two digital cameras, respectively with projection centers C_l and C_r as shown in Fig. 3.3. The stereoscopic system permits to simultaneously obtain two projections of the same scene from two different points of view [56].

Given a 3D scene point Q with coordinate $Q^W = \{X^W, Y^W, Z^W\}$ in the world coordinate system, it can be respectively projected onto the left and right image planes as points $q_l(u_l, v_l)$ and $q_r(u_r, v_r)$. q_l and q_r are defined from their own image coordinate systems associated with the left and right images.

- Epipolar line and epipole. As seen in Fig. 3.3, the projection of the right projection center C_r (resp. C_l) on the left (resp. right) image plane Π_l (resp. Π_r) is called the left (resp. right) epipole e_l (resp. e_r). The plane through the two epipoles e_l , e_r and 3D point Q is called an epipolar plane. Lines $q_l e_l$ and $q_r e_r$ (between the epipoles and the image points) are called epipolar lines. Given a point q_l (resp. q_r) in one image, its corresponding point viewed in the other image must lie on its corresponding epipolar line $q_r e_r$ (resp. $q_l e_l$). This constraint is called the epipolar geometry constraint.

1. <http://www.vision.caltech.edu/bouguetj/calib.doc>

2. http://opencv.itseez.com/trunk/doc/tutorials/calib3d/camera_calibration/camera_calibration.html

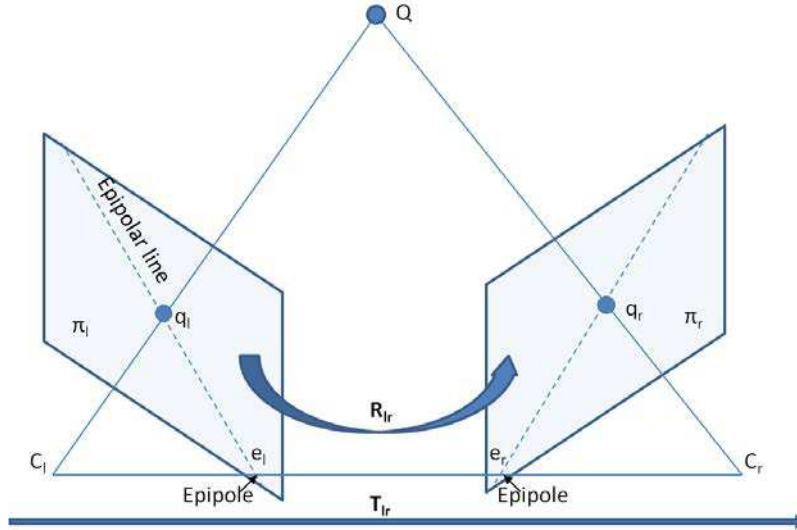


FIGURE 3.3 – Epipolar geometry of a stereoscopic system

- **Essential matrix.** Essential matrix E describes the location of a camera with respect to the other camera in the system attached to the first camera. It contains the information which relates the coordinates of two projections q_l and q_r of the point Q on the left and right image planes. This relation is written as :

$$(q_r^{C_r})^T E q_l^{C_l} = 0 \quad (3.16)$$

where $q_l^{C_l}(x_l^{C_l}, y_l^{C_l}, z_l^{C_l})$ and $q_r^{C_r}(x_r^{C_r}, y_r^{C_r}, z_r^{C_r})$ are respectively the coordinates of two projections q_l and q_r in the left and right camera frames.

- **Fundamental matrix.** Just as the essential matrix, fundamental matrix F also builds links between the two cameras. F is defined in terms of pixel coordinates with the intrinsic information of both cameras, while essential matrix E is defined in terms of camera coordinates.

The coordinates of a point in image frame can be obtained with its coordinates in the camera frame and the camera intrinsic parameters, $q_l^I = K_l q_l^{C_l}$, $q_r^I = K_r q_r^{C_r}$, where q_l^I and q_r^I are respectively the coordinates of image points q_l and q_r in the left and right image frames ; K_l , K_r are respectively the intrinsic models of the left and right cameras.

Replacing $q_r^{C_r} = K_r^{-1} q_r^I$ and $q_l^{C_l} = K_l^{-1} q_l^I$, Eq.3.16 can then be written as :

$$(q_r^I)^T (K_r^{-1})^T E K_l^{-1} (q_l^I) = 0 \quad (3.17)$$

Note :

$$F = (K_r^{-1})^T E K_l^{-1} \quad (3.18)$$

we can obtain :

$$(q_r^I)^T F q_l^I = 0 \quad (3.19)$$

In Eq. 3.19, F directly establishes a mapping from pixels on an image to the corresponding epipolar lines on the other image, without any prior knowledge of the geometric relation between the two cameras.

Remarks : the coordinates of image points q_l and q_r respectively in the left and right images frames are abbreviated as $\{q_l(u_l, v_l), q_r(u_r, v_r)\}$ in the following texts.

Calibration of a stereoscopic system

Stereo calibration is the process of recovering the geometric relation (relative position and orientation) between two cameras in a stereoscopic system [70]. As seen in Fig. 3.3, for a point Q in 3D space, its two coordinate vectors Q^{C_l} and Q^{C_r} respectively observed by the left and right cameras are related to each other by a rigid transformation : $Q^{C_r} = R_{lr}Q^{C_l} + T_{lr}$, where R_{lr} is a 3×3 rotation matrix, T_{lr} is a 3×1 translation vector. R_{lr} and T_{lr} describe the relative location of the left camera with respect to the right one.

-Implementation. Several methods [59] [70] have been proposed to find the parameters R_{lr} and T_{lr} of a stereoscopic system. Hartley [70] proposed to use the fundamental matrix with uncalibrated cameras : corresponding image points $\{q_l(u_l, v_l), q_r(u_r, v_r)\}$ are used to estimate the fundamental matrix F [93][114] through Eq.3.19, then recover the rotation matrix R_{lr} and translation vector T_{lr} by factorization of the fundamental matrix with SVD (Singular Value Decomposition). Bouguet [25] proposed to separately calibrate the two cameras at first, then estimate the rotation and translation parameters by minimizing the reprojection errors of all the corresponding corners. The second method is used in our work to calibrate the stereoscopic system with the toolbox developed by Bouguet.

-Stereo rectification. Rectification is a process to correct an image pair (Fig. 3.3) such that the images taken by the two cameras are row-aligned, as shown in Fig. 3.4. In such configuration, the optical axes of the two cameras are parallel and the baseline $B(C_l C_r)$ is perpendicular to the optical axes.

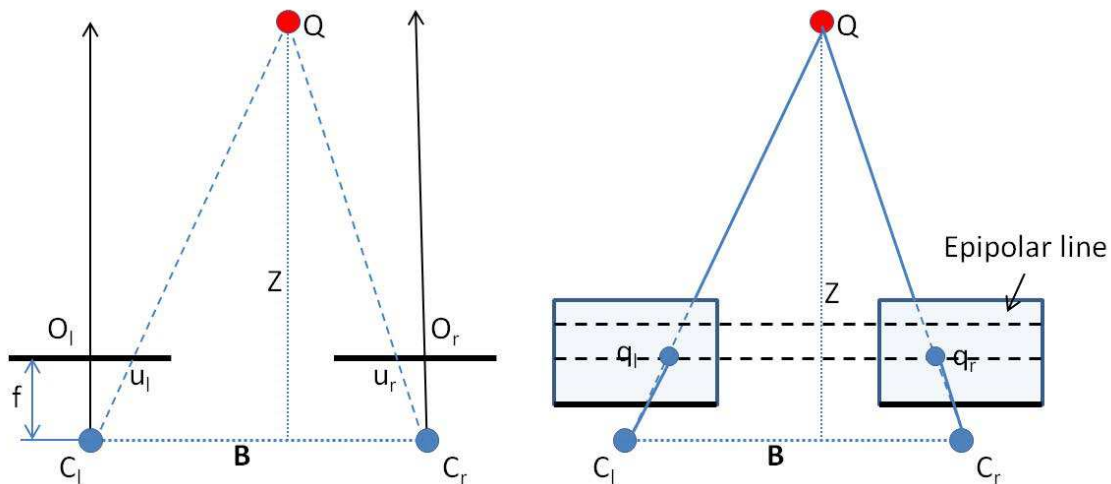


FIGURE 3.4 – Ideal stereoscopic configuration after stereo rectification

After rectification, all the epipolar lines are collinear and parallel to the horizontal image lines. We can then search the correspondence of an image point on the same row of another image. As shown in Fig. 3.4, $d = u_l - u_r$ is the horizontal-disparity between the two corresponding points. In such a configuration, the depth Z of point Q can be derived by the disparity information with $Z = \frac{fB}{d}$ (see section 3.2.2.3).

We use Bouguet's algorithm to rectify the stereo images with the rotation and translation parameters between the two calibrated cameras. After image rectification, the stereoscopic system is calibrated again to obtain new parameters of the two cameras and new relationship between them. These parameters are used for stereovision based vi-

sual odometry in the next section.

3.2.2/ STEREOVISION BASED VISUAL ODOMETRY

Since the baseline between the left and right cameras can be known by calibration, the scale of Euclidean reconstruction can be directly provided. Therefore, stereovision based visual odometry method is adopted in our system.

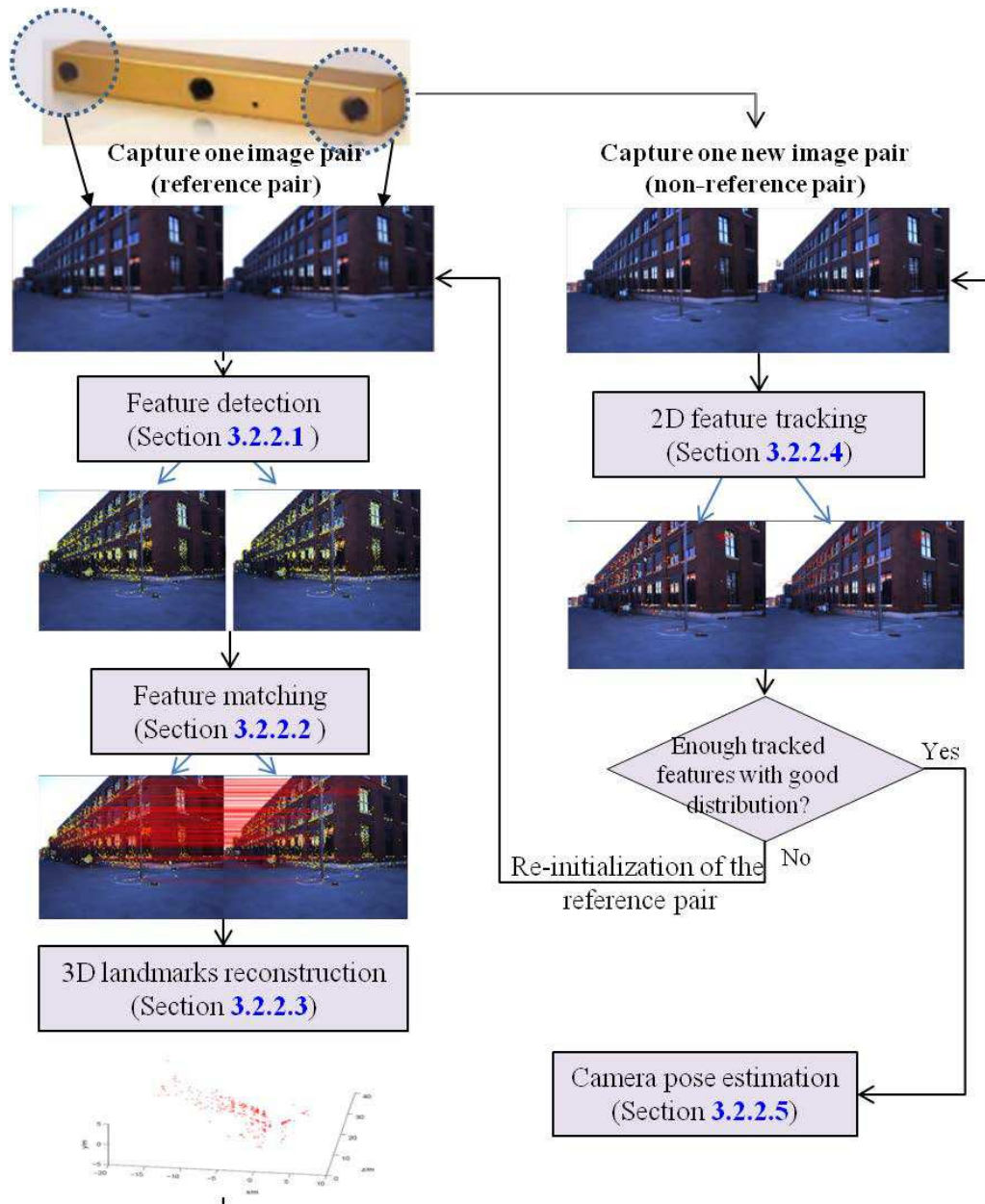


FIGURE 3.5 – Working flow of stereovision based visual odometry method

As shown in Fig.3.5, when an image pair is acquired :

1) If it is the first image pair of a reference, features (e.g., Harris, SURF, CenSurE) are extracted from the image pair (section 3.2.2.1), and corresponding features are found

between the left and right images (section 3.2.2.2) ; then, the corresponding features are reconstructed into 3D space by triangulation (section 3.2.2.3) ;

2) If it is not a reference image pair, the detected features in the reference image pair are tracked from frame to frame (section 3.2.2.4) till the reference stereo pair updates (section 3.2.2.4).

3) Then, the camera motion of current image pair can be estimated with RANSAC by incorporating several 2D and 3D outlier rejection strategies (section 3.2.2.5).

3.2.2.1/ FEATURE EXTRACTION

In order to track image features in multiple views, the extracted features and descriptors have to be robust and distinctive under various conditions (e.g., different illumination conditions, positions, angles of view and image scales). For real-time application, feature tracking also requires that the features and descriptors could be quickly extracted.

A lot of feature types have been proposed in the computer vision community, such as point, line, or region features. For example, the widely used Harris corners [69] with good detection rate. But as Harris corners rely on image gradient information, it is sensitive to the change of image scales and viewpoints. Several other detectors have also been proposed, e.g., Shi and Tomasi corners [144], Scale Invariant Feature Transform (SIFT) [96], FAST feature [134], MSERs feature (Maximally Stable Extremal Regions) [44], SURF (Speeded Up Robust Features) [12] and CenSurE (Center Surround Extremas) [1].

Characteristics of these features are compared in Tab 3.1 based on the work of Fraundorfer et al. [58] and Mikolajczyk et al. [105].

Feature \ Criteria	C1	C2	C3	C4	C5	C6	C7	C8	C9
Harris	×		×			***	***	**	**
Shi-Tomasi	×		×			***	***	**	**
SIFT		×	×	×	×	***	**	***	*
FAST	×		×	×		**	**	**	****
MSERs		×	×	×	×	***	**	***	*
SURF		×	×	×	×	***	**	**	**
CenSurE		×	×	×	×	***	**	***	***

TABLE 3.1 – Comparison of feature detectors (based on [58]) for different criteria. **C1** : corner detector ; **C2** : blob detector ; **C3** : rotation invariant ; **C4** : scale invariant ; **C5** : affine invariant ; **C6** : repeatability ; **C7** : localization accuracy ; **C8** : robustness ; **C9** : efficiency

We have tested Harris, Shi-Tomasi, SIFT, SURF and CenSurE features listed in Tab 3.1 with a set of images to choose a feature with better repeatability and robustness for our work. Results show that most SURF features can be repeatedly detected and matched in multiple images for motion estimation. But it should mention that this experiment was implemented in indoor environments with manually designed camera displacements. If we want to compare the robustness, efficiency and accuracy of different features for localization in long sequences, we need to test these features with more image sequences (under

different illumination and weather conditions, and different environment structures, obstacles, etc.), then evaluate the results of estimated camera trajectory with accurate ground truth like RTK-GPS.

In this work, SURF features³ (Fig 3.6) and descriptors are respectively extracted from the left and right images in three steps :

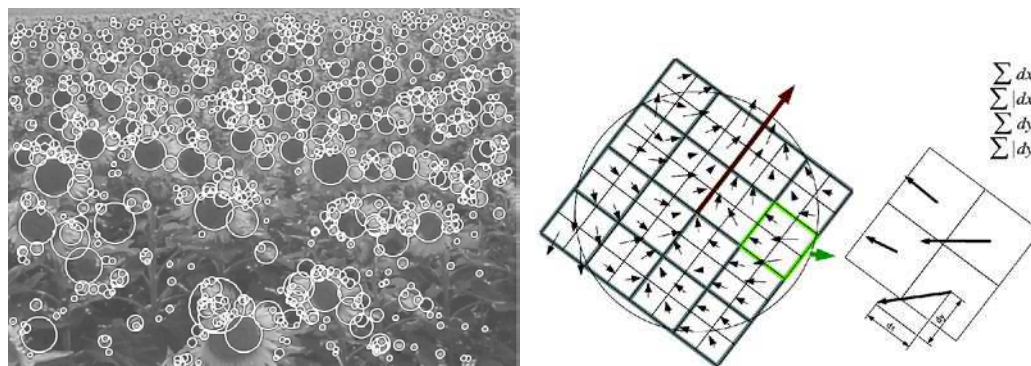


FIGURE 3.6 – Left) detected SURF points in a sunflower field image ; right) an oriented quadratic grid with 4×4 square sub-regions is placed around the interest point, and the wavelet responses for each square are computed to build the feature descriptor [12]

1. Scale spaces are implemented as image pyramids. Taking into account the discrete nature of integral images and the specific structure of image filters, layers I_δ are obtained by filtering the image with gradually bigger masks.
2. Then, given a point $q = (u, v)$ in an image I , the determinant of approximated Hessian matrix $H(u, I_\delta)$ in u at scale I_δ is calculated ; then a non-maximum suppression is applied in a $3 \times 3 \times 3$ neighborhood to localize interest points on the image over all scales.
3. After that, the descriptor D of each SURF feature is estimated based on pixel properties. Instead of computing the image gradients, first order Haar-wavelet responses are calculated in u and v directions based on integral images for fast filtering. Thus, the descriptor can describe the distribution of Haar-wavelet responses within the neighborhood of the interest point.

3.2.2.2/ FEATURE MATCHING

After extracting the features from two images, image feature matching is to measure the similarity of two features using pixel information or descriptors, then to find corresponding feature couples between two images. Two types of matching methods are discussed in the following part : area similarity based (window-based) matching method and feature descriptor based matching method.

- **Area similarity based matching.** Area based matching algorithms solve the correspondence problem by taking a window area around the considered pixels. With the assumption that all the pixels inside a matching window belong to the same object and their

3. <http://www.vision.ee.ethz.ch/~surf/index.html>

intensities (and disparity variation) are constant and continuous, the similarity of two extracted intensity windows is computed to represent the similarity of the considered central pixels.

Let I_1 be a local window around an image feature in one view, and I_2 one window around an image feature in another view. The similarity of the two areas can be measured by different metrics, as summarized in Tab 3.2.

Similarity metrics	Definitions
Sum of absolute differences	$SAD = \sum_{u,v \in W} I_1(u, v) - I_2(u, v) $
Zero-mean sum of absolute differences	$ZSAD = \sum_{u,v \in W} (I_1(u, v) - \bar{I}_1) - (I_2(u, v) - \bar{I}_2) $
Sum of squared differences	$SSD = \sum_{u,v \in W} (I_1(u, v) - I_2(u, v))^2$
Zero-mean sum of squared differences	$ZSSD = \sum_{u,v \in W} ((I_1(u, v) - \bar{I}_1) - (I_2(u, v) - \bar{I}_2))^2$
Normalized cross correlation	$NCC = \frac{\sum_{u,v \in W} I_1(u, v)I_2(u, v)}{\sqrt{\sum_{u,v \in W} I_1(u, v)^2 \sum_{u,v \in W} I_2(u, v)^2}}$
Zero-mean normalized cross correlation	$ZNCC = \frac{\sum_{u,v \in W} (I_1(u, v) - \bar{I}_1)(I_2(u, v) - \bar{I}_2)}{\sqrt{\sum_{u,v \in W} (I_1(u, v) - \bar{I}_1)^2 \sum_{u,v \in W} (I_2(u, v) - \bar{I}_2)^2}}$

TABLE 3.2 – Area similarity based matching metrics

In Tab 3.2, W is the matching window centered around an image feature point, $I_1(u, v)$ (and $I_2(u, v)$) is the intensity of a pixel within the window, \bar{I}_1 (and \bar{I}_2) is the average intensity of pixels inside the window. The limitation of SSD and SAD is their high sensitivity to changes of illumination conditions. NCC and ZNCC improve the matching performance by dividing the measuring result with intensity variance in matching window. ZNCC further improves NCC by removing the changes of intensity between two images.

Remarks : in order to improve the matching accuracy and to reduce the computation cost, size and shape of the matching window should be appropriately chosen : the window size should be large enough to incorporate enough intensity variations for matching, but also small enough to avoid the effects of projective distortion. Various methods have been proposed to adaptively choose the window size or window shape [84] [161].

• **Feature descriptor matching.** Matching two sets of descriptors (e.g., SIFT descriptor or SURF descriptor) between two images is to evaluate the similarity of features in their descriptor space. Different distances can be used to measure the difference of two descriptor vectors, e.g., Euclidean distance, Mahalanobis distance [100]. Then, after measuring the similarity between two descriptors, two approaches can be used to fix the corresponding feature for a feature point $q(u, v)$:

1) **the nearest method** finds the corresponding feature which has the closest distance to the descriptor vector of q ;

2) **the nearest distance ratio method** compares the distances between the closest and the second closest descriptor vectors to q . Then, the closest correspondence is accepted only if the ratio between these two distances is less than a defined threshold.

In our work, SURF descriptors with smallest Euclidean distance are considered to find the corresponding features. Though the descriptors based feature matching is reliable, some false correspondences are still unavoidable. So, after finding the closest corresponding features by SURF descriptors, *ZNCC* with 13×13 window size (window size is empirically chosen) under several geometric constraints is used to check the similarity between the two corresponding features.

• **Feature matching constraints.** In order to improve the matching precision and computation time, some geometric constraints are considered to reduce the search space or to refine the matching result :

- i) Epipolar constraint : for one image feature, its corresponding feature in the other image must lie on the relative epipolar line (Fig. 3.3). For rectified image pairs, a simplification of the general epipolar geometry is shown in Fig. 3.4 (right). The two cameras are arranged in parallel with identical focal length, and all the epipolar line are parallel to the horizontal scan-lines. Considering image noises, we define the epipolar constraint for features as $|v_l - v_r| \leq 2$;
- ii) Maximum and minimum disparity constraint : according to the length of baseline and the maximum and minimum depth of the 3D objects in the environment, the minimum and maximum disparity ranges are defined ;
- iii) Threshold of correlation score : for *ZNCC*, larger value indicates a closer relationship, thus, only the feature couples with the largest value of $ZNCC > 0.9$ are chosen as the potential corresponding features ;
- iv) Uniqueness constraint : one feature can only be matched with another one ;
- v) Ordering constraint : for two corresponding feature couples $q_{1,l} \leftrightarrow q_{1,r}$ and $q_{2,l} \leftrightarrow q_{2,r}$, if $q_{1,l}$ lies on the left (or right) side of $q_{2,l}$, $q_{1,r}$ should be on the same side of $q_{2,r}$;
- vi) Inverse matching (mutual checking) : after obtaining a potential feature correspondence ($q_l \leftrightarrow q_r$), taking the corresponding feature q_r in the right image as reference, we can look for its corresponding feature in the left image. If the same left image feature q_l is found to be the correspondence, this matching result ($q_l \leftrightarrow q_r$) is considered as an inlier.

3.2.2.3/ 3D LANDMARK RECONSTRUCTION

When the geometric arrangement of a stereoscopic system is known, local 3D position $Q^C(X^C, Y^C, Z^C)$ of a point Q relative to the camera center can be recovered based on its corresponding image points [56]. But, due to the influence of various noises, the left and right rays passing through the camera centers and the corresponding features might not intersect at the same 3D point, as shown in Fig. 3.7. An alternative method is to respectively obtain the left and right rays with left and right image features, then find the shortest segment that connects these two rays, and take the middle point of this segment as the 3D position of Q .

Taking the left camera system as the reference system, the left and right camera centers are respectively $C_l(0, 0, 0)$ and $C_r(-B, 0, 0)$. Let r_1 be a 3×1 unit vector of the ray that

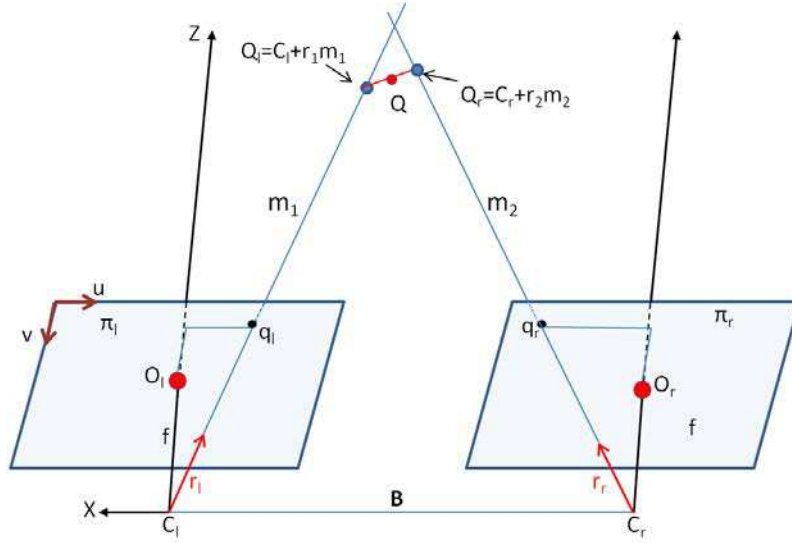


FIGURE 3.7 – Triangulation by intersecting rays passing through corresponding image points and camera centers

connects the left camera center and the left image feature $q_l^{C_l}(x_l, y_l, f)$, r_r be a 3×1 unit vector of the ray that connects the right camera center and the right image feature $q_r^{C_r}(x_r, y_r, f)$, Q_l and Q_r be the endpoints of the shortest line segment connecting the two rays [37]. B is the baseline between C_l and C_r , r_l and r_r are calculated with the coordinates of image features through :

$$\begin{aligned} r_l &= [x_l, y_l, f]^T / \|C_l q_l\| \\ r_r &= [x_r, y_r, f]^T / \|C_r q_r\| \end{aligned} \quad (3.20)$$

then, the coordinates of the 3D points Q_l and Q_r can be written as :

$$Q_l = C_l + r_l m_1, Q_r = C_r + r_r m_2 \quad (3.21)$$

where $m_1 = \|Q_l C_l\|$ and $m_2 = \|Q_r C_r\|$. In order to find the parameters m_1 and m_2 , we need to minimize :

$$\|Q_l - Q_r\|^2 = \|(C_l + r_l m_1) - (C_r + r_r m_2)\|^2 \quad (3.22)$$

this function can be written in the matrix format, as :

$$\begin{bmatrix} r_l & -r_r \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} = C_r - C_l \quad (3.23)$$

with the least-square approach, we can obtain :

$$\begin{bmatrix} m_1 \\ m_2 \end{bmatrix} = \left(\begin{bmatrix} r_l & -r_r \end{bmatrix}^T \begin{bmatrix} r_l & -r_r \end{bmatrix} \right)^{-1} \begin{bmatrix} r_l & -r_r \end{bmatrix}^T (C_r - C_l) \quad (3.24)$$

Since $B_0 = C_r - C_l = [-B \ 0 \ 0]^T$, $r_l^T r_l = 1$ and $r_r^T r_r = 1$, we can have :

$$m_1 = \frac{B_0 \cdot r_l - (B_0 \cdot r_r)(r_l \cdot r_r)}{1 - (r_l \cdot r_r)^2}, m_2 = \frac{(B_0 \cdot r_l)(r_l \cdot r_r) - B_0 \cdot r_r}{1 - (r_l \cdot r_r)^2} = (r_l \cdot r_r)m_1 - B_0 \cdot r_r \quad (3.25)$$

As seen in Fig. 3.7, the midpoint of the line segment $Q_l Q_r$ is supposed to be the coordinates of the 3D point Q :

$$Q = (Q_l + Q_r)/2 \quad (3.26)$$

For a given stereo system with fixed intrinsic and extrinsic parameters, if the depth Z of a point is estimated by direct triangulation model $Z = fB/d$, the derivative of Z to horizontal-disparity d is $\frac{\partial Z}{\partial d} = -\frac{fB}{d^2}$. Replacing d by $d = \frac{fB}{Z}$, the relationship of depth and disparity can be written as : $\frac{\partial Z}{\partial d} = -\frac{Z^2}{fB}$, $\partial Z = -\frac{Z^2}{fB} \times \partial d$. Though the variance of depth in triangulation process is not a simple scalar function of distance to the point [148], we can note that the influence becomes greater when the depth increases, and the accuracy of the depth estimation will decrease. Therefore, in our experiments, reconstructed 3D points with negative depth or with depth more than 50 meters are eliminated.

3.2.2.4/ 2D FEATURE TRACKING AND REFERENCE PAIR UPDATING

In order to estimate the camera motion with image sequences, corresponding image features in continuous frames can be found by two alternative methods : the first one is to detect features in every image frame, then match them between two frames ; the second method is to track previously detected features in current frame with techniques like optical flow. As the geometric relationship between consecutive frames is unknown, both the computation of epipolar line and the use of area based matching method are time-consuming, thus we choose Kanade-Lucas-Tomasi (KLT) feature tracker [98] to track key features.

2D feature tracking

When a new stereo pair is acquired, the previous matched key features between left and right images are separately tracked in the current image pair by KLT feature tracker [98]. KLT feature tracker uses local information in a small window around the image points with three assumptions :

- Brightness constancy between corresponding pixels in two frames ;
- Pixel intensities constancy in a local window around the pixel ;
- Small movement between two consecutive frames.

A pixel located at (u, v) in image frame at time t with intensity $I(u, v)$ will move to $(u + \Delta u, v + \Delta v)$ in image frame at time $t + \Delta t$ with intensity $I(u + \Delta u, v + \Delta v)$. The pixel intensity at time $t + \Delta t$ can be approximated by first-order Taylor series as :

$$I(u + \Delta u, v + \Delta v, t + \Delta t) = I(u, v, t) + \frac{\partial I}{\partial u} \Delta u + \frac{\partial I}{\partial v} \Delta v + \frac{\partial I}{\partial t} \Delta t \quad (3.27)$$

where $\frac{\partial I}{\partial u}$, $\frac{\partial I}{\partial v}$, $\frac{\partial I}{\partial t}$ are derivatives of image intensity at (u, v, t) . With the brightness constancy assumption, we can have :

$$\frac{\partial I}{\partial u} \Delta u + \frac{\partial I}{\partial v} \Delta v + \frac{\partial I}{\partial t} \Delta t = \frac{\partial I}{\partial u} V_u + \frac{\partial I}{\partial v} V_v + \frac{\partial I}{\partial t} = 0 \quad (3.28)$$

where V_u and V_v are respectively image velocities on u and v directions. With multiple pixels in a window, an optimal (V_v, V_u) can be found by solving the over-constrained function (Eq.3.28) with least square method.

During the tracking process, several constraints are applied to remove outliers of tracked features :

- i) Intensity constraint : the intensity difference between patches (13×13 pixels) around the image features in two consecutive frames should be less than 500 ;

- ii) Search space constraint : a tracked feature should not move out of the tracked image plane ;

After obtaining the tracked features in the left and right frames, the tracked image features in the current camera coordinate system are triangulated into 3D space. The distribution of features in the current stereo frame is calculated as $distribut_{tracking}$ by dividing the left image plane into a series of 20×20 (pixels) squares and calculating the number of squares which contain the extracted features.

Reference stereo pair updating

When the camera moves, some features will move out of FOV of the camera, as in Fig. 3.8(a). Therefore, only features that can be tracked in the previous frame will be tracked sequentially until the number or distribution conditions cannot be satisfied.

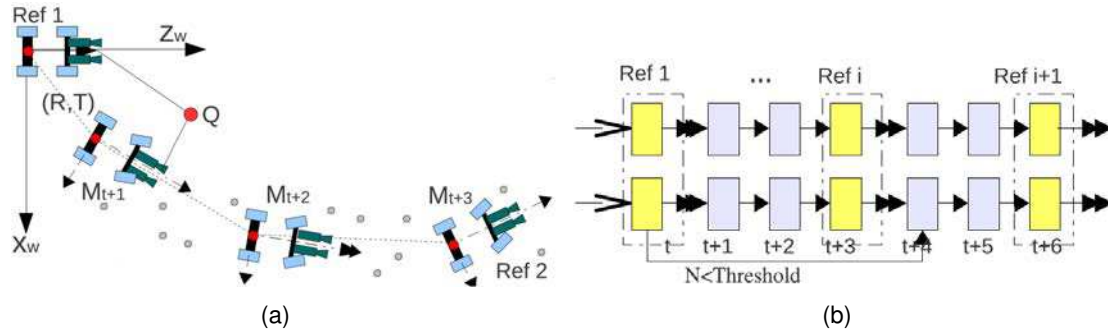


FIGURE 3.8 – (a) Vehicle model (left camera center is considered to be the vehicle reference system center) ; (b) selection and re-initialization of reference image pair

The reference stereo pair selection and updating mechanism is illustrated in Fig. 3.8(b). The first stereo pair is selected as initial reference image pair. After reconstructing the features into 3D space, the distribution of image features in the reference pair is calculated as $distribut_{reference}$, like $distribut_{tracking}$. For our experiments, the thresholds of reference re-initialization is empirically set : if the number of matched features in the reference stereo pair is n_m , the number threshold is set as $T_{number} = n_m * 60\%$; the distribution threshold is set as $T_{distribut} = (distribut_{reference} * 60\%)$. If the number or distribution value is less than the defined threshold, the previous stereo pair is selected as the new reference stereo pair, then new features are detected for matching and tracking.

3.2.2.5/ MOVING FEATURE ELIMINATION AND RIGID MOTION ESTIMATION

Taking the first left camera position as the origin of the global coordinate system W , the global positions of every camera pose and 3D landmarks can be obtained with the positions and orientations of their reference stereo pairs.

Vision based relative motion estimation

Let $\{Q_i^t\}$ and $\{Q_i^r\}$ ($i = 1 \dots N$) be two corresponding point sets respectively obtained by the camera system at time t and reference frame. These two point sets can be related by :

$$Q_i^t = R_{r \rightarrow t} Q_i^r + T_{r \rightarrow t} + V_i \quad (3.29)$$

where $R_{r \rightarrow t}$ is a 3×3 rotation matrix composed of three rotation angles : yaw, pitch and roll (θ, ψ, ϕ) , $T_{r \rightarrow t}$ is a 3D translation vector (T_x, T_y, T_z) , and V_i is a noise vector. In order to find the optimal transformation $[R, T]_{r \rightarrow t}$ between the reference set and current set, we need to minimize the residual error :

$$\varepsilon^2 = \sum_{i=1}^N \|Q_i^t - R_{r \rightarrow t} Q_i^r - T_{r \rightarrow t}\|^2 \quad (3.30)$$

With the assumption that the environment is rigid, the centroids \bar{Q}^t and \bar{Q}^r of two point sets should be the same :

$$\bar{Q}^t = \frac{1}{N} \sum_{i=1}^N Q_i^t \quad (3.31)$$

$$\bar{Q}^r = \frac{1}{N} \sum_{i=1}^N Q_i^r \quad (3.32)$$

We note $\{Q_{ci}^t\} = \{Q_i^t\} - \bar{Q}^t$ and $\{Q_{ci}^r\} = \{Q_i^r\} - \bar{Q}^r$ as the two centered point sets corresponding to $\{Q_i^t\}$ and $\{Q_i^r\}$. Then,

$$\begin{aligned} \varepsilon^2 &= \sum_{i=1}^N \|Q_{ci}^t - R_{r \rightarrow t} Q_{ci}^r\|^2 \\ &= \sum_{i=1}^N (Q_{ci}^{tT} Q_{ci}^t + Q_{ci}^{rT} Q_{ci}^r - 2Q_{ci}^{tT} R_{r \rightarrow t} Q_{ci}^r) \end{aligned} \quad (3.33)$$

This equation is minimized when the last term is maximized. It is equivalent to maximizing the trace (RH) of this term, the cost expression is written as :

$$H = \sum_{i=1}^N Q_{ci}^r Q_{ci}^{tT} \quad (3.34)$$

The optimal rotation matrix can be found by SVD (singular value decomposition) of matrix $H = US D^T$ with Arun's solution [3], together with Umeyama's complements [160] for some degenerated cases :

$$S = \begin{cases} I_{3 \times 3}, & \text{if } \det(D) \times \det(U) = 1 \\ \text{diag}(1, 1, -1), & \text{if } \det(D) \times \det(U) = -1 \end{cases} \quad (3.35)$$

Then the rotation matrix can be obtained by $R_{r \rightarrow t} = DS U^T$. Since the optimal translation vector can align the centroid of the point set $\{Q_i^t\}$ with the rotated centroid of the point set $\{Q_i^r\}$, translation vector $T_{r \rightarrow t}$ is estimated by : $T_{r \rightarrow t} = \bar{Q}^t - R_{r \rightarrow t} \bar{Q}^r$.

Elimination of moving features

Some 3D points with large depth noises have already been rejected before motion estimation. In outdoor environments, the dynamic nature of vehicle motion and scenery modeling make the use of vision methods challenged to apply. Natural landmarks in areas with amount of repeating textures (e.g., trees, fences) might make feature detection and matching complicated. During the procedure of vehicle motion estimation, the use of features on non-stationary or non-rigid objects may arouse unreal vehicle motion, such as suddenly appeared pedestrians or moving vehicles (Fig. 3.9). It is important to eliminate the false tracking features and features belonging to moving obstacles. Here, outlier rejection mechanisms with RANSAC and 3D/2D optical flow are applied to remove features



FIGURE 3.9 – Urban environment with moving vehicles or strong sunlight

belonging to the non-static or non-rigid objects. Only static features are used to estimate vehicle ego-motion.

• **Iterative motion estimation with RANSAC.** RANdom SAmple Consensus (RANSAC) is a robust estimator proposed by Fischler et al. [55]. Instead of using all data for model estimation, RANSAC statistically uses as small initial data set as necessary and enlarges this set with consistent data when possible. The steps of RANSAC algorithm [70] are listed in Algorithm 1.

Algorithm 1: RANSAC

Input : A data set S , the smallest sample size n , error distance threshold l , number N_{sample} of sampling times, threshold N of an acceptable consensus set

Output : Fitted model

- 1 Randomly select a sample of n data from the data set S , and estimate the model based on this subset;
 - 2 Determine which data in S are within the distance threshold l of the model. Data within the distance are considered to be inliers and form a consensus set S_i of the sample;
 - 3 If the number of inliers in S_i is greater than the threshold of an acceptable consensus set N , the model is estimated again using all data in this consensus set;
 - 4 After N_{sample} trials, the largest consensus set S_i is used to re-estimate the model.
-

In our experiments, iterative motion estimation with RANSAC is to iteratively select a random subset of 3 tracked image points in the current left image frame; then, generate one camera motion hypothesis R and T with the point correspondences between consecutive images. The reconstructed points are transformed from the reference coordinate system to the current camera system with the new motion hypothesis. The Euclidean differences between each two 3D positions are calculated, and the corresponding points within the threshold are considered to be inliers. The best motion hypothesis is the one with the largest number of inliers. Then, the final solution is estimated with the largest inlier subset.

To ensure that the randomly selected three points are well distributed in the image, every two image features must have a distance larger than the square size 24×24 (pixels). The other parameters are dynamically chosen [70] :

1. For probability of 95% that a point correspondence is an inlier, the distance threshold l used is set to $5.99 \times \delta^2$, where δ is the standard deviation of Euclidean differences between the two 3D point sets ;
2. The number N_{sample} of sampling times is dynamically estimated by $N_{sample} = \frac{\log(1-p)}{\log(1-(1-\epsilon)^n)}$ to ensure a probability p that at least one of the random samples is free from outliers, where n is the smallest sample size, p is set to 0.99, and ϵ is the percentage of outliers, $\epsilon = 1 - (\text{number}_{inliers})/(\text{number}_{points})$.

• **3D/3D method to remove outlier.** When the vehicle moves in rigid and static environments, all static objects (which are considered as the background) should have similar 3D motion relative to the moving vehicle. For the corresponding local 3D positions $\{Q_i^r\}$ and $\{Q_i^t\}$ ($i = 1 : N$) at time t and *reference* instant, the velocity of 3D displacement can be obtained by [145] :

$$\widetilde{\Delta Q}_i = \begin{bmatrix} \widetilde{X}_i^r \\ \widetilde{Y}_i^r \\ \widetilde{Z}_i^r \end{bmatrix} = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} + \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \begin{bmatrix} X_i^r \\ Y_i^r \\ Z_i^r \end{bmatrix} \quad (3.36)$$

where $\omega = [\omega_x, \omega_y, \omega_z]^T$ is the angular velocity and $\tau = [\tau_x, \tau_y, \tau_z]^T$ is the translational velocity. 2D projection of the point Q_i^r on the left image plane can be approximately written as $\{x_i = fX_i^r/Z_i^r, y_i = fY_i^r/Z_i^r\}$. Then, the 2D optical flow can be obtained by computing the derivation of $\{x_i, y_i\}$ to time and represented by 3D object position Q_i^r , 3D motion $\widetilde{\Delta Q}_i$ and disparity $d_i = x_{i,l} - x_{i,r}$ as :

$$\begin{bmatrix} V_{x,i} \\ V_{y,i} \end{bmatrix} = \begin{bmatrix} \frac{d_i}{B} & 0 & -\frac{d_i * x_i}{f * B} & -\frac{y_i * x_i}{f} & \frac{f^2 + x_i^2}{f} & -y_i \\ 0 & \frac{d_i}{B} & -\frac{d_i * y_i}{f * B} & -\frac{f^2 + y_i^2}{f} & \frac{y_i * x_i}{f} & x_i \end{bmatrix} \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (3.37)$$

For every three couples of tracked features, Eq.3.37 is solved to obtain a motion parameter vector. The points fit the initial motion parameters are measured by Euclidean distance (suppose that every variables are independent), the mean error $\bar{\epsilon}$ and standard deviation σ of errors are used to remove the tracked features whose deviation is more than 2σ .

Global pose estimation based on incremental procedures

Taking the first camera position as the origin of global coordinate system, the camera pose and position of the 3D landmarks in the global system can be obtained with the pose of its reference stereo pair :

$$\begin{aligned} R_t^{global} &= R_r R_{r \rightarrow t} \\ T_t^{global} &= R_r T_{r \rightarrow t} + T_r \\ \{Q_{i,t}^{global}\} &= R_t^{global} \{Q_i^t\} + T_t^{global} \end{aligned} \quad (3.38)$$

where :

- R_r, T_r : the rotation matrix and translation vector of the reference stereo pair in the global system ;
- $R_{r \rightarrow t}, T_{r \rightarrow t}$: the rotation matrix and translation vector of camera at time t with respect to

the reference stereo pair ;

– $R_t^{global}, T_t^{global}$: the current camera rotation matrix and translation vector in the global system ;

– $\{Q_i^t\}$: the set of coordinates of 3D points $\{Q_i\}$ relative with the local stereo pair at time t ;

– $\{Q_{i,t}^{global}\}$: the set of coordinates of 3D points $\{Q_i\}$ in the world coordinate system.

3.2.3/ ERROR MODELING OF STEREO VISUAL ODOMETRY

For the procedure of a vision algorithm, it is important to know how the random perturbation is propagated to the estimated quantity through the vision algorithm [68] [108]. Precision of the estimation is essential when we want to integrate the vision based estimation with other measurements. For example, the error propagation problem of stereo reconstruction and motion estimation is discussed in the following part.

3.2.3.1/ OVERVIEW OF UNCERTAINTY MODELING AND PROPAGATION

• **Covariance** : considering a random variable x with density P , expectation of x is defined as :

$$E\{x\} = \int_{-\infty}^{+\infty} xp(x)dx \quad (3.39)$$

and the variance of x is :

$$Var\{x\} = E\{(x - E\{x\})^2\} \quad (3.40)$$

The variance describes the quality (error) around the expected value $E\{x\}$. If we consider another random variable y , the covariance between these two random variables x and y is defined as :

$$Cov(x, y) = E\{(x - E\{x\})(y - E\{y\})\} \quad (3.41)$$

The covariance describes how much the two random variables are correlated with each other.

• **Properties of expectation and variance** :

- operation of the expected value E is linear, written as : $E\{ax\} = aE\{x\}$;
- if all values of x are scaled by a constant a , the variance of ax is scaled by the square of constant a , as : $Var\{ax\} = a^2Var\{x\}$;
- the expectation and variance of the sum of two random variables x and y are given by :

$$\begin{aligned} E\{ax + by\} &= aE\{x\} + bE\{y\} \\ Var\{ax + by\} &= a^2Var\{x\} + b^2Var\{y\} + 2abCov(x, y) \end{aligned} \quad (3.42)$$

More generally, denote the n variables related to a variable f by a vector $s(s_1, s_2, \dots, s_n)$, the variance of f can be written as :

$$Var\{f\} = \sum_{i=1}^n \left(\frac{\partial f}{\partial s_i}\right)Var\{s_i\} + \sum_i^n \sum_{j,j \neq i}^n \left(\frac{\partial f}{\partial s_i}\right)\left(\frac{\partial f}{\partial s_j}\right)2Cov(s_i, s_j) \quad (3.43)$$

When f is a multi-dimension vector, $Var\{f\}$ becomes the covariance of f , as :

$$Cov\{f\} = JCov(s)J^T \quad (3.44)$$

where J is the Jacobian matrix (first partial derivative) of f to s , $Cov(s)$ is the covariance of the random variables $\{s_1, s_2, \dots, s_n\}$.

If f is nonlinear to the variables s , the covariance of f can be approximated by first-order error propagation of the true value with first-order Taylor series expansion at point s_i , as :

$$Y \approx f(s_i) + \left. \frac{\partial f}{\partial s} \right|_{s=s_i} (s - s_i) \quad (3.45)$$

Since we suppose that the error of s is small, higher orders in Taylor series expansion are neglected. When f is too nonlinear, the probability distribution of f might not be well approximated.

3.2.3.2/ UNCERTAINTY OF STEREO RECONSTRUCTION

Sources of 3D landmark errors vary from inaccurate camera calibration parameters, physical image noises, illumination conditions to feature detection and matching errors, these errors will lead to some 3D points that do not really exist [32]. In this work, we consider only the covariance of stereo reconstruction propagated from the image feature position noise through the 3D reconstruction function.

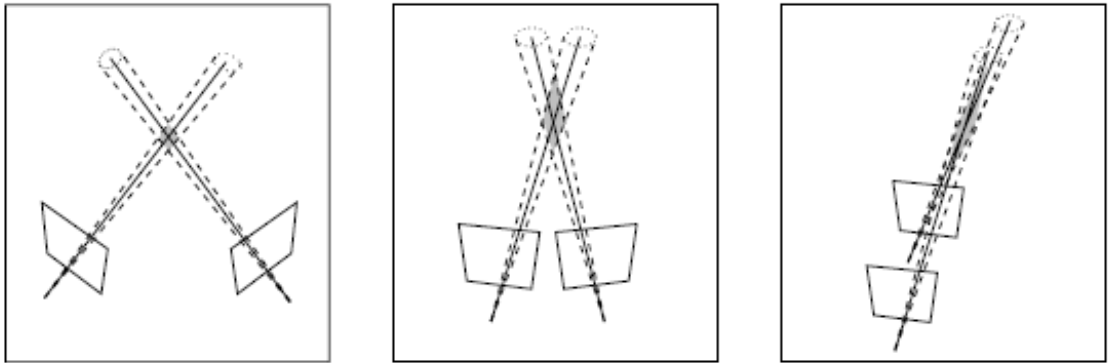


FIGURE 3.10 – Uncertainty region of a reconstructed 3D point [70]

• **Covariance of image feature position.** Due to the noise of image points caused by image quantization [102], the estimated coordinates of a 3D point Q might lie anywhere inside the shaded region, as shown in Fig. 3.10. Assume that the normally distributed error of image coordinates are uncorrelated on horizontal and vertical directions, the zero-mean Gaussian position error of image feature $q_l(x_l, y_l)$ and $q_r(x_r, y_r)$ can be respectively written as 2×2 covariance matrices :

$$H_{left} = \begin{bmatrix} \delta_{x_l}^2 & 0 \\ 0 & \delta_{y_l}^2 \end{bmatrix}, H_{right} = \begin{bmatrix} \delta_{x_r}^2 & 0 \\ 0 & \delta_{y_r}^2 \end{bmatrix} \quad (3.46)$$

where $\delta_{x_l}^2$, $\delta_{x_r}^2$, $\delta_{y_l}^2$, and $\delta_{y_r}^2$ are the standard deviations of pixel coordinates x_l , x_r , y_l , and y_r . As the two corresponding points are uncorrelated, covariance matrix H_{pair} of the image points pair can be written as a 4×4 diagonal matrix :

$$H_{pair} = \begin{bmatrix} \delta_{x_l}^2 & 0 & 0 & 0 \\ 0 & \delta_{y_l}^2 & 0 & 0 \\ 0 & 0 & \delta_{x_r}^2 & 0 \\ 0 & 0 & 0 & \delta_{y_r}^2 \end{bmatrix} \quad (3.47)$$

The covariance of a scale invariant feature (e.g., SURF feature used in our work) can also be estimated from the detector response map in the neighborhood of feature point [169]. Covariance of the tracked 2D feature position is supposed to be proportional to the distance between the reference and tracked image frames (though the 2D locations of features are not independent from each other when features are tracked from one frame to another ; considering the computation time for independently detecting new features in every image frame, feature tracking is used here and the cross-correlation between features is not considered).

• **Covariance of the triangulated 3D point propagating from the image feature location covariance.** Covariance of the reconstructed 3D points is propagated from the covariance H_{pair} of 2D image positions through the nonlinear triangulation model (see Eq.3.20 to Eq.3.26). According to Eq.3.44, the Jacobian matrix Q' of a 3D point Q with respect to 2D image coordinates $\{x_l, y_l, x_r, y_r\}$ can be estimated by [37] :

$$Q' = (r'_l m_1 + r_l m'_1 + r'_r m_2 + r_r m'_2)/2 \quad (3.48)$$

$$\text{with : } \begin{aligned} m'_1 &= \frac{[B_0 \cdot r'_l - (B_0 \cdot r_r)(r_l \cdot r_r) - (B_0 \cdot r_r)(r'_l \cdot r_r + r_l \cdot r'_r)][1 - (r_l \cdot r_r)^2] + 2[B_0 \cdot r_l - (B_0 \cdot r_r)(r_l \cdot r_r)][(r_l \cdot r_r)(r'_l \cdot r_r + r_l \cdot r'_r)]}{[1 - (r_l \cdot r_r)^2]^2} \\ m'_2 &= (r_l \cdot r_r)m'_1 + (r'_l r_r + r_l r'_r)m_1 - B_0 \cdot r'_r \end{aligned}$$

where r'_l and r'_r are respectively the first derivatives of r_l and r_r with respect to $\{x_l, y_l, x_r, y_r\}$. Then, the covariance matrix of the reconstructed 3D point can be approximately measured by :

$$p = Q' H_{pair} Q'^T \quad (3.49)$$

3.2.3.3/ UNCERTAINTY OF STEREO VISUAL ODOMETRY

Covariance of the estimated transformation between two camera poses depends on many factors, including the number, the accuracy and geometric distribution of point correspondences, etc. In this section, the covariance of the estimated pose transformation is propagated from the covariance of the used 3D points sets.

Error of relative motion

Let $(R_{r \rightarrow t}, T_{r \rightarrow t})$ denote the motion parameters from *reference* frame to frame t . The position of a landmark Q_i^t at time t can be estimated by its position Q_i^r at the reference time and the motion parameters

$$Q_i^t = g(R_{r \rightarrow t}, T_{r \rightarrow t}, Q_i^r) \quad (3.50)$$

$i = 1, \dots, N$, where N is the number of point pairs used for motion estimation. Assume that every landmark Q_i^r follows a Gaussian distribution with mean Q_i^r and covariance matrix p_i^r , $Q_i^r \sim N(Q_i^r, p_i^r)$; and coordinates Q_i^t follow a Gaussian distribution with mean Q_i^t and covariance matrix p_i^t , $Q_i^t \sim N(Q_i^t, p_i^t)$. If the estimated camera motion parameters are optimal, covariance matrix $\sum_{r \rightarrow t}$ of the motion parameters can be approximated through error propagation [70][107] as :

$$(\sum_{r \rightarrow t})^{-1} = S^T P^{-1} S = \sum_{i=1}^N (S_i^T p_i^{-1} S_i) \quad (3.51)$$

where S is the Jacobian matrix of $\{Q^t\}$ with respect to the motion parameters (Eq. 3.50), P is the covariance of landmarks positions. S_i is the Jacobian matrix of the i -th point $\{Q_i^t\}$ with respect to the motion parameters, and p_i is the covariance of the landmarks i at the reference frame and frame t , $p_i = p_i^t + p_i^r$.

When the ground is flat, the camera motion parameters are the translation vector $(\Delta x, \Delta y)$ and rotation θ , as $(\Delta x, \Delta y, \theta)$, Equation 3.50 can be written as $Q_i^t = [\cos\theta X_i^r - \sin\theta Y_i^r + \Delta x, \sin\theta X_i^r + \cos\theta Y_i^r + \Delta y]^T$.

Error of current vehicle pose

As described in Eq.3.38, the current vehicle pose $(R_t^{global}, T_t^{global})$ is obtained by integrating the pose at the reference frame (R_r, T_r) (with covariance Σ_r) and the relative motion $(R_{r \rightarrow t}, T_{r \rightarrow t})$ (with covariance $\Sigma_{r \rightarrow t}$) between the reference and current frames.

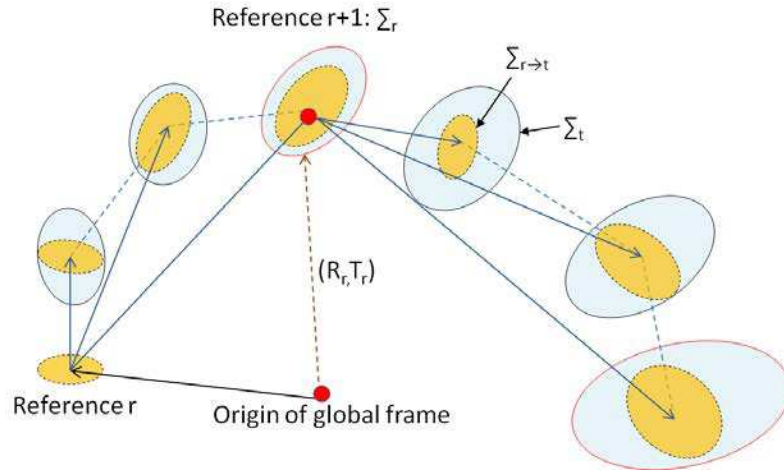


FIGURE 3.11 – Error propagation of stereovision based odometry

Supposing that the relative motion errors are small and independent with the reference pose error [147], the covariance Σ_t of current vehicle pose P_t^{global} can be propagated from the reference frame and the relative motion error by first-order Taylor approximation, the higher orders in Taylor series expansion are neglected :

$$\Sigma_t = J_{(r,r \rightarrow t)} \begin{bmatrix} \Sigma_r & 0 \\ 0 & \Sigma_{r \rightarrow t} \end{bmatrix} J_{(r,r \rightarrow t)}^T \quad (3.52)$$

where $J_{(r,r \rightarrow t)}$ is the Jacobian matrix of the global vehicle pose with respect to (R_r, T_r) and $(R_{r \rightarrow t}, T_{r \rightarrow t})$ (see Eq.3.38). Σ_t can be written as :

$$\Sigma_t = J_{(R_r, T_r)} (\Sigma_r) (J_{(R_r, T_r)})^T + J_{(R_{r \rightarrow t}, T_{r \rightarrow t})} (\Sigma_{r \rightarrow t}) (J_{(R_{r \rightarrow t}, T_{r \rightarrow t})})^T \quad (3.53)$$

As shown in Fig. 3.11, the yellow ellipse centered at each pose estimation represents the covariance of the relative vehicle motion. The ellipse in light blue color represents the vehicle pose covariance in the global frame. The covariance of the vehicle pose in the global frame grows gradually if no other measurements are applied.

3.2.4/ CONCLUSION

In this section, stereovision based visual odometry is presented. Stereoscopic system is used to capture the stereo video flow, recover the Euclidean 3D environments and estimate the vehicle motion on the basis of image feature detection and tracking. Several frames are selected as references and the optimal rotation and translation between the current and reference frames are computed using a RANSAC based minimization method. Covariance of the visual odometry estimation is also measured in this section. Noises of the landmarks are propagated from the image points through triangulation process. Accuracy of the estimated transformation parameters is dependent on the particular landmarks used for pose estimation. In the future works, other factors like precision of the camera calibration and stereo calibration process might also be incorporated for measuring the VO accuracy [120]. Besides, the precision of visual odometry is still challenged in outdoor environments, especially when the light condition changes, e.g., if the camera looks directly to the sun, light spots might appear and a large part of the images might be totally white (Fig. 3.9(right)). In these situations, it is difficult to detect enough features for accurate motion estimation.

3.3/ LASER RANGE FINDER BASED VEHICLE MOTION ESTIMATION

The basic principles of laser range finder (LRF) systems were introduced in 2.2.4. A laser range finder system measures the distance of an object by sending a laser pulse and calculating the traveling time of the pulse. Classified by the range of FOV, some LRF systems are listed in Fig. 3.12.

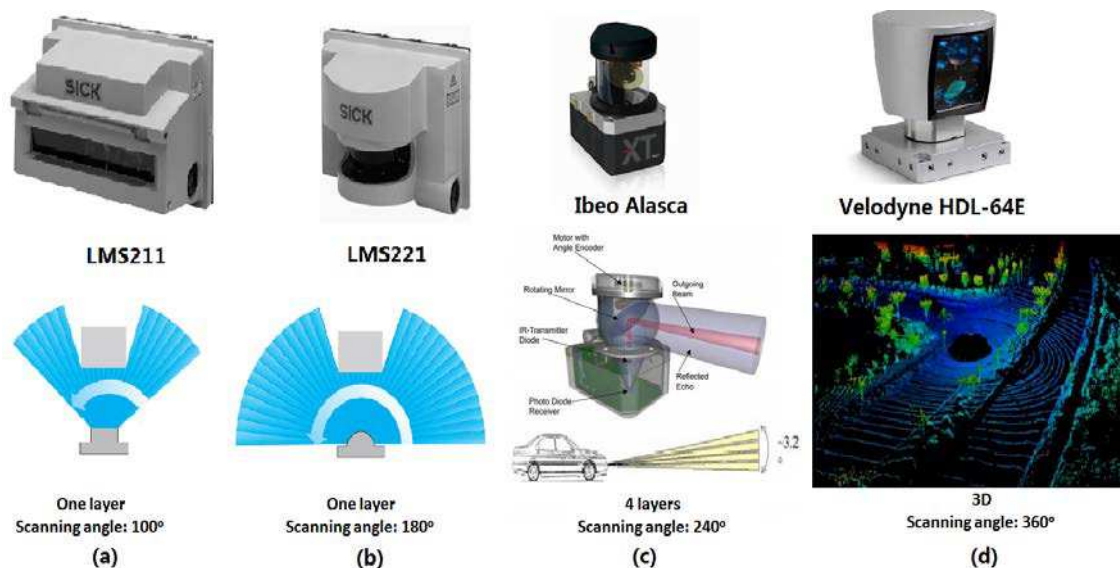


FIGURE 3.12 – Some LRF systems used in autonomous vehicle research

- The most often used LRF system in robotics area is 2D LRFs with a single layer, such as SICK LMS211 in Fig. 3.12 (a) and SICK LMS221 in Fig. 3.12 (b). These two systems scan their surroundings in two dimensions respectively with a scanning angle of 100° and 180° . In the radial field, a light impulse can be emitted every 0.25° , 0.5° or 1° , the maximum range can be measured is $80m$.

- Multi-layer LRF is used to scan more planes. For example, IBEO Alasca XT⁴ in (Fig. 3.12(c) scans 4 individual planes with a vertical angle of 3.2°. 3D LRF systems with multiple lasers have also been used, like Velodyne HDL-32E and Velodyne HDL-64E⁵ in Fig 3.12 (d), which is used on Google self-driving car [66] and some other autonomous vehicles. Velodyne HDL-64E uses an array of 64 lasers to scan the environment around the vehicle, with 360° horizontal field of view and 26.8° vertical field of view, the maximum range can be measured is 120m.

In this section, a single layer LRF is used for vehicle motion estimation. Several LRF based motion estimation methods (i.e., laser scan alignment methods) are introduced in section 3.3.1 ; then, covariance of ICP process is discussed in section 3.3.2.

3.3.1/ LASER SCAN ALIGNMENT METHODS

A LRF measurement is represented by a point (d_i, ϕ_i) (Fig. 3.13) in polar coordinates system. It can be converted to coordinates in Cartesian coordinates system (x_i, y_i) through :

$$(x_i, y_i) = (d_i \cos \phi_i, d_i \sin \phi_i) \quad (3.54)$$

where d_i is the measured range and ϕ_i is the incident angle of a laser beam. Coordinates used in the following alignment process are all in Cartesian coordinate system.

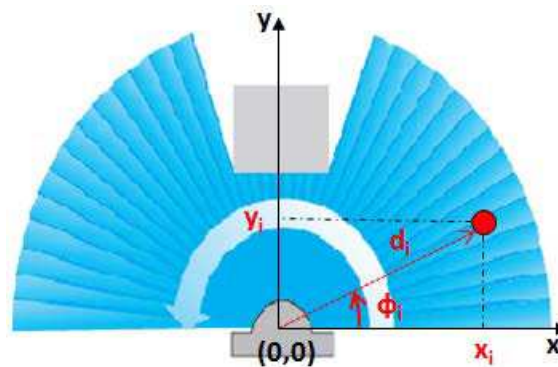


FIGURE 3.13 – A point with coordinates (d_i, ϕ_i) in polar coordinates system can be converted into Cartesian coordinates (x_i, y_i)

Scan alignment is the process to align a LRF scan data (2D or 3D) set with another LRF scan data set, which is considered to be the reference model. ICP (Iterative Closest Point) method [13] is one of the most used alignment methods to find the rigid transformation between a model point set and a reference point set. If a LRF sensor is rigidly installed on a moving platform, the movement of this platform can then be obtained by LRF scan alignment [54].

3.3.1.1/ CLASSIC ITERATIVE CLOSEST POINT (ICP) METHOD

Let $Model = \{q_1, q_2, \dots, q_{n_m}\}$ denote a model points set, let $Data = \{p_1, p_2, \dots, p_{n_d}\}$ denote a data set (n_m and n_d are respectively the number of points in the *Model* set and in the

4. <http://www.ibeo-as.com/>

5. <http://velodynelidar.com/lidar/hdlproducts/hdl64e.aspx>

Data set). ICP tries to find the optimal 2D transformation T between two point sets by iterative data association of corresponding points with nearest neighbor algorithm and least square minimization.

Algorithm of ICP method is shown in Algorithm 2. Let T_0 denote the initial transformation between two data sets, each point p_i in *Data* set is transformed by T_0 , then the closest point of the transformed p_i in the *Model* set is searched. For a set of n matched couples, the optimal transformation is estimated by minimizing the sum of point difference squares in Eq 3.55 through least mean square method :

$$\hat{T} = \arg \min \frac{1}{n} \sum_{i=1}^n \|T_0 p_i - q_i\| \quad (3.55)$$

Then, the transformation T_0 is replaced by \hat{T} and the previous steps are repeated till the number of iterations reaches the maximum threshold, or the difference of minimization errors between two iterations is below a predefined threshold.

Algorithm 2: ICP (Iterative Closest Point)

Input : $Model = \{q_1, q_2, \dots, q_{n_m}\}$, $Data = \{p_1, p_2, \dots, p_{n_d}\}$, initial transformation T_0 , threshold ξ

Output : The best transformation \hat{T} between two data sets

- 1 **do**
 - 2 Apply the transformation T_0 to points p_i in *Data* set, then find the closest point q_i of each transformed p_i in *Model* set by their Euclidean distance
 $e_i(p_i, q_i, T_0) = \|T_0 p_i - q_i\|$;
 - 3 For a set of n matched couples, the optimal transformation is estimated by minimizing the following equation through least square method :
 $\hat{T} = \arg \min \epsilon = \arg \min \frac{1}{n} \sum_{i=1}^n \|T_0 p_i - q_i\|$;
 - 4 Replace the transformation T_0 with $\hat{T} : T_0 \leftarrow \hat{T}$
 - 5 **while** $\epsilon > \xi$;
-

In our case, the *Model* and *Data* sets are 2 LRF scans, the transformation T is the relative 2D translation and rotation angle, $T = \{\Delta x, \Delta y, \Delta \theta\}$. However, as the 2D LRF scans are sparse and uncertain in outdoor applications, the estimated motion and corresponding couples from ICP algorithm are not always reliable. Thus several modifications are integrated into the classic ICP algorithm to enhance motion estimation accuracy. A comparison of several ICP variants can be found in [138]. We evaluate an outlier-rejection ICP, and two image aided scan alignment methods in the two following parts.

3.3.1.2/ OUTLIER-REJECTION-ICP (OR-ICP)

During the matching procedure of ICP, an outlier reduction strategy with an adaptive distance error threshold during each iteration step is added before applying line 3 in Algorithm 2.

For every new estimated transformation T_0 , the standard deviation σ of errors between the transformed data set and model set is calculated. n couples of point correspondences between the previous and current laser scans $\{(p_1, q_1); \dots (p_i, q_i) \dots; (p_n, q_n)\}$ are used for the

distance calculation as follows :

$$\begin{aligned}
 e_i(p_i, q_i, T_0) &= \| T_0 p_i - q_i \|, i = 1, 2, \dots, n \\
 \bar{e} &= \frac{1}{n} \sum_{i=1}^n e_i \\
 \sigma &= \sqrt{\frac{1}{n} \sum_{i=1}^n (e_i - \bar{e})^2}
 \end{aligned} \tag{3.56}$$

Data points which satisfy the following threshold relation are considered to be outlier and eliminated before the next iteration :

$$|e_i - \bar{e}| > 3\sigma \tag{3.57}$$

Then, the ICP process continues with the rest points. As seen in Fig. 3.14, when there are a lot of unstructured objects in the environment, the data set transformed by the outlier-rejection ICP (right image) can fit the model set better than with the classic ICP (left image). Comparison of the estimation accuracy of ICP and OR-ICP in long trajectory will be shown in the experiments part of this chapter (section 3.4.3).

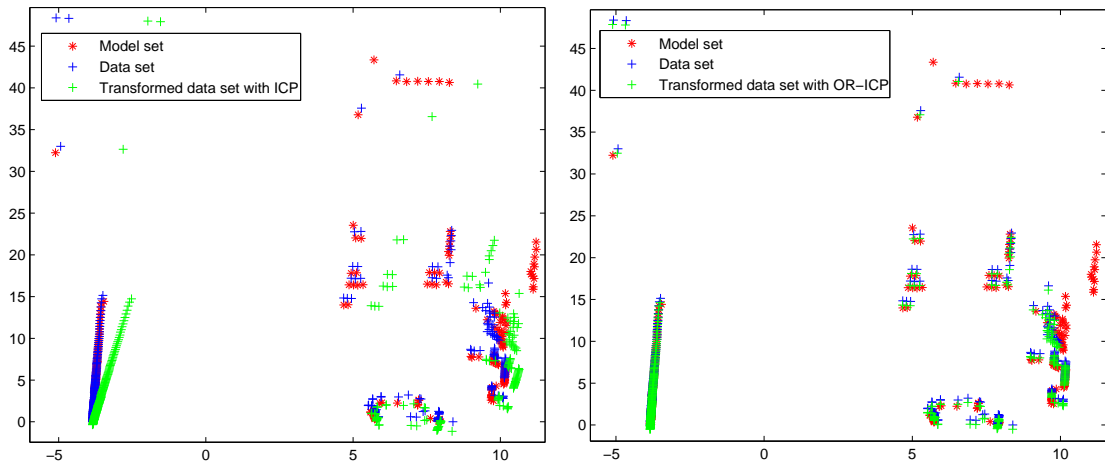


FIGURE 3.14 – Two LRF scans are taken at consecutive positions, data sets are respectively transformed with the pose estimation from classic ICP (left) and OR-ICP (right)

If the initial transformation T_0 is not close enough to the truth, this algorithm might converge to a local minimum. In order to avoid the local convergence of the minimization process, the optimal transformation of the previous scan is used as the initial transformation of the current scan alignment as the vehicle is supposed to move with constant speed.

3.3.1.3/ IMAGE-AIDED SCAN ALIGNMENT

Compared with camera, LRF has the advantage that it can directly measure the detected environment in Euclidean space. However, using only the geometric information for scan alignment might fail in some areas with few structures. Considering the photometric information provided by images (color, intensity, etc.), onboard cameras can be adopted to assist the performance of LRF scan alignment.

With corresponding laser - image points, the depth information of laser points can be given to their corresponding 2D image coordinates to avoid 3D reconstruction of the im-

age points, then 3D/2D method (in section 2.2.3.2) can be used to estimate the ego-motion of camera in the framework of visual odometry [18]. Or, the photometric attributes of image points can be given to their corresponding LRF points to augment the measured distance $e_i(p_i, q_i, T_0)$.

In order to find the corresponding image pixel of a LRF point, the extrinsic calibration between a LRF system and a camera system is required. In the following part, the extrinsic calibration method between a LRF system and a camera system is introduced ; then, two variants of image-aided ICP methods are respectively.

Extrinsic calibration between a LRF and a camera

The objective of extrinsic calibration between a LRF system and a camera system is to find their relative position and orientation. The coordinates Q^C of a 3D point Q in the camera frame can be transformed to coordinates Q^L in the LRF frame by :

$$Q^L = R_{C,L}Q^C + T_{C,L} \quad (3.58)$$

where $R_{C,L}$ is a 3×3 rotation matrix corresponding to the camera orientation with respect to the LRF system, $T_{C,L}$ is a 3-dimensional vector corresponding to the camera position in the LRF system.

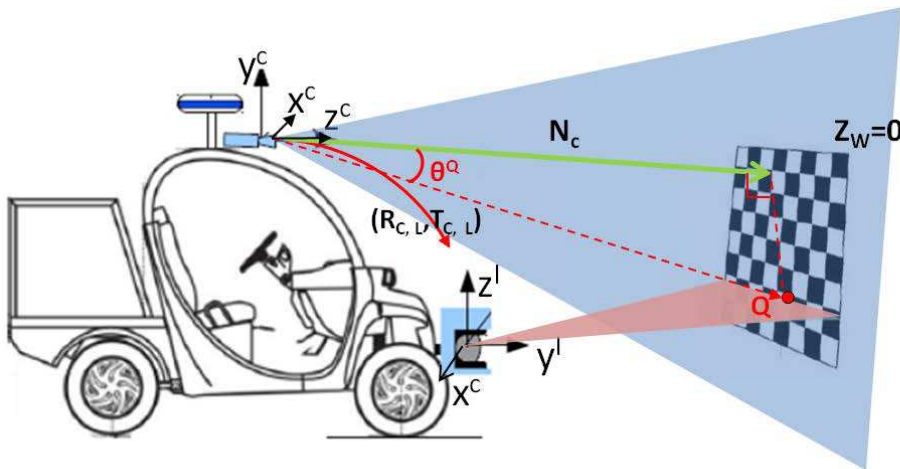


FIGURE 3.15 – A chessboard is placed in front of the camera and laser range finder, we need to find the rotation $R_{C,L}$ and translation $T_{C,L}$ to transform the point coordinates from camera frame to LRF frame

The calibration method proposed in [170] is used in our work to determine the rigid transformation from the camera system to LRF system. In this method, a chessboard (planar calibration pattern) is placed with different poses in front of the camera and LRF systems. The chessboard should be visible for both systems, as shown in Fig. 3.15. The calibration process is in three steps :

- 1) For each chessboard pose, the laser points on the chessboard are extracted from the laser scan ; meanwhile, the chessboard grid points are detected in the image.
- 2) Then, assume that the chessboard is on the plane $Z^W = 0$ in the chessboard coordinate system, the chessboard plane can be represented by a 3-dimension vector N_c in the

camera coordinate system. N_c is parallel to the normal of the chessboard plane and $\|N_c\|$ equals the distance from the camera center to the chessboard plane. This vector can be estimated by the camera extrinsic parameters in the world coordinate frame by :

$$N_c = -R_3(R_3^T T) \quad (3.59)$$

where R_3 is the third column of the camera rotation matrix R , T is the camera translation vector with respect to the world frame. The extrinsic parameters of the camera (the rotation matrix R and translation vector T) can be known by camera calibration (details can be found in section 3.2.1.1).

3) With the coordinates Q^L of a point Q in the laser coordinate system, its coordinates Q^C in the camera reference frame can be derived from Eq.3.58, as :

$$Q^C = R_{C,L}^{-1}(Q^L - T_{C,L}) \quad (3.60)$$

Since point Q is on the calibration plane defined by the vector N_c , the dot product of Q^C and N_c is :

$$N_c \cdot Q^C = \|N_c\| \|Q^C\| \cos\theta^Q = \|N_c\|^2 \quad (3.61)$$

where θ^Q is the angle between the point vector and the normal vector (see Fig. 3.15). Replacing Q^C by Eq. 3.60, this geometric constraint can be written as :

$$N_c \cdot R_{C,L}^{-1}(Q^L - T_{C,L}) = \|N_c\|^2 \quad (3.62)$$

Then, Eq. 3.62 is applied to find the relative rotation $R_{C,L}$ and translation $T_{C,L}$ between the camera and laser range finder, using both linear solution and nonlinear Levenberg-marquardt method.

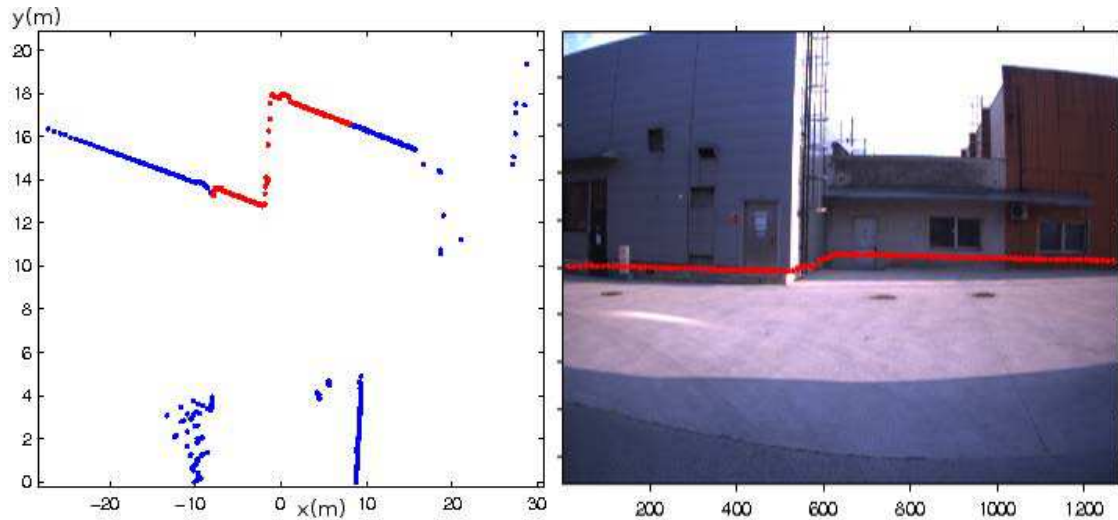


FIGURE 3.16 – One laser scan and its corresponding camera image frame. (Left) LRF scan : the points are in red if they are in the FOV of the camera, the other points are in blue ; (Right) Laser points (red) in the FOV of the camera are transformed into the image frame using LRF/camera extrinsic parameters

With the estimated extrinsic parameters between these two sensors, LRF points which are in the FOV of the camera can be transformed into the corresponding camera frame, then to the image frame with camera intrinsic parameters, as shown in Fig. 3.16. We

know then the corresponding image pixel of each LRF scan point. For a LRF scan point, we have then both its geometric information (from LRF measurement) and photometric information (from image pixel).

ICP - Color

Based on the color information provided by images, ICP can be improved by directly removing the outlier laser points or reducing the searching area of corresponding laser points with their color information. For example, [49] proposed to classify the laser points by color information before applying ICP association and apply the classic ICP with some selected sub clouds. [45] proposed to use the classic ICP at first, then apply a color distance threshold to remove the outliers in color space. [61] proposed to use the additional viewpoint-invariant attributes (e.g., visibility, reflected intensities) to reduce the search space of the ICP association.

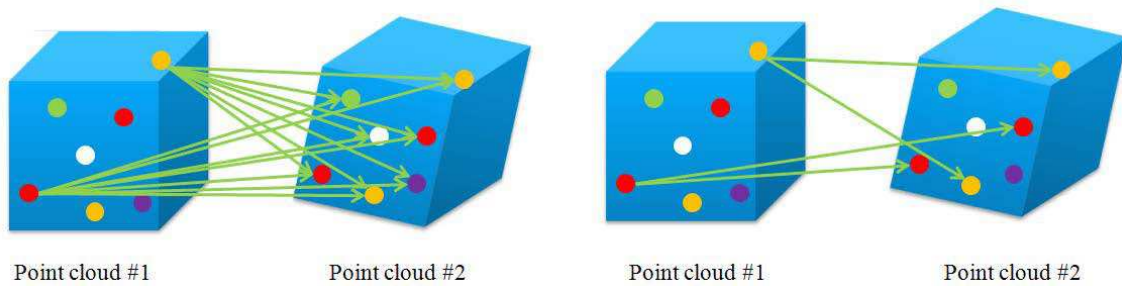


FIGURE 3.17 – ICP with corresponding image color constraint : instead of only searching for the closest point, color information of the image points is also taken into account [81]

Another approach is to use color information together with the coordinates of LRF points [80] (Fig. 3.17). Distance defined by the LRF coordinates in Eq.3.56 can be augmented by image color information through :

$$e_i = \|T_0 p_i - q_i\| + a \sqrt{a_1(c1_{p_i} - c1_{q_i})^2 + a_2(c2_{p_i} - c2_{q_i})^2 + a_3(c3_{p_i} - c3_{q_i})^2} \quad (3.63)$$

where $c1$, $c2$ and $c3$ are color components of an image pixel according to a specific color model ; a_1 , a_2 and a_3 are weights for the different color components ; and a is the weight for color component.

Implementation. In order to use color information, we need to find a proper color space to represent it. In our work, we tested several color models (RGB, HSV, YIQ, and gray scale) with different coefficients for each color model component. Since HSV color model can separate the illumination and true color value of color information⁶, HSV information of image points is added into the Euclidean distance e_i (in Eq 3.63) to measure the similarity of points in ICP data association step. In outdoor environment, since the shadow might affect the color value or the color saturation component, while the hue component which indicates the intrinsic color is not affected, the weight coefficient a_1 for hue is set to 100, a_2 for saturation and a_3 for the value (brightness) are both set to 0 to reduce the influence of brightness changes.

6. In the HSV color space, **H** stands for hue (intrinsic color such as red, yellow, green), **S** is for saturation (strength of the color), and **V** for value (or brightness)

Remarks : generally, we consider that laser points in one LRF scan are acquired instantly. But actually, the raw laser scan provided by a LRF system are acquired during a period (depending on the LRF frequency), and the camera system captures one image at one instant. If we want to integrate the LRF points and image pixel information in this way above (ICP - color), the time latency between two sensors cannot be neglected. We have tested this method with two LRF data sets of different frequencies. The first data set is only with $5Hz$ frequency, the laser points on the left part of the scan cannot be directly well associated with the image pixels, we have tried to compensate the coordinates of LRF scan points with incident angles, LRF frequency and the predicted vehicle speed, but the precision is still limited. The second LRF data set tested is with about $75Hz$ frequency, the time latency problem between the two system are much better solved than the first data set.

ICP - Descriptor

Considering the attribute information of image, the descriptor of an image point can be used to describe the point by its neighborhood, e.g., SIFT descriptor, SURF descriptor. For a laser point, the descriptor of its corresponding image point can also be used to constrain the ICP association process. As detailed in section 3.2.2.1, the SURF descriptor D of each laser-corresponding image point is extracted from the image, then added into the distance measuring function, as :

$$e_i = \|T_0 p_i - q_i\| + \alpha_D \|D_{p_i} - D_{q_i}\| \quad (3.64)$$

where D_{p_i} and D_{q_i} are respectively the SURF descriptors of image points corresponding to the LRF scan points p_i and q_i , α_D is the weight of image descriptor. We tested different values for α_D and set it to 0.05 in our work after comparing the localization results with RTK-GPS.

For example, in Fig. 3.18, LRF scans in the FOV of the camera are respectively projected onto their corresponding image planes at time $t - 1$ and t . Then, SURF descriptors with 64 dimensions are extracted from the image. Green circles in the figures represent the extracted descriptors, the green segments inside the circles indicate the orientation of the descriptor. After that, the laser points are associated by both their geometric coordinates and descriptor attributes. The corresponding laser points are connected by red lines in Fig. 3.18. Comparison of OR-ICP, ICP-Color and ICP-Descriptor for vehicle localization in long sequence will be presented in experiment section 3.4.3.

3.3.2/ ERROR MODELING OF ICP PROCESS

3.3.2.1/ UNCERTAINTY OF LASER POINTS

As shown in Fig. 3.13, the coordinates of a LRF scan point (x_i, y_i) are related to the incidence angle ϕ_i of the laser beam and the distance d_i between the emitter and the object. Therefore, the uncertainty of every laser point can be measured by the range and orientation of the laser beam. The noise of a LRF range measurement is assumed to be a zero-mean Gaussian error with variance δd_i^2 , the noise of a LRF incidence angle measurement is assumed to be a zero-mean Gaussian error with variance $\delta \phi_i^2$.

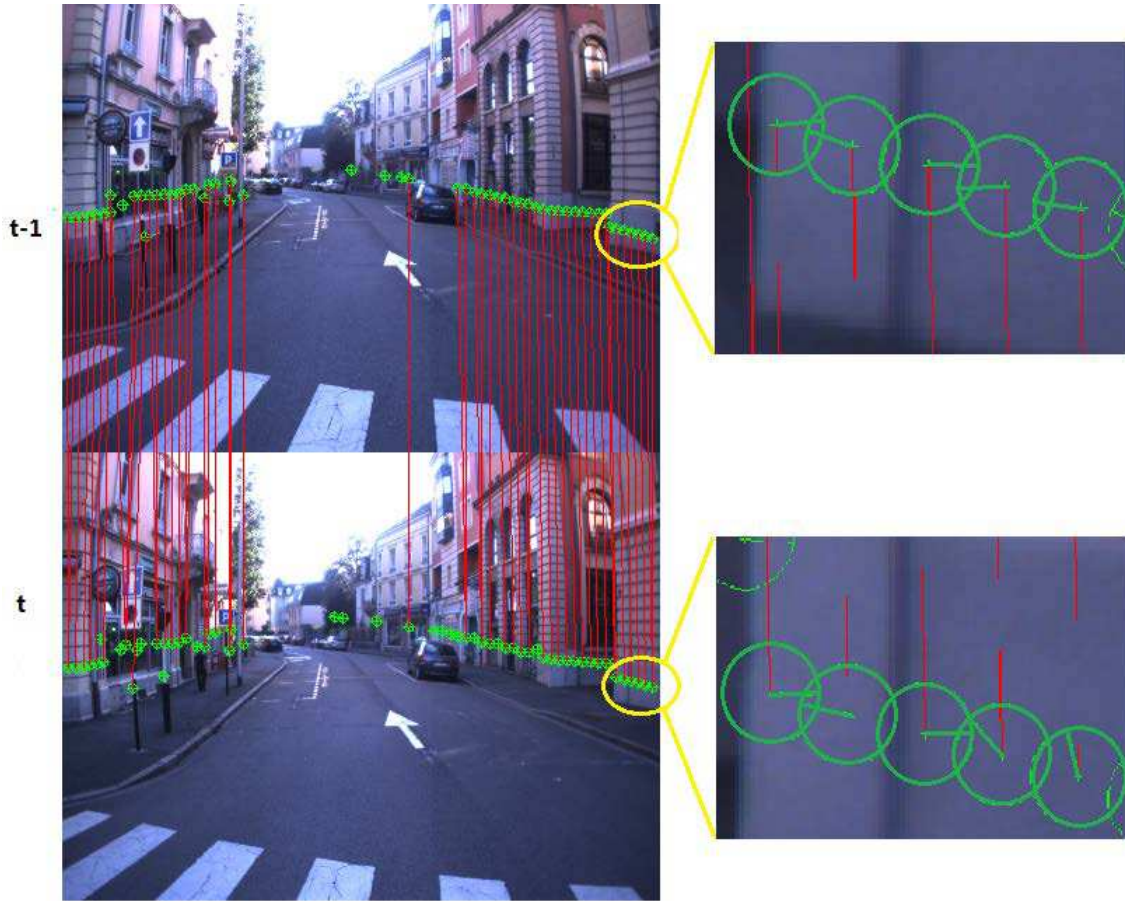


FIGURE 3.18 – LRF points in the FOV of the camera are projected onto the image ; and SURF descriptors are extracted from the image ; then, the laser points are associated by both their geometric coordinates and descriptor attributes

According to the sensor construction data, the systematic error of LMS221 is about $\pm 5\text{cm}$ for range $1\text{m} \sim 20\text{m}$ when choosing *cm-mode*, the standard deviation of range d_i is assumed to be proportional to the distance and measured by $\delta d_i = d_i/400$. The standard deviation of orientation ϕ_i is measured by $\delta\phi_i = 0.01$ rad (in the existing robotics implementations, the angle errors are usually not considered to be proportional to the angle. More detailed discussion about laser range and angle error can be found in [43]). With the variance of range and incidence angle, the covariance of a laser point $p_i = (x_i, y_i) = (d_i \cos\phi_i, d_i \sin\phi_i) = f_i(d_i, \phi_i)$ in Euclidean space can be propagated from $\delta^2 d_i$ and $\delta^2 \phi_i$ with the Jacobian matrix of function f :

$$\text{cov}(x_i, y_i) = H_p \begin{bmatrix} \delta d_i^2 & 0 \\ 0 & \delta \phi_i^2 \end{bmatrix} H_p^T \quad (3.65)$$

where H_p is the Jacobian matrix, as $H_p = \begin{bmatrix} \frac{\partial x_i}{\partial d_i} & \frac{\partial x_i}{\partial \phi_i} \\ \frac{\partial y_i}{\partial d_i} & \frac{\partial y_i}{\partial \phi_i} \end{bmatrix} = \begin{bmatrix} \cos\phi_i & -d_i \sin\phi_i \\ \sin\phi_i & d_i \cos\phi_i \end{bmatrix}$.

3.3.2.2/ UNCERTAINTY OF OR-ICP ESTIMATION

Here, covariance of the predicted vehicle pose from OR-ICP method is estimated with the covariances of the LRF points and the minimization process. If \hat{T} is the solution of a minimization process with a cost function ϵ and variable A , the covariance of \hat{T} could be estimated in closed form with the covariance of variable A [33] :

$$\hat{T} = \hat{T}(A) = \arg \min_A \epsilon(A, T) \quad (3.66)$$

$$\text{cov}(\hat{T}) = \left(\frac{\partial^2 \epsilon}{\partial T^2} \right)^{-1} \frac{\partial^2 \epsilon}{\partial T \partial A} \text{cov}(A) \left(\frac{\partial^2 \epsilon}{\partial T \partial A} \right)^T \left(\frac{\partial^2 \epsilon}{\partial T^2} \right)^{-1} \quad (3.67)$$

For OR-ICP minimization process, the solution \hat{T} is the relative translation and rotation estimated by the OR-ICP method, written as $\hat{T} = (\Delta\hat{x}, \Delta\hat{y}, \Delta\hat{\theta})$; A is the point vector composed of k couples of corresponding LRF scan points in the previous and current LRF scans : $A = ((p_1, q_1), \dots, (p_k, q_k))$, where $p_i = (x_i^p, y_i^p)$ and $q_i = (x_i^q, y_i^q)$. The error function ϵ is written as :

$$\epsilon(\text{Model}, \text{Data}, \hat{T}) = \sum_{i=1}^k \| m(\hat{T}, p_i) - q_i \|^2 = \sum_{i=1}^k \{(m_i^x)^2 + (m_i^y)^2\} \quad (3.68)$$

where k is the number of the point correspondences after OR-ICP convergence, $m(\hat{T}, p_i)$ is the transformed location of p_i with the estimated solution \hat{T} , and

$$\begin{aligned} m_i^x &= \cos(\Delta\hat{\theta})x_i^p - \sin(\Delta\hat{\theta})y_i^p + \Delta\hat{x} - x_i^q, \\ m_i^y &= \sin(\Delta\hat{\theta})x_i^p + \cos(\Delta\hat{\theta})y_i^p + \Delta\hat{y} - y_i^q, \\ n_i^x &= -\sin(\Delta\hat{\theta})x_i^p - \cos(\Delta\hat{\theta})y_i^p, \\ n_i^y &= \cos(\Delta\hat{\theta})x_i^p - \sin(\Delta\hat{\theta})y_i^p. \end{aligned}$$

The partial derivative of ϵ with respect to T at \hat{T} is :

$$\frac{\partial \epsilon}{\partial T} \Big|_{T=\hat{T}} = 2 \left[\sum_{i=1}^k (m_i^x) \quad \sum_{i=1}^k (m_i^y) \quad \sum_{i=1}^k \{m_i^x n_i^x + m_i^y n_i^y\} \right] \quad (3.69)$$

Then, the second-order partial derivative is :

$$\frac{\partial^2 \epsilon}{\partial T^2} \Big|_{T=\hat{T}} = 2 \begin{bmatrix} \sum_{i=1}^k 1 & 0 & \sum_{i=1}^k (n_i^x) \\ 0 & \sum_{i=1}^k 1 & \sum_{i=1}^k (n_i^y) \\ \sum_{i=1}^k (n_i^x) & \sum_{i=1}^k (n_i^y) & J_{33} \end{bmatrix} \quad (3.70)$$

where $J_{33} = \sum_{i=1}^k \{n_i^x n_i^x + m_i^x (-n_i^y) + n_i^y n_i^y + m_i^y n_i^x\}$. Then, the second-order mixed derivative is a $3 \times 4k$ matrix, written as :

$$\frac{\partial^2 \epsilon}{\partial T \partial A} \Big|_{T=\hat{T}} = 2 \begin{bmatrix} \epsilon_{TA}(1) & \cdots & \epsilon_{TA}(1) \\ \epsilon_{TA}(2) & \cdots & \epsilon_{TA}(2) \\ \epsilon_{TA}(3)^1 & \epsilon_{TA}(3)^i & \epsilon_{TA}(3)^k \end{bmatrix}_{3 \times 4k}, \quad i = 1, 2, \dots, k \quad (3.71)$$

where :

$$\begin{aligned} \epsilon_{TA}(1) &= [\cos(\Delta\hat{\theta}), -\sin(\Delta\hat{\theta}), -1, 0], \\ \epsilon_{TA}(2) &= [\sin(\Delta\hat{\theta}), \cos(\Delta\hat{\theta}), 0, -1], \\ \epsilon_{TA}(3)^i &= [\cos(\Delta\hat{\theta})(n_i^x) + (m_i^x)(-\sin(\Delta\hat{\theta})) + \sin(\Delta\hat{\theta})(n_i^y) + (m_i^y)(\cos(\Delta\hat{\theta})), \\ &\quad -\sin(\Delta\hat{\theta})(n_i^x) + m_i^x(-\cos(\Delta\hat{\theta})) + \cos(\Delta\hat{\theta})(n_i^y) + m_i^y(-\sin(\Delta\hat{\theta})), -n_i^x, -n_i^y]. \end{aligned}$$

Assume that all the laser points are independent from each other, the covariance matrix $\text{cov}(A)$ of k point correspondences are obtained by :

$$\text{cov}(A)_{4k \times 4k} = \text{diag}\{\text{cov}(x_1^p, y_1^p); \text{cov}(x_1^q, y_1^q); \dots; \text{cov}(x_k^p, y_k^p); \text{cov}(x_k^q, y_k^q)\}.$$

3.3.3/ CONCLUSION

In this section, several laser range finder based motion estimation methods were presented in order to provide a robust LRF based vehicle motion estimation. An outlier-rejection ICP (OR-ICP) and two image-aided ICP methods (ICP-Color, ICP-Descriptor) are proposed to reduce the matching ambiguities of scan alignment in outdoor environment. For the purpose to be integrated with other sensor measurements, the uncertainty of the alignment was also discussed. These methods are compared with real experimental data in section 3.4.3. But it should be mentioned that as the LRF system used is two dimensional and the vehicle cannot move on absolutely flat ground in real experiments, the precision of 2D LRF system based outdoor vehicle ego-motion estimation is still limited.

3.4/ IMPLEMENTATION AND EXPERIMENTAL RESULTS

3.4.1/ EXPERIMENTAL PLATFORM

The proposed method is tested with real data obtained by an experimental vehicle *SeT-Car* developed within Laboratoire Systèmes et Transports of IRTES (IRTES- SET) at UTBM⁷. *SeT-Car* is based on an electric GEM car⁸ with speed between 24km/h and 40km/h. It (Fig. 3.19) is equipped with a GPS receiver, a RTK-GPS receiver, two laser range finders and a stereoscopic system, together with an embedded hard disk and a computer system (PC) to log the acquired data and to implement autonomous navigation task.

GPS receiver. The GPS receiver used on *SeT-Car* is a ProFlex 500 Magellan RTK-GPS receiver⁹ with frequency $10Hz$. This receiver permits to use both GPS and GLONASS networks for quick position initialization. When the RTK-GPS mode is chosen, a fixed RTK-GPS base should be settled nearby to send corrections to the mobile GPS receiver by UHF radio antenna (Fig. 3.20). The RTK-GPS receiver can provide a position up to centimeter precision on horizontal plane, it is used as ground-truth to evaluate the proposed localization method. Another GPS receiver of the same type is also mounted to provide normal precision GPS positions without choosing the RTK-GPS mode.

Stereoscopic system. The stereoscopic system used in the experiment is a Bumblebee XB3 system¹⁰. It captures 16 images pairs every second, respectively with size of 1280×960 pixels. The baseline of the stereoscopic system is $0.24m$, with a FOV (Field of View) of 66° for each camera.

Laser range finder. A horizontal SICK LMS221 laser range finder is mounted on the bottom front of the vehicle. This LRF provides about 5 scans every second and each LRF scan provides 361 laser points in a 180° arc with 0.5° angular resolution and maximum range of 80 meters.

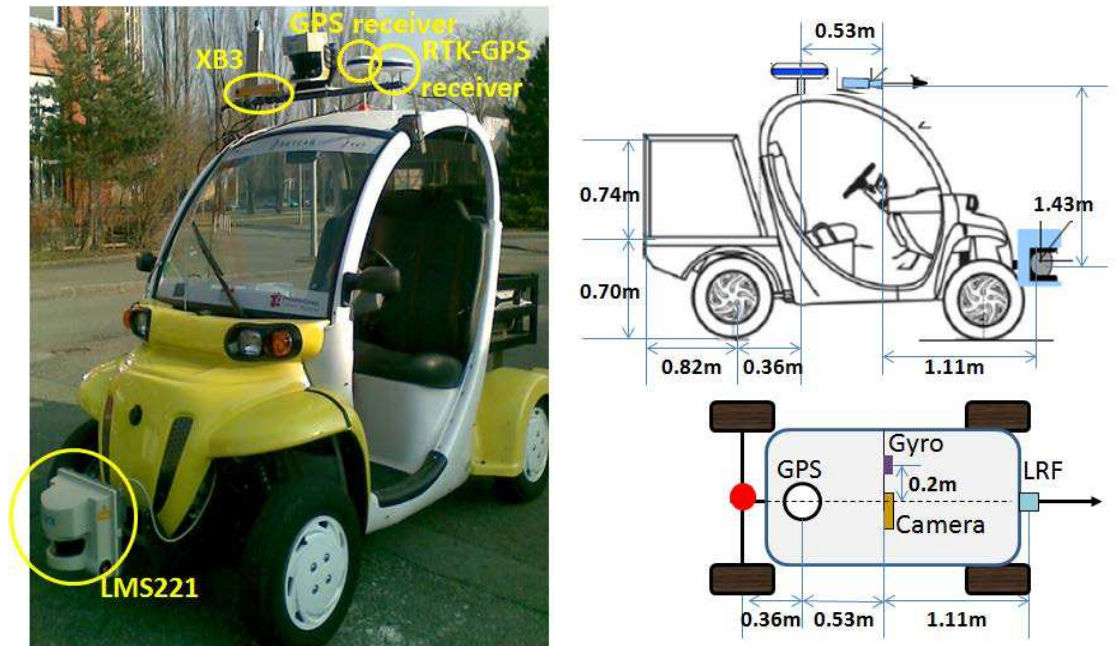
Two experimental data sets were respectively captured in March, 2011 and September,

7. <http://epan.fr/Accueil>

8. <http://www.gemcar.com/>

9. <http://www.magellangps.com/>

10. www.ptgrey.com



(a) The experimental GEM vehicle equipped with multiple sensors
 (b) Position of sensors on the GEM vehicle : side view and top view of the sensor configuration in the 2nd experiment

FIGURE 3.19 – Configuration of the experimental vehicle



FIGURE 3.20 – Fixed GPS base used for sending corrections to the mobile GPS receiver

2012 with this experimental vehicle. For the two experiments, the positions of the GPS receiver and the LRF sensor are fixed on the vehicle, while the position of the stereoscopic system was changed during the second experiment. The intrinsic and extrinsic parameters of the stereoscopic system were obtained by calibration as presented in section 3.2.1.1 and section 3.2.1.2. The relative poses between the stereoscopic system and the LRF system were obtained by chessboard based calibration as presented in previous section.

3.4.2/ EXPERIMENTAL RESULTS OF STEREOVISION ODOMETRY AND LRF ODOMETRY

For the experimental data set captured in March 2011, the vehicle was driven in an industrial area with buildings around (Fig. 3.21). Recorded GPS positions of the vehicle were transformed from WGS84 system to Extended Lambert II system which covers the experimental area (see Appendix A). The whole trajectory length measured by RTK-GPS receiver is 603.73 meters. Landmarks reconstructed from the stereovision odometry and 2D LRF scan alignment are overlaid on satellite image with the vehicle positions provided by the RTK-GPS receiver, as shown in Fig. 3.21.

In this part, we compare the localization results obtained using : the stereoscopic system (visual odometry approach described in section 3.2), and the LRF system (comparison of classic ICP and OR-ICP based scan alignment described in section 3.3).

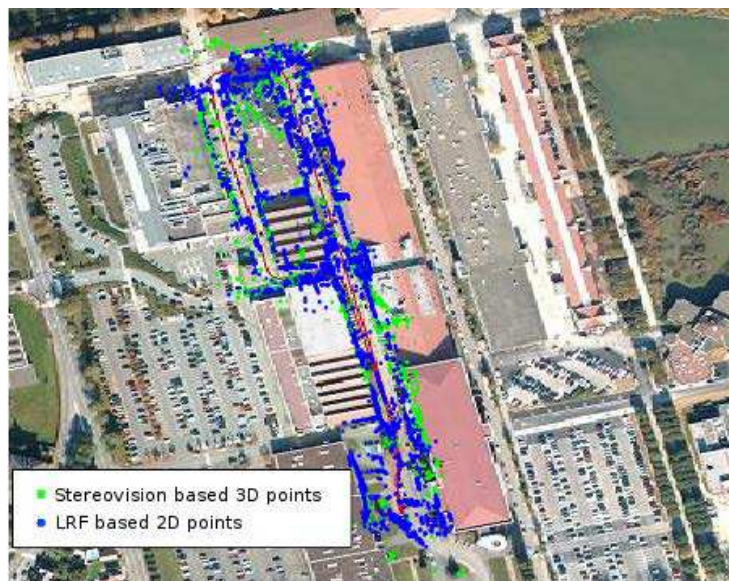
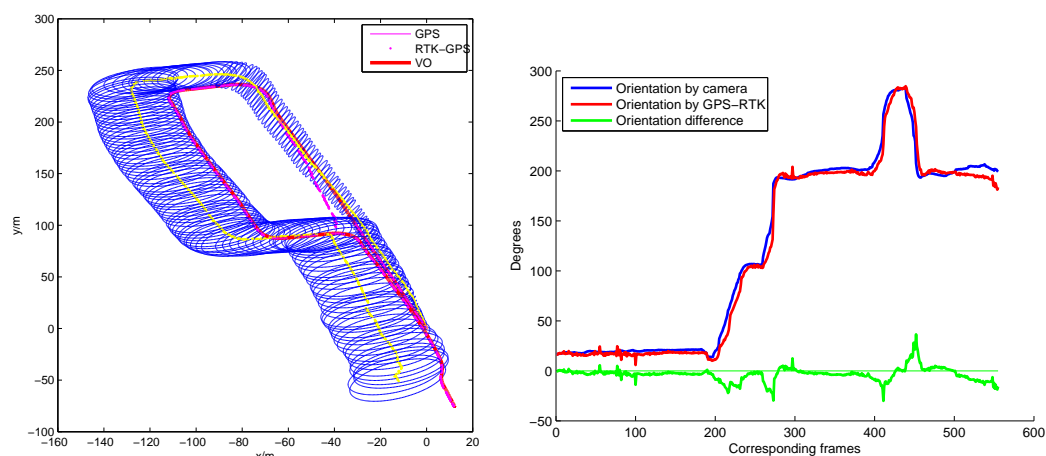


FIGURE 3.21 – Landmarks from the stereoscopic system and from the LRF are overlapped on Google aerial image

In Fig. 3.22(a), the vehicle positions are respectively estimated by the stereoscopic system with 2D visual odometry and 3D visual odometry after camera calibration and images rectification. The stereovision based vehicle yaw angles are compared with the ground truth provided by RTK-GPS receiver in Fig. 3.22(b).

It is noted that the vehicle localization error of stereovision based visual odometry is small till the second big turning of the vehicle. When the vehicle encounters sharp turns where

the movement of the vehicle is large and the illumination condition changes quickly, it is difficult to detect enough image features and apply feature matching. During these situations, the inaccurate translation and roll angle estimations lead to unreal vehicle motion, and the vehicle trajectory gradually drifts due to error accumulation.



(a) Vehicle trajectory obtained by the stereo- (b) Vehicle yaw angles obtained from the stereo-
 scopic system (95% confidence error ellipse) scopic system and from the RTK-GPS

FIGURE 3.22 – Comparison of the vehicle trajectories and yaw angles obtained from the stereoscopic system and the RTK-GPS receiver

When the ground truth is available, the covariance estimation algorithm is consistent if the predicted covariance matrix can accurately represent the estimation error within its 3σ bound. From Fig. 3.4.2, the estimated position error is mostly inside the bounds. In the future works, other factors like precision of the camera calibration and stereo calibration process might also be incorporated for measuring the VO accuracy.

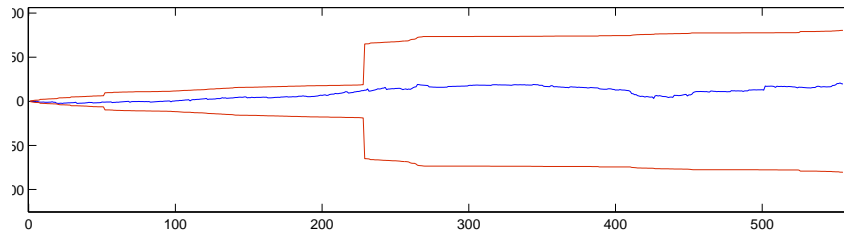
Then, the vehicle positions are respectively estimated by the classic ICP and the proposed outlier-rejection ICP (OR-ICP).

The localization results in Fig. 3.24(a) show that for this data set, the OR-ICP approach can better estimate the vehicle moving distance than the classic ICP method, and can improve the localization precision of LRF subsystem. The vehicle orientations estimated by the LRF OR-ICP are compared with the ground truth given by the RTK-GPS in Fig. 3.24(b). It is noted that from the starting point to the second turning, the vehicle orientation is almost consistent with the ground truth, but the trajectory begins to drift from the second turning. As the LRF used in this experiment is installed on the bottom front of the vehicle and is close to the ground, when the vehicle moves in the field with a certain slope, it might scan on the ground and the consecutive scans could not be associated.

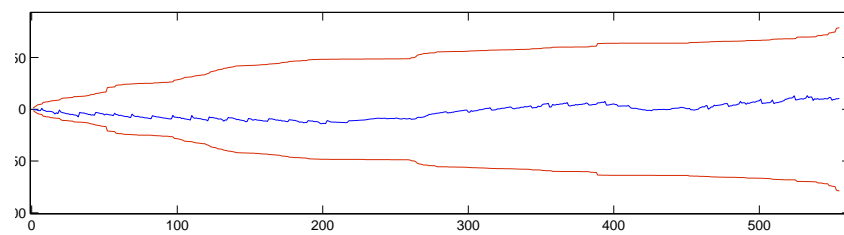
3.4.3/ COMPARISON OF DIFFERENT LRF BASED ICP METHODS

The second data sequence was acquired in the old town center of Belfort, in September, 2012. The whole trajectory measured by RTK-GPS is about 800m, as shown in Fig. 3.25.

After associating different sensor data by their logged time, the estimated vehicle translation and rotation from the visual odometry method and from the different ICP methods

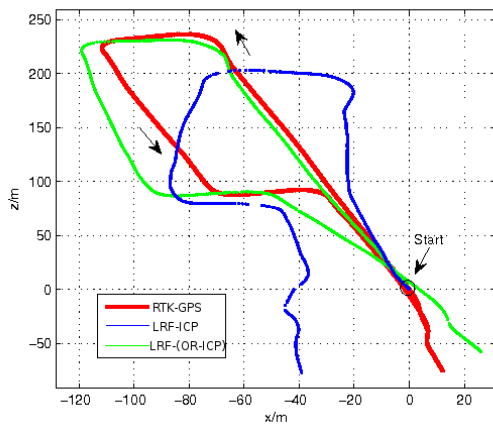


(a) Position errors on x direction with 3σ bound

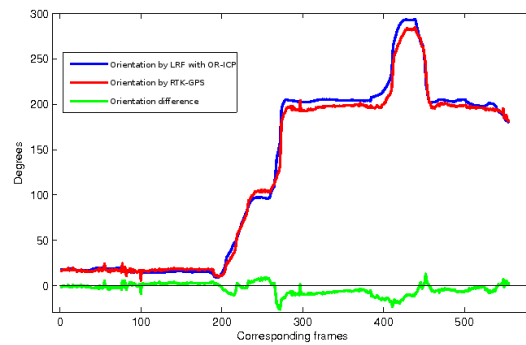


(b) Position errors on y direction with 3σ bound

FIGURE 3.23 – Position error of stereoscopic system based estimation



(a) Vehicle trajectory obtained by the LRF using classic ICP and OR-ICP algorithms



(b) Vehicle yaw angles obtained from the LRF with OR-ICP and from the RTK-GPS

FIGURE 3.24 – Comparison of the vehicle trajectories and yaw angles obtained from the LRF and the RTK-GPS receiver



FIGURE 3.25 – Vehicle trajectory overlapped on aerial image (Google)

are compared with the ground truth.

For ICP-Color and ICP-Descriptor methods, the left camera of the stereoscopic system is used together with the LRF sensor. Extrinsic parameters from the left camera to the LRF sensor are obtained by calibration before the experiment. The rotation matrix and translation vector are respectively :

$$R_{C,L} = \begin{bmatrix} 0.9980 & 0.0574 & -0.0274 \\ -0.0539 & 0.9922 & 0.1124 \\ 0.0336 & -0.1107 & 0.9933 \end{bmatrix}, T_{C,L} = \begin{bmatrix} -0.1438 & -1.4329 & -1.1094 \end{bmatrix} \text{ meter.}$$

Then, the vehicle translation and rotation parameters between every two instants are respectively estimated by different sensors : laser range finder using ICP and OR-ICP, LRF and left camera using ICP-Color and ICP-Descriptor methods, and also the stereovision based visual odometry. The vehicle trajectories are respectively obtained by the different methods and compared with the RTK-GPS trajectory, as shown in Fig. 3.26.

With the assumption that the time interval δt is small, we can obtain the average speed and angular velocity of the vehicle during each period by dividing the estimated rotation and translation by δt . The mean and standard deviation of the speed difference $(\overline{\varepsilon_v}, \sigma_{\varepsilon_v})$, mean and standard deviation of the angular velocity difference $(\overline{\varepsilon_\omega}, \sigma_{\varepsilon_\omega})$ during the whole trajectory, and the estimated trajectory length are shown in Tab 3.3. The vehicle angular velocity and speed are respectively calculated and compared with the ground truth provided by the gyro and the RTK-GPS receiver (Fig 3.27 and Fig 3.28).

From the comparison shown in Tab 3.3 and Fig 3.27, Fig. 3.28, neither the LRF nor the stereoscopic system based approaches could provide accurate absolute localization results in long term due to the error accumulation. There is no big difference between the different ICP methods for the average speed error. ICP-Color has smaller angular velocity error than the classic ICP, while ICP-Descriptor has smaller angular velocity standard deviation than the classic ICP. OR-ICP has the smallest average angular velocity error among the four ICP methods. For the total length of trajectory, ICP-Descriptor has the

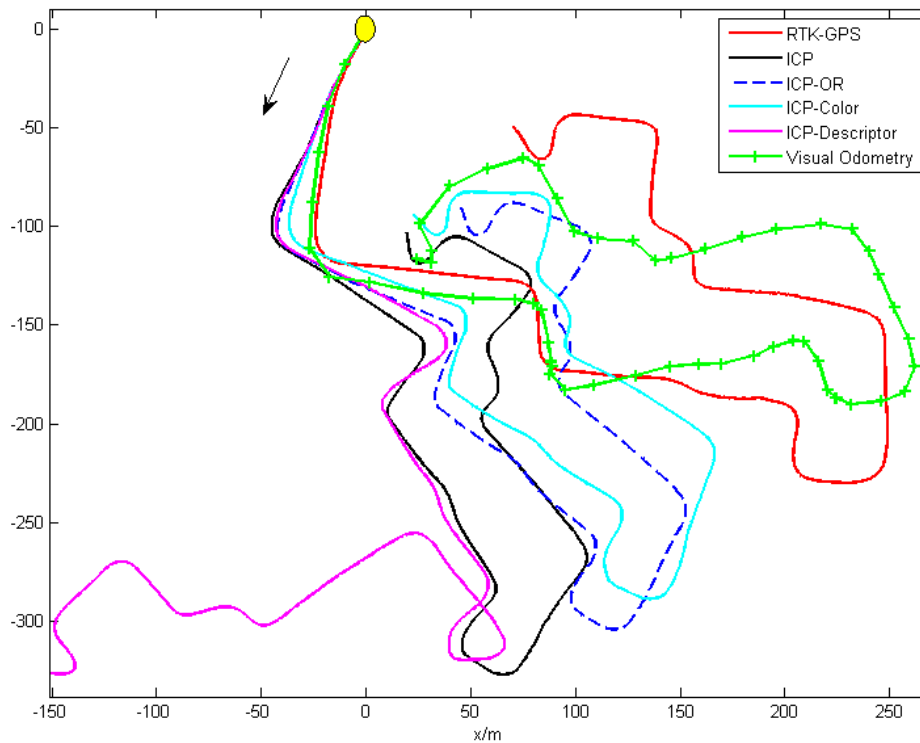


FIGURE 3.26 – Vehicle trajectories obtained by LRF with ICP, OR-ICP, ICP-Color (with HSV color model), ICP-descriptor (SURF descriptor), stereovision odometry, commercial GPS and RTK-GPS

	$\bar{\varepsilon}_v$ (m/s)	σ_{ε_v} (m/s)	$\bar{\varepsilon}_\omega$ (degree/s)	$\sigma_{\varepsilon_\omega}$ (degree/s)	Total length(m)
ICP	3.47	3.56	2.90	7.23	735.52
OR-ICP	3.47	3.56	2.80	5.97	742.30
ICP-Color	3.60	5.06	2.87	5.69	746.90
ICP-Descriptor	3.46	3.58	3.79	6.98	769.98
Visual odometry	3.90	4.00	2.21	4.31	865.35
RTK-GPS	-	-	-	-	805.00

TABLE 3.3 – Comparison of vehicle speed with RTK-GPS based ground truth, and angular velocity with gyro based ground truth

closest estimation.

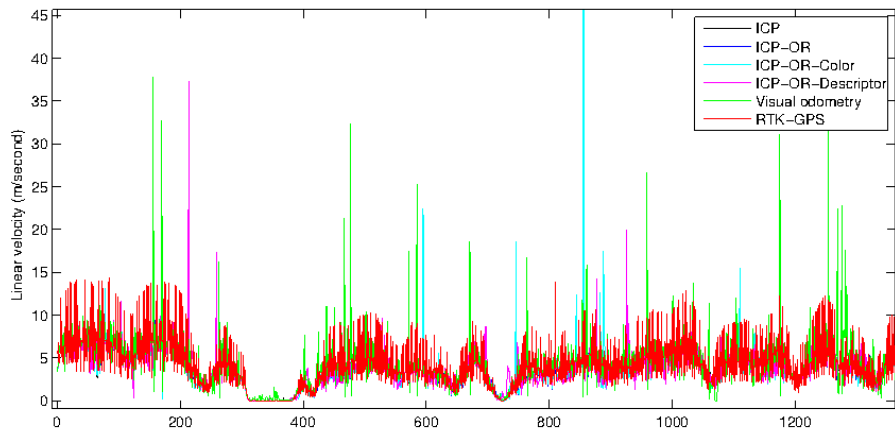
Image-aided ICP alignment methods only show small improvements over the classic ICP method, this might due to the precision of extrinsic calibration of the LRF and camera systems, to the parameters chosen to represent the weights of image color attributes in the distance measurement function, or to the limited FOV of the camera (66° in our experiment, it is probable that some important geometric or structure information in the LRF scan might be neglected).

3.5/ CONCLUSION

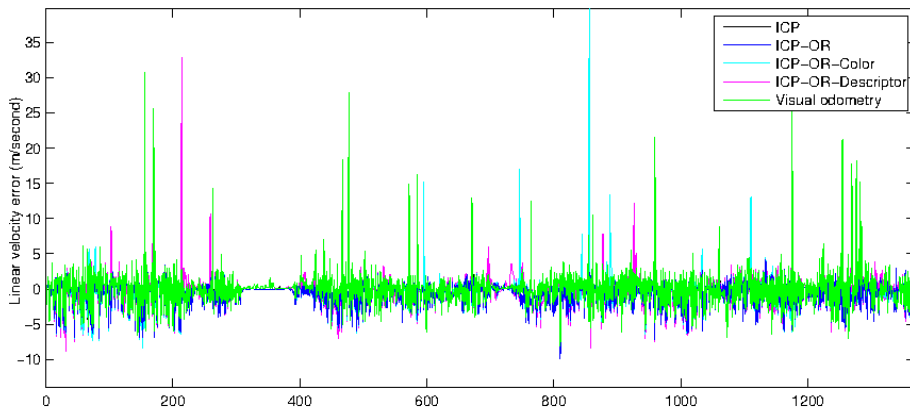
In this chapter, stereovision based visual odometry and laser range finder (LRF) based motion estimation methods are presented. Stereovision based visual odometry can estimate the vehicle motion based on corresponding image features. The error of wheel encoder odometry due to wheel slippage in bad soil conditions can be avoided by visual odometry. LRF scans based outlier-rejection ICP (OR-ICP) and two image-aided ICP methods (ICP-Color, ICP-Descriptor) are presented and compared with the classic ICP method, in order to obtain an accurate and robust LRF based vehicle motion estimation in outdoor environment. The two methods were tested with real data and evaluated by RTK-GPS as ground truth.

Several research perspectives are summarized for the localization methods in this chapter :

- The precision of visual odometry method could be improved by improving camera calibration results, or using more precise and robust features. The precision and robustness of different features can be tested with more data sequences under different illumination and weather conditions, and in more complex environments with different types of obstacles, etc. Detection of dynamic obstacles and feature detection and matching time should also be measured and improved.
- For the image aided ICP methods, the work on how to dynamically choose appropriate coefficients for the image attributes needs to be continued in the future work ; the incorporation of camera with larger FOV (e.g. fish eye) can also be considered.
- Due to the error accumulated from point to point, vehicle trajectory estimated by the relative methods might gradually drift. Therefore, apart from improving the performance of every sensor, we can take use of redundancy and complementarity of multiple sensors to provide a more accurate and robust vehicle pose estimation.

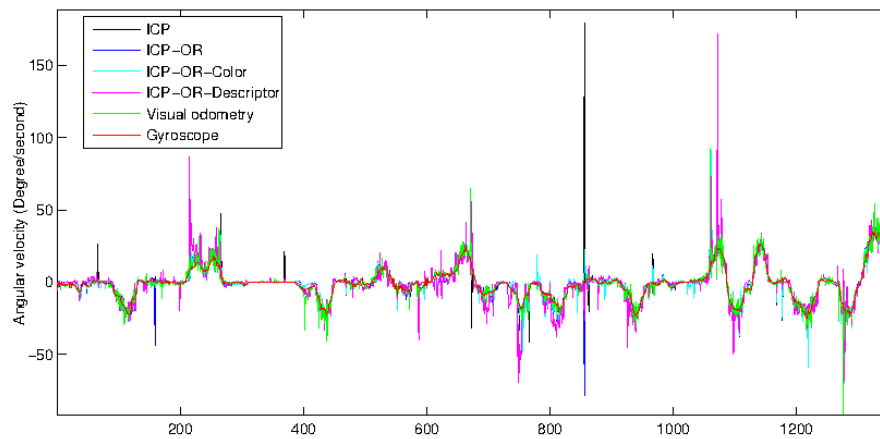


(a) Vehicle speed estimated by the different methods

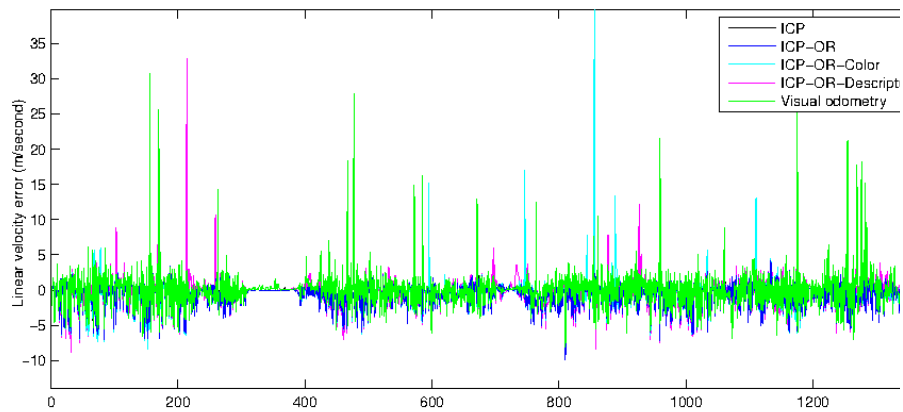


(b) Error of vehicle speed compared with RTK-GPS based ground truth

FIGURE 3.27 – Comparison of the vehicle speed with the ground truth



(a) Angular velocity estimated by the different methods



(b) Error of vehicle angular velocity compared with gyro based ground truth

FIGURE 3.28 – Comparison of the angular velocity with the ground truth

GPS-STEREOVISION-LRF BASED DATA FUSION FOR VEHICLE LOCALIZATION

4.1/ OVERVIEW

As discussed in chapter 2, GPS (Global Positioning System) has been the most used vehicle localization system in outdoor environments. If a vehicle onboard GPS receiver can receive direct signals from (at least) four well distributed satellites in the sky, it can provide current vehicle position by trilateration. Although localization errors due to atmospheric conditions and radio signal noises can be corrected by equipments like Differential GPS (DGPS), the error due to satellite visibility or multi-path problems cannot be corrected by these methods, especially in dense urban environments. Relative localization methods, like dead-reckoning sensors based wheel-encoder odometry, inertial navigation or vision based ego-motion estimation, have been applied to provide odometry and orientation information of vehicle by integration. Small errors in short term might result in unbounded error of integrated measurements.

By considering advantages and drawbacks of each localization system, apart from improving the performance of every sensor, the redundant measurement information can be used to evaluate the coherence of different systems and to continuously provide pose measurement if any system fails to work, while the complementary information can be used to provide a complete state estimation (e.g., a gyro can provide the vehicle yaw angle information and a GPS receiver can provide the absolute vehicle position).

In this chapter, the vehicle motion information estimated from stereoscopic system (see section 3.2) and laser range finder system (see section 3.3), and vehicle global position from a GPS receiver are integrated within a probabilistic framework. This chapter is organized as follows : an introduction of data fusion approaches is presented in section 4.2 ; the sensor coherence validation method is introduced in section 4.3 ; after that, vehicle state are estimated from the process model and observations from three sensors (GPS, stereoscopic system and LRF) in section 4.4 ; finally, the proposed method is tested with real experimental data in section 4.5.

4.2/ DATA FUSION APPROACHES

Based on Bayesian theory, Kalman filter and its derivations (e.g., EKF, UKF, interacting-multiple-model (IMM) system) [11][82][38] have been used for data fusion under the assumption that both the process and the observation noises obey white Gaussian distribution. The vehicle state is predicted with its previous state and a new input vector according to a vehicle process model. The observations from different sensors are used to correct this prediction by taking into account their uncertainties. These observations are supposed to be independent to each other.

Let X_t denote the state of a system at time t , s_t denote a measurement at time t , the transition model f_t of the system and the measurement model h_t can be written as :

$$\begin{aligned} X_t &= f_t(X_{t-1}, u_t) + \alpha_t \\ s_t &= h_t(X_t) + q_t \end{aligned} \quad (4.1)$$

where u_t is the input data, α_t is the noise of transition process and q_t is the noise of measurement.

In the following section, four Gaussian probabilistic based data fusion approaches are briefly introduced : Kalman filter (section 4.2.1), extended Kalman Filter (section 4.2.2), unscented Kalman Filter (section 4.2.3) and information filter (section 4.2.4). Some other filters are also presented in section 4.2.5.

4.2.1/ KALMAN FILTER

For a linear discrete-time dynamic system of which the transition model f_t in Eq. 4.1 is linear at X_{t-1} and u_t and α_t is Gaussian distributed, and the measurement model h_t is linear at X_t with a Gaussian noise q_t , Kalman filter [83] can be used to provide a closed form recursive solution for the system state. The linear transition model f_{t-1} and the measurement model h_t can be written as :

$$\begin{aligned} X_t &= F_t X_{t-1} + U_t u_t + G_t r_t \\ s_t &= H_t X_t + q_t \end{aligned} \quad (4.2)$$

where F_t is the state transition matrix, U_t is the input transition matrix, $\alpha_t = G_t r_t$, G_t is the transition model of process noise ($(n \times m)$ -dimensional noise matrix, where n is the size of the state, m is the size of the noise), r_t is the Gaussian process noise with $r_t \sim N(0, R_t)$, H_t is the measurement matrix, q_t is the measurement noise at time t with $q_t \sim N(0, Q_t)$.

At time $t = 0$, the initial system state X_0 is assumed to be Gaussian distributed with known mean X_0 and covariance P_0 , $X_0 \sim N(X_0, P_0)$. Let (X_{t-1}, P_{t-1}) the updated vehicle state and noise at time $t - 1$, the system state and covariance (X_t, P_t) at time t can be estimated by Kalman filter with a prediction step and an update step :

– **Prediction** : the predicted mean $\widehat{X}_{t|t-1}$ and covariance $\widehat{P}_{t|t-1}$ of the state at time t are :

$$\begin{aligned} \widehat{X}_{t|t-1} &= F_t X_{t-1} + U_t u_t \\ \widehat{P}_{t|t-1} &= F_t P_{t-1} F_t^T + G_t R_t G_t^T \end{aligned} \quad (4.3)$$

– **Update** : then, the innovation v_t between a measurement and the prediction is calculated by :

$$v_t = s_t - H_t \widehat{X}_{t|t-1} \quad (4.4)$$

Covariance of the innovation is :

$$S_t = H_t \widehat{P}_{t|t-1} H_t^T + Q_t \quad (4.5)$$

Kalman gain is then defined by :

$$K_t = \widehat{P}_{t|t-1} H_t^T S_t^{-1} \quad (4.6)$$

Then, the predicted system state is updated with the Kalman gain K_t and innovation v_t as :

$$\begin{aligned} X_t &= \widehat{X}_{t|t-1} + K_t v_t \\ P_t &= \widehat{P}_{t|t-1} - K_t S_t K_t^T \end{aligned} \quad (4.7)$$

4.2.2/ EXTENDED KALMAN FILTER

If the transition model f_t or the measurement model h_t is nonlinear, or α_t, q_t are not Gaussian noises, extended Kalman filter (EKF) have been proposed to estimate the state of such system. EKF extends the scope of Kalman filter to nonlinear optimal filtering problems by forming a Gaussian approximation to the joint distribution of the state X_t and the measurements s_t using a Taylor series based transformation. The first-order approximation based extended Kalman filter is as follows :

– **Prediction :**

$$\begin{aligned} \widehat{X}_{t|t-1} &= \widehat{F}_t X_{t-1} + U_t u_t \\ \widehat{P}_{t|t-1} &= \widehat{F}_t P_{t-1} \widehat{F}_t^T + G_t R_t G_t^T \end{aligned} \quad (4.8)$$

where \widehat{F}_t is the Jacobian matrix of the nonlinear transition model f_t with respect to X_{t-1} :

$$\widehat{F}_{t-1} = \left. \frac{\partial f_t}{\partial X_t} \right|_{X_t = X_{t-1}} \quad (4.9)$$

– **Update :**

the difference between a measurement observation and the predicted observation is :

$$v_t = s_t - \widehat{H}_t \widehat{X}_{t|t-1} \quad (4.10)$$

where \widehat{H}_t is the Jacobian matrix of h_t with respect to $\widehat{X}_{t|t-1}$:

$$\widehat{H}_t = \left. \frac{\partial h_t}{\partial X_t} \right|_{X_t = \widehat{X}_{t|t-1}} \quad (4.11)$$

The covariance of the innovation at time t is :

$$S_t = \widehat{H}_t \widehat{P}_{t|t-1} \widehat{H}_t^T + Q_t \quad (4.12)$$

The Kalman gain is defined by :

$$K_t = \widehat{P}_{t|t-1} \widehat{H}_t^T S_t^{-1} \quad (4.13)$$

Then, the system state is updated by :

$$\begin{aligned} X_t &= \widehat{X}_{t|t-1} + K_t v_t \\ P_t &= \widehat{P}_{t|t-1} - K_t S_t K_t^T \end{aligned} \quad (4.14)$$

4.2.3/ UNSCENTED KALMAN FILTER

In many cases, the nonlinear system cannot produce reliable results if the process are not well approximated by linear approximation, the estimation of the filter might diverge. Instead of using linear approximation of the system, unscented transform (UT) [82] chooses a fixed number of sigma points to represent the desired moment of the original distribution of X_{t-1} . The sigma points are then propagated through the non-linear function f_t and used to estimate the moment of the transformed variable X_t . The advantage of UT over Taylor series based approximation is that UT is better at capturing the higher order moments caused by the non-linear transform. Based on the unscented transform of the system state X_{t-1} at time $t - 1$, steps of unscented Kalman filter are as follows :

• **Calculate the approximation points** : for a variable X_{t-1} with dimension n , its mean and covariance at time t are augmented by mean and covariance of the process noise $r_t \sim N(0, R_t)$ through :

$$m_{t-1} = \begin{bmatrix} X_{t-1} \\ 0 \end{bmatrix}, P_{0,t-1} = \begin{bmatrix} P_{t-1} & 0 \\ 0 & R_t \end{bmatrix} \quad (4.15)$$

then, the distribution of X_{t-1} is approximated by a set of $(2n + 1)$ sigma points $X_{t-1}^{(i)}$ through :

$$\begin{aligned} X_{t-1}^{(0)} &= m_{t-1} \\ X_{t-1}^{(i)} &= m_{t-1} + [\sqrt{(n + \lambda)P_{0,t-1}}]_i, \quad i = 1, \dots, n \\ X_{t-1}^{(i)} &= m_{t-1} - [\sqrt{(n + \lambda)P_{0,t-1}}]_{i-n}, \quad i = n + 1, \dots, 2n \end{aligned} \quad (4.16)$$

where $\sqrt{(n + \lambda)P_{0,t-1}}$ could be calculated by Cholesky decomposition [82], and $[\sqrt{(n + \lambda)P_{0,t-1}}]_i$ is the i^{th} column of the matrix square root.

Associated weights $W_m^{(i)}$ of the state, and weights $W_c^{(i)}$ of the covariance are respectively estimated by :

$$\begin{aligned} W_m^{(0)} &= \lambda / (n + \lambda) \\ W_c^{(0)} &= \lambda / (n + \lambda) + (1 - \alpha^2 + \beta) \\ W_m^{(i)} &= W_c^{(i)} = 1 / (2(n + \lambda)), i = 1, \dots, 2n \end{aligned} \quad (4.17)$$

where β is a positive constant related to the distribution of the state vector, λ is a scaling factor defined by : $\lambda = \alpha^2(n + \kappa) - n$, α and κ are positive constants controlling the spread of sigma points.

• **Prediction** : the current state vector $X_{t|t-1}^{(i)}$ is predicted by propagating the sigma points $X_{t-1}^{(i)}$ through the nonlinear function f_t :

$$X_{t|t-1}^{(i)} = f_t(X_{t-1}^{(i)}, u_t), i = 0, \dots, 2n \quad (4.18)$$

The mean and covariance of prediction are :

$$\widehat{X}_{t|t-1} = \sum_{i=0}^{2n} W_m^{(i)} X_{t|t-1}^{(i)} \quad (4.19)$$

$$\widehat{P}_{t|t-1} = \sum_{i=0}^{2n} W_c^{(i)} [X_{t|t-1}^{(i)} - \widehat{X}_{t|t-1}][X_{t|t-1}^{(i)} - \widehat{X}_{t|t-1}]^T \quad (4.20)$$

• **Update** : the measurement is augmented by mean and covariance of the measurement noise $\sim N(0, Q_t)$ and approximated by a set of sigma points $\widehat{X}_{t|t-1}^{(i)}$ as in the prediction stage. Then, these sigma points are propagated to predict the measurement through :

$$y_t^{(i)} = h_t(\widehat{X}_{t|t-1}^{(i)}), i = 0, \dots, 2n \quad (4.21)$$

the predicted measurement mean \tilde{y}_t is obtained by :

$$\tilde{y}_t = \sum_{i=0}^{2n} W_m^{(i)} y_t^{(i)} \quad (4.22)$$

the covariance of the measurement is :

$$S_t = \sum_{i=0}^{2n} W_c^{(i)} [y_t^{(i)} - \tilde{y}_t][y_t^{(i)} - \tilde{y}_t]^T \quad (4.23)$$

the cross-covariance of the predicted state and the measurement is :

$$C_t = \sum_{i=0}^{2n} W_c^{(i)} [X_{t|t-1}^{(i)} - \widehat{X}_{t|t-1}][y_t^{(i)} - \tilde{y}_t]^T \quad (4.24)$$

where the associated weights $W_m^{(i)}$ for the mean and $W_c^{(i)}$ for the covariance are defined by the actual dimension of the measurement. The filter gain K_t is :

$$K_t = C_t S_t^{-1} \quad (4.25)$$

and the state and covariance of the system are updated by :

$$\begin{aligned} X_t &= \widehat{X}_{t|t-1} + K_t(s_t - \tilde{y}_t) \\ P_t &= \widehat{P}_{t|t-1} - K_t S_t K_t^T \end{aligned} \quad (4.26)$$

4.2.4/ INFORMATION FILTER BASED SENSOR FUSION

Kalman filter represents the belief of Gaussian with mean and covariance, while information filter (IF) [146] deals with the information state vector and information matrix (inverse of the covariance matrix) associated with the Fisher information. Both representations are duals of each other, and each of them can be recovered from the other by matrix inversion [156]. Since the information state is related to the underlying likelihood of the vehicle state, it is able to update the predicted state by directly integrating the information state vectors and information matrices from multiple observations.

Although the prediction stage of information filter is more complex than Kalman filter, information filter has the advantage that the update stage is computationally easier since no gain or innovation covariance matrices need to be calculated. The maximum size of the matrix to be inverted in information filter is the dimension of the state vector. When the number or size of the observations largely increases, the update stage of information filter is simpler than the usually used multi-filter approach [51]. For nonlinear systems, first-order approximation based extended information filter (EIF) [158][2][155] or unscented transformation (UT) based unscented Kalman filter (UIF) [90][6] can be used.

4.2.4.1/ INFORMATION FILTER FOR LINEAR PROCESS MODEL

For a linear state transition model $X_t = F_t X_{t-1} + U_t u_t + G_t r_t$ with the previous information vector i_{t-1} and information matrix I_{t-1} , the current information vector can be directly predicted.

• **Information state/matrix prediction.** The current information vector \tilde{i}_t and the information matrix \tilde{I}_t are predicted by :

$$\begin{aligned} A_t &= (F_t^{-1})^T (I_{t-1}) F_t^{-1} \\ E_t &= G_t^T A_t G_t + R_t^{-1} \\ B_t &= A_t G_t E_t^{-1} \\ \tilde{I}_t &= A_t - B_t E_t B_t^T \\ \tilde{i}_t &= [I - B_t G_t^T] (F_t^{-1})^T (i_{t-1}) + (\tilde{I}_t) U_t u_t \end{aligned} \quad (4.27)$$

The predicted vehicle state $\widehat{X}_{t|t-1}$ and covariance $\widehat{P}_{t|t-1}$ can be derived from the information vector and information matrix by :

$$\widehat{X}_{t|t-1} = (\tilde{I}_t)^{-1} \tilde{i}_t, \quad \widehat{P}_{t|t-1} = (\tilde{I}_t)^{-1} \quad (4.28)$$

• **Information state/matrix update.** For a sensor measurement with a linear measurement model $s_t = H_t X_t + q_t = \tilde{s}_t + q_t$, its information state vector i_t and information matrix I_t at instant t can be obtained by :

$$i_t = H_t^T Q_t^{-1} s_t, \quad I_t = H_t^T Q_t^{-1} H_t \quad (4.29)$$

where H_t is the linear observation model matrix, Q_t is the observation uncertainty. The information state vector and information matrix can be obtained by combination of the prediction and the contribution from the observation :

$$i_t = \tilde{i}_t + i_t, \quad I_t = \tilde{I}_t + I_t \quad (4.30)$$

The vehicle state X_t and covariance P_t can be derived from the information vector and information matrix through :

$$X_t = i_t (I_t)^{-1}, \quad P_t = (I_t)^{-1} \quad (4.31)$$

4.2.4.2/ UNSCENTED INFORMATION FILTER FOR NONLINEAR PROCESS MODEL

For a nonlinear system, instead of approximating the process and observation models by Taylor series, unscented transform is embedded to approximate the prediction and observation by a set of sigma points.

• **Information state/matrix prediction.** Unscented transform chooses a fixed number of sigma points from the original distribution of the state, propagates the sigma points through the non-linear process model, and estimates the mean and covariance of the current state on the basis of the sigma points (Eq.4.15 to Eq.4.20). Then, the predicted information state vector \tilde{i}_t and information matrix \tilde{I}_t can be recovered from the predicted mean and covariance of the state [51] :

$$\tilde{I}_t = (\widehat{P}_{t|t-1})^{-1}, \quad \tilde{i}_t = (\tilde{I}_t) \widehat{X}_{t|t-1} \quad (4.32)$$

• **Information state/matrix update.** The observation vector can be approximated by the predicted sigma points $X_{t|t-1}^{(i)}$ with error propagation. Then, the observation vector is used to approximate the measurement \tilde{y}_t . The information matrix I_t in Eq.4.29 can be written as :

$$I_t = H_t^T Q_t^{-1} H_t = (\widehat{P}_{t|t-1})^{-1} \widehat{P}_{t|t-1} H_t^T Q_t^{-1} H_t (\widehat{P}_{t|t-1})^T (\widehat{P}_{t|t-1}^T)^{-1} \quad (4.33)$$

Replacing $\widehat{P}_{t|t-1} H_t^T$ by C_t , the information matrix becomes :

$$I_t = \widehat{P}_{t|t-1}^{-1} C_t Q_t^{-1} C_t^T (\widehat{P}_{t|t-1}^{-1})^T \quad (4.34)$$

where C_t is the cross covariance of the state vector and the measurement vector. It is approximated by :

$$C_t = \sum_{i=0}^{2n} W_c^{(i)} [X_{t|t-1}^{(i)} - \tilde{X}_{t|t-1}] [y_t^{(i)} - \tilde{y}_t]^T \quad (4.35)$$

Then, the information state vector is written as :

$$\begin{aligned} i_t &= H_t^T Q_t^{-1} s_t \\ &= H_t^T Q_t^{-1} [v_t + H_t \widehat{X}_{t|t-1}] \\ &= \widehat{P}_{t|t-1}^{-1} \widehat{P}_{t|t-1} H_t^T Q_t^{-1} [v_t + H_t \widehat{P}_{t|t-1}^T (\widehat{P}_{t|t-1}^T)^{-1} \widehat{X}_{t|t-1}] \\ &= \widehat{P}_{t|t-1}^{-1} C_t Q_t^{-1} [v_t + C_t^T (\widehat{P}_{t|t-1}^T)^{-1} \widehat{X}_{t|t-1}] \end{aligned} \quad (4.36)$$

where s_t is the observation, v_t is the innovation vector. A pseudo-measurement matrix \widetilde{H}_t can be defined as [90] :

$$\widetilde{H}_t \equiv (\widehat{P}_{t|t-1}^{-1} C_t)^T \quad (4.37)$$

The information contribution i_t in Eq.4.36 and I_t in Eq.4.34, can be respectively expressed by :

$$i_t = \widetilde{H}_t^T Q_t^{-1} [v_t + \widetilde{H}_t \widehat{X}_{t|t-1}], \quad I_t = \widetilde{H}_t^T Q_t^{-1} \widetilde{H}_t \quad (4.38)$$

• **With N observation sensors.** Local information from different sensors are directly combined to obtain the information state vector i_t and the information matrix I_t of the system :

$$i_t = \tilde{i}_t + \left(\sum_{sen=1}^N i_t^{sen} \right), \quad I_t = \tilde{I}_t + \left(\sum_{sen=1}^N I_t^{sen} \right) \quad (4.39)$$

where i_t^{sen} and I_t^{sen} are respectively the information state and information matrix of the $(sen)^{th}$ sensor.

The vehicle state X_t and covariance P_t can be derived from the information vector and information matrix through : $X_t = i_t (I_t)^{-1}$, $P_t = (I_t)^{-1}$.

4.2.5/ OTHER FUSION METHODS FOR VEHICLE LOCALIZATION

Besides the fusion methods presented above, a large number of methods also exist in the literature especially for handling the non-Gaussian distributed noises, such as :

• Particle filter [46][65][129]. Instead of linearizing and representing the noise distribution by an exponential function (Gaussian noise), particle filter approximates the prior distribution of the state in state space by a set of random state samples, called particles, denoted as $\{\tilde{x}_t^i, i = 1, \dots, M$ [156]. Each particle has an importance factor $\{w_t^i\}$ to

incorporate the measurement s_t into the particle set. These weighted particles are independently propagated through the transition model to represent the posterior of x_t , then replaced in an importance re-sampling process with their important factors $\{w_t^i\}$. Particles in low-probability regions will be filtered out gradually.

- Interval method. An alternative solution to estimate the state of dynamic processes is the set-membership estimation approach [86][104][106]. In this method, the inaccuracy of the measurements is not expressed by mean and covariance, but in terms of bounds on the possible errors.

4.2.6/ CONCLUSION

The classic Kalman filter permits to predict and update the system state of a linear system. When the transition model or the measurement model are nonlinear, EKF forms a Gaussian approximation to the joint distribution of the state and the measurement using a Taylor series based transformation. However, the calculation of Jacobian matrix might be difficult or not accurate for approximating the system. Unscented transform (UT) chooses a fixed number of sigma points to represent the desired moments of the original distribution of state. The advantage of UT over the first-order Taylor series based approximation is that UT is better for capturing the higher order moments caused by the non-linear transform. When more than two measurements are provided, information filter has the advantage that the update stage is computationally easy : the inverse-covariance form of the information filter is able to update the prediction by directly integrating the information state vectors and the information matrices from multiple observations. Therefore, in the following section, UT is chosen to approximate the nonlinear system, and information filter is chosen to integrate the observation from different sensors.

4.3/ SENSOR COHERENCE VALIDATION BY EXTENDED NIS

During the multi-sensor fusion process, measurements might be contaminated by disturbances. Erroneous measurements would result in unreliable data. Therefore, measuring the coherence of different localization measurements is important for reliable location demands. The coherence and integrity of multiple sensor measurements are validated for checking and removing the biased measurements [101].

4.3.1/ NORMALIZED INNOVATION SQUARED (NIS)

- **Chi-squared distribution** : if x_1, x_2, \dots, x_k are k independent standard normal random variables, the sum of their squares $d_x = \sum_{i=1}^k x_i^2$ is supposed to be Chi-squared distributed with k degrees of freedom, denoted as $d_x \sim \chi^2(k)$. For a k -dimensional Gaussian random vector s with mean \bar{s} and covariance P_s , $s \sim N(\bar{s}, P_s)$, the variable :

$$d_s = (s - \bar{s})^T P_s^{-1} (s - \bar{s}) \quad (4.40)$$

can be considered as the squared sum of k Gaussian random variables with mean 0 and variance 1, thus the variable d_s is considered to be Chi-squared distributed with k degrees of freedom.

• **Normalized innovation squared (NIS) test** : based on Chi-squared distribution, the normalized estimation error squared test (NEES) and the normalized innovation squared test (NIS) [10] are popular for testing the consistency of two statistical distributions. When a sensor measurement is available, as the true state is not known, the innovation between the observation data and predicted state is supposed to be Gaussian distributed with covariance P_v . Let v_t denote the difference between the observed measurement s_t and the predicted state \tilde{s}_t at time t :

$$v_t = s_t - \tilde{s}_t \quad (4.41)$$

The covariance matrix of v_t is :

$$P_v = H_t(\tilde{P}_{|t-1})^{-1}H_t^{-1} + Q_t \quad (4.42)$$

The normalized innovation squared (NIS) d_M between the predicted state \tilde{s}_t and the measurement is :

$$d_M = v_t^T(P_v)^{-1}v_t \quad (4.43)$$

d_M follows a Chi-squared distribution $d_M \sim \chi^2(m)$, where m is the dimension of the measurement vector (degrees of freedom). A measurement will be rejected if d_M is outside (greater than a threshold) the confidence region defined by the χ^2 table. Vice versa, if d_M is smaller than the threshold, this measurement is considered to be reliable and coherent with the prediction.

4.3.2/ EXTENDED NIS FOR MULTIPLE MEASUREMENTS VALIDATION

When multiple sensors are used to provide the measurements (e.g., GPS, wheel encoder, gyro), besides the coherence between each sensor measurement and the process prediction, the coherence between different measurements also needs to be validated in case the process model prediction is not correct. Therefore, an extended NIS measurements validation method is used to verify the coherence of the sensors.

For N sensors $\{S^i\}, i = 1, \dots, N$ respectively with measurement s_t^i and covariance Q_t^i , a set of parity relations are calculated for every two sensors. This method is under the assumption that only one fault happens at a time and the different sensor measurements are uncorrelated with each other. The fault in any one of the sensors will cause a unique subset of these relations to increase ; then, the fault sensor can be detected [97].

As shown in Fig.4.1, d_{ij} is the Mahalanobis distance between sensors S^i and S^j for selected measurement elements at instant t (by Eq.4.44 to Eq.4.46). In each column d_{ij} , number **1** means the Mahalanobis distance d_{ij} is corresponding to sensor i and j , while **0** in the table means d_{ij} is not correlated with these sensors. For example, in (row S^1 , column d_{12}) and (row S^2 , column d_{12}), **1** means that d_{12} is correlated with sensors S^1 and S^2 . Let v_t^{ij} denote the difference between the measurements s_t^i and s_t^j at time t :

$$v_t^{ij} = f_i s_t^i - f_j s_t^j \quad (4.44)$$

where the matrices f_i and f_j are defined by the two sensor measurements to find their observation for the same element. For example, if S^i is a gyro sensor, it can provide an orientation measurement $s_t^{gyro} = [\theta_t]$ at instant t ; if S^j is an inertial navigation sensor, it can provide a measurement $s_t^{ins} = [x_t, y_t, \theta_t]^T$. The two sensors provide redundant information for the vehicle orientation, thus the distance d_{ij} is calculated based on this orientation

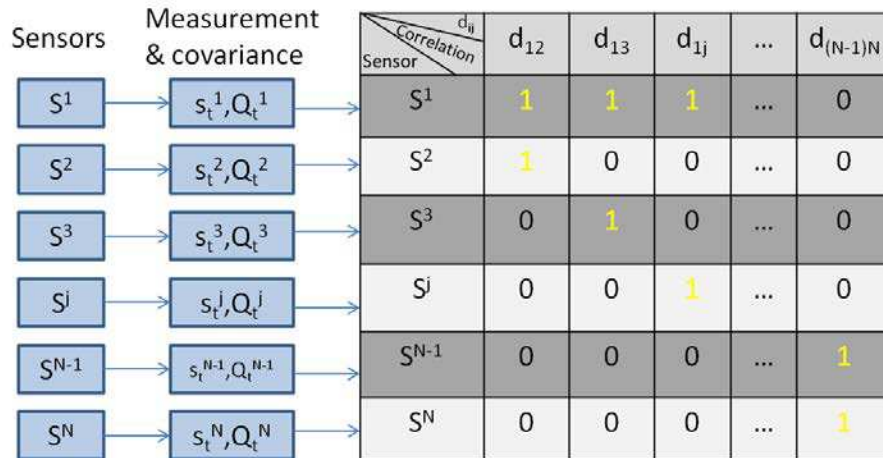


FIGURE 4.1 – Extended NIS between multiple sensors

element θ . By setting $f_i = [1]$ and $f_j = [0 \ 0 \ 1]$, the covariance matrix P_v^t of v_t^{ij} and the distance d_{ij} are written as :

$$P_v^t = f_i Q_t^i f_i^T + f_j Q_t^j f_j^T \quad (4.45)$$

$$d_{ij} = (v_t^{ij})^T (P_v^t)^{-1} (v_t^{ij}) \quad (4.46)$$

Then, a parity checking step is applied to detect sensor fault : the change of a special subset of $\{d_{ij}\}$ should be aroused by a sensor fault. For example, if the distances d_{12}, d_{1j}, \dots and d_{1N} increase simultaneously, the fault should correspond to the first sensor ; else if $d_{21}, d_{2,j}, \dots$ and d_{2N} increase simultaneously, the fault sensor is the second one, etc. If two sensors do not provide any redundant information, no measuring distance is calculated. After the validation process, the information contributions of the sensors which are consistent with each other, are integrated together to provide a final estimation.

4.4/ INTEGRATION OF GPS-STEREOVISION-LRF FOR VEHICLE LOCALIZATION

In this work, a GPS receiver, a stereoscopic system and a LRF are integrated for vehicle localization. Stereovision based visual odometry can be integrated with GPS measurements by using direct registration method, EKF or UKF (as presented in section 4.2.2 and section 4.2.3). The rotation angle and translation information provided by the visual odometry method is used to predict the vehicle state. GPS positions are used as measurements to update the state prediction. If the GPS signals could not be received or the GPS measured position is only available with low precision, the vision based method is used alone.

However, it is difficult to decide which sensor is correct if only two sensors are used in the system. In this section, the vehicle state information provided by a GPS receiver, an on-board stereoscopic system, and an on-board horizontal laser range finder are integrated for vehicle localization with an unscented information filter (UIF). Other localization information, such as wheel encoder, or gyro can also be used.

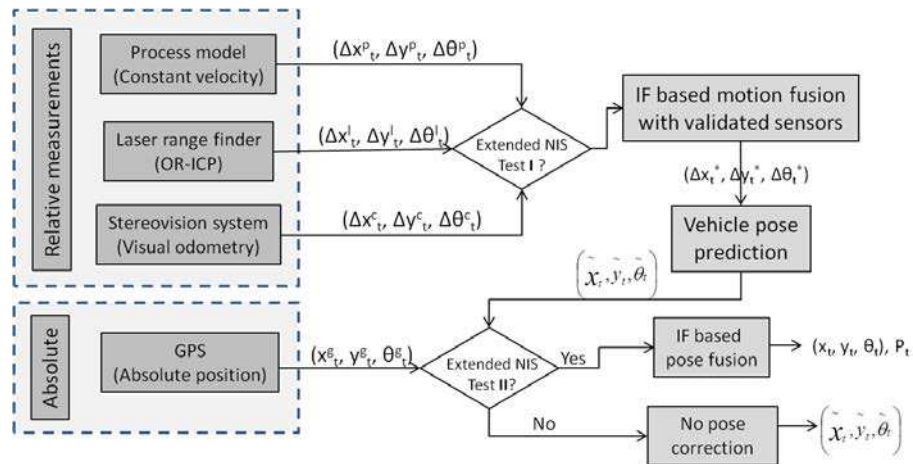


FIGURE 4.2 – Overview of the proposed UIF based vehicle localization method

As shown in Fig. 4.2, a constant-speed process model is considered as a “virtual sensor” and used to predict the vehicle motion at first ; the vehicle motion data estimated by the LRF system and the stereoscopic system are validated to provide a vehicle motion ; after that, this motion vector is used to predict the vehicle state ; finally, GPS data is used as measurement to update the prediction if it can pass the NIS checking test.

4.4.1/ COORDINATE FRAMES FOR VEHICLE LOCALIZATION

For a multi-sensor based vehicle localization system, there are three types of coordinate frames : global reference frame, vehicle body frame, and sensor frames. In our system, the frames are defined as shown in Fig.4.3.

-Global reference frame. The global system is denoted as $R_W(W_0, X_W, Y_W, Z_W)$. We take the initial vehicle position W_0 as the origin of the global system, and the initial forward orientation Y_W of the vehicle as the positive direction, the axis Z_W is pointing upward, X_W direction obeys the right hand rule. At time t , the vehicle position is represented by (x_t, y_t, z_t) in the world frame W with heading orientation θ_t .

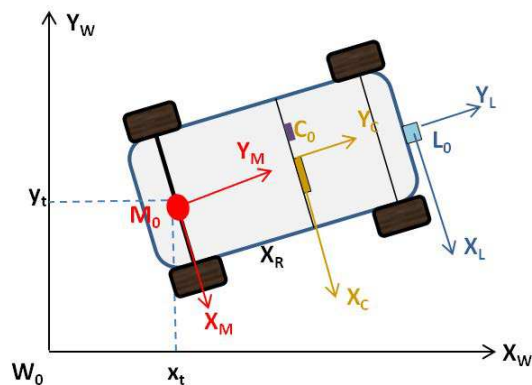


FIGURE 4.3 – Different coordinate frames of the vehicle system

-Vehicle body frame. Origin M_0 of the mobile frame M is attached to the center of the rear

axle of vehicle, as shown in Fig. 4.3, and denoted as (M_0, X_M, Y_M, Z_M) . Y_M is determined by the moving direction of the vehicle, Z_M is pointing upward and X_M direction obeys the right hand rule.

-Sensor coordinate frames. There are three sensors on the experimental vehicle. The GPS receiver and camera system are mounted on the roof of the vehicle, the LRF is mounted in front of the vehicle (see Fig. 3.15). The sensor frames are :

1. **GPS frame R_G .** The outputs of GPS receiver are vehicle longitude and latitude in the World Geodetic System (WGS84). In order to be integrated with other measurements, GPS longitude and latitude information are converted from the WGS84 system to local Cartesian space (e.g., Lambert II). Origin of the GPS frame is a fixed local point. The plane $X_G - Y_G$ is parallel to the local earth surface. Details of GPS coordinates transformation can be found in Appendix A.
2. **Stereoscopic system frame R_C .** Coordinate frame of the (rectified) stereoscopic system is defined by the left camera. Its origin C_0 is at the projection center of the left camera, X_C axis points to the direction of the right camera center (parallel to the baseline), Z_C axis points upwards and Y_C axis points along the camera optical axis.
3. **Laser range finder frame R_L .** The origin L_0 of LRF frame is at the center of the LRF sensor, the plane $X_L - Y_L$ is on the laser scanning plane.

-Position coordination and time synchronization. In order to obtain the vehicle motion by different sensors, the relative position and orientation between sensor frames and the vehicle body frame should be known before the experiments. In our work, during the procedure of vehicle position fusion, this transformation is not taken into account since all the sensors are mounted close to the central line of the vehicle. However, for higher-level sensor fusion (e.g., image-aided ICP methods in section 3.3.1.3), accurate extrinsic calibration between the different sensor systems is essential. Furthermore, as the frequencies of different sensors are not the same, the measurements from different sensors are synchronized according to their logged time into the system. Then, there are two strategies available for information fusion :

- 1) Fusion when at least one sensor observation is available. In this case, the vehicle state information is updated when at least one measurement is provided ;
- 2) Fusion when all the sensors' observations are available. In this case, when the slowest sensor provides an observation, the vehicle state information can be updated with all the sensor measurements.

In order to have more redundant information and to check the coherence between different measurements, the second fusion strategy is applied in the following work.

4.4.2/ VEHICLE MOTION PREDICTION

Assume that the ground is flat, the vehicle pose can be represented by $x - y$ coordinates and yaw angle θ . Let $X_t = [x_t, y_t, \theta_t]$ denote the vehicle state vector at time t , where (x_t, y_t) and θ_t are respectively the vehicle position and orientation in the reference navigation system. The current vehicle pose can be predicted by a nonlinear transition model. A simple kinematic model of the vehicle has two rear wheels, and two front wheels represented by a single point in the center of the front wheel axle. This transition vehicle model is given

by $X_t = f(X_{t-1}, \delta t) + \alpha_t$, and written as :

$$\begin{aligned} x_t &= x_{t-1} + \Delta d_t \cos(\theta_{t-1} + \Delta\theta_t/2) + \alpha_{1t} \\ y_t &= y_{t-1} + \Delta d_t \sin(\theta_{t-1} + \Delta\theta_t/2) + \alpha_{2t} \\ \theta_t &= \theta_{t-1} + \Delta\theta_t + \alpha_{3t} \end{aligned} \quad (4.47)$$

where $X_{t-1} = [x_{t-1}, y_{t-1}, \theta_{t-1}]^T$ is the vehicle state at time $t-1$, Δd_t and $\Delta\theta_t$ are respectively the vehicle movement and rotation from time $t-1$ to t , and $\alpha_t = [\alpha_{1t}, \alpha_{2t}, \alpha_{3t}]^T$ is the process noise.

For a constant speed model, $v_{x,t-1}$, $v_{z,t-1}$ and ω_{t-1} are respectively vehicle linear velocities on x and z directions and vehicle angular velocity at time $t-1$, calculated by the previous vehicle positions and headings at time $t-1$ and $t-2$. The previous vehicle speed is used to predict the current state with time interval δt :

$$\Delta d_t = \sqrt{v_{x,t-1}^2 + v_{z,t-1}^2} \delta t, \quad \Delta\theta_t = \omega_{t-1} \delta t \quad (4.48)$$

Thus, the vehicle relative movement from $t-1$ to t is denoted as :

$$s_t^p = \{\Delta d_t \cos(\Delta\theta_t), \Delta d_t \sin(\Delta\theta_t), \Delta\theta_t\} = \{\Delta x_t^p, \Delta y_t^p, \Delta\theta_t^p\} \quad (4.49)$$

Then, the information vector $i\tilde{f}_t$ and information matrix $I\tilde{F}_t$ are predicted according to Eq.4.32.

4.4.3/ SUBSYSTEM ESTIMATIONS

- Laser range finder subsystem based observation : as shown in section 3.3.1.2, the direct output of LRF alignment at time t is the relative vehicle motion :

$$s_t^l = \hat{T} = [\Delta\hat{x}, \Delta\hat{y}, \Delta\hat{\theta}]^T = [\Delta x_t^l, \Delta y_t^l, \Delta\theta_t^l]^T \quad (4.50)$$

and the motion covariance matrix $cov(\hat{T})$ (section 3.3.2).

The information contribution i_t^l and I_t^l of the LRF system can be calculated using Eq.4.29 :

$$i_t^l = (H_t^l)^T (Q_t^l)^{-1} s_t^l \text{ and } I_t^l = (H_t^l)^T (Q_t^l)^{-1} H_t^l, \text{ with } H_t^l = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

- Stereoscopic subsystem based observation : as described in section 3.2.2, the output of stereovision based odometry at time t is the relative vehicle motion with respect to the previous vehicle local frame :

$$s_t^c = [\Delta x_t^c, \Delta y_t^c, \Delta\theta_t^c]^T \quad (4.51)$$

and the motion covariance matrix P_t (section 3.2.3.3).

The information contribution i_t^c and I_t^c of the stereoscopic sensor can be calculated using Eq.4.29 :

$$i_t^c = (H_t^c)^T (Q_t^c)^{-1} s_t^c \text{ and } I_t^c = (H_t^c)^T (Q_t^c)^{-1} H_t^c, \text{ with } H_t^c = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

• GPS subsystem based observation : position provided by the GPS receiver is in latitude and longitude (ϕ, λ) , the coordinates are converted to local Cartesian coordinates (x_t^g, y_t^g) through Algorithm 5 in Appendix A. If the vehicle moves only with rotation, the movement of the vehicle body could not be detected by GPS sensor. But when the vehicle moves with translation, the change of the vehicle orientation could be obtained by two consecutive GPS positions through $\theta_t^g = \arctan(\frac{x_t^g - x_{t-1}^g}{y_t^g - y_{t-1}^g})$. Thus, the GPS observation is written in the form :

$$s_t^g = [x_t^g, y_t^g, \theta_t^g]^T \quad (4.52)$$

In this work, we use the NMEA GST sentence from roving GPS receivers to represent the confidence level of a GPS position (though it might not be reliable in some special cases [14]). The covariance matrix Q_t^g of the GPS position is estimated with GPS NMEA sentence "GST" by :

$$Q_t^g = \begin{pmatrix} \delta_x^2 & \rho\delta_x\delta_y \\ \rho\delta_x\delta_y & \delta_y^2 \end{pmatrix} \quad (4.53)$$

where δ_x and δ_y are the standard deviations of the longitude error and the latitude error provided by the GPS NMEA sentence "GST", ρ is the spatial correlation coefficient calculated according to the method in [111] :

$$\rho = \begin{cases} \frac{\tan(2\varphi)(\delta_x^2 - \delta_y^2)}{2\delta_x\delta_y} & 0 < \varphi < \frac{\pi}{2} \\ \frac{\tan(2\varphi)(\delta_y^2 - \delta_x^2)}{2\delta_x\delta_y} & -\frac{\pi}{2} < \varphi < 0 \end{cases} \quad (4.54)$$

where φ is the orientation of the semi-major axis of the error ellipse in degrees from true North in GPS NMEA sentence "GST". The noise of the GPS orientation θ_t^g can be approximated by unscented transform with the known noise of $\{x_t^g, y_t^g, x_{t-1}^g, y_{t-1}^g\}$.

The information contribution i_t^g and I_t^g of GPS observation can be calculated using

$$\text{Eq.4.29 : } i_t^g = (H_t^g)^T (Q_t^g)^{-1} s_t^g \text{ and } I_t^g = (H_t^g)^T (Q_t^g)^{-1} H_t^g, \text{ with } H_t^g = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

4.4.4/ VALIDATION OF DIFFERENT SENSOR MEASUREMENTS

For the four sensors in our system (laser range finder, stereoscopic system, a GPS receiver, and process model which is considered as a virtual sensor), there are two strategies to validate their coherence :

1. the first strategy is in two steps : the coherence between relative measurements are validated and applied to predict the global pose ; then, the coherence between the prediction and other absolute measurements are validated.
2. the second strategy is to directly compare the absolute vehicle poses from different sensors after pose integration.

The parity relations between different sensors are calculated considering the uncertainties of their observations with Eq. 4.44 to Eq. 4.46.

Relative measurements validation : parity relations between the relative measurements from process model, LRF and stereoscopic system are shown in Tab 4.1. d_{ij} is the distance between the vehicle motion measurements. The value of a special subset of $\{d_{ij}\}$

should be aroused by a unique sensor fault, then this detected sensor measurement is rejected and not fused.

Sensor measurements which are validated by the test above are integrated through :

$$i_t = \tilde{i}_t + \left(\sum_{sen=1}^N i_t^{sen} \right), \quad I_t = \tilde{I}_t + \left(\sum_{sen=1}^N I_t^{sen} \right) \quad (4.55)$$

where N is the number of validated relative measurements.

Then, the vehicle relative movement and its covariance matrix P_a can be recovered from the information vector i_t and information matrix I_t through :

$$\begin{aligned} [\Delta\tilde{x}_t, \Delta\tilde{y}_t, \Delta\tilde{\theta}_t] &= (I_t)^{-1} i_t \\ P_a &= (I_t)^{-1} \end{aligned} \quad (4.56)$$

On the basis of the vehicle movement $[\Delta\tilde{x}_t, \Delta\tilde{y}_t, \Delta\tilde{\theta}_t]$ and the previous vehicle pose $[x_{t-1}, y_{t-1}, \theta_{t-1}]$, the current vehicle absolute pose $\tilde{s} = [\tilde{x}_t, \tilde{y}_t, \tilde{\theta}_t]^T$ can be predicted with the vehicle motion model in Eq.4.47, the covariance \tilde{P} is approximated by unscented transform according to Eq.4.16 to Eq.4.20. The information contribution $(\tilde{i}_f, \tilde{I}_f)$ is calculated using Eq. 4.32 : $\tilde{i}_f = \tilde{P}^{-1} \tilde{s}$ and $\tilde{I}_f = (\tilde{P})^{-1}$.

Absolute measurements validation : then, the absolute measurements are validated. Here, we have two absolute measurements, one is $\tilde{s} = [\tilde{x}_t, \tilde{y}_t, \tilde{\theta}_t]^T$ predicted above, another one is the GPS measurement $s^g = [x_t^g, y_t^g, \theta_t^g]^T$. Other absolute measurements could also be added. In Table 4.2, d_{f4} is the distance between the relative measurement based estimation and the GPS measurement.

As there are only two measurements, the validation test is like the classic NIS test : if d_{f4} meets the test condition, the information contribution of the relative measurement based estimation and the GPS measurement are integrated to provide a final vehicle pose and covariance ; otherwise, the GPS measurements is not used to update the prediction.

Sensor \ d_{ij}	d_{12}	d_{13}	d_{23}
S1-Process model	1	1	0
S2-LRF	1	0	1
S3-VO (Stereovision)	0	1	1

TABLE 4.1 – Parity relations between every two motion measurements

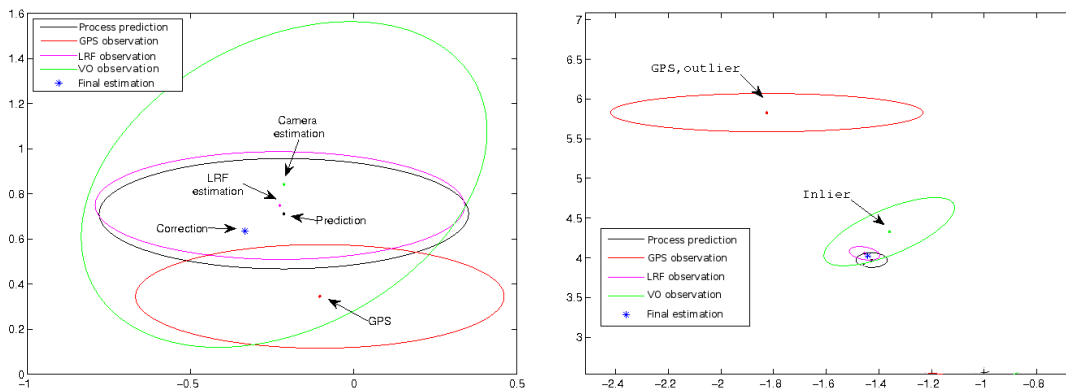
Sensor \ d_{ij}	d_{f4}
S_f -vehicle pose prediction	1
S4-GPS	1

TABLE 4.2 – NIS relation between every two sensors (1)

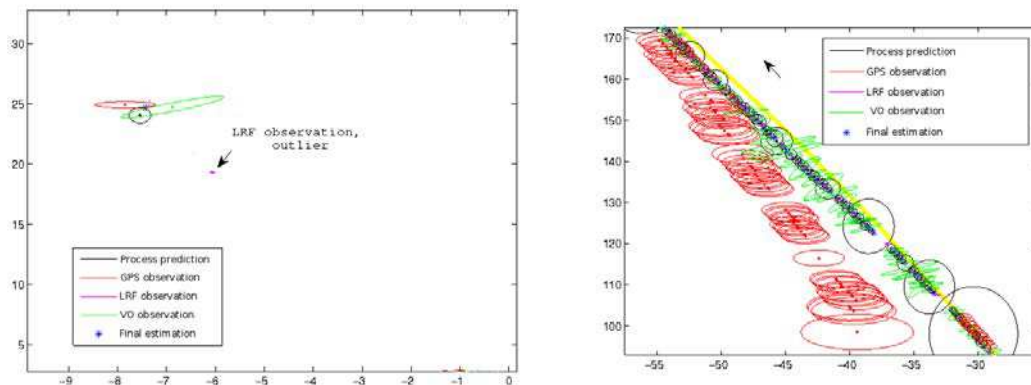
4.5/ IMPLEMENTATION AND EXPERIMENTAL RESULTS

The proposed UIF based fusion method is tested with the same data sequences obtained by our experimental vehicle *SeTCar* (section 3.4.1), as presented in section 3.4.2 and section 3.4.3. The initial orientation and speed of the vehicle are set by GPS observation. The initial state covariance matrix is set by the covariance matrix of the first GPS position.

4.5.1/ EXPERIMENTAL RESULTS OF THE INDUSTRIAL PARK SEQUENCE



(a) Frame 1, all the four estimations are considered to be reliable (b) Frame 6, GPS is considered as an outlier, the other three sensors are used for current estimation



(c) Frame 19, LRF observation is considered as an outlier, the other three sensors are used for current estimation (d) Since frame 151, GPS observations jumped to the left side with large errors, they are treated as outliers and not used for estimation.

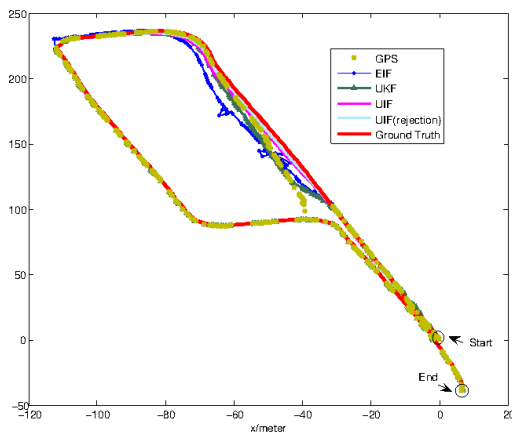
FIGURE 4.4 – Some results of vehicle state and covariance estimation with UIF method

Unscented transform is used in several parts of this method and the parameters of the filter are empirically chosen as detailed in the following parts. Some examples of state and covariance estimation (using the different sensors) at different instants are shown in Fig. 4.4(a) to Fig. 4.4(d).

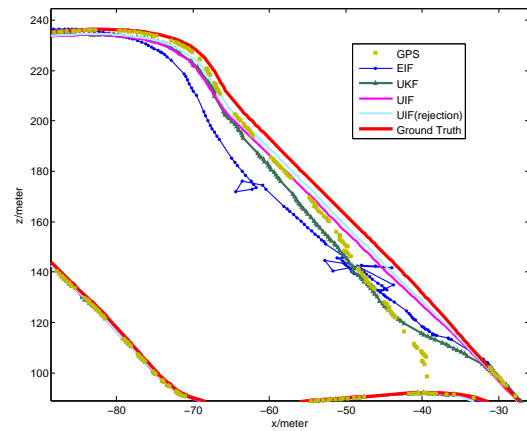
In Fig. 4.4(a), all the sensor measurements are accepted as their observation differences are small. In Fig. 4.4(b), the distances between GPS and all the other sensors increase at the same time, thus the GPS measurement is considered to be an outlier, only the other three sensors are used for the current estimation. In Fig. 4.4(c), as the LRF measure-

ment is not coherent with the other observations, and the error distances in the subset (d_{12}, d_{23}, d_{24}) of LRF increase, the LRF is not used for the estimation. In Fig. 4.4(d), the trajectory in yellow is the ground truth provided by the RTK-GPS. When the low cost GPS receiver fails to provide accurate observations, the fusion results without GPS are more accurate than the GPS observations.

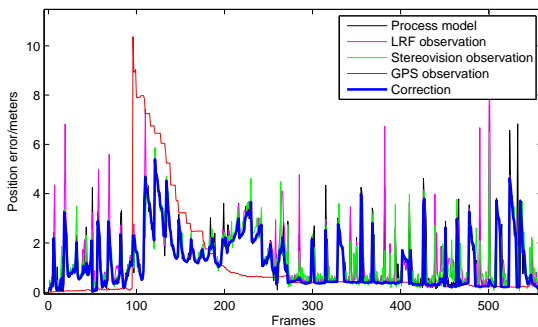
Vehicle position and orientation errors with the different fusion methods are shown in Fig. 4.5(c) and Fig. 4.5(d). As shown in Fig. 4.5(b), GPS positions jump to the left side with an error of about 10m. This GPS failure lasts about 70 meters during the period between frame 100 and frame 200 (see Fig. 4.5(a) and Fig. 4.5(b)). It is noted that neither the LRF nor the stereoscopic system based approaches could provide an accurate absolute localization results alone in long term due to the error accumulation. GPS works in long term, with an error of 1.83% for the whole trajectory.



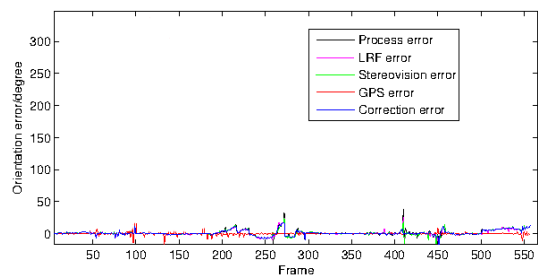
(a) Comparison of vehicle trajectories estimated by EIF, UKF, UIF and UIF (rejection) based multi-sensor fusion



(b) Zoom of the estimated trajectories



(c) Vehicle position errors observed by the different sensors (compared with RTK-GPS)



(d) Vehicle orientation errors observed by the different sensors (compared with RTK-GPS)

FIGURE 4.5 – Estimated vehicle trajectories and comparison of the fusion based vehicle position and orientation with the ground truth

The performance of the proposed UIF based localization method is also compared with the multiple updates UKF methods. The comparison is implemented using the estimated trajectory length error (trajectory length is calculated by the integration of movement between consecutive frames), mean and standard deviation of the error between corresponding estimated positions and ground truth. The results are shown in Tab 4.3.

In order to quantify the performance gain of the unscented approach than extended transformation, the proposed UIF method is compared with extended information filter (EIF) method. Results are shown in Table 4.3 (the 5th row) and Fig. 4.5(a). By comparing the whole trajectory with the ground truth, it is noted that the unscented transformation can provide better localization results in long term. Nevertheless, if the sampling frequency of the system is high enough, the extended transformation should be efficient.

And secondly, in order to quantify the performance gain of the information formulation, a classical unscented Kalman filter (UKF) with multiple updates is implemented and compared with the results of UIF. The prediction is firstly updated by GPS observation, then the corrected result is updated by LRF estimation, and finally by stereo visual odometry. In order to compare the information formulation and the covariance matrix formulation, no outlier rejection step is used when comparing the two methods. Results are shown in Table 4.3 (the 6th row) and Fig. 4.5(a).

Sensors	Traveling distance (m)	Mean (m)	Std.(m)
LRF scan alignment	621.67	1.74	1.52
Stereovision odometry	685.43	1.69	1.38
GPS	614.78	1.08	1.90
EIF fusion	662.11	3.73	6.89
UKF fusion (three updates)	603.07	1.38	1.87
UIF fusion	601.20	1.11	1.17
UIF fusion (with rejections)	603.88	1.18	1.08

TABLE 4.3 – Localization results with the different sensors and UKF, EIF, UIF

It can be seen that for the whole length of trajectory, there is no big difference between UKF and UIF. However, the performance of UIF is better than UKF when comparing the standard deviation and mean error. Besides, the information form has the advantage that the update stage is computationally easier because no gain or innovation covariance matrices need to be calculated. Even though this advantage is not significant for our experiment (because we only have three sensor observations), other sensors can be easily added into the same framework in the future.

Without the error rejection step, information from all the subsystems are integrated together to provide a final estimation. After applying the error rejection step in the fusion process, the GPS failure is detected by the validation test. Nevertheless, without reliable GPS information, the localization system can continue to correctly estimate the vehicle positions by integrating the process model, stereoscopic system and LRF. The error rejection step makes the estimated trajectory more accurate in long term.

4.5.2/ EXPERIMENTAL RESULTS OF THE OLD TOWN CENTER SEQUENCE

The UIF based fusion method is tested with the same data set acquired in the old town center of Belfort (as in section 3.4.3) by integrating the GPS positions, LRF based OR-ICP, stereoscopic system based visual odometry and constant speed model. Stereovision based visual odometry is used to provide an initial transformation for OR-ICP to reduce the searching area of scan correspondences.

4.5.2.1/ WITH SIMULATED GPS MASKS

We added 11 GPS masks into different parts of the vehicle trajectory to simulate the problem of GPS signal blockage in urban environment. In order to find which extended NIS check method is more effective for the vehicle localization, we have tested the two methods : 1) if distances of a subset $\{d_{ij}\}$, $i, j = 1 \dots N, j \neq i$ increase simultaneously, the fault should correspond to the sensor i ; 2) if all distances of a subset $\{d_{ij}\}$, $i, j = 1 \dots N, j \neq i$ meet the condition $d_{ij} > threshold$, the measurement i is considered to be an outlier, where N is the number of measurements. The threshold is defined by the Chi-square table, as 7.8 in our tests.

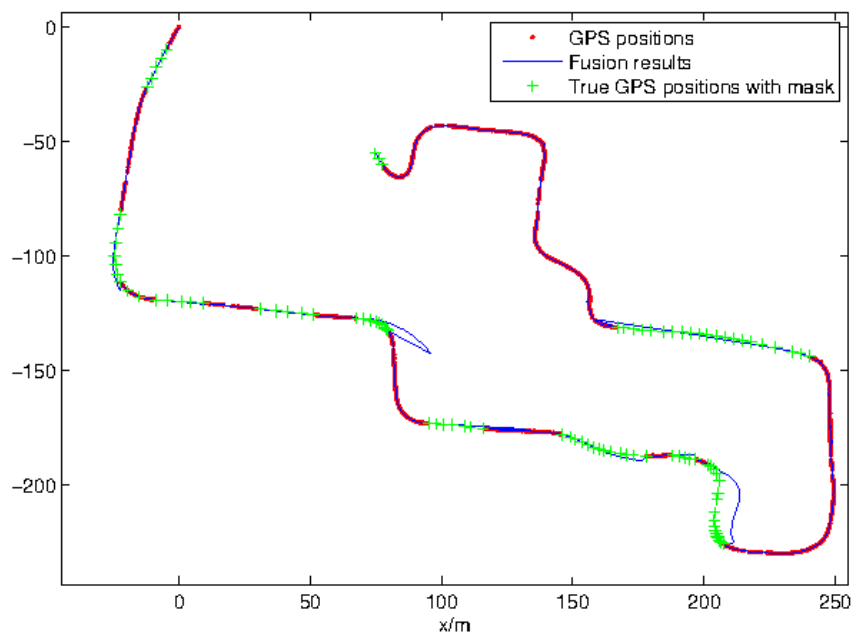


FIGURE 4.6 – Vehicle trajectory during the GPS masks : extended NIS increasing method

As seen in Fig. 4.6 and Fig. 4.7, red points represent the GPS positions without GPS masks, green crosses represent the true GPS positions where GPS masks are added, and blue lines represent the vehicle trajectories estimated by the UIF based methods. Performance of the proposed UIF based localization methods is compared with the ground truth, results are shown in Fig. 4.8 (in order to see the difference more clearly, the error on x and y directions are respectively shown) and Tab 4.4.

	Average error(m)	standard deviation(m)
Sensor fusion (NIS-increase)	4.47	4.16
Sensor fusion (NIS-threshold)	2.53	3.57

TABLE 4.4 – Localization results of the fusion methods in the area with GPS masks

As seen in Tab 4.4 and Fig. 4.8, for the same sensor measurements and the same GPS masks, the second NIS method with a defined threshold (if all distances of a subset are larger than the threshold, the measurement i is considered to be an outlier) can provide

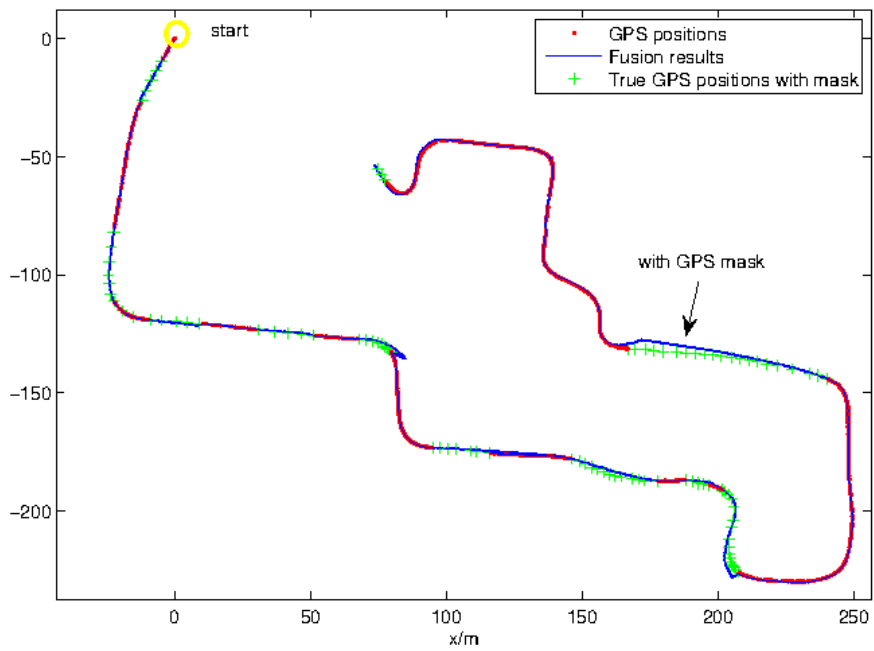


FIGURE 4.7 – Vehicle trajectory during the GPS masks : extended NIS threshold method

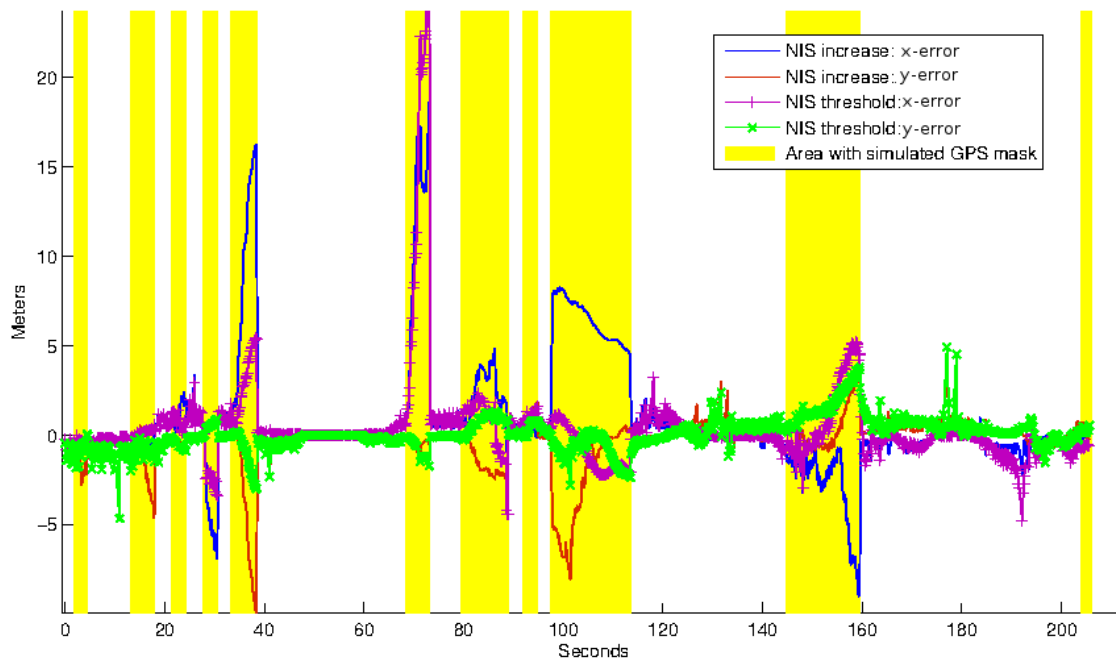


FIGURE 4.8 – Comparison of position error after adding GPS masks

more accurate result for this sequence. It can be noted that in the GPS masked areas, visual odometry and OR-ICP can continue to estimate the vehicle trajectory though the error increases gradually after long term.

4.5.2.2/ WITH SIMULATED GPS JUMPS

In order to test if the erroneous sensor measurements could be detected by other measurements, white noises are simulated and randomly added to 36 GPS positions. The position error on x -direction is between $0 \sim 10\text{m}$, and the position error on y -direction is between $0 \sim 2\text{m}$. As seen in Fig. 4.9, the erroneous GPS positions are shown in green crosses, the positions estimated by the UIF based method is shown in red circles. It can be seen that the other sensors can continue to estimate vehicle positions.

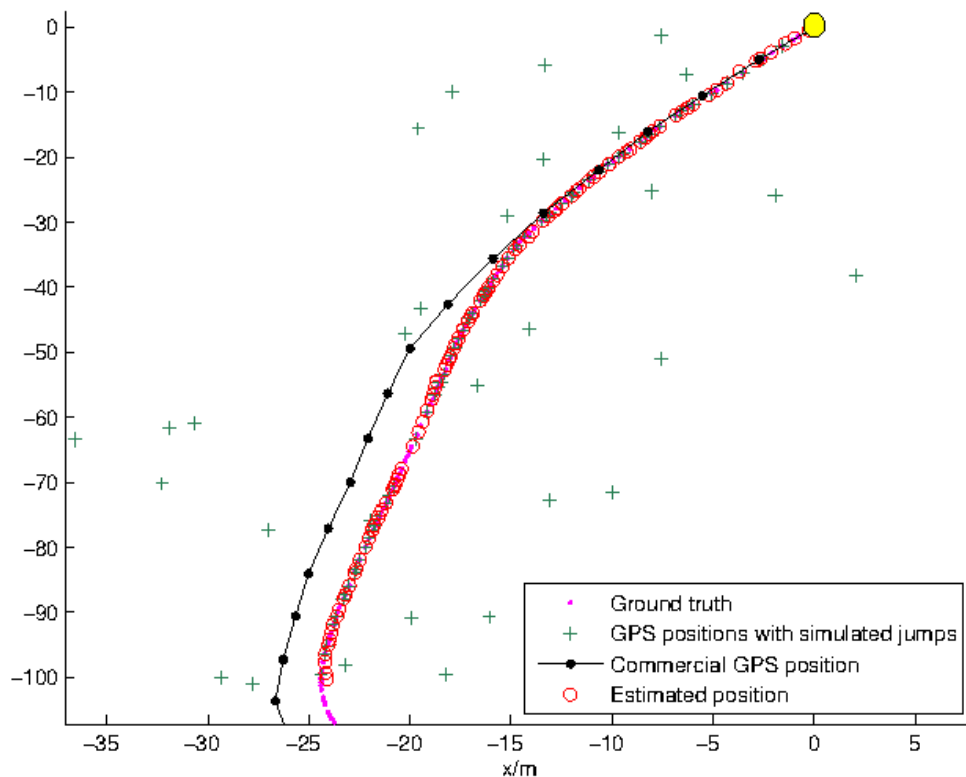


FIGURE 4.9 – Vehicle trajectory with simulated erroneous GPS positions

As described in Table 4.1, the parity checking constraint is applied for information fusion. Firstly, the normalized innovation squared between the three relative measurements (process model, OR-ICP and VO) are shown in Fig. 4.10. The prediction from the relative sensors are compared with the absolute GPS measurements to provide a final vehicle position. It can be seen that most of the time LRF and stereovision based estimations are coherent with each other.

During the whole trajectory, different sensors are accepted for fusion. As seen in Fig. 4.11, the points above the black line represent the sensors used for estimation at every time instant, while the points below the line represent the measurements which are rejected for fusion. Most of the GPS positions with the simulated jumps are rejected for fusion.

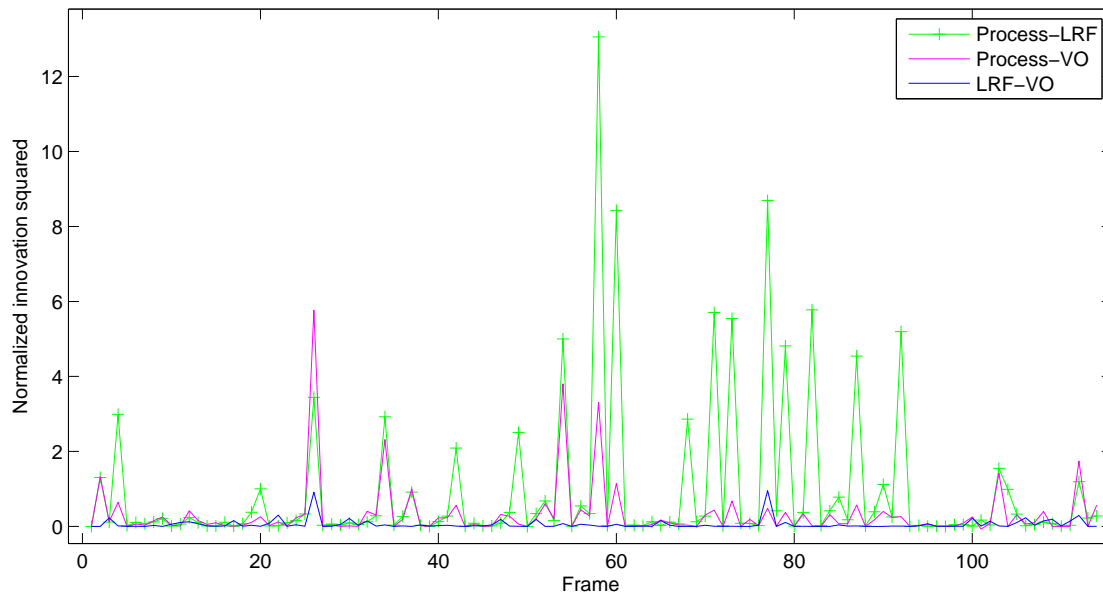


FIGURE 4.10 – NIS changes of the relative measurements

The position error of the fusion method is compared with the simulated GPS error in Fig. 4.12. When GPS observations are rejected in short term, the other relative sensors can continue to provide accurate positions.

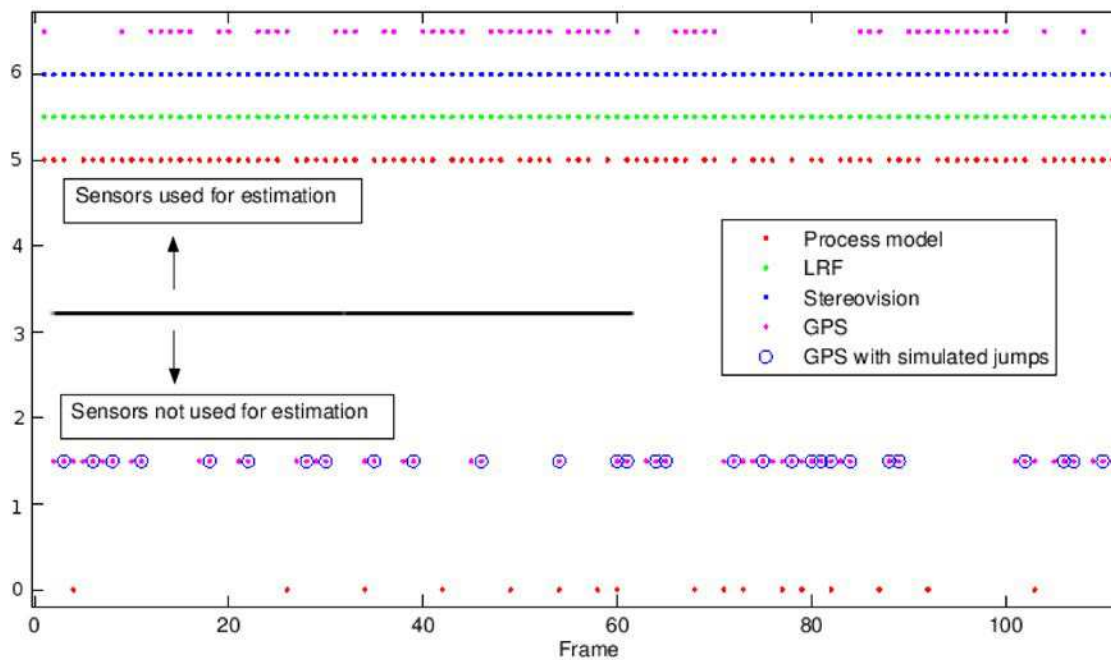


FIGURE 4.11 – Sensors validated for sensor fusion after adding simulated GPS jumps

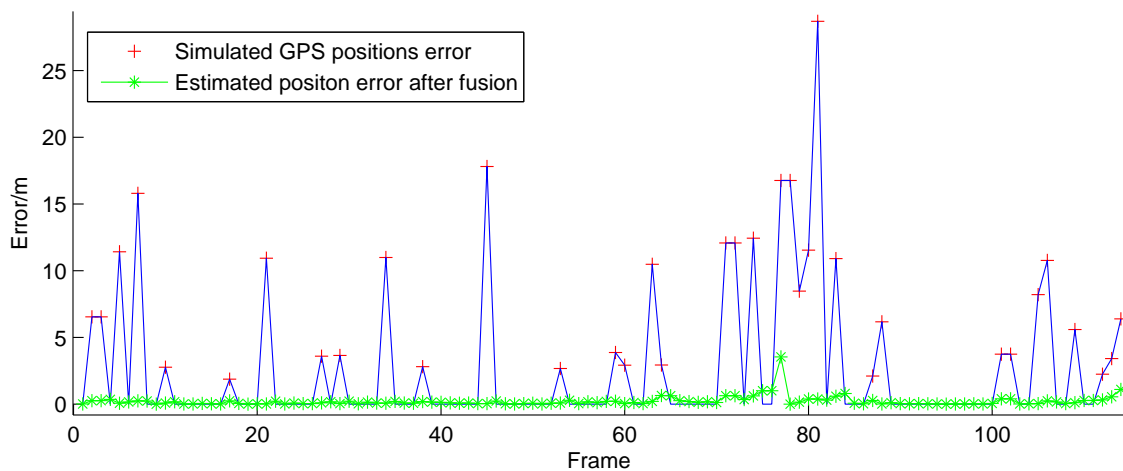


FIGURE 4.12 – Comparison of position error after adding random GPS position errors

4.6/ CONCLUSION

In this chapter, a vehicle localization method is presented by integrating vehicle global information from onboard GPS receiver and vehicle odometry information respectively from a stereoscopic system (visual odometry) and a LRF system (scan alignment). The coherence of different sensor measurements are validated with extended NIS method before being integrated by an unscented information filter. Information from other sensors could also be easily incorporated into the system if needed. The proposed method was tested with real data and evaluated by RTK-GPS as ground truth. Results show that the coherence validation step makes the estimated trajectory more accurate in long term as GPS failures can be detected by the validation test. Fusion of the stereoscopic system and LRF can continue to localize the vehicle during GPS outages, while GPS measurements permit to avoid the trajectory drift when only the stereoscopic system or the LRF is used.

For the proposed vehicle localization method in this chapter, several research perspectives are summarized :

- In future works, more sensors like IMU or odometry can also be directly integrated thanks to the convenience of information filter. And tight coupling approach between LRF and image data is also envisaged instead of loose coupling of their estimated motions.
- If the GPS signals are lost for long period, the trajectory might gradually drift if only using the relative sensors. Therefore, another kind of global information should also be incorporated into the system to adjust the vehicle pose, we will discuss about the use of maps in the next chapter.

HORIZONTAL/VERTICAL LRFs AND GIS MAPS AIDED VEHICLE LOCALIZATION

5.1/ OVERVIEW

In order to solve the problem of blockage/reflection of GPS signals within urban areas, dead-reckoning methods like inertial navigation, wheel encoder odometry or camera/laser based visual odometry have been used to compensate GPS outages by continuously estimating the vehicle motion. After these steps, the predicted vehicle pose can be corrected when the GPS receiver returns to work. However, dead-reckoning methods can provide accurate relative vehicle movements only in short period. If GPS receiver cannot provide any absolute vehicle positions for long time, the vehicle trajectory might gradually drift and the localization error cannot be bounded. Therefore, other global information sources are needed to correct the accumulated localization error.

Nowadays, a lot of vehicles have been equipped with GPS navigation systems : a GPS receiver, an itinerary planning software and a series of digital maps which can be displayed in human readable format, as shown in Fig. 5.1. Digital maps are provided by cartographers like professional companies NAVTEQ¹, TeleAtlas², or Chinese companies like NavInfo³, AutoNavi⁴, etc.

The digital maps (like 2D/3D/DEM maps) can provide static global environment information, such as shapes of urban roads (line-style landmarks), positions of building footprint (polygon-style landmarks), trees and street lamps (point-style landmarks), as well as attributes of these objects (e.g., width of a road, height of a building). If the environment information around the vehicle can be observed by on-board exteroceptive sensors like cameras or laser range finder, and be well associated with the information provided by the initial maps, the vehicle pose can be corrected.

Different features have been proposed to make use of the information from maps together with a laser range finder. Scheunert et al. [141] proposed to use a horizontal laser scanner to detect point style landmarks in the environment and use GIS map as measurements

1. <http://www.navteq.com>
2. http://www.tomtom.com/en_gb/licensing/
3. <http://www.navinfo.com/en/>
4. <http://www.autonavi.com/en/index>



FIGURE 5.1 – In-car GPS navigation system and digital map provided by NAVTEQ

(Fig. 5.2(a)). Like the sensor installation of LRF sensor in [118], Jabbour et al. [74] [75] [76] proposed to mount a laser range finder at the bottom front of the vehicle, then extract the sidewalk edges from the vertical LRF scan (Fig. 5.2(b)). The detected sidewalk landmarks are grouped into road boarder segments and stored in an enhanced map layer for vehicle pose correction and landmark updating in a SLAM-like strategy. Qin et al. [124] also proposed to use a tilt-down lidar to scan the road in front of the vehicle and extract the curb features on the left and right sides.

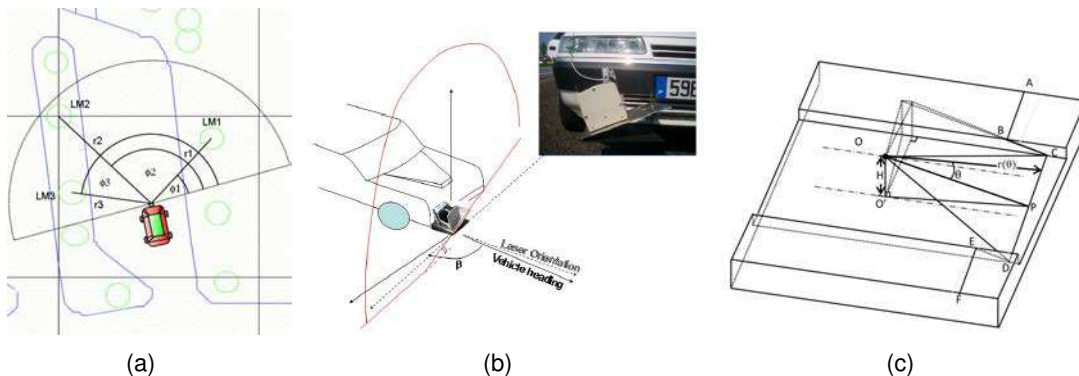


FIGURE 5.2 – (a) Use horizontal LRF to detect point features in GIS map [141] ; (b)(c) use tilt-down LRF to scan the sidewalk edges [74] [124]

Considering that in urban environments, a lot of vehicles are parked by the roadside of the streets, it is very likely that some sidewalk edges might be blinded by vehicles and cannot be detected by LRF scan. Thus, we propose to use building facades as landmarks since there are less obstacles in the air than on the ground. Top part of the building facades which is higher than the vehicle can still be scanned by the vertical LRF (with long enough detecting range) even though there are obstacles around. Besides, as the footprints of buildings have already been marked in a digital map layer, this *a priori* information can be directly used for initial vehicle pose correction.

5.1.1/ PROPOSED METHOD

In the proposed method, two LRF systems are respectively horizontally and vertically installed on the roof of an experimental vehicle (see Fig. 5.3 and Fig. 5.34) to provide comprehensive representation of the environment around the vehicle :



FIGURE 5.3 – Two LRF systems are mounted on the roof of the vehicle : the vertical LRF scan is in blue and the horizontal LRF scan is in red

- One LRF (LRF-H) looks forward and scans horizontally the environment in front of the vehicle. The use of LRF-H scans is in twofold : it can be used to provide vehicle odometry information as presented in section 3.3, and also be used to detect the building facades. If there is few obstacles or no obstacle in front of the vehicle, lines corresponding to the building facades can be extracted from the LRF-H scans and associated with the GIS building map. If there are a lot of obstacles in the view of the LRF-H, the LRF-H cannot provide the structure information around the vehicle.
- Another LRF (LRF-V) scans vertically and looks upward. Even though LRF-H cannot observe the building facades due to obstacles around, top part of the building facades which is higher than the vehicle can still be scanned by the LRF-V. The intersection lines of the building facades and the LRF-V scan plane can be extracted from the vertical LRF scan, and associated with the building map to correct the vehicle lateral position.

The proposed approach is in three steps, as shown in Fig. 5.4.

The first vehicle position is initialized by GPS position for global positioning. Then, the vehicle position is estimated with the relative localization measurements from a gyro sensor and LRF-H alignment through vehicle transition model, and updated by the absolute location provided by the GPS receiver (if available), together with road map-matching result from GIS road map layer (section 5.2). After that, the vehicle pose is refined with observations from the two LRFs and the (original) GIS building map layer (section 5.3). Considering the inherent noises of maps during surveying and map producing processes, the proposed LRF and building map based pose correction method is used only if GPS readings are not provided. Furthermore, in order that the landmarks observed by the LRF-V can be reused for vehicle localization in the future, two new map layers are generated to store these landmarks data (section 5.4) : the building facade landmarks detected by the LRF-V at every instant are grouped into building segments and stored in a facade

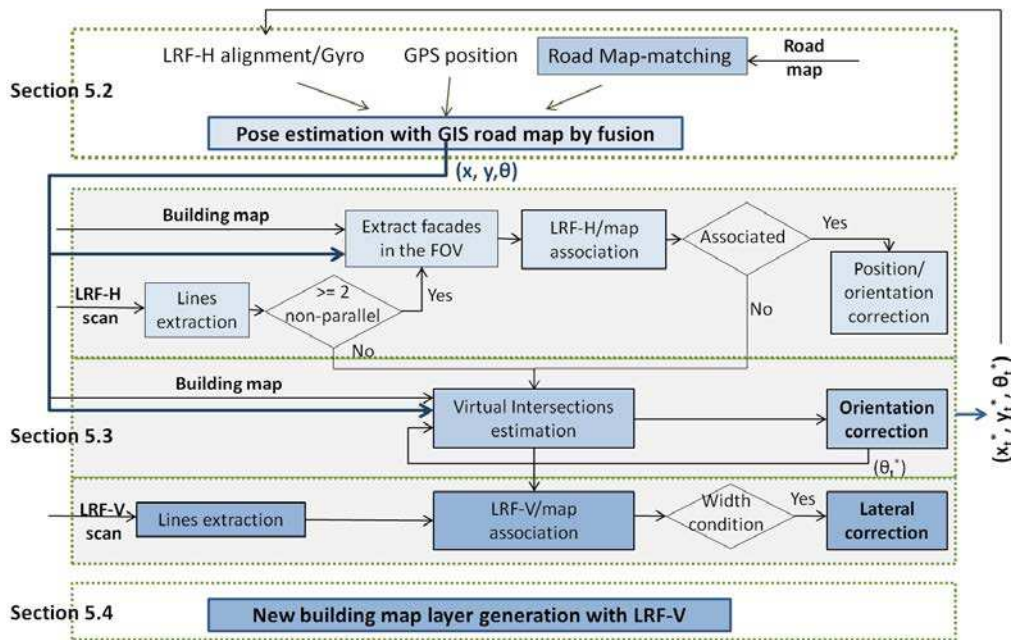


FIGURE 5.4 – The proposed GIS aided localization method

map layer on the basis of the original building map ; another map layer is used to store the independent objects.

The structure of this chapter is organized as follows : section 5.1.2 introduces the principle of data organization and main applications of GIS ; section 5.2 details the method of vehicle pose estimation with GIS road map layer ; section 5.3 presents the vehicle pose refining method with horizontal/vertical LRFs and the original GIS building map layer ; section 5.4 introduces the new map layer generating method ; section 5.5 presents some experimental results ; and finally, conclusions and perspectives are presented in section 5.6.

5.1.2/ GEOGRAPHICAL INFORMATION SYSTEM (GIS)

For a geographical information system (GIS), there are several GIS organizational schemes in which all data of a particular class, such as roads, buildings, woodland or water types, are grouped into a same layer (or coverages), as shown in Fig. 5.5⁵. In this spatial database, the geometry and attribute information of the spatial features are stored in table files, with unique identifiers (ID) linking the corresponding spatial object and its attribute data.

The geometry data model treats environment objects as a set of primitives and spatial entities, such as point, lines and areas in 2D models. As seen in Tab. 5.1, the geometry of a primitive entity is stored in a shape comprising a set of vector coordinates. The location of a point is described by a set of coordinates. A line is defined by an ordered sequence of two or more sets of point coordinates. An area is defined by a boundary of one or more lines which form a closed, non-self-intersecting loop. If the area has holes in it, more than one such loop might be used to describe it.

As the range of information which can be placed in the geographical context is large,

5. <http://resources.arcgis.com/en/help/getting-started/articles/026n000000q000000.htm>

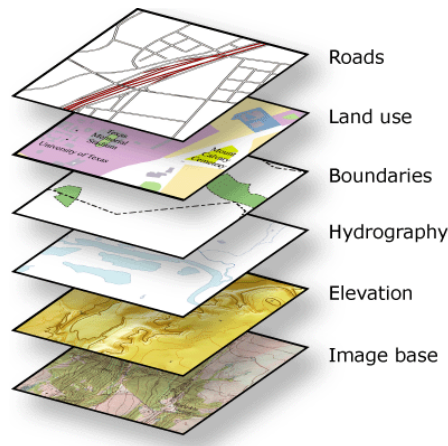


FIGURE 5.5 – Organization of GIS layers

Type	Graphic representation	Digital representation
Point	•	Coordinates : (x, y) in 2D
Line	~	Ordered list of coordinates
Area	⬠	a) One line : if the first point equals the last one ; b) A set of lines : if an area has wholes

TABLE 5.1 – Types of GIS objects

there is continuing growth of GIS usage across many disciplines. It has been widely used for earth survey and monitoring, such as land survey, hydrology and marine survey, soil surveys, geological surveys ; it has also been used for public administration, logistics or transportation management, etc. The possible use of GIS for vehicle localization can be summarized as follows :

- Displaying the vehicle position and trajectory information.
- Providing structure information and attribute information of the environment.
- Spatial analysis and real-time road traffic analysis for vehicle navigation assistance [166], like path-planning.
- Storing and managing sensor data with GIS, such as the vehicle positions provided by global positioning system (GPS) receivers, and visual landmarks observed by visual sensors in the urban environments [131] [74], etc.

Generally, digital maps used for vehicle navigation focus on providing detailed road network information and road attributes (length, surface materials, driving directions, obstacles, vehicle speed, etc.) in various map formats, such as GDF (Geographical Data File) which are used to describe the transportation network in Europe, standard KIWI proposed by Japan KIWI-W Consortium for vehicle navigation, standard SDAL (Shared data access library) and NAVSTREET defined by NavTeQ. Digital maps in different formats can be transformed to SDAL format and used for navigation application development with toolbox NAVTOOLS. In this thesis, GIS maps in general format : shapefile⁶ are used. This format is defined by ESRI and each layer includes three data files : shp/shx/dbf. The road network map and building footprint map provided by IGN (BD TOPO map - Pays of Territoire de Belfort, IGN, 2002) are used to provide *a priori* environmental information for

6. <http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>

vehicle localization.

5.2/ VEHICLE POSE ESTIMATION WITH GIS ROAD MAP

The process of vehicle pose estimation with GIS road map layer are briefly described in Fig. 5.6. Vehicle pose is at first predicted by LRF-H based scan alignment and a gyro through vehicle motion model, then corrected by observations from a GPS receiver (if available) and a road map based map-matching method (section 5.2.1) within an information filter (IF, EIF, UIF, etc.) framework (section 5.2.2).

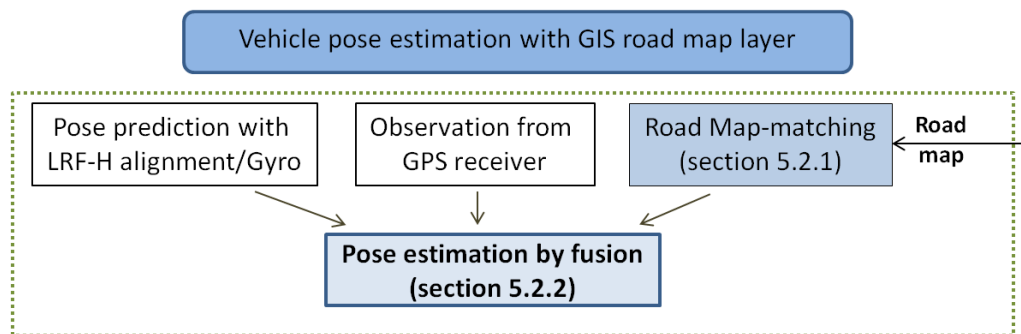


FIGURE 5.6 – Vehicle pose estimation with GIS road map layer

- **Pose prediction with LRF-H based scan alignment and gyro.** The vehicle model used for pose prediction is the same as in Chapter 4. The odometry information d_t of the vehicle is provided by LRF-H scan alignment (detailed in section 3.3). Vehicle yaw angle θ_t is estimated with the angular velocity ω^{gyro} from gyro sensor :

$$\theta_t = \theta_{t-1} + \omega_t^{gyro} \delta t \quad (5.1)$$

Variance of the gyro rate is set to 0.01^2 rad , according to the construction datasheet of the sensor. With the assumption that the road ground is flat and the vehicle moves with constant speed, the vehicle state $[\tilde{x}_t, \tilde{y}_t, \tilde{\theta}_t]$ and state covariance Q_t are approximated with the vehicle transition model $X_t = f(X_{t-1}, d_t, \theta_t, \delta t) + \alpha_t$ by unscented transform.

- **Observation from GPS sensor.** The observation from GPS receiver was detailed in section 4.4.3 with observation vector $s_t^g = [x_t^{gps}, y_t^{gps}]$ and covariance matrix Q_t^g of the GPS position.

5.2.1/ OBSERVATION FROM ROAD MAP-MATCHING METHOD

Road map is an ITN layer (Integrated Transportation Network) which describes the road topology using two types of features : road link and road node. Road map-matching is to find the correspondence between a vehicle trajectory position (e.g., from a GPS receiver, or a GPS receiver integrated with dead-reckoning sensors) and a position in the road network (provided by a GIS map).

An example of map-matching is shown in Fig. 5.7. Numerous approaches have been proposed to solve the road map matching problem in ambiguous situations (e.g., road

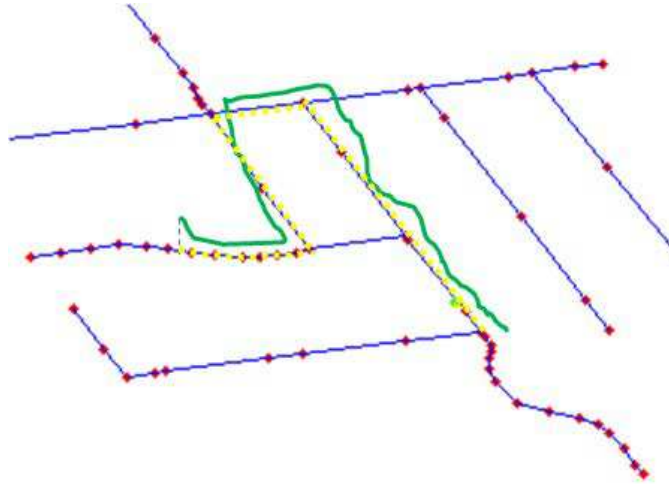


FIGURE 5.7 – An example of road map based map-matching method. In this figure, the red points are the road nodes, the blue line segments are the road links with node feature at each end of the segment. The green line indicates a vehicle trajectory provided by the GPS receiver, and the yellow points are the correspondences of vehicle positions on road network obtained by map-matching method.

intersections, parallel close roads), such as the multi-hypothesis map-matching algorithm [63][112][125][165].

Though vehicles might not remain on the road central line represented by the road network map, the map-matching method can provide a relatively accurate position by re-localizing the vehicle on road, especially when the GPS encounters large errors or the predicted position is off the road. Then, the map-matching result can be integrated with other vehicle pose measurements from an INS, a wheel encoder or other sensors [73] to restrict the vehicle position around the road.

5.2.1.1/ ROAD MAP-MATCHING OBSERVATION

Measurement provided by the map-matching method is the corresponding position of the vehicle on the road map, as :

$$s_t^m = [x_t^{map}, y_t^{map}] \quad (5.2)$$

In this work, the classic nearest road map-matching method is used under a speed constraint and an orientation constraint. The distances between the predicted vehicle position and all the road segments in the neighborhood of the vehicle are calculated (for a predefined zone with size $50m \times 50m$) at first. E.g., in Fig. 5.8, the distance between the current vehicle position and the road segment $[ba]$ is calculated in two steps :

- At first, the perpendicular distance between the vehicle position and each road segment is calculated. Then,

a) if the perpendicular foot $p(p_x, p_y)$ (see Fig. 5.8) is on the road segment, this distance is considered as the closest distance between the vehicle and the road segment. The corresponding position of the vehicle on road is the point p , written as : $(x_t^{map}, y_t^{map}) = (p_x, p_y)$;

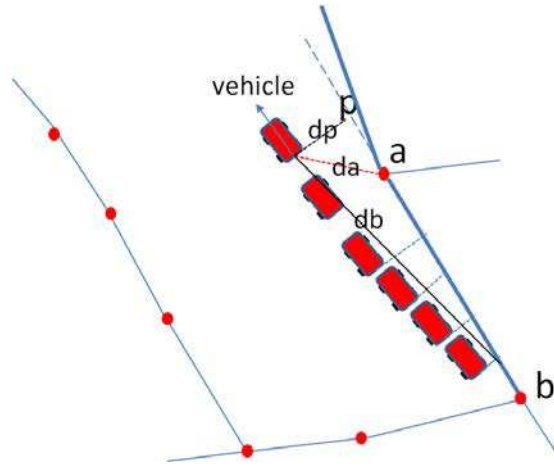


FIGURE 5.8 – Choosing the corresponding road segment

b) if the perpendicular foot p lies outside of the road segment (see Fig. 5.8), the distances $[da]$ and $[db]$ between the vehicle and the two endpoints $a(a_x, a_y)$ and $b(b_x, b_y)$ are calculated. The point a (or b) with smaller distance $[da]$ (or $[db]$) is selected as the corresponding position of the vehicle, written as : $(x_t^{map}, y_t^{map}) = (a_x, a_y)$ or (b_x, b_y) .

- After that, the road segment which is closest to the vehicle is chosen as the corresponding road segment.

The slope of the corresponding road segment θ_t^{map} (direction of the road segment is chosen according to the previous vehicle moving orientation) can also be calculated.

In order to guarantee the accuracy of the map-matching method, the vehicle is firstly tracked on the previous corresponding road segment till it moves out of this segment, with the following equations :

$$\begin{aligned}
 r_x &= \tilde{x}_t - b_x \\
 r_y &= \tilde{y}_t - b_y \\
 \lambda_x &= a_x - b_x \\
 \lambda_y &= a_y - b_y \\
 \lambda_u &= (\lambda_x r_x + \lambda_y r_y) / ((\lambda_x)^2 + (\lambda_y)^2)
 \end{aligned}
 \tag{5.3}$$

where a and b are the endpoints of a road segment shown in Fig. 5.9.

- If $0 \leq \lambda_u \leq 1$, the vehicle is on the same segment as the previous position, as in Fig. 5.9(a) ;
- If $\lambda_u < 0$ or $\lambda_u > 1$, it indicates that the vehicle moves outside of the current road segment, we need to search for the vehicle position on a new road segment, as in Fig. 5.9(b). When the vehicle is tracked on another road segment, the tracked segment should be close to or connected with the previous segment by considering the vehicle velocity.

In our experiments, the map-matching method is based on GPS position if a GPS reading is available ; if not, it is based on the predicted position from the transition model.

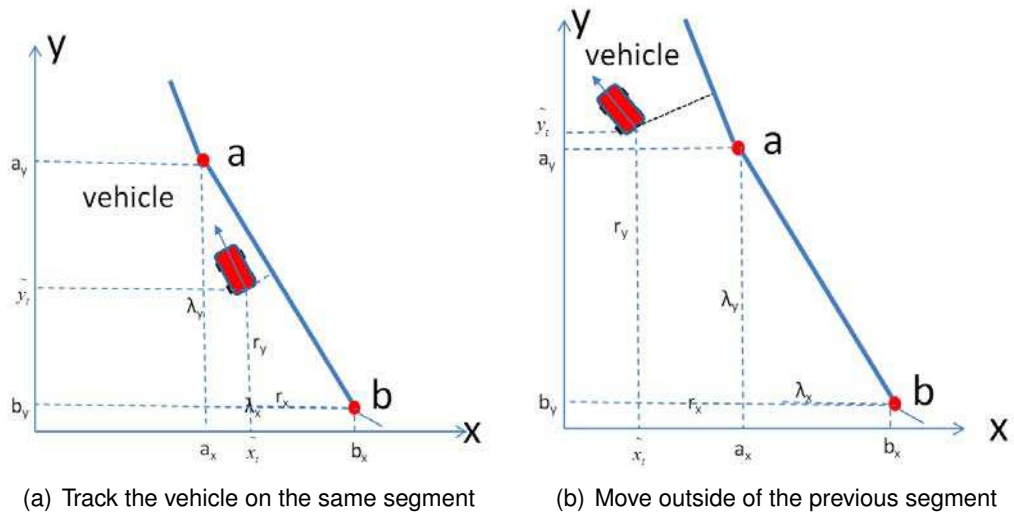


FIGURE 5.9 – Track vehicle position on the previous road segment

5.2.1.2/ UNCERTAINTY OF MAP-MATCHING POSITION

For the corresponding map matching position (x_t^{map}, y_t^{map}) of the vehicle, it is essential to estimate its covariance matrix before being integrated with other measurements. In the local frame attached to a road segment, let x^l -axis be collinear to the road direction (Fig. 5.10), y^l be perpendicular to x^l , the error ellipse of a map-matching position is along the direction of road segment with center (x_t^{map}, y_t^{map}) .

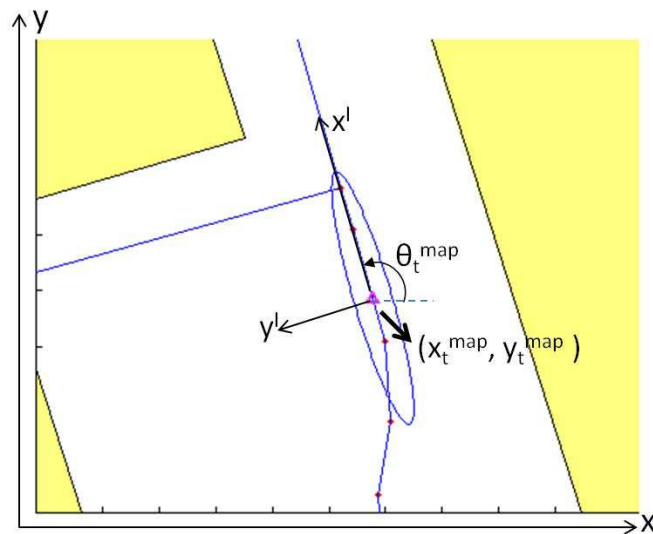


FIGURE 5.10 – Representation of the error ellipse of a map-matching observation (3σ)

Covariance matrix $Q_t^{m,l}$ of the map-matching observation in the local frame is approximated by a large longitudinal error and a lateral error, which is represented by the the

width attribute of the road segment, written as [111] :

$$Q_t^{m,l} = \begin{bmatrix} (\sigma_x^l)^2 & 0 \\ 0 & (\sigma_y^l)^2 \end{bmatrix} \quad (5.4)$$

where σ_x^l and σ_y^l are respectively the longitudinal and lateral standard deviations of map observation. Since the vehicle is not always driven on the segment corresponding to the road central line, the road width is taken into account to represent the lateral standard deviation σ_y^l , as :

$$\sigma_y^l = w_r/2k + e_m \quad (5.5)$$

where w_r is the width attribute of the road segment stored in the attribute table of the GIS road layer ; k is a constant associated with the Gaussian probability error ellipse with $P = 0.9$, $k = \sqrt{-2\ln(1-P)}$, e_m is the map error provided by the map cartographer. It is set to $0.5m$ in this work. The longitudinal error σ_x^l is set big enough compared to the error in y^l -axis (it is set to $10m$ for our map database in this work).

Then, the covariance Q_t^m of the map observation in the global reference frame is obtained with the local covariance matrix $Q_t^{m,l}$, and the orientation of road segment with respect to the global reference frame, i.e., θ_t^{map} . Q_t^m is written as :

$$Q_t^m = \begin{bmatrix} \sigma_x^2 & \sigma_{xy}^2 \\ \sigma_{xy}^2 & \sigma_y^2 \end{bmatrix} \quad (5.6)$$

where :

$$\begin{aligned} \sigma_x^2 &= (\sigma_x^l)^2 \cos^2(\theta_t^{map}) + (\sigma_y^l)^2 \sin^2(\theta_t^{map}) \\ \sigma_y^2 &= (\sigma_x^l)^2 \sin^2(\theta_t^{map}) + (\sigma_y^l)^2 \cos^2(\theta_t^{map}) \\ \sigma_{xy}^2 &= ((\sigma_x^l)^2 - (\sigma_y^l)^2) \cos(\theta_t^{map}) \sin(\theta_t^{map}) \end{aligned} \quad (5.7)$$

As shown in Fig. 5.10, the corresponding point of vehicle position is on a road segment with $4m$ width, and the ellipse covariance (95%) corresponding to the map-matching position observation is along the road segment.

5.2.2/ POSE UPDATE AND COVARIANCE ESTIMATION WITH UIF FUSION

As described in section 4.2, different fusion strategies could be used to update the vehicle pose with the prediction and observations from the GPS sensor and the map-matching method. Since the inverse-covariance form of information filter is able to update the prediction by directly integrating the information state vectors and information matrices of multiple observations, the information filter based fusion (section 4.2.4) is applied in this section to integrate the prediction from LRF-H scan alignment/gyro, and the observations from GPS receiver and road map-matching method. Procedure of the vehicle pose updating is in five steps :

1. The vehicle state $[\tilde{x}_t, \tilde{y}_t, \tilde{\theta}_t]$ is predicted with the process model. The information vector $i\tilde{f}_t$ and information matrix $I\tilde{F}_t$ are predicted according to Equation 4.32.
2. Together with the GPS observation $s_t^g = [x_t^{gps}, y_t^{gps}]$, the information contribution i_t^g and I_t^g of GPS can be calculated using Equation 4.29.
3. With the map-matching observation $s_t^m = [x_t^{map}, y_t^{map}]$ in Equation 5.2 and its covariance matrix Q_t^m in Equation 5.6, the information contribution i_t^m and I_t^m of the road map observa-

tion can be calculated using Equation 4.29 : $i_t^m = (H_t^m)^T (Q_t^m)^{-1} s_t^m$ and $I_t^m = (H_t^m)^T (Q_t^m)^{-1} H_t^m$, with $H_t^m = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$.

4. Validation of different sensor observations. The vehicle process model is considered as a virtual sensor. Thus, three observations need to be validated : the process prediction, the GPS observation, and the map-matching observation.

5. With N validated observations, the information state vector and information matrix are obtained by linear combination of the local information contributions from the considered sensors :

$$if_t = \sum_{s=1}^N i_t^s, \quad IF_t = \sum_{s=1}^N I_t^s \quad (5.8)$$

where i_t^s and I_t^s are respectively the information state and information matrix of the s^{th} validated observation (here, $N = 3$ if all sensors are used and $if_t = i_t^{\tilde{f}} + i_t^g + i_t^m$, $IF_t = I_t^{\tilde{f}} + I_t^g + I_t^m$). Then, the vehicle pose $\widehat{X}_t = (\widehat{x}_t, \widehat{y}_t, \widehat{\theta}_t)$ can be recovered by : $\widehat{X}_t = (IF_t)^{-1} if_t$, $P_{\widehat{X}_t} = (IF_t)^{-1}$.

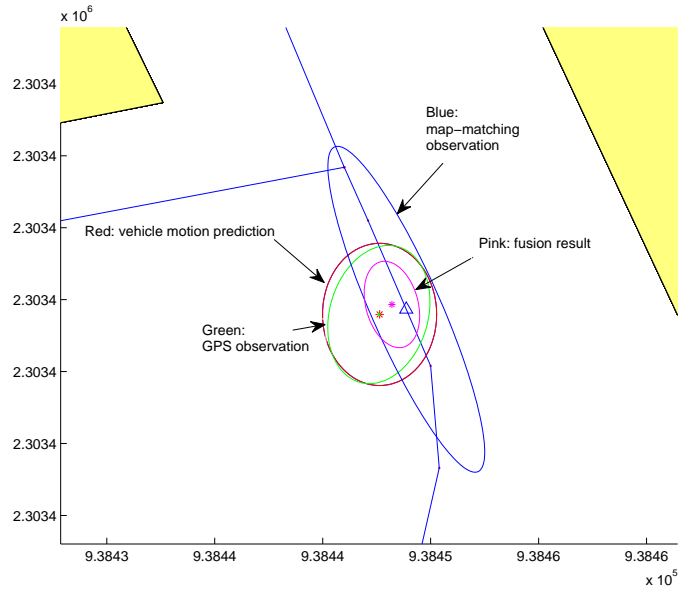


FIGURE 5.11 – Representation of the uncertainty ellipse of the vehicle positions obtained by different methods

Fig. 5.11 shows the covariances of vehicle positions respectively estimated by the vehicle motion model, the GPS sensor, and the road map based map-matching. It can be seen that the covariance is reduced after fusion of multiple measurements (shown in pink color).

This work assumes that a road map-matching observation exists if it is coherent with other observations ; otherwise, this observation is rejected. But the vehicle might be on a new road not existing in the road map, it is also possible to update the road network map like in the work of [23] [24], especially for the missing highway since GPS delivers accurate positions in these open areas.

5.3/ VEHICLE POSE REFINEMENT WITH LRFs AND ORIGINAL GIS BUILDING MAP

As presented in section 5.2.1, if GPS subsystem cannot provide vehicle positions for long time, map-matching method with road map layer can help to re-localize the vehicle on road. However, due to the width of road, this vehicle pose is still coarse. If we want to control the vehicle action on road (e.g., lane-changing), more accurate estimation of vehicle position on road is needed. Therefore, with the vehicle pose initially estimated in section 5.2, we propose to use two laser range finders (LRFs) and *a priori* GIS building map layer to make the localization from coarse-to-fine, especially in structured urban environments. This step would be useful for a more accurate vehicle lateral position.

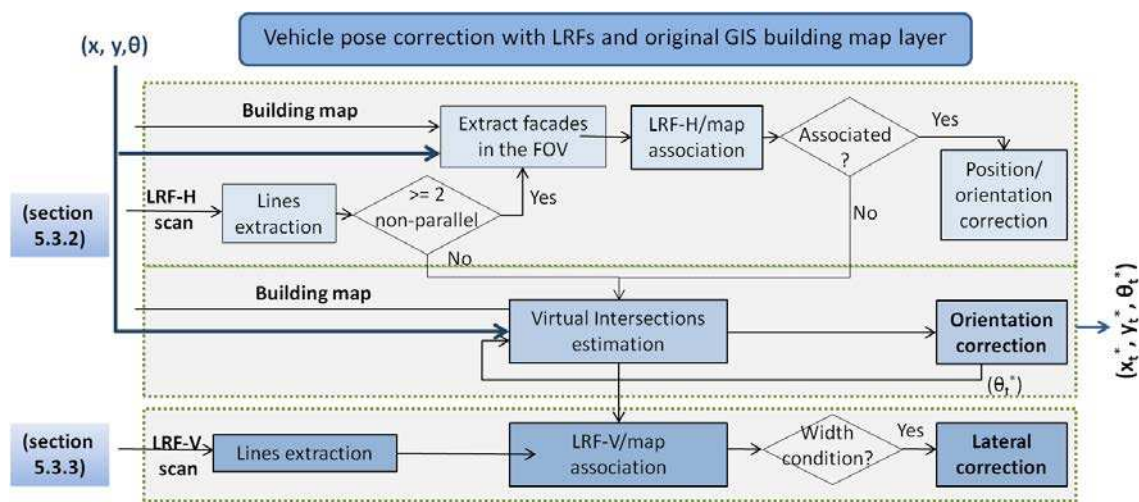


FIGURE 5.12 – Method of vehicle pose refining with LRFs and original GIS building map

The vehicle pose refinement method with LRFs and original building map layer is shown in Fig. 5.12. At first, line features are respectively extracted from the horizontal and vertical LRF scans from the onboard LRFs systems (section 5.3.1.1). Then, the environment around the vehicle is analyzed (section 5.3.1.2). After that, the detected lines in the LRF-H scan are associated with the GIS building map. If the lines are well associated, the vehicle longitudinal/lateral/orientation pose is corrected (section 5.3.2.3); if the vehicle is in a narrow street with one/two parallel building facades, the orientation of the vehicle might be corrected (section 5.3.2.4); if the non-parallel lines solution (section 5.3.2) fails to work, the lateral position of the vehicle is corrected with the vertical LRF (section 5.3.3).

5.3.1/ ENVIRONMENT ANALYSIS WITH HORIZONTAL AND VERTICAL LRFs

In order to analyze the environment configuration, we first have to extract the line features from LRF scans in section 5.3.1.1.

5.3.1.1/ LINE FEATURE EXTRACTION FROM LRF SCANS

In order to extract lines from a LRF scan, a lot of line extraction algorithms using 2D range data have been proposed [162], such as the point distance based method using

distance between every two consecutive points [9], Split-and-Merge algorithm [20], line segmentation based invariant parameters (SIP) [57]. Line extraction algorithm used in our work is shown in Algorithm 3.

Algorithm 3: *Line_extraction*

Input : a set of laser points : $\{(x_k, y_k)\}, k = 1, \dots, n$, interval of radial distance r_{th} , interval of polar angle α_{th} , number threshold of a line n_{th}

Output : fitted line uncertainty P_L , fitted lines $L(R, \alpha)$, points on every fitted line N_{upx} , uncertainty Q_x of every point

- 1 Extract lines using Hough transform based line extraction method;
 - 2 Delete points far away from each other in each line cluster;
 - 3 Merge co-planar line segments according to R and α ;
 - 4 Weighted line fitting based on selected line segments from Hough transform.
-

Hough transform (HT) for line extraction

Polar coordinate system is a two-dimensional coordinate system, in which each point is represented by the distance d_k from a fixed point O (called pole), and the angle ϕ_k from a fixed direction Ox (see Fig. 5.13).

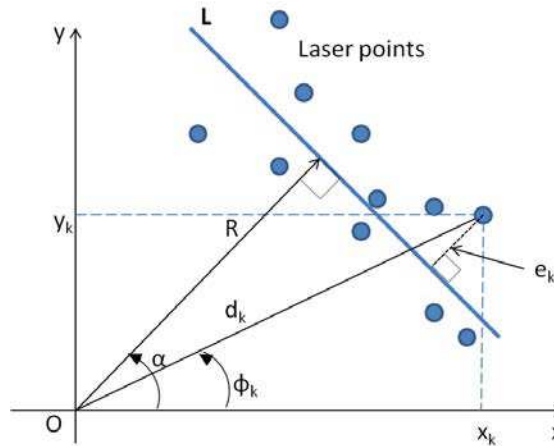


FIGURE 5.13 – Line parameters in polar coordinates

The polar coordinates d_k and ϕ_k of a point can be converted from its Cartesian coordinates (x_k, y_k) by :

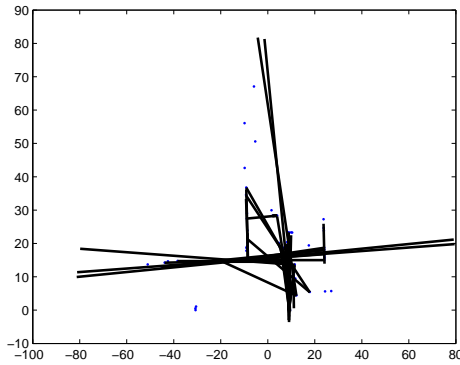
$$d_k = \sqrt{x_k^2 + y_k^2}, \quad \phi_k = \text{atan}(y_k/x_k) \quad (5.9)$$

As seen in Fig. 5.13, a straight line L in the polar coordinate system is represented by its perpendicular distance R (called the radial distance) from the origin (pole) to the line, and by the angle α (called the polar angle) from the Ox direction to the perpendicular line of the line L . The points on a line can be written as :

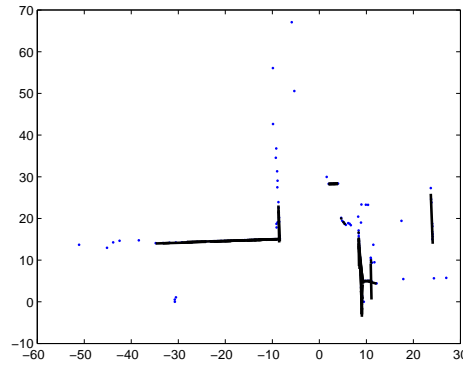
$$R = x \cos \alpha + y \sin \alpha \quad (5.10)$$

Hough transform is firstly used to detect and locate straight lines $L(R, \alpha)$ in the polar coordinate system. Since Hough transform fits lines in parameter space without using

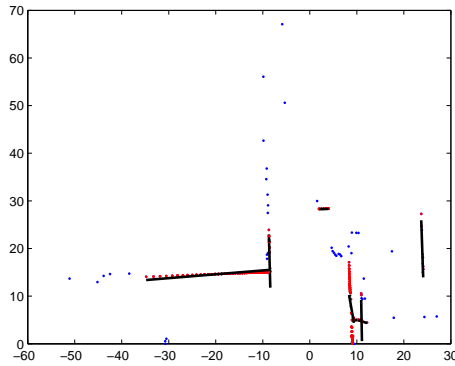
local information of the points, the points belonging to the same parameter combination (R_k, α_k) might be far away from each other (see Fig. 5.14(a)). A distance constraint is added to delete those points far away from the adjacent points (line 2 in Algorithm 3, see Fig. 5.14(b)). Besides, in order to avoid the problem due to discretization of R and α , co-planar lines for which $|\Delta R| < 0.2m$ and $|\Delta\alpha| < 0.5^\circ$ are grouped as a new line (line 3 in Algorithm 3, see Fig. 5.14(c)). Finally, the point number threshold n_{th} is used again to guarantee the number of points on every extracted line (see Fig. 5.14(d)).



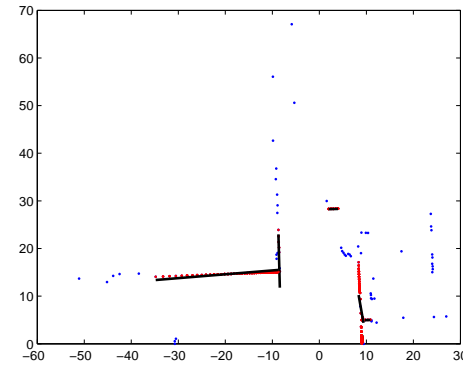
(a) Lines extracted with HT



(b) Lines extracted with HT and distance constraint



(c) Lines extracted with HT and distance constraint after line merging (red lines)



(d) Lines extracted with HT and distance constraint after line merging and point number constraint, $n_{th} = 10$ (red lines)

FIGURE 5.14 – Lines extracted from a horizontal LRF scan with constrained Hough transform

Weighted line fitting with HT results

As show in Fig. 5.13, e_k is the distance from the k^{th} laser point to the fitted line L , written as :

$$e_k = d_k \cos(\alpha - \phi_k) - R \quad (5.11)$$

The line fitting problem to estimate the line $L(R, \alpha)$ can be solved by minimizing the error e_k for a set of laser points. Least-square method fits the line under the assumption that each data point carries the equal weight. Due to the noise inherent to the laser system and the uncertainties caused by the environmental effects, the uncertainties of different

laser range measurements are not the same. This inherent noise can be considered using weighted line fitting method [122] with maximum likelihood based line fitting approach by taking into account the covariance of each e_k . Thus, the contribution of every point for the line fitting is not uniform, but linked to the covariance matrix P_k of each e_k . The cost function is written as :

$$\epsilon = \sum_{k=1}^n r_k (e_k)^2 \quad (5.12)$$

where n is the number of laser points used for line fitting, r_k is the weight for each e_k , $r_k = (P_k)^{-1}$. P_k is approximated by first-order propagation of the laser measurement error with the function e_k in Equation 5.11 :

$$P_k = H_e \begin{bmatrix} \delta d_k^2 & 0 \\ 0 & \delta \phi_k^2 \end{bmatrix} H_e^T \quad (5.13)$$

where $H_e = \begin{bmatrix} \cos(\alpha - \phi_k) & d_k \sin(\alpha - \phi_k) \end{bmatrix}$. $(\delta d_k, \delta \phi_k)$ are the standard deviations of the laser measurement (as discussed in section 3.3.2.1). As the true position (d_k, ϕ_k) of a laser point is unknown, the estimated position of laser point observed by the LRF sensor is used in the above equation. Therefore, the covariance of e_k is approximated by :

$$P_k = \cos^2(\alpha - \phi_k) \delta d_k^2 + d_k^2 \delta \phi_k^2 \sin^2(\alpha - \phi_k) \quad (5.14)$$

In order to estimate the line parameters $L(R, \alpha)$, we have to minimize the cost function of Eq. 5.12, written as :

$$\epsilon = \sum_{k=1}^n \frac{(d_k \cos(\alpha - \phi_k) - R)^2}{\cos^2(\alpha - \phi_k) \delta d_k^2 + d_k^2 \delta \phi_k^2 \sin^2(\alpha - \phi_k)} \quad (5.15)$$

Given an initial estimate of the orientation $\widehat{\alpha}$ of the line L by the Hough transform method (in section 4), the radial distance R of the line L can be estimated by Eq. 5.16.

$$R = P_{RR} \left(\sum_{k=1}^n \frac{d_k \cos(\widehat{\alpha} - \phi_k)}{P_k} \right) \quad (5.16)$$

where P_{RR} is the variance of the radial distance, calculated by $P_{RR} = (\sum_{k=1}^n P_k^{-1})^{-1}$, P_k is calculated by setting $\alpha = \widehat{\alpha}$ in Eq. 5.14.

Then, the orientation α of the line L is updated by $\alpha = \widehat{\alpha} + \delta\alpha$, where $\delta\alpha$ is defined by :

$$\delta\alpha = - \frac{\sum_{k=1}^n (b_k(0) a'_k(0) - a_k(0) b'_k(0)) / b_k(0)^2}{\sum_{k=1}^n G''_T(0)} \quad (5.17)$$

where a'_k and b'_k are respectively the derivatives of a_k and b_k with respect to $\delta\alpha$.

$$\begin{aligned} c_k &= \cos(\widehat{\alpha} + \delta\alpha - \phi_k) \\ s_k &= \sin(\widehat{\alpha} + \delta\alpha - \phi_k) \\ a_k(\delta\alpha) &= (d_k c_k - R)^2 \\ b_k &= \delta d_k^2 c_k^2 + \delta \phi_k^2 d_k^2 s_k^2 \\ a'_k(\delta\alpha) &= -2d_k s_k * (d_k c_k - R) \\ a''_k(\delta\alpha) &= 2d_k^2 s_k^2 - 2 * d_k c_k (d_k c_k - R) \\ b'_k(\delta\alpha) &= 2(d_k^2 \delta \phi_k^2 - \delta d_k^2) c_k s_k \\ b''_k(\delta\alpha) &= 2(d_k^2 \delta \phi_k^2 - \delta d_k^2) (c_k^2 - s_k^2) \\ G'_T(0) &= (b_k(0) a'_k(0) - a_k(0) b'_k(0)) / b_k(0)^2 \\ G''_T(0) &= (((a''_k(0) b_k(0) - a_k(0) b''_k(0)) b_k(0) - 2(a'_k(0) b_k(0) - a_k(0) b'_k(0)) b'_k(0)) / (b_k(0))^3) \end{aligned} \quad (5.18)$$

After that, R and α are iteratively calculated with Eq. 5.16 and Eq. 5.17 till $\delta\alpha < 1e^{-6}$.

The covariance P_L of the extracted line L is approximated by :

$$P_L = \begin{bmatrix} P_{RR} & P_{R\alpha} \\ P_{R\alpha} & P_{\alpha\alpha} \end{bmatrix} \quad (5.19)$$

where :

$$\begin{aligned} P_{R\alpha} &= \frac{P_{RR}}{G_T''} \sum_{k=1}^n \left(\frac{2d_k \sin(\alpha - \phi_k)}{b_k(0)} \right) \\ P_{\alpha\alpha} &= \frac{1}{(G_T'')^2} \sum_{k=1}^n \left(\frac{4d_k^2 \sin^2(\alpha - \phi_k)}{b_k(0)} \right) \end{aligned} \quad (5.20)$$

Finally, lines extracted from the current LRF scan are merged according to the difference between every two lines and their covariances. For a line $L_i = (R_i, \alpha_i)$ and a line $L_j = (R_j, \alpha_j)$ respectively with the line covariances P_{L_i} and P_{L_j} , the difference of their line parameters is :

$$\begin{aligned} \Delta L &= L_i - L_j \\ \delta_L &= \Delta L^T (P_{L_i} + P_{L_j})^{-1} \Delta L \end{aligned} \quad (5.21)$$

If the normalized difference δ_L is less than a predefined threshold from chi-square table, the two lines are merged as a new line L with covariance P_L by :

$$\begin{aligned} P_L &= ((P_{L_i})^{-1} + (P_{L_j})^{-1})^{-1} \\ L &= P_L ((P_{L_i})^{-1} L_i + (P_{L_j})^{-1} L_j) \end{aligned} \quad (5.22)$$

Line detection in horizontal LRF scan Line features are extracted from every horizontal LRF scan with the above line extraction method. In our work, the interval of radial distance r_{th} is set to $0.02m$ and the resolution of polar angle α_{th} is set to 1° , the number threshold of a line cluster n_{th} is set to 5.

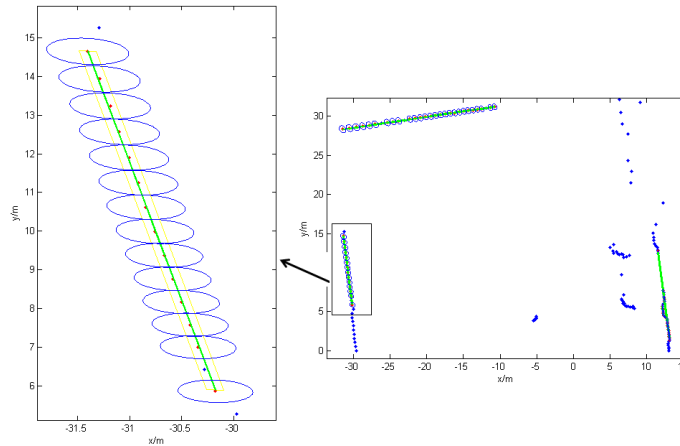


FIGURE 5.15 – Lines detected in a horizontal LRF scan

In Fig. 5.15, the red points are those points chosen to estimate the straight line by Hough transform, the estimated lines are shown in blue. The green lines are obtained after taking into account the uncertainties of each laser point with the weighted line fitting approach. And the yellow zone represents the radial distance uncertainty of each fitted line. One line is zoomed as shown in the left part of Fig. 5.15.

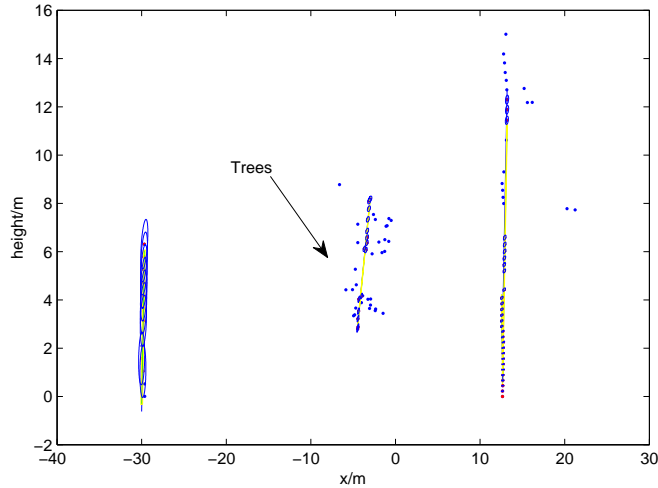


FIGURE 5.16 – Lines extracted from a vertical LRF scan : two building lines are respectively detected on the left and right sides of the vehicle, the line on the tree is eliminated

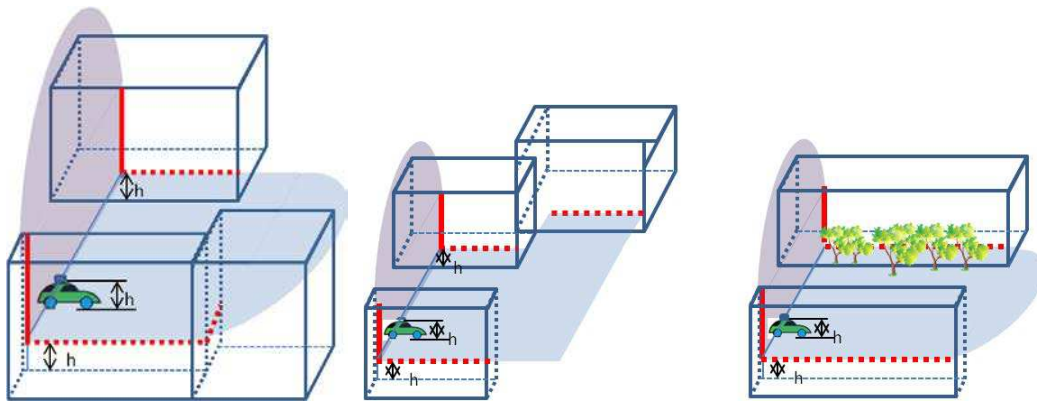
Line detection in vertical LRF scan

Straight lines in vertical LRF scan are also extracted, by setting the interval of radial distance as $0.05m$ and the number threshold of line cluster as 5. Since we assume that the building facades are vertical, the extracted lines from vertical LRF scan should be perpendicular to the ground, with the polar angle $\alpha : |\alpha| < 10^\circ$. Besides, the extracted lines are merged as a new line L if the Mahalanobis distance δ_L between two lines is less than a predefined threshold (Fig. 5.16).

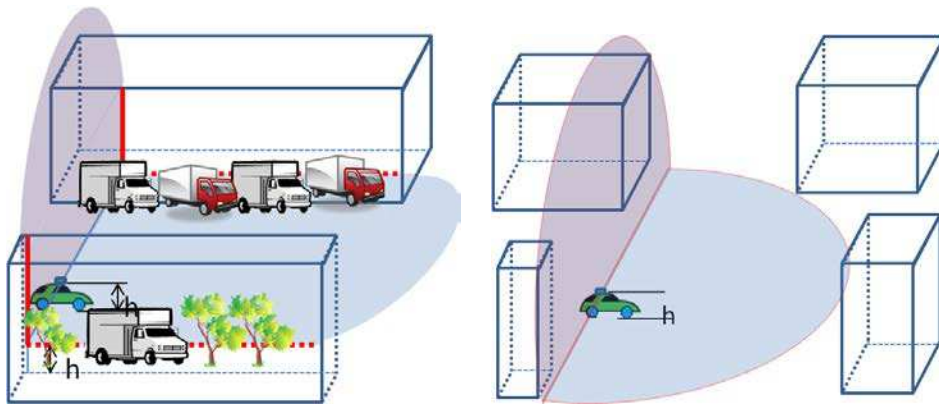
5.3.1.2/ ENVIRONMENT ANALYSIS USING DETECTED FEATURES

As shown in Fig. 5.17, the horizontal LRF mounted on the roof of the vehicle scans on a plane which is h meters above the ground, and another LRF scans vertically upward. The ideal situation occurs when there is no dynamic obstacle around the vehicle, as shown in Fig. 5.17(a) and Fig. 5.17(b).

The relative position between the horizontal and vertical LRFs can be known before the experiment by calibration. After extracting the straight lines from the horizontal and vertical



(a) If more than 2 lines are detected in LRF-H scan : apply orientation and lateral/longitudinal lateral pose correction
 (b) If parallel lines or 1 line are detected in LRF-H scan : apply orientation and lateral/longitudinal lateral pose correction ; and lines detected in LRF-V scan : apply position correction



(c) If vertical lines are detected in LRF-V scan and no line is detected in LRF-H scan : apply RF-V based lateral correction
 (d) If no lines are detected in LRF-H or LRF-V scan : no pose correction

FIGURE 5.17 – Environment analysis with horizontal and vertical LRFs

LRF scans, the selection of the pose correction method is described in Algorithm 4.

Algorithm 4: *To_analyze_the_environment*

Input : *LRF – H scan, LRF – V scan*

- 1 **if** *Non-parallel lines are detected in LRF-H then*
 - 2 Apply orientation/lateral/longitudinal error correction (case (a) of Fig. 5.17, section 5.3.2);
 - 3 **else if** *Only one line or parallel lines are detected in LRF-H then*
 - 4 Apply orientation correction (section 5.3.2.4);
 - 5 **if** *Two building lines are detected in LRF-V then*
 - 6 Apply lateral correction (case (b) of Fig. 5.17, sections 5.3.2.4 and 5.3.3)
 - 7 **else if** *no line is detected in LRF-H then*
 - 8 **if** *Two building lines are detected in LRF-V then*
 - 9 Apply lateral correction (case (c) of Fig. 5.17, section 5.3.3);
 - 10 **else**
 - 11 No correction (case (d) of Fig. 5.17);
 - 12 **end**
 - 13 **end**
-

- If few obstacles or small obstacle is around the vehicle (Fig. 5.17(a)), according to the current sensor configuration of our LRFs, the horizontal LRF can scan the building facades on the left and right sides and also in the front of the vehicle, the extracted building facades can then be associated with the building map and used to correct the vehicle longitudinal/lateral position and the orientation (see section 5.3.2).
- When there are a lot of obstacles in front of the vehicle, if only one line or parallel lines are detected in the horizontal LRF scan (Fig. 5.17(b)) and no line can be detected in the LRF-V scan, the orientation of the vehicle is corrected (section 5.3.2.4).
- After the previous step, if there are no high obstacles beside the vehicle, the building facades can be detected by the LRF-V scan and associated with the building map (Fig. 5.17(c)), then the vehicle lateral position can be corrected (section 5.3.3).
- If the buildings on the left and right sides of the vehicle are detected in both the horizontal and vertical scans, redundant information are provided to correct the vehicle lateral position. We prefer to use the vertical scan because there are less obstacles in the air.
- If neither the horizontal LRF nor the vertical LRF can detect any building facade (Fig. 5.17(d)), the vehicle is assumed to be in an open area. In this situation, the buildings are supposed to be far away, the GPS signals can usually be well received and thus no correction is needed.

5.3.2/ HORIZONTAL LRF BASED VEHICLE POSE CORRECTION

Steps of horizontal LRF based vehicle pose correction method are shown in Fig. 5.18.

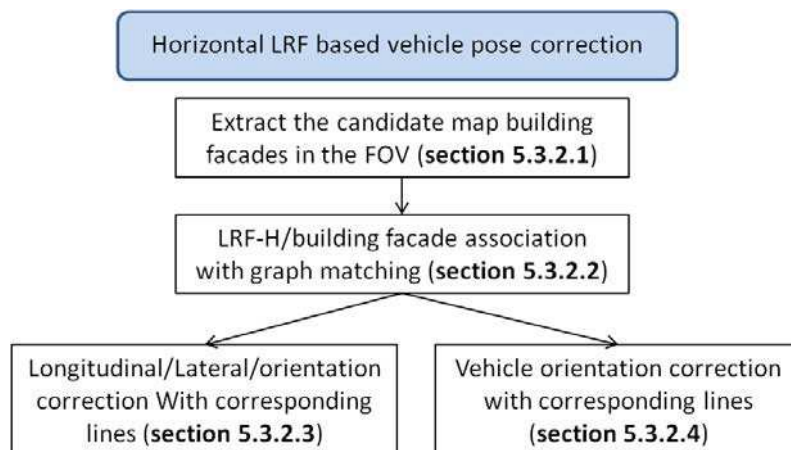


FIGURE 5.18 – Method of vehicle pose correction with non-parallel lines

Assuming that the initially estimated vehicle pose $\widehat{X}_t = (\widehat{x}_t, \widehat{y}_t, \widehat{\theta}_t)$ is with small error, candidate building facades in the field of view (FOV) of the horizontal LRF are extracted from the GIS building map (section 5.3.2.1), and transformed from the global coordinate system to the current vehicle LRF frame. The accuracy of the extracted building facades is provided by the GIS cartographer (the radial distance error is set to $0.5m$).

Then, the extracted lines from the LRF-H scan and GIS map are associated by graph matching and maximum clique searching method (section 5.3.2.2). If non-parallel lines are associated, the correction step is to find the best rotation $\delta\theta$ and translation vector

$(\delta x, \delta y)$ to transform the LRF features to the corresponding map features in the global reference frame (section 5.3.2.3) ; if only one line or parallel lines are detected in the LRF-H scan, the vehicle orientation can be corrected (section 5.3.2.4) ; if no line is detected in the LRF-H scan, the pose might be corrected with the vertical LRF in the next step (section 5.3.3).

5.3.2.1/ CANDIDATE BUILDING FACADES EXTRACTION

In order to extract the candidate building facades in the FOV of the horizontal LRF, two constraints are used :

1. the maximum range constraint of the LRF system : the maximum range of a LRF system (e.g. 180° and $80m$ range) is intersected with the building map to extract candidate building facades in the FOV ;
2. the occlusion constraint : the candidate building facades should be in the LOS (line of sight) of the laser beam.

Then, the lines extracted from the buildings map are transformed from the global map system into the current vehicle LRF frame.

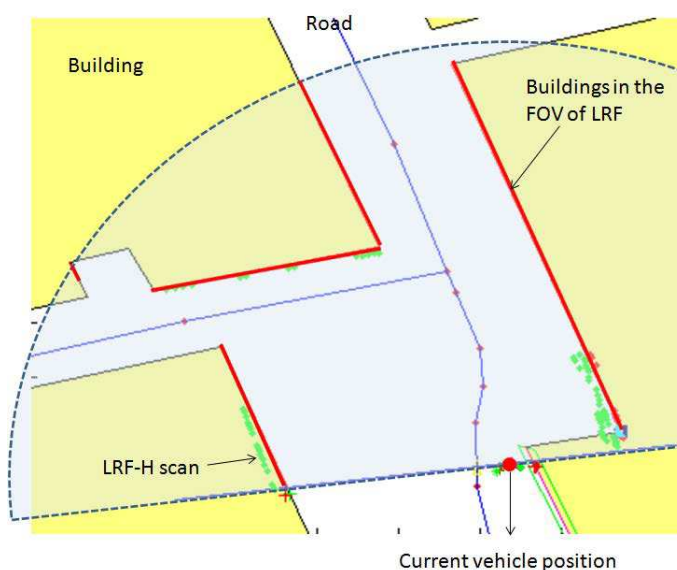


FIGURE 5.19 – Extraction of candidate building facades in the FOV of LRF-H scan

As seen in Fig. 5.19, for a horizontal LRF with maximum range of $80m$, six candidate building facades are in the field of view of the LRF-H. The extracted lines are marked in red.

5.3.2.2/ ASSOCIATION OF LRF-H AND MAP OBSERVATIONS

Line features detected in the LRF-H scan are associated with the candidate lines from the building map by graph matching and maximum clique searching method [9]. A graph is an ordered pair $G = (V, E)$ comprising a set of nodes V together with a set of edges E . A complete subgraph is a part of graph G in which all the nodes are connected to each

other. A clique C of graph G is the maximal complete subgraph of G . A maximal clique is a clique which includes the largest possible number of connected nodes [62].

At first, the feature graphs of the LRF-H observation and the map candidates are respectively generated (section 5.3.2.2); then, a correspondence graph is generated on the basis of the two feature graphs (section 5.3.2.2); after that, the maximum clique is searched in the correspondence graph to find the clique with the largest number of line correspondences between the LRF-H and the map observations (section 5.3.2.2).

The feature graphs

The feature graphs of LRF scan and map are respectively generated. For the two feature graphs, the graph nodes are features (lines), graph edges are defined by the geometric relationship between every two features (the difference of the radial distances of lines and the difference of polar angles of lines).

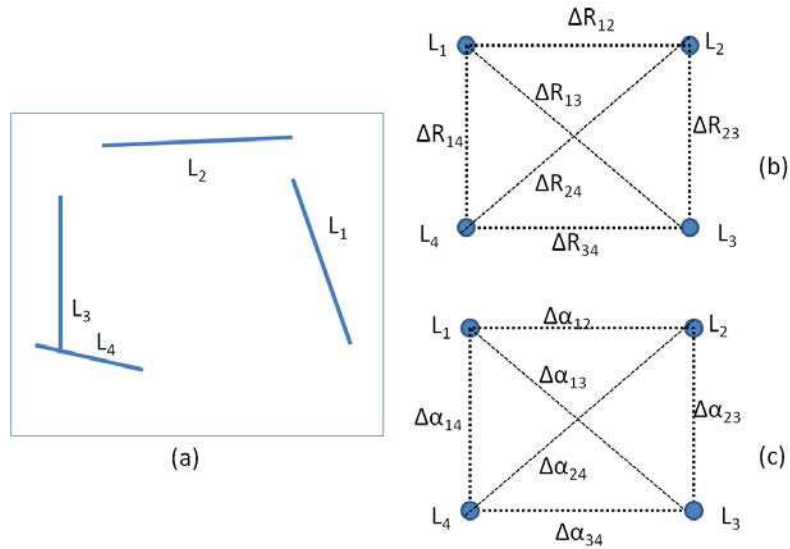


FIGURE 5.20 – (a) four line features in a LRF scan ; (b) line feature graph G_{A1} defined by the radial distances ; (c) line feature graph G_{A2} defined by the polar angles

Let $\{L_i(R_i, \alpha_i)\}, i = 1, \dots, n_L$ be the lines in the LRF-H scan, and $\{L_i^m(R_i^m, \alpha_i^m)\}, i = 1, \dots, n_L^m$ be the lines extracted from the map. For example, in Fig. 5.20(a), there are four line features extracted from the LRF scan, two feature graphs are generated : G_{A1} (Fig. 5.20(b)) is the radial distance graph with four nodes and six edges, where the edges ΔR_{ij} are defined by the difference of radial distances between every two lines L_i and L_j , $\Delta R_{ij} = R_i - R_j$; the edges G_{A2} (Fig. 5.20(c)) is the polar angle graph with the same nodes as G_{A1} , where the six edges $\Delta \alpha_{ij}$ are defined by the difference of the polar angles between every two lines, as $\Delta \alpha_{ij} = \alpha_i - \alpha_j$.

Fig. 5.21(a) shows the eight lines extracted from the building map. Then feature graphs G_{B1} and G_{B2} are generated as shown in Fig. 5.21(b)(c). The edges ΔR_{ij}^m in G_{B1} are defined by the difference of radial distances between two lines L_i^m and L_j^m , as $\Delta R_{ij}^m = R_i^m - R_j^m$; $\Delta \alpha_{ij}^m$ in G_{B2} are defined by the difference of the polar angles, as $\Delta \alpha_{ij}^m = \alpha_i^m - \alpha_j^m$.

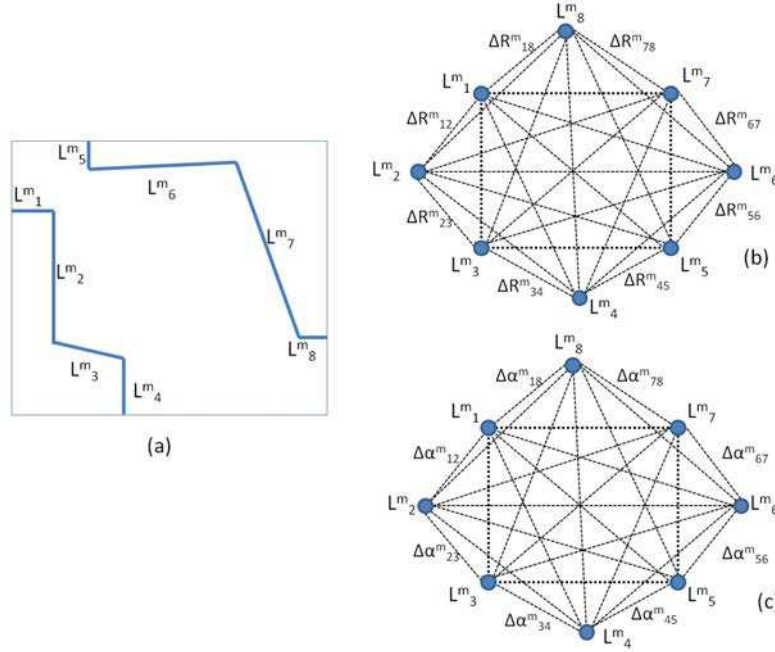


FIGURE 5.21 – (a) eight building lines are extracted from the GIS building map layer ; (b) line feature graph G_{B1} defined by the radial distances ; (c) line feature graph G_{B2} defined by the polar angles

The correspondence graph

The correspondence graph of LRF scan and building map observation is generated by finding the edge matches in the feature graphs $\{G_{A1}, G_{B1}\}$ and in the graphs $\{G_{A2}, G_{B2}\}$. If G_{A1} and G_{A2} have n_a nodes, G_{B1} and G_{B2} have n_b nodes, the correspondence graph G_{AB} is initialized as a zero array with $(n_a n_b) \times (n_a n_b)$ dimension. G_{AB} is an undirected graph in which edges have no directions and the adjacency matrix is symmetric.

If edge $\Delta R_{i_1 j_1}$ in the feature graph G_{A1} , and edge $\Delta R^m_{i_2 j_2}$ in the feature graph G_{B1} obey the following constraint :

$$\left| \Delta R_{i_1 j_1} - \Delta R^m_{i_2 j_2} \right| < 2.0m \quad (5.23)$$

and edge $\Delta \alpha_{i_1 j_1}$ in the feature graph G_{A2} and $\Delta \alpha^m_{i_2 j_2}$ in the feature graph G_{B2} obey the following constraint :

$$\left| \Delta \alpha_{i_1 j_1} - \Delta \alpha^m_{i_2 j_2} \right| < 0.1 \text{ radians} \quad (5.24)$$

The above equations suggest that there is one edge that connects the node (A_{i_1}, B_{j_2}) and the node (A_{j_1}, B_{j_2}) , together with one edge that connects the node (A_{i_1}, B_{j_2}) and the node (A_{j_1}, B_{i_2}) . This information is added into the undirected correspondence graph G_{AB} . The correspondence graph of the graphs in Fig. 5.20 and Fig. 5.21 is shown in Fig. 5.22. There are 16 nodes and 12 edges in the correspondence graph.

The maximum clique of the correspondence graph

The maximum clique of the correspondence graph is searched to find the maximum common subgraph. Given a graph's Boolean adjacency matrix G_{AB} , *Bron – Kerbosch* algorithm

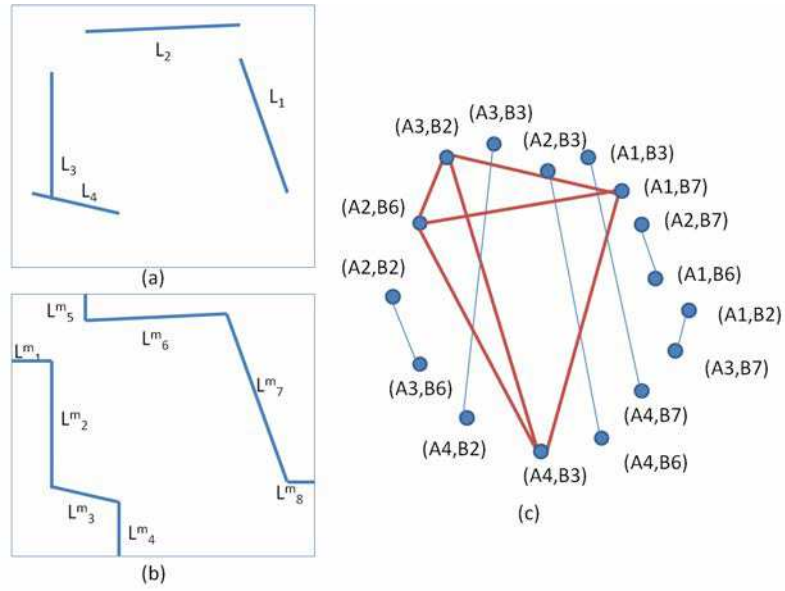


FIGURE 5.22 – (a) observed building lines in the LRF-H scan ; (b) extracted building lines in the FOV of LRF-H from GIS building map layer ; (c) the correspondence graph and the maximum clique of the correspondence graph (red lines)

[40] is used to find all maximal cliques of the undirected graph. This algorithm applies a recursive backtracking procedure to augment a candidate clique by considering one node at a time. The considered node is either added to the candidate clique or to the excluded nodes. Fig. 5.22 (c) shows the maximum clique of the correspondence graph in which all the four nodes are connected with each other, this result indicates four line correspondences between the LRF-H and map observations : $\{(L_3, L_2^m), (L_2, L_6^m), (L_1, L_7^m), (L_4, L_3^m)\}$.

5.3.2.3/ LONGITUDINAL/LATERAL POSITION AND ORIENTATION CORRECTION

Based on the initially estimated vehicle pose, the correction step is to find the best rotation $\delta\theta$ and translation $(\delta x, \delta y)$ to transform the LRF features to the corresponding map features in the global reference frame. The maximum likelihood estimation method is used by considering both the precision of the lines detected in the LRF-H scan (see Equation 5.19) and the precision of the line extracted from the map. The precision of the extracted building facades is provided by the GIS cartographer (as described in section 5.2.1.2, it is set to $0.5m$).

With N pairs of line correspondences : $\{L_i(R_i, \alpha_i), L_i^m(R_i^m, \alpha_i^m)\}$, $i = 1, \dots, N$ in the LRF scan and in the map, the error Δl_i between two lines is written as :

$$\Delta l_i = \begin{bmatrix} R_i + \delta x \cos(\alpha_i + \delta\theta) + \delta y \sin(\alpha_i + \delta\theta) - R_i^m \\ \alpha_i + \delta\theta - \alpha_i^m \end{bmatrix} \quad (5.25)$$

The cost function E to be minimized is written as :

$$E = \sum_{i=1}^N \frac{1}{2} (\Delta l_i)^T w_i (\Delta l_i) = \sum_{i=1}^N \frac{1}{2} (\Delta l_i)^T (P_{\Delta l_i})^{-1} (\Delta l_i) \quad (5.26)$$

where w_i is the inverse of covariance matrix $P_{\Delta l_i}$ of Δl_i , $w_i = (P_{\Delta l_i})^{-1}$. As the radial distance and the polar angle are not in the same scale, we separate this problem into two steps :

- 1) The cost function E_α is written as :

$$E_\alpha = \sum_{i=1}^N \frac{1}{2} w_i^\alpha (\alpha^i + \delta\theta - \alpha_m^i)^2 \quad (5.27)$$

The local minimum of Equation 5.27 with respect to the rotation angle θ satisfies :

$$\frac{\partial E_\alpha}{\partial \delta\theta} = \sum_{i=1}^N w_i^\alpha (\alpha^i + \delta\theta - \alpha_m^i) = 0 \quad (5.28)$$

where $w_i^\alpha = (P_{\alpha^i} + P_{\alpha_m^i})^{-1} = \frac{1}{P_{\alpha^i} + P_{\alpha_m^i}}$. Thus,

$$\delta\theta = \frac{\sum_{i=1}^N w_i^\alpha (\alpha_m^i - \alpha^i)}{\sum_{i=1}^N w_i^\alpha} \quad (5.29)$$

The variance of orientation is $P_{\delta\theta} = (\sum_{i=1}^N (P_{\alpha^i} + P_{\alpha_m^i})^{-1})^{-1}$.

- 2) After estimating the value of orientation $\delta\theta$, the translation δx is estimated with cost function E_x , as :

$$E_x = \sum_{i=1}^N \frac{1}{2} w_i^x (R^i \cos(\alpha^i + \delta\theta) + \delta x - R_m^i \cos(\alpha_m^i))^2 \quad (5.30)$$

then, the vehicle x-translation is estimated by :

$$\delta x = \frac{\sum_{i=1}^N w_i^x (R_m^i \cos(\alpha_m^i) - R^i \cos(\alpha^i + \delta\theta))}{\sum_{i=1}^N w_i^x} \quad (5.31)$$

where

$$\begin{aligned} \epsilon_{x_i} &= R^i \cos(\alpha^i + \delta\theta) + \delta x - R_m^i \cos(\alpha_m^i) \\ &= (R^i + \sigma_R) \cos(\alpha^i + \sigma_\alpha + \theta + \sigma_\theta) + \delta x - (R_m^i + \sigma_{R_m}) \cos(\alpha_m^i + \sigma_{\alpha_m}) \\ &= \sigma_R \cos(\alpha^i + \delta\theta) - R^i \sin(\alpha^i + \delta\theta) (\sigma_\alpha + \sigma_\theta) - \sigma_{R_m} \cos(\alpha_m^i) + \sigma_{\alpha_m} R_m^i \sin(\alpha_m^i) \end{aligned} \quad (5.32)$$

Then,

$$P(\epsilon_{x_i}) = P_{RR^i} \cos(\alpha^i + \delta\theta)^2 + P_{\alpha^i} (R^i)^2 \sin(\alpha^i + \delta\theta)^2 + P_{\delta\theta} (R^i)^2 \sin(\alpha^i + \delta\theta)^2 + P_{R_m^i} \cos(\alpha_m^i)^2 + P_{\alpha_m^i} (R_m^i)^2 \sin(\alpha_m^i)^2 \quad (5.33)$$

and $w_i^x = 1/P(\epsilon_{x_i})$, $P_{\delta x} = \sum_{i=1}^N (1/w_i^x)$.

- 3) The cost function E_y is written as :

$$E_y = \sum_{i=1}^N \frac{1}{2} w_i^y (R^i \sin(\alpha^i + \delta\theta) + \delta y - R_m^i \sin(\alpha_m^i))^2 \quad (5.34)$$

then, the vehicle y-translation is estimated by :

$$\delta y = \frac{\sum_{i=1}^N w_i^y (R_m^i \sin(\alpha_m^i) - R^i \sin(\alpha^i + \delta\theta))}{\sum_{i=1}^N w_i^y} \quad (5.35)$$

where

$$P(\epsilon_{y_i}) = P_{RR^i} \sin(\alpha^i + \delta\theta)^2 + P_{\alpha\alpha^i} (R^i)^2 \cos(\alpha^i + \delta\theta)^2 + P_{\delta\theta} (R^i)^2 \cos(\alpha^i + \delta\theta)^2 + P_{R_m^i} \sin(\alpha_m^i)^2 + P_{\alpha_m^i} (R_m^i)^2 \cos(\alpha_m^i)^2 \quad (5.36)$$

and $w_i^y = 1/P(\epsilon_{y_i})$, $P_{\delta y} = \sum_{i=1}^N (1/w_i^y)$.

With the estimated rotation matrix and translation vector, the corrected vehicle pose is written as : $(x, y, \theta) = (\hat{x} + \delta x, \hat{y} + \delta y, \hat{\theta} + \delta\theta)$.

5.3.2.4/ VEHICLE ORIENTATION CORRECTION

If only one straight line or more than two parallel lines are detected in the LRF-H scan, and no line is detected in the LRF-V scan, the vehicle orientation $\hat{\theta}$ is corrected by the following steps as shown in Fig. 5.23.

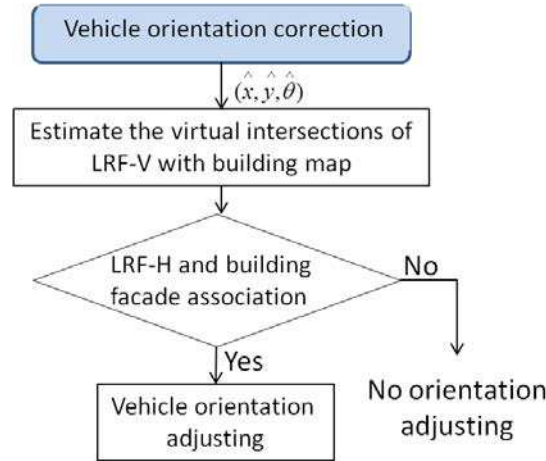


FIGURE 5.23 – Method of orientation correction with one line/parallel lines from LRF-H scan

Virtual intersections estimation

This step is to estimate the virtual intersections of the vertical LRF scan with the building map on the left and right sides. The vertical LRF scans are in a vertical plane determined by the current vehicle position and orientation. Since we need to extract the building facades in the FOV of the LRF-H, the intersections of the vertical laser scanning plane and the buildings are two vertical lines on the building facades as seen in Fig. 5.24. The candidate intersections are shown by two red points.

As seen in Fig. 5.25, let $(\overline{x_{left}}, \overline{y_{left}})$ and $(\overline{x_{right}}, \overline{y_{right}})$ respectively denote the virtual intersections of the vertical LRF scan with the building map on the left and right sides.

With two building segments $[AB]$ and $[CD]$ (Fig. 5.25), the candidate intersections are calculated by respectively setting the laser incident angle θ to $(\hat{\theta} + 0)$ and $(\hat{\theta} + \pi)$. The coordinates of the intersection points $(\overline{x_{left}}, \overline{y_{left}})$ and $(\overline{x_{right}}, \overline{y_{right}})$ are respectively estimated

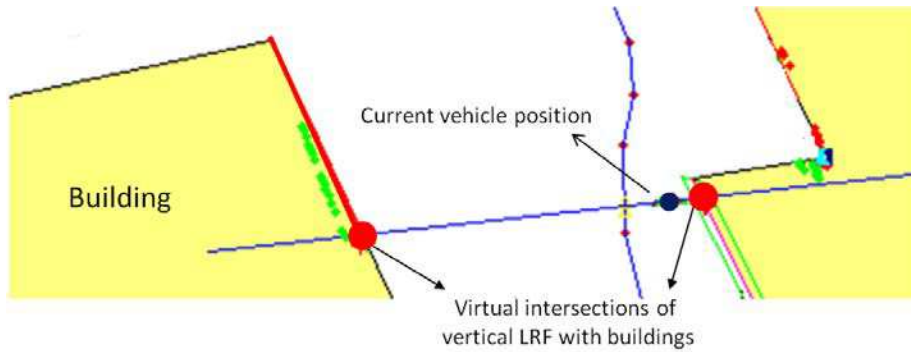


FIGURE 5.24 – Virtual intersections of the LRF-V scan with building facades

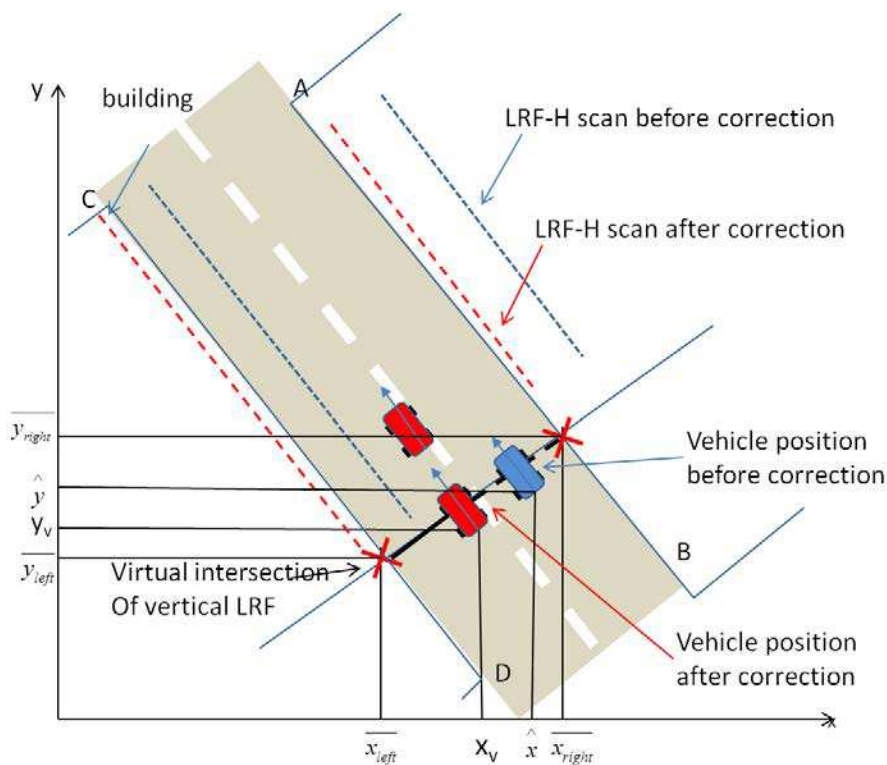


FIGURE 5.25 – Lateral position correction with parallel lines

by Equation 5.37.

$$\left\{ \begin{array}{ll} a = \tan(\hat{\theta}) & b = -(a_2\hat{x} - \hat{y}) \\ a_0 = \frac{A_y - B_y}{A_x - B_x} & b_0 = \frac{A_x B_y - B_x A_y}{A_x - B_x} \\ \hat{x}_{right} = \frac{b - b_0}{a_0 - a} & y_{right} = \frac{a_0 b - a_2 b}{a_0 - a} \\ a_1 = \frac{C_y - D_y}{C_x - D_x} & b_1 = \frac{C_x D_y - D_x C_y}{C_x - D_x} \\ \hat{x}_{left} = \frac{b - b_1}{a_1 - a} & y_{left} = \frac{a_1 b - a b_1}{a_1 - a} \end{array} \right. \quad (5.37)$$

Vehicle orientation adjusting

After knowing the intersections, the vehicle orientation is adjusted by LRF-H based building facades and map based facades where the intersection points belong to. The Maha-

lanobis distance between the LRF-H based facades and map building facades are calculated. Then, the polar angles of validated building facades and LRF-H based facades whose Mahalanobis distance is less than a threshold are used for vehicle orientation correction with Equation 5.38.

$$\theta = \widehat{\theta} + \frac{\sum_{i=1}^2 w_i^\alpha (\alpha_m^i - \alpha^i)}{\sum_{i=1}^2 w_i^\alpha} \quad (5.38)$$

where N_v is the number of validated facades, $w_i^\alpha = (P_{\alpha^i} + P_{\alpha_m^i})^{-1} = \frac{1}{P_{\alpha^i} + P_{\alpha_m^i}}$. If no corresponding lines are found, the vehicle orientation remains the same.

5.3.3/ VERTICAL LRF BASED VEHICLE LATERAL POSITION CORRECTION

If two building facades are detected in the vertical LRF scan, the vehicle lateral position can be corrected with the building map as shown in Fig. 5.26.

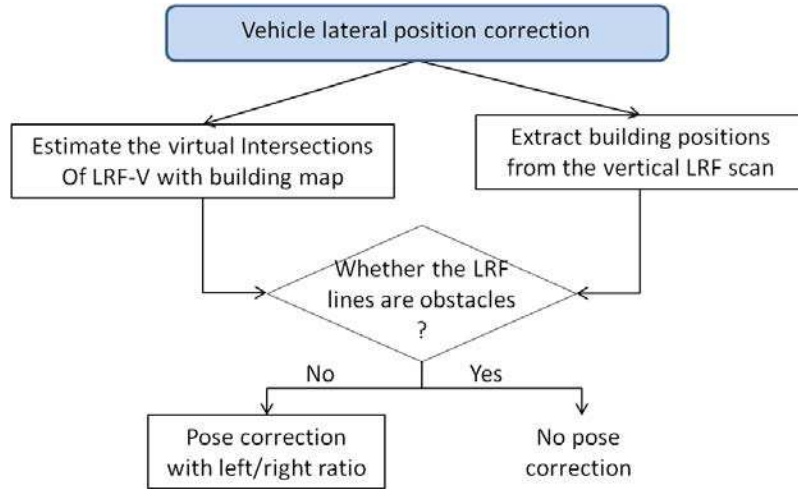


FIGURE 5.26 – Method of lateral position correction with vertical lines in LRF-V scan

- At first, the intersections of the vertical LRF scan with the building map are estimated based on the vehicle position and the corrected orientation with Equation 5.37 (if the vehicle orientation $\widehat{\theta}$ is corrected in section 5.3.2.4). The two intersections are \overline{x}_{left} and \overline{x}_{right} in Fig. 5.25, respectively with error $0.5m$;
- Then, with the extracted vertical lines $L^l(R^l, \alpha^l)$ and $L^r(R^r, \alpha^r)$ from the vertical LRF scan (as shown in Fig. 5.16), the lateral positions of building facades in local frame and their variances can be obtained by :

$$\begin{aligned} x_{left} &= R^l \cos \alpha^l \\ x_{right} &= R^r \cos \alpha^r \end{aligned} \quad (5.39)$$

Variance of the two positions $\sigma_{x_{left}}^2$ and $\sigma_{x_{right}}^2$ are propagated from the covariance of the two extracted lines P_L^l and P_r by first-order approximation ;

- After that, whether the detected LRF-V points are obstacles or not is verified by the width of road W_r which is stored in the GIS attribute table :

$$\left| |x_{right} - x_{left}| - \sqrt{(\bar{x}_{right} - \bar{x}_{left})^2 + (\bar{y}_{right} - \bar{y}_{left})^2} \right| \leq W_r \quad (5.40)$$

- If the above Eq. 5.40 is satisfied, the current estimated vehicle pose $(\hat{x}, \hat{y}, \hat{\theta})$ is adjusted by the ratio $\Psi = 1 / |x_{left} / x_{right}|$:

$$\begin{aligned} x &= (\bar{x}_{right} + \Psi \bar{x}_{left}) / (1 + \Psi) \\ y &= \tan(\hat{\theta})(x - \hat{x}) + \hat{y} \end{aligned} \quad (5.41)$$

It should be noted that as shown in Fig. 5.25, only the lateral position of the vehicle can be corrected (from the blue position to the red position), the longitudinal error cannot be bounded by the vertical LRF lines.

- With Eq. 5.41, the covariance of the corrected pose (x, y) can be measured by the covariance of current vehicle pose $P_{x,y,\theta}$, variances of the LRF-V building points $\sigma_{x_{left}}^2, \sigma_{x_{right}}^2$ and the error of map intersections with unscented transform (section 4.2.3), written as $diag(P_{\hat{x},\hat{y},\hat{\theta}}, \sigma_{x_{left}}^2, \sigma_{x_{right}}^2, 0.5^2, 0.5^2)$.

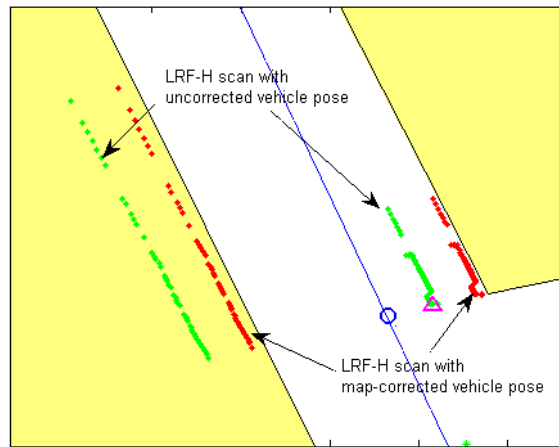


FIGURE 5.27 – LRF-H in global reference before and after the map based pose correction

Fig. 5.27 shows one horizontal laser scan before and after the map based vehicle pose correction step. The yellow polygons are buildings on the map, the blue line segments are roads on the map. The green points are laser scans transformed into the global reference system with the initial estimated vehicle pose. After the map based pose correction step, the laser observation (red points) are more consistent to the map. The initial estimated vehicle position is shown by blue circle, and the corrected vehicle position is shown by triangle in magenta color.

5.4/ NEW MAP LAYER GENERATING WITH LRFs AND GIS

In the previous sections, the vehicle pose is corrected such that the local observation from the LRF sensors can be matched with the building map. The above method can meet most of our demands for continuous vehicle localization, especially in countries with well

surveyed maps or regions with small city infrastructure changes. However, sometimes the information from different data sources might not be coherent with each other or the environment changes.

A new map layer of building facades can be generated with the onboard LRF sensors. Since there are less obstacles in the air (with LRF-V) than on the ground (with LRF-H), the new map layer is generated with the onboard vertical LRF sensor. This method looks like SLAM, which solves the problem of building a map of an unknown environment (or update an existing map) by a mobile robot while at the same time navigating the robot using the map [8]. For our work, since there is no common data between consecutive vertical scans, the SLAM method cannot be directly used. Therefore, our work is implemented in two steps as follows :

1. A new map layer of building facade landmarks is generated at first by extracting features from the vertical scan (section 5.4.1) ;
2. Then, the reconstructed map is used as measurements to correct the vehicle pose when the vehicle moves again around the same experimental area (section 5.4.2).

5.4.1/ NEW MAP LAYER OF BUILDING LANDMARKS

The vehicle is driven in an experimental area, and the vehicle poses are obtained by vision based odometry information, gyro, RTK-GPS, and map-matching method. Then, the building facade lines are detected in each vertical LRF scan, and transformed from the vehicle attached frame into the global reference system with the vehicle pose (section 5.4.1.1). These extracted landmarks are associated with the original building map (section 5.4.1.2). Then, points which are considered to be on the building facades are connected as building segments, and stored in a map layer as facade landmarks. If a point is considered to be an independent object, it is stored in an independent object map layer (section 5.4.1.3).

5.4.1.1/ UNCERTAINTY OF LRF-V OBSERVATION

The covariance of every building facade landmark extracted from the vertical LRF scan is estimated in four steps :

1. Covariance of every laser scan measurement is related to the laser range and incidence angle, the covariance matrix of k^{th} point is written as $\{\delta d_k^2, \delta \phi_k^2\}$, as described in section 3.3.2.1.
2. Covariance P_L of every fitted vertical line $L(R_i, \alpha_i)$ is related to the covariance matrices of all the laser points used for line fitting $\{\delta d_k^2, \delta \phi_k^2\}, k = 1 \dots n$, as described in Equation 5.19 ;
3. Local covariance of the extracted facade landmark in the laser scan frame is related to the covariance P_L of the extracted line. As shown in Fig. 5.28, the projection of the building line on 2D map is calculated by the parameters of the fitted line L with :

$$x_{v,i} = R_i \cos \alpha_i \quad (5.42)$$

The variance of this projection in local frame is propagated from the covariance P_L of the extracted line through :

$$\sigma x_{v,i}^2 = [\cos(\alpha_i) \quad -R_i \sin(\alpha_i)] P_L [\cos(\alpha_i) \quad -R_i \sin(\alpha_i)]^T \quad (5.43)$$

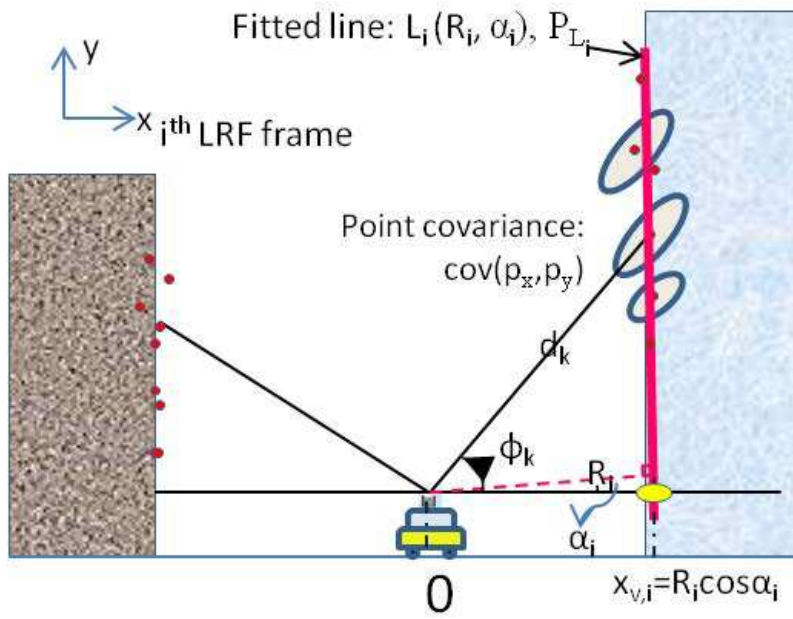


FIGURE 5.28 – Covariance of facade landmarks in i^{th} LRF frame

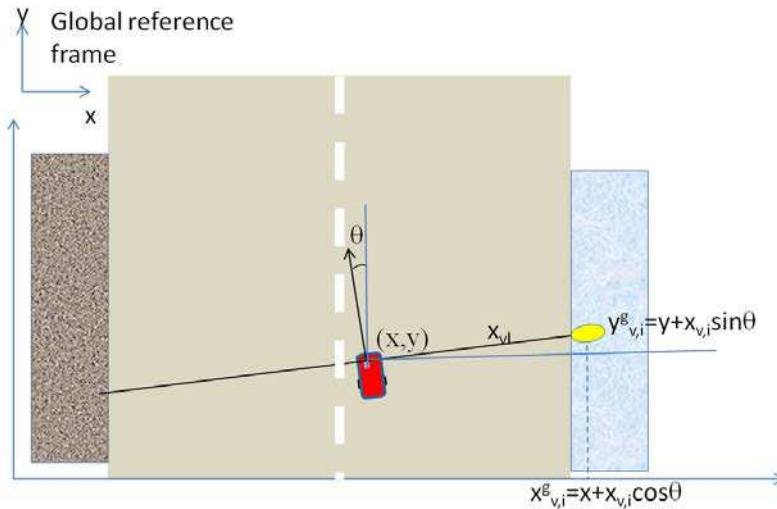


FIGURE 5.29 – Covariance of facade landmarks in global frame

4. After that, this projection is transformed into the global reference system with the current estimated vehicle pose $(\hat{x}, \hat{y}, \hat{\theta})$, as shown in Fig. 5.29 :

$$\begin{cases} x_{v,i}^g = \hat{x} + x_{v,i} \cos(\hat{\theta}) \\ y_{v,i}^g = \hat{y} + x_{v,i} \sin(\hat{\theta}) \end{cases} \quad (5.44)$$

Since the covariances $P_{\hat{x}, \hat{y}, \hat{\theta}}$ of the vehicle pose and the variance $\sigma x_{v,i}^2$ of the LRF observation are independent, the global covariance of the facade landmark is approximated with the covariance $P_{\hat{x}, \hat{y}, \hat{\theta}}$ of vehicle pose and the LRF observation variance

$\sigma x_{v,i}^2$ through first-order propagation.

$$P_{v,i}^g = H_{v,i} \begin{bmatrix} P_{\widehat{x,y,\theta}} & 0_{3 \times 1} \\ 0_{1 \times 3} & \sigma x_{v,i}^2 \end{bmatrix} H_{v,i}^T \quad (5.45)$$

where $H_{v,i} = \begin{bmatrix} 1 & 0 & -x_{v,i} \sin(\widehat{\theta}) & \cos(\widehat{\theta}) \\ 0 & 1 & x_{v,i} \cos(\widehat{\theta}) & \sin(\widehat{\theta}) \end{bmatrix}$.

5.4.1.2/ ASSOCIATION WITH THE ORIGINAL BUILDING MAP

The lines extracted from the vertical LRF are projected onto the 2D map as a set of points (Fig. 5.28 and Fig. 5.29). All the projections of the vertical laser lines are associated with a corresponding building facade in the GIS building map, as shown in Fig. 5.30. The association step is implemented by finding the projection of LRF-V points on the nearest building facade in the GIS map.

• **Detection of independent objects.** In order to separate the independent objects (like trees, street lamp, traffic sign, etc.) from the building facade landmarks, a continuity constraint is used on the basis of the original building map. The points corresponding to the same building facade should have the same distance from the building. Those landmarks far from their neighbors are considered to be independent objects (Fig. 5.31). They are stored in an “independent object” layer (Fig. 5.32).

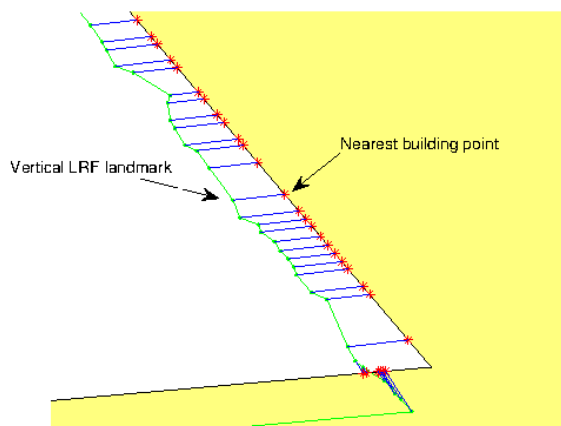


FIGURE 5.30 – Nearest building point of LRF-V on GIS building map

5.4.1.3/ ORGANIZATION OF OBJECTS IN THE NEW MAP

After obtaining the points on building facades, they are grouped into several segments to represent new building facades. Every two consecutive points are connected till :

- If an independent object is found, the previous building facade segment ends ;
- If the distance between two consecutive points is more than 4 meters, a new building facade segment starts.

A map layer is created to store the new detected facade landmarks from LRF-V. Each landmark is composed of a series of line segments as shown in Tab. 5.2 (left). The global covariance matrices of all the facade points are stored. Another map is created to store the independent objects, as in Tab 5.2 (right). Each object is represented by one point.

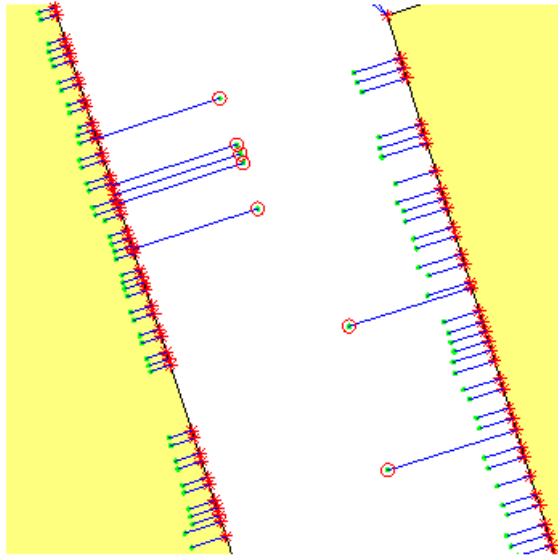


FIGURE 5.31 – Detected independent objects by LRF-V

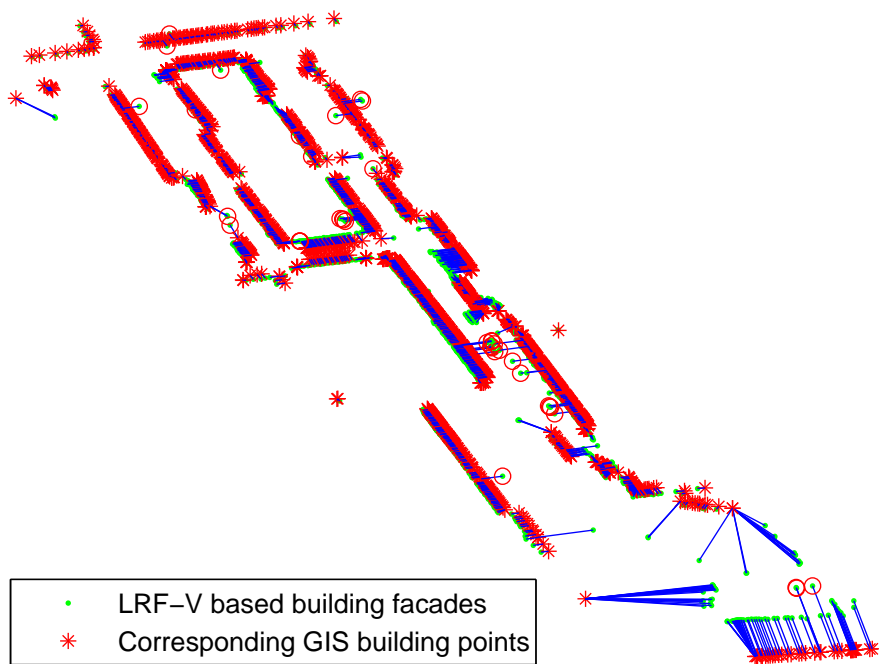


FIGURE 5.32 – Corresponding positions of LRF-V based features on GIS building map

5.4.2/ LOCALIZATION WITH NEW BUILDING FACADES MAP

Due to the appearance of obstacles, the changes in the environment, or the inherent error of the original GIS map, the LRF observations might not be associated with the original map in some area. The newly generated building facade map can better represents the real environment observed by the vertical LRF, and can be used for more accurate localization in the experimental area. In order to use the new map for localization, we update both the vehicle pose and the facade landmark map.

An outline of the localization process is given in Fig. 5.33. When the vehicle arrives at a new position, an expected observation is searched within the new building facades map : if there exists building line segments, an intersection of the building facade segment is calculated ; or an independent observation might be extracted. The localization process is detailed in the following parts.

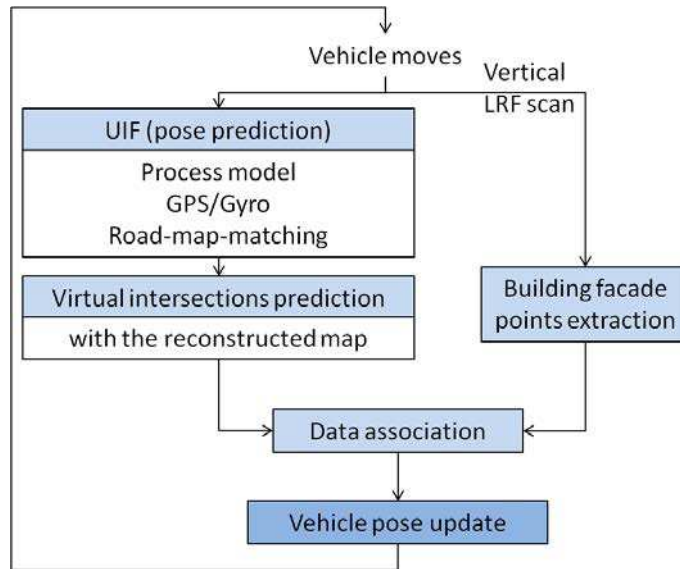


FIGURE 5.33 – Process of vehicle localization with the new generated building facade map

- **Pose prediction** : when the vehicle moves, the vehicle pose $(\hat{x}, \hat{y}, \hat{\theta})$ and its covariance are predicted with the validated sensors as presented in section 5.2.
- **Landmarks prediction** : building facade landmarks are then extracted from the re-

Building facade landmarks	Independent objects
Geometry : Line ;	Geometry : Point ;
ID ;	ID ;
X $(X_1, X_2, \dots, X_n, NaN)$;	X ;
Y $(Y_1, Y_2, \dots, Y_n, NaN)$;	Y ;
Height ;	Height ;
Covariance matrices ;	Covariance matrix ;
Corresponding original facade ;	

TABLE 5.2 – Organization of landmarks : n is the number of points on a facade segment

constructed map with the vehicle's new position, as described in section 5.3.2.4. The intersections of the vertical beams with the reconstructed facade map are calculated. One or two landmarks might be extracted, or no landmarks can be extracted if there is no intersection, $N_{landmark} = 0, 1, 2$.

The position of the intersection is not directly used, it is calculated with the end-points of the facade segments $\{A(A_x, A_y), B(B_x, B_y)\}$, or $\{C(C_x, C_y), D(D_x, D_y)\}$ as shown in Fig. 5.25. The vehicle state is a $(3 + 4 \times N_{landmark})$ dimension vector, written as : $X = \{\widehat{x}, \widehat{y}, \widehat{\theta}, A_x, A_y, B_x, B_y, C_x, C_y, D_x, D_y\}$. The covariances of the endpoints are extracted from the reconstructed building map.

• **Observation model** : if a landmark is extracted from the segment AB , the measurement used is the distance between the building facade and the vehicle, written as :

$$\begin{aligned} distance &= \sqrt{(\widehat{x} - x_{right})^2 + (\widehat{y} - y_{right})^2} + Q_m \\ &= \sqrt{1 + \tan(\widehat{\theta})^2} \frac{-(A_x - B_x)\widehat{y} + (A_x B_y - B_x A_y) + (A_y - B_y)\widehat{x}}{(A_y - B_y) - (A_x - B_x)\tan(\widehat{\theta})} \end{aligned} \quad (5.46)$$

where Q_m is the observation noise. If a landmark is extracted from the segment CD , the measurement model is the same as Eq. 5.46. Since the measurement model is nonlinear, an UKF is used to update the vehicle pose and map uncertainty (the algorithm of UKF can be found in section 4.2).

• **Measurement** : the measurements are the 2D projections of the building facade points extracted from the current vertical LRF scan, $x_{v,i}$ (Equation 5.42). One or two building points might be extracted, or no observation is available. Measurement variance $\sigma x_{v,i}^2$ is obtained as presented in Equation 5.43.

• **Update** : then the landmarks on the left and right sides of the vehicle can be associated with the current LRF observations of the building facades. They are used to update the vehicle pose and map uncertainty in the UKF.

Landmarks which have not been seen before are added into the map such that they can be re-observed later. All the new building facade points are stored without modifying the initial facade points. When the number of facade points related to a building facade reaches a specified density threshold, a new building facade segment is calculated with respect to the covariances of the points. Methods like split-and-merge can also be used to segment and merge the landmarks.

5.5/ IMPLEMENTATION AND EXPERIMENTAL RESULTS

5.5.1/ EXPERIMENTAL PLATFORM

Two experiments were implemented with the same vehicle *SeTCar* introduced in previous section 3.4. A ProFlex 500 Magellan RTK-GPS receiver, a DSP 3000 Fibre optic gyro, and two SICK *LMS291* laser range finders are mounted on the roof of the vehicle (Fig. 5.34). An embedded hard disk is installed to acquire multi-sensor data. The measurements from the different sensors are associated according to their logged time. Beside the sensors presented in section 3.4, a gyro and two LRF as used in this experiment.

Gyro. A DSP 3000 Fibre optic gyro⁷ (10Hz, Fig. 5.35) is mounted on the roof of the vehicle

7. <http://www.kvh.com/dsp3000>



FIGURE 5.34 – The experimental vehicle - SeTCar

beside the horizontal laser range finder. This gyro can sense the rotation of the vehicle around an axis perpendicular to the base plane.



FIGURE 5.35 – DSP 3000 Fibre optic gyro

Laser range finders (Fig. 5.36). In our experiments, two SICK *LMS 291* laser range finders⁸ are mounted on the roof of the vehicle : one looks forward and scans horizontally, the other one looks upward and scans vertically. By setting the angular resolution of the two LRF systems to 1° and the maximum range to 80 meters, the LRFs can send 75 scans every second, with 181 range data (a LRF scan) in 180° field of view.

In order to evaluate the effectiveness of the proposed map-aided vehicle localization approach, two experiments were designed and implemented with data acquired by our experimental vehicle.

8. <http://www.sick.com/>



FIGURE 5.36 – SICK *LMS221* and *LMS291*

5.5.2/ EXPERIMENTAL RESULTS OF THE OLD TOWN CENTER SEQUENCE

The first data sequence was acquired in the old town center of Belfort, in September, 2012 (Fig. 5.37). The whole trajectory measured by RTK-GPS is about 972.49m.



FIGURE 5.37 – Vehicle trajectory overlapped on the OpenStreetMap and Google aerial image

The streets are narrow and the buildings are tall in the old town center. It is easily to note that in several areas (Fig. 5.38), positions provided by the RTK-GPS are situated on the buildings, though the HDOP (Horizontal dilution of precision) of all the GPS positions are between 1 and 2 during this acquisition (provided by the NMEA GPGGA sentences), this means that the distribution of the satellites are good enough to provide accurate position measurements. Therefore, this error might due to the influence of the local environment.

As seen in Fig. 5.39, several RTK-GPS positions are with large noises, even overlapped on the buildings. Thus, the RTK-GPS positions cannot be used as reference for evaluation. Since no ground truth is provided in this area, we only show the corrected vehicle positions and the local perception of the vertical LRF.

In Fig. 5.39, the red lines are the GPS positions provided by the RTK-GPS receiver. The road and building maps are used to correct the vehicle pose with the local perception from the laser range finders. In our experiments, the map-matching method is based on the

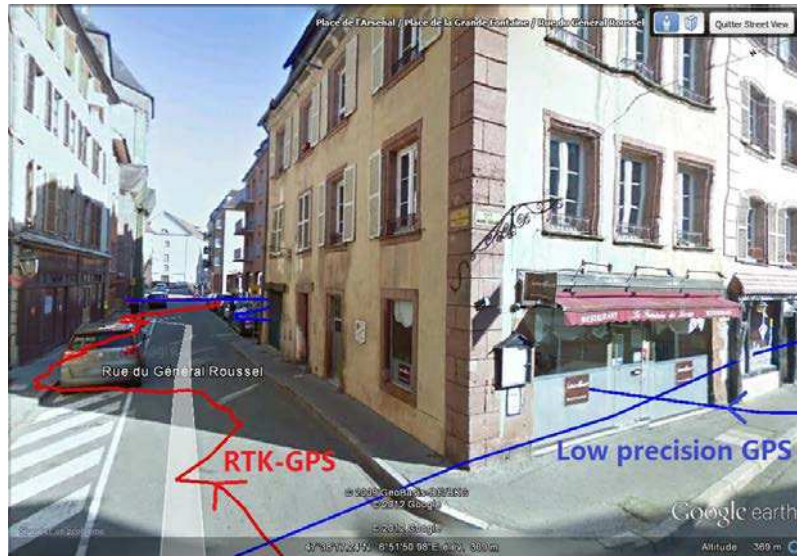


FIGURE 5.38 – Trajectory (in area 1) of RTK-GPS and low precision GPS shown in Google Street View

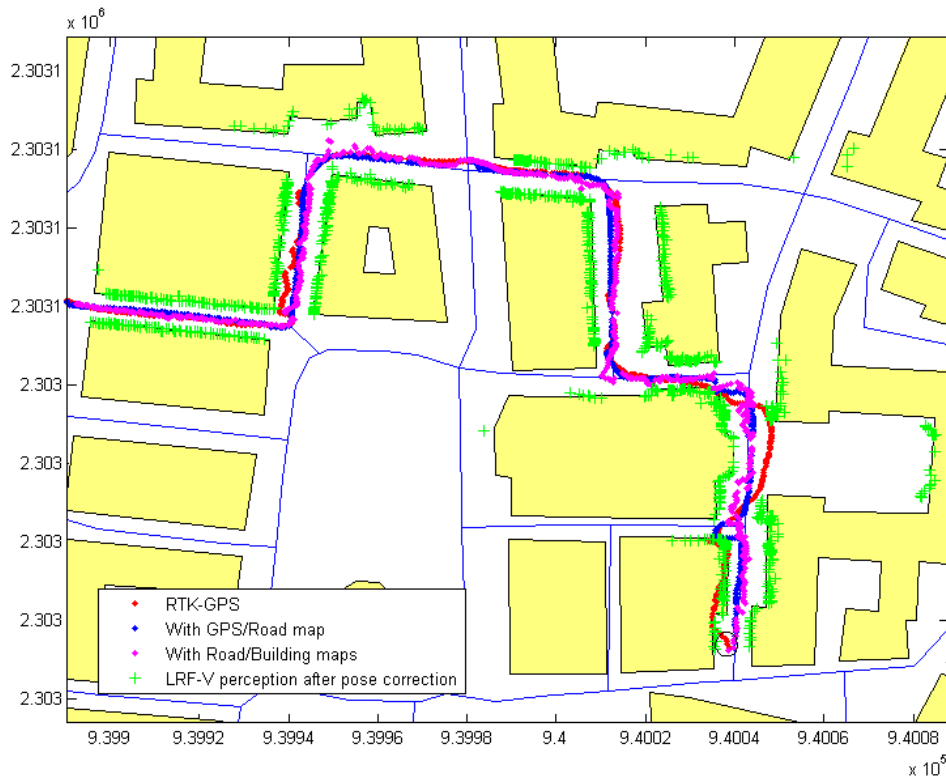


FIGURE 5.39 – Vehicle trajectories estimated by different approaches : vehicle position used for map-matching method is based on the GPS position if the GPS reading is available ; if not, it is based on the predicted position from the transition model

GPS position if the GPS reading is available ; if not, it is based on the predicted position from the transition model. The vehicle positions corrected by the road map are shown in

blue points, the vehicle positions corrected by the road and building maps are shown in magenta color. The local perception extracted from the vertical LRF are shown in green crosses. After correcting the vehicle pose, the local LRF perception is more coherent with the original building map.

5.5.3/ EXPERIMENTAL RESULTS OF THE INDUSTRIAL PARK SEQUENCE

The second data sequence was captured in an industrial area - the Techn'hom at Belfort, France in February, 2012. As seen in Fig. 5.40, the vehicle trajectory is with buildings around and there are no big altitude changes on the road. The trajectory length measured by the RTK-GPS sensor is about 720 meters. In order to test our proposed method in the circumstances of GPS failure, four GPS masks were added into GPS data to simulate the blockage of GPS signals (Fig. 5.41). Each GPS mask lasts about 25 seconds.

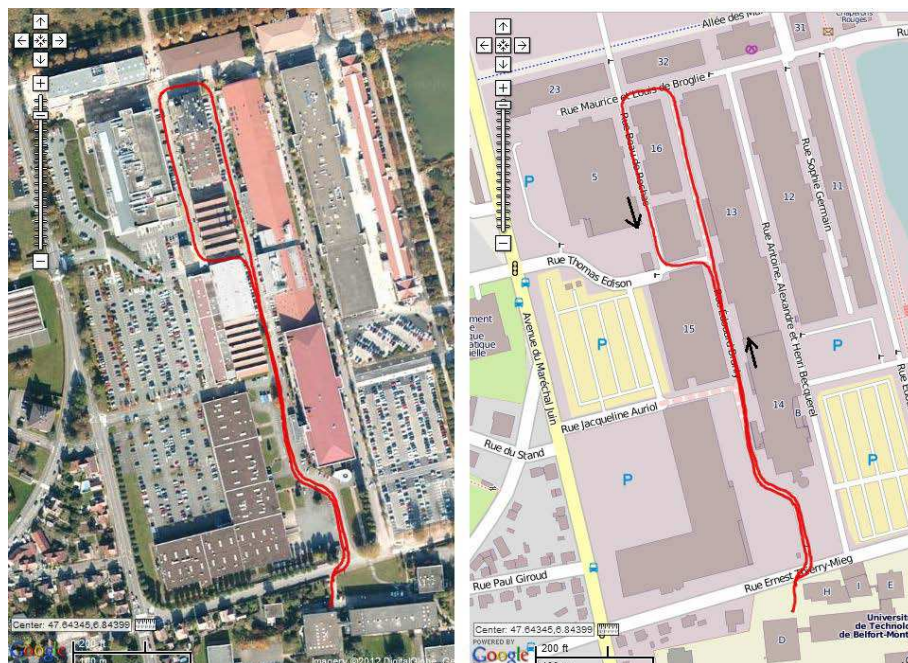


FIGURE 5.40 – The second vehicle trajectory respectively overlapped on Google aerial image and OpenStreetMap

- **1. Localization without map-aiding.** As shown in Fig. 5.41, when GPS readings are not provided, the vehicle poses are estimated by LRF scan alignment information and the orientation from gyro (green trajectory). It is noted that without any global information, the vehicle trajectory drift gradually and the covariance of the vehicle position enlarges.
- **2. Localization with the road map.** Then, the road map layer is added to assist vehicle localization by map-matching method. The blue trajectory in Fig. 5.41 demonstrates that the erroneous vehicle positions are dragged back onto the road and the pose covariance is bounded around the road after adding the road map.
- **3. Localization with the original building map.** After that, the building map layer is added to correct both the vehicle position and orientation with the horizontal and vertical LRFs. It is noted that when the observed building facades can be associated with the map,

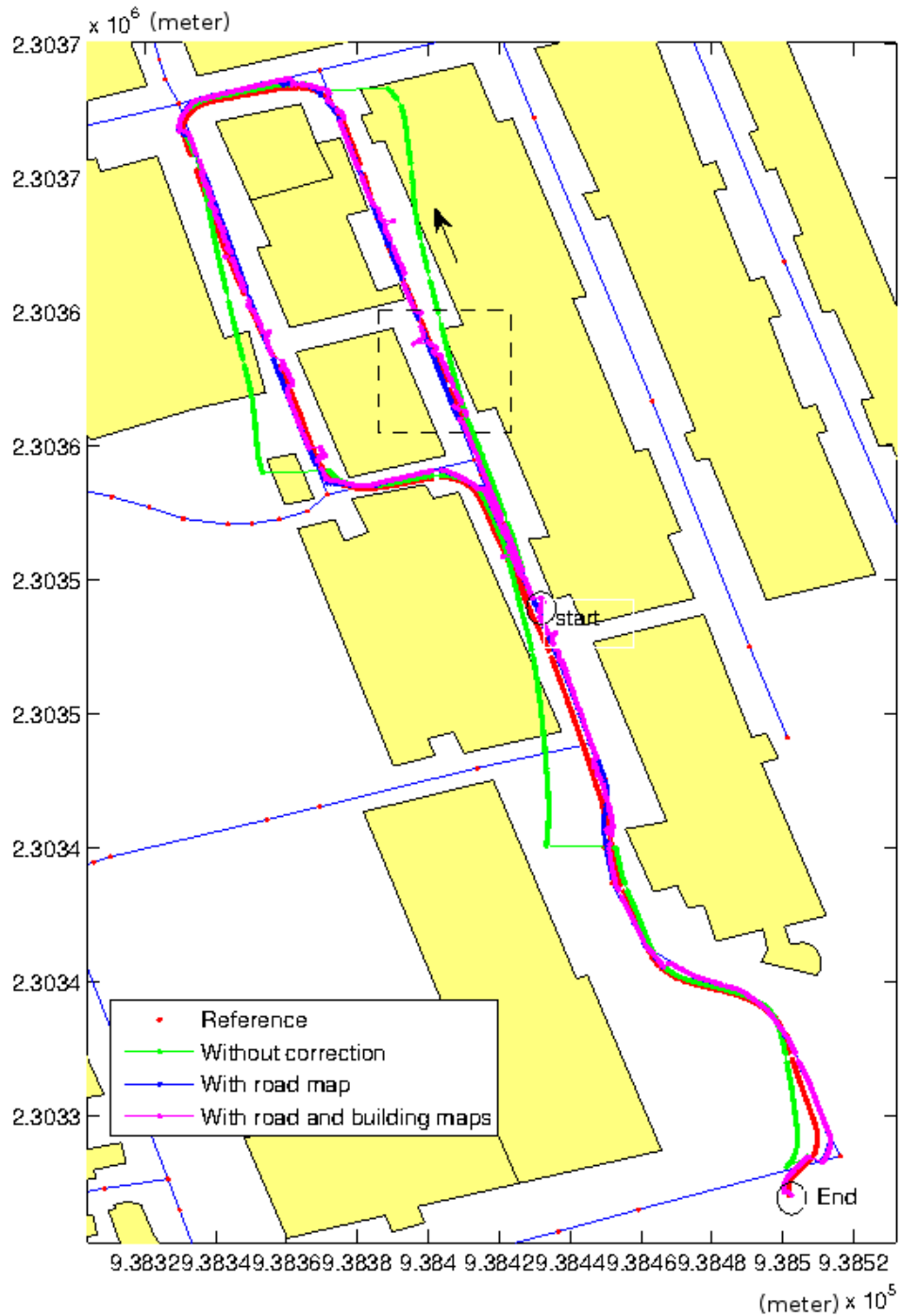


FIGURE 5.41 – Vehicle positions respectively obtained by RTK-GPS, LRF-H scan alignment/gyro prediction, road map based correction, and road/building maps based correction methods. The rectangle area is zoomed in Fig. 5.42

the horizontal LRF observation can be used to correct the vehicle orientation. Then, if the road width condition is met (Equation 5.40), the vehicle lateral position is also corrected. The whole corrected trajectory is shown in magenta color in Fig. 5.41.

For example, as shown in Fig. 5.42, the green points are the horizontal LRF scans before

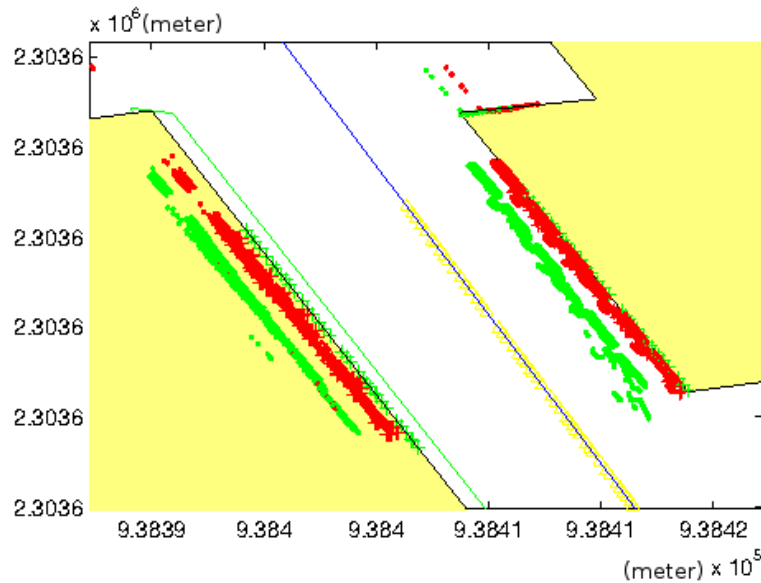


FIGURE 5.42 – Zoom of rectangle area in Fig. 5.41 (green points are LRF-H scans before the vehicle pose correction, and red points are LRF-H scans after the pose correction)

the vehicle pose correction, and the red points are the LRF scans after pose correction. It can be seen that after the pose correction step with the building map, the local LRF observations are more consistent with the map and the vehicle position.

In Fig. 5.44 and Fig. 5.45, several experimental results are presented in different environment conditions. In Fig. 5.44, the correspondences between the lines in the horizontal LRF scan and the map are found by the graph matching as : $\{(L_1 : L_7^m), (L_2 : L_2^m), (L_4 : L_4^m)\}$, the vehicle pose is corrected by the three line correspondences. In Fig. 5.45, only parallel lines are detected in the horizontal LRF scan, the observation from the vertical LRF is used to correct the vehicle lateral position.

Taking RTK-GPS as the ground truth, the localization errors by different methods during the periods with simulated GPS masks are compared, as shown in Fig. 5.43 and Tab. 5.3.

Method	mean error	standard deviation of error
LRF-H alignment/Gyro	7.3776	5.1813
LRF-H alignment/Gyro/Road map	3.8493	1.3807
LRF-H alignment/Gyro/Road map/ LRFH-LRFV-Building maps	3.0980	1.2649

TABLE 5.3 – Localization error during GPS mask (/meter)

During the periods with simulated GPS mask, the average localization error can be reduced after using the road map. Since the vehicle trajectory is close to the road central line in this sequence, the advantage of adding building map is not significant over only using road map. However, if the road is much wider or the vehicle is not close to the road central line, the usefulness of building map for lateral position correction would be more important.

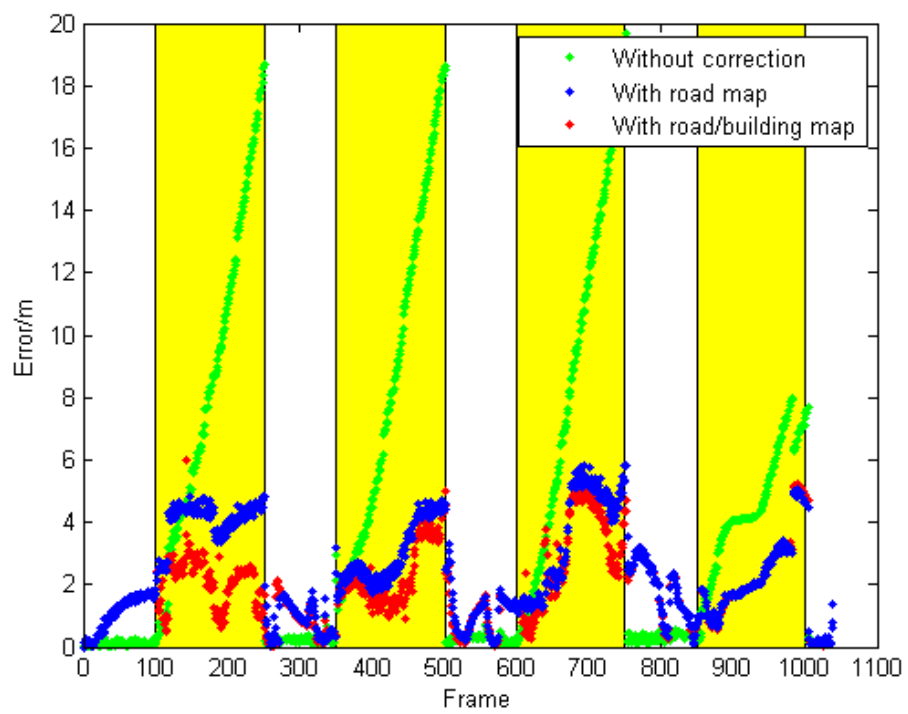
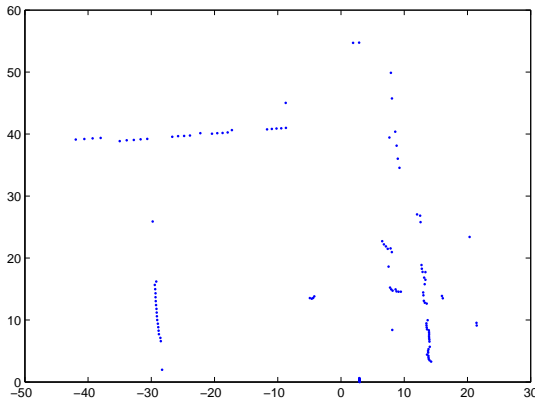
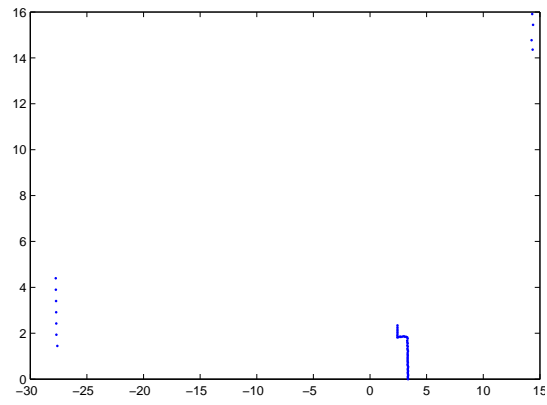


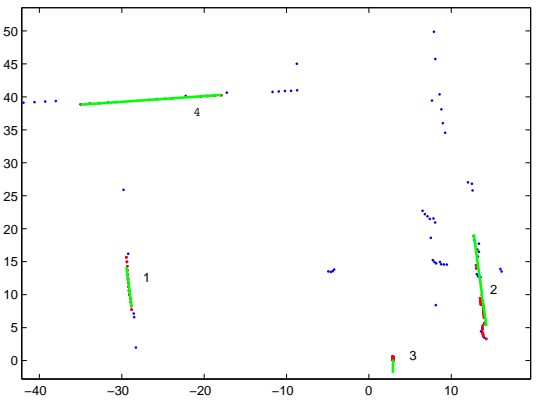
FIGURE 5.43 – Comparison of vehicle localization errors with different approaches



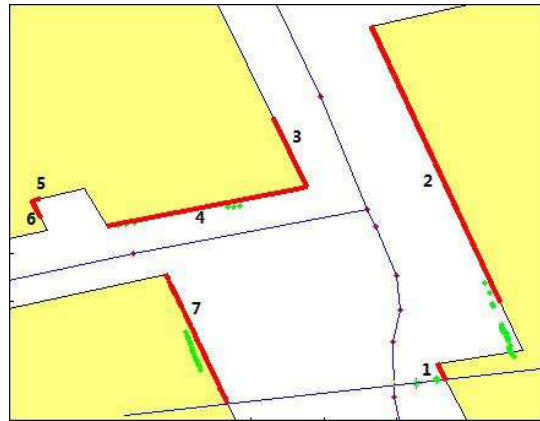
(a) A horizontal laser scan



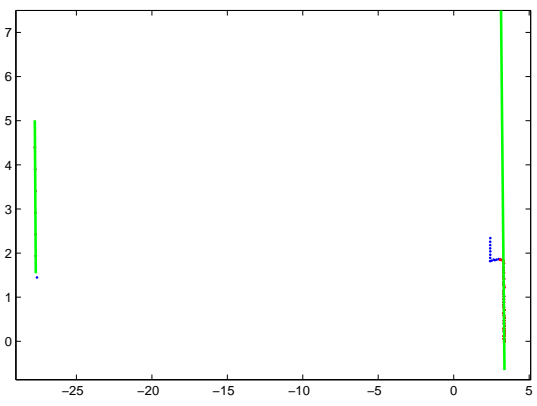
(b) A vertical laser scan



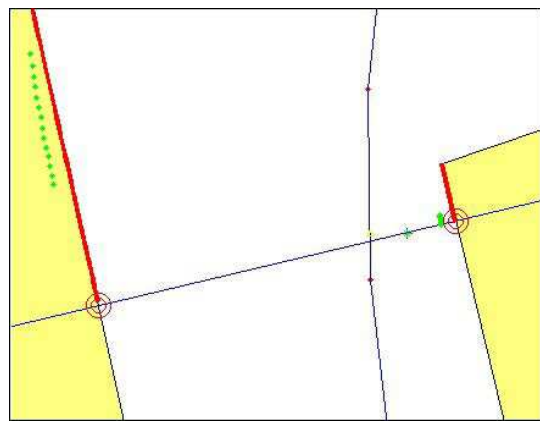
(c) Detected lines (in green) in the horizontal laser scan



(d) Building facades in the FOV of the horizontal LRF

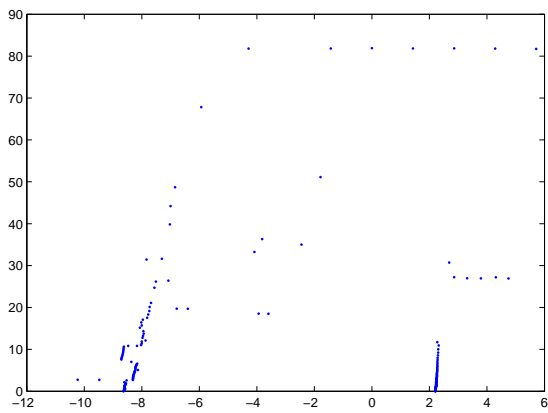


(e) Detected lines (in green) in the vertical laser scan

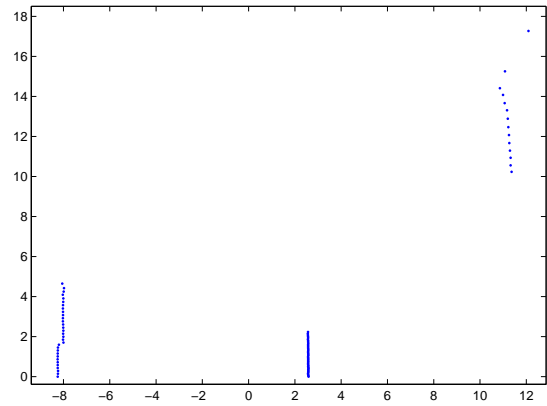


(f) Virtual intersections of the vertical LRF with the building map

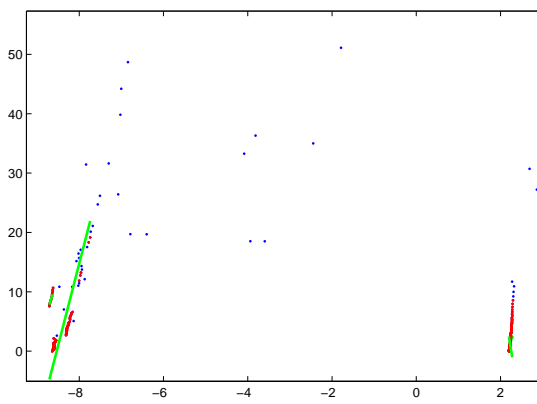
FIGURE 5.44 – Observation from LRFs and GIS map - result 1 (area with non-parallel lines)



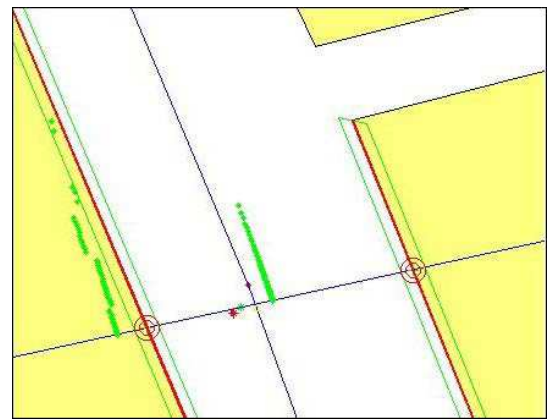
(a) A horizontal laser scan



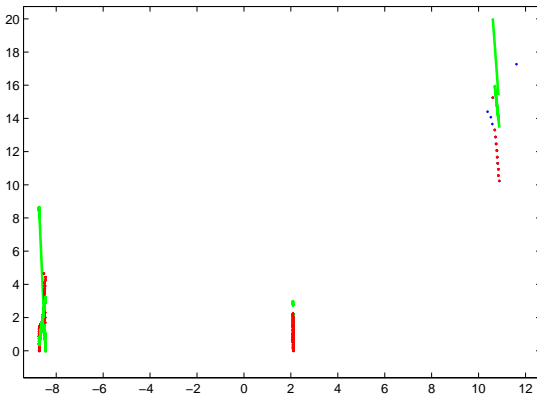
(b) A vertical laser scan



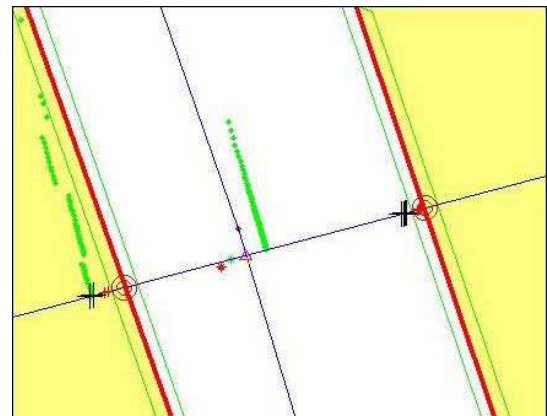
(c) Detected lines (in green) in the horizontal laser scan



(d) Virtual intersections of the vertical LRF with the building map



(e) Detected lines (in green) in the vertical laser scan



(f) Vertical LRF facade landmarks before and after the pose correction (black crosses '+' : the vertical LRF landmarks with the predicted pose ; red crosses : the vertical LRF landmarks with the corrected vehicle pose ; green star : the predicted vehicle position ; magenta triangle : the corrected vehicle position ; green points : the horizontal LRF scan which is not coherent with the vertical LRF and the map)

FIGURE 5.45 – Observation from LRFs and GIS map - result 2 (in narrow street with parallel lines) : lines detected in the horizontal LRF scan are neither coherent with the building map nor the vertical LRF scan

• 4. Localization with the reconstructed map

1) Reconstructed map. Fig. 5.46 (left) shows the reconstructed scene in this experiment from the vertical LRF. The blue lines represent the grouped building facades along the vehicle trajectory, the red crosses represent the detected independent objects. A rectangle area is zoomed in and shown on the right side : the green points are the detected points on the facades, and the red circles are the endpoints of the building facades.

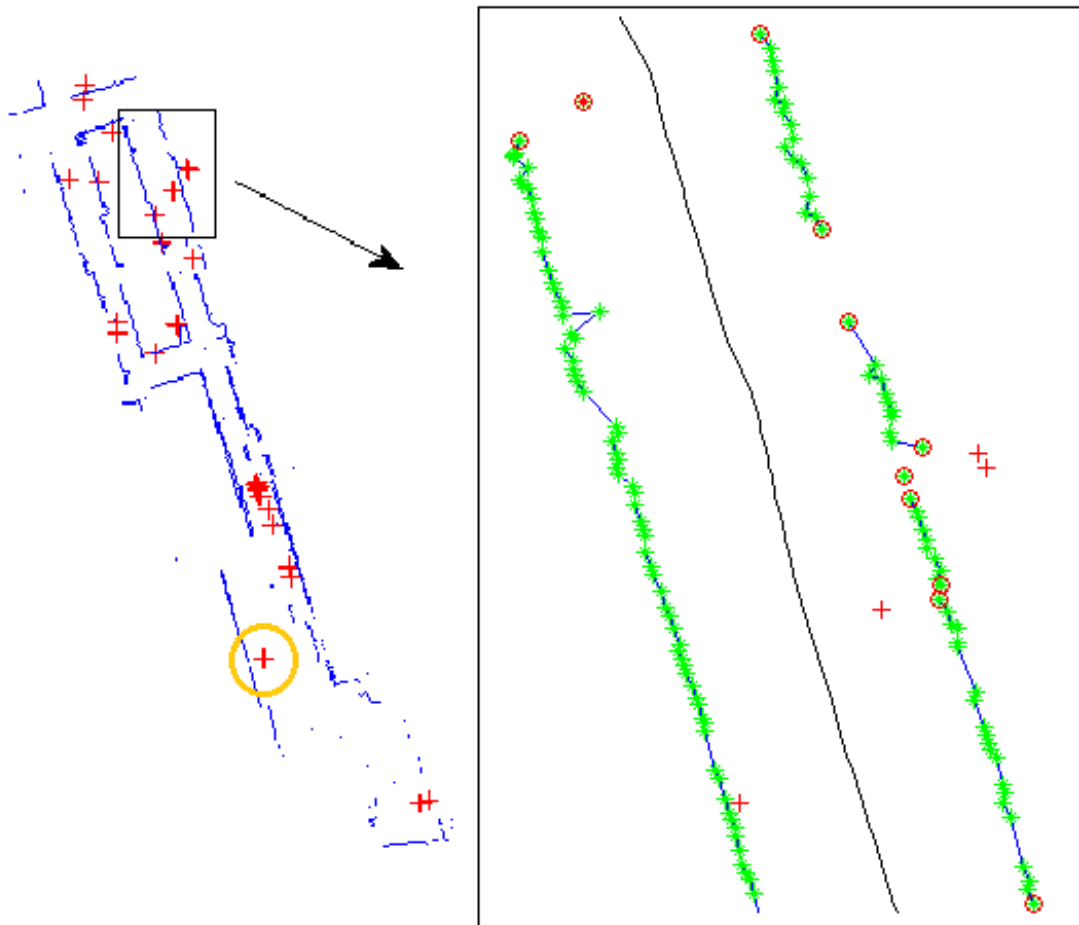


FIGURE 5.46 – New maps with extracted building facade landmarks and independent objects

LRF and camera data are associated by their logged time. In order to verify the correctness of the detected lines in laser scan, we take an image around the position where the independent object in the yellow circle of Fig. 5.46 (left) is observed by the vertical LRF. As seen in Fig. 5.47, the object detected by the vertical LRF is the tree in front of the building in the orange ellipse.

2) Localization with the reconstructed building map. This work is still in progress and some preliminary results are presented. We tested the proposed method in three steps : at first, with the known ground truth, noises are manually simulated and added into the GPS positions at several points to simulate GPS jumps ; then, a GPS mask during a period of time is added to simulate the blockage of GPS signals ; and finally, the vehicle is driven through the same experimental area, the map reconstructed at the first time is



FIGURE 5.47 – One tree in front of the building is treated as an independent object by the LRF-V (corresponding to the object in the yellow circle of Fig. 5.46)

used to correct the vehicle pose.

In order to simulate GPS jumps with large errors, we randomly degraded the precision of several GPS measurements. Fig. 5.48 shows the localization results when GPS encounters large jumps. In Tab 5.4, the randomly added GPS position errors are listed. Some erroneous GPS positions are rejected during the initial fusion step with the process model/gyro and road map-matching. Then, the vehicle position is corrected by the new reconstructed building map. Tab 5.4 shows that the building map can help to correct the vehicle pose such that the vertical LRF observations are consistent with the reconstructed map.

	Position noise	Rejected by fusion	Corrected by new map	Error on x	Error on z
P1	19.2846	rejected	corrected	0.4450	1.9051
P2	-0.4006	not rejected	corrected	-0.0506	0.9510
P3	-15.9633	rejected	corrected	0.0482	0.5408
P4	7.3357	rejected	not corrected	-0.0704	0.9733
P5	27.0277	rejected	corrected	0.0783	0.3970
P6	-11.7806	rejected	corrected	-0.1577	0.8420
P7	-16.9585	rejected	corrected	-0.1849	0.4077
P8	50.5200	rejected	corrected	-0.4135	0.8765
P9	6.1507	rejected	corrected	-0.2921	0.3524

TABLE 5.4 – Localization results with simulated GPS jumps (/meter)

In order to simulate GPS signal blockage in urban environments, we added a GPS mask for about 30 seconds (the ground truth lasts about 101.36m). Fig. 5.50 show the localization results when a GPS mask is added. Without any map information, the trajectory gradually drifts to the right side. After incorporating the map information, the vehicle is closer to the ground truth.

As seen in Fig. 5.50 and Tab. 5.5, the localization accuracy is about 1m when using the

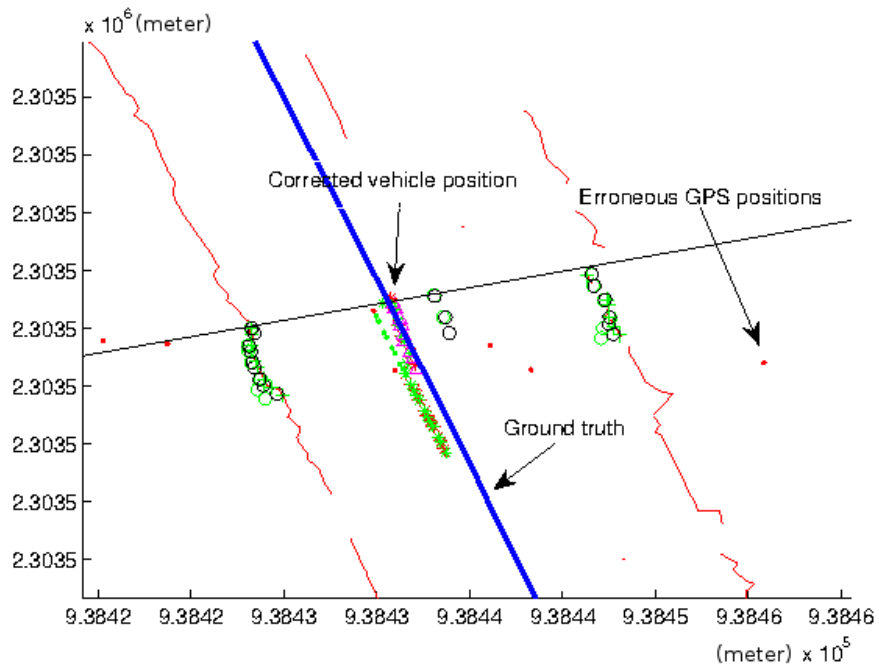


FIGURE 5.48 – Vehicle localization results by using the reconstructed building map (with simulated GPS jumps)

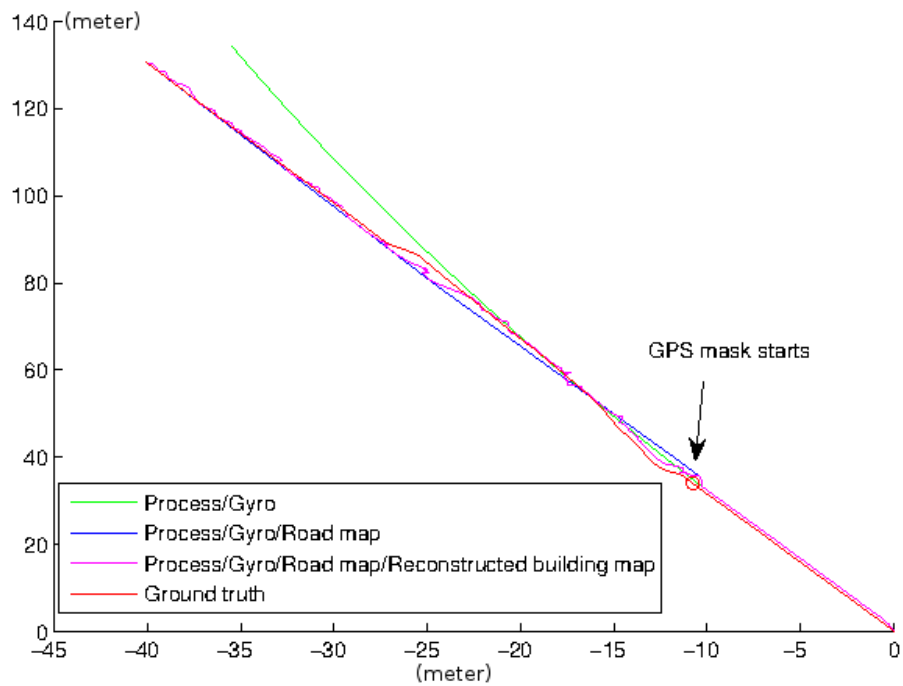


FIGURE 5.49 – Vehicle localization results with the reconstructed building map (with about 30s simulated GPS mask)

road map observation. Because both the vehicle position and orientation can affect the predicted landmarks, the localization error changes when using the reconstructed building map. However, in several areas, the reconstructed building map can help to reduce the localization error of road map based estimation.

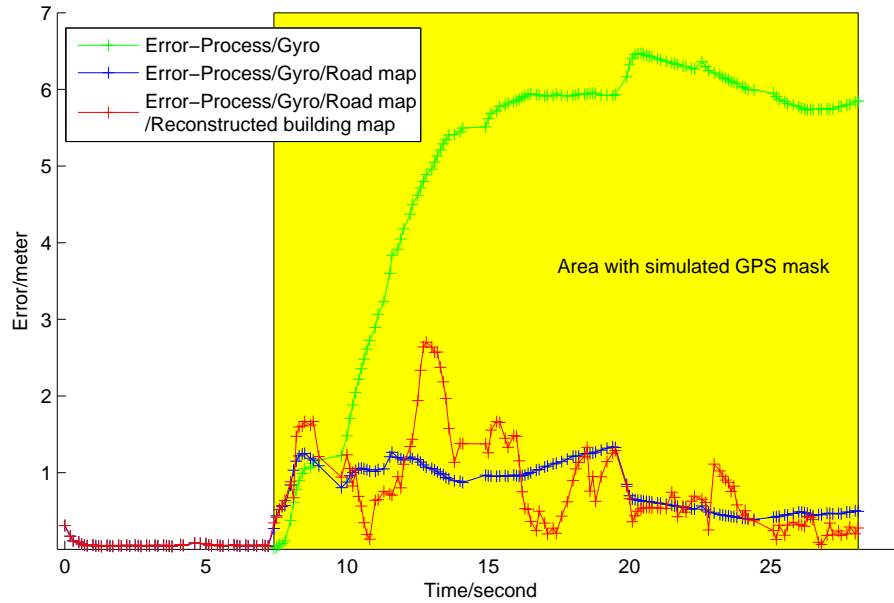


FIGURE 5.50 – Comparison of vehicle position errors after adding GPS masks

All the LRF-V observations acquired at the current pose are stored with the initial facade points. They are used to update the map when the number of facade points around a facade segment reaches a specified density threshold. This work will be continued by acquiring much more experimental data of the same area.

5.6/ CONCLUSION AND PERSPECTIVES

In this chapter, we presented a geographical information system (GIS) aided vehicle localization method with a GPS receiver, a gyro, two LRF systems and 2D GIS maps (road network map, building map). Two types of map information are used : the first one is the original GIS maps, including the road network map and the building map ; in order that the landmarks can be reused for vehicle localization in the future, the building facade landmarks detected by the perception sensor are stored in a new building facade map

	Min error	Max error	Average error/m
Process/Gyro	0.0140	6.4698	4.8999
Process/Gyro/Road map	0.0411	1.3361	0.8161
Process/Gyro/Road map /LRFH-LRFV-New map	0.0411	2.7014	0.8504

TABLE 5.5 – Localization results with the different approaches (/meter)

layer ; another map layer is used to store the independent objects.

The proposed approaches were tested with two real data sequences, one in an old town center and the other one in an industrial area. Both experimental results demonstrate that when GPS signals are blocked or with large jumps, the road map can help to bound the vehicle pose around the road ; and the building map can help to adjust the vehicle pose from coarse-to-fine such that the local perception of the two LRFs are consistent with the building facade ; and the newly reconstructed building map can help to keep the map up to date and to correct the vehicle pose.

Perspectives : based on the preliminary results in this chapter, several research perspectives are summarized as follows :

- When the vehicle goes through the same place, the problem becomes a SLAM problem to update both the vehicle pose and the map, together with their covariances. We can also pass the same area more times, 8 times, 10 times or even more, to obtain more laser points, and to extract the facades from this large set of points.
- Our current work only takes use of the LRF-V for mapping and localization. The LRF-V provides only information of the buildings which are parallel to the vehicle trajectory, while the building facade perpendicular to the vehicle trajectory cannot be observed. Therefore, we can use the LRF-H and LRF-V for mapping within a probabilistic 3D occupancy grid framework. The same landmark can be seen at different consecutive poses and can be associated to update the predicted vehicle pose.
- A LRF can also be installed at the bottom back of the vehicle to scan the ground behind the vehicle. In this work, we use a vertical LRF to test the feasibility of the method. Stereovision or the fusion of stereovision and LRF are also envisaged in the future. Plane features can also be extracted if we use the historical information of the left or right building scans.
- Many methods have been proposed to solve the problem of road matching when cross-roads appear. The local perception of the environment provided by the camera or LRF can help to solve the problem of road map-matching (road/building combination pattern) and to determine the position of the vehicle on road.
- We can take use of the GIS map to dynamically manage the objects in urban environments, which are extracted from the horizontal and vertical LRFs, like static objects, and moving objects.
- We can also take use of the other attribute information provided by the GIS database, such as the slope of ground.

CONCLUSIONS AND FUTURE WORKS

6.1/ CONCLUSIONS

The problem addressed in this thesis is how to provide precise and robust localization service in urban environments by integrating multi-source information. Accurate localization is one requirement for intelligent vehicles applications. From map provider's point of view, localization is also one of the key points in geo-referencing process for mobile mapping systems. After a detailed review of the existing relative and absolute vehicle localization approaches in **Chapter 2**, vehicle localization methods were proposed in this thesis to assist vehicle localization in urban environments.

At first, stereovision based visual odometry and laser range finder based scan alignment methods are presented in **Chapter 3**. Stereovision based visual odometry predicts the vehicle movement on the basis of image feature detection and tracking. Compared with wheel encoder based odometry, it can provide vehicle motion in 6 degrees of freedom and avoid the wheel slippage problem in bad soil conditions. Nevertheless, especially during bad illumination condition like in the night, image based visual odometry method cannot be used. This method is then completed by adding the vehicle motion estimated by consecutive scans in horizontal LRF system. Experimental results show that these two relative localization methods can provide vehicle movement information in short term like other inertial sensors, but their localization accuracy decreases in long term due to error accumulation from frame to frame. Thus, we propose to integrate GPS and stereovision based visual odometry, and horizontal LRF based scan alignment together in **Chapter 4**. The redundant measurement information are used to evaluate the coherence of different systems and to continuously provide pose measurement if any system fails to work. This method to integrate GPS with relative localization methods is with the assumption that GPS signals are lost in short term. If the GPS signals are lost for long period, the accumulated localization error of dead-reckoning method cannot be bounded and the trajectory might gradually drift if only using relative approaches.

Therefore, in **Chapter 5**, static environment information stored in digital maps of a geographical information system (GIS) is used to bound the localization error of dead-reckoning methods if GPS receiver fails for long time. At first, GPS/DR (LRF-H based scan alignment and a gyro) measurements are fused with a 2D GIS road network map to provide a coarse pose estimation. This step is like the classic road map-matching method. Then, two complementary LRF systems (horizontal and vertical LRFs) mounted on the roof of the vehicle are used to detect building facades in urban environments. Building facade features in the vertical LRF scan are chosen for lateral pose correction since there

are less obstacles in the air than on the ground. The detected vertical building facades are projected onto the 2D plane and associated with the GIS building map layer to correct the vehicle pose error, especially for the lateral pose error. The experimental results with real data show that the road map can help to obtain an approximate estimation of the vehicle position by projecting the vehicle position on the corresponding road segment, then the integration of the building information can help to refine this first pose estimation.

6.2/ PERSPECTIVES

A lot of perspectives are envisaged to improve and complete the vehicle localization system proposed in this thesis in future works :

- At first, real-time implementation of the localization system is being developed. As the time synchronization between different sensor systems is important for real-time fusion, the processing time of each subsystem (e.g., feature detection and matching, and vehicle motion estimation in visual odometry ; scan alignment in laser method ; map extraction and building feature association) and data transmission should be considered.
- The precision of visual odometry method might be improved by improving camera calibration results, or using more robust features. We can test and compare the precision and robustness of different kind of features with more data sequences under different illumination and weather conditions, or in more complex environments with different types of obstacles, etc. Since detection of dynamic obstacles and vision based relative motion estimation are dual-processes, the work on dynamic obstacles detection should be improved.
- Other uncertainty representation might also be considered in future works, like bounded intervals. In this thesis, covariance of the visual odometry based estimation is propagated from the image point noise, 3D point reconstruction process and pose estimation. Covariance of LRF-H scan alignment estimation is propagated from the laser point noise and minimization process. But the noise or error in point matching/association step is not yet well measured and this should be considered in future works. For the image aided ICP methods, the work on how to dynamically choose appropriate coefficients for the image attributes needs to be continued in the future ; the incorporation of camera with larger FOV (e.g. fish eye) can also be considered.
- In future works, sensors like IMU or odometry can also be directly integrated thanks to the convenience of information filter. Tight coupling approaches between LRF and image data, inertial sensor and image data, GPS pseudo-ranges with image data are also envisaged instead of loose coupling of their estimated motions.

The work on combining GIS for intelligent vehicle localization in this thesis is still very preliminary, some research perspectives are summarized as follows :

- In this work, we have tested the possibility of using building facade features for vehicle pose correction, especially for vehicle lateral pose. This method can be completed by combining other continuous features. For example, a LRF sensor can be installed at the bottom back of the vehicle to scan the ground and detect curbs beside the road (due to the special LRF configuration in our experiment, these features cannot be extracted) ; camera system can be used to detect lanes on the ground. Then, these three continuous

line features : lanes, curbs and building footprints can be stored in three map layers and used for vehicle lateral pose correction in case that one of them might not be well detected.

- The idea of our current work is simple and easy to implement, we only take use of the vertical LRF scan for new map generation and localization. But LRF-V can provide only information of the buildings which are parallel to the vehicle trajectory, while the building facades perpendicular to the vehicle moving direction cannot be observed. Therefore, we can use the LRF-H and LRF-V for mapping within a probabilistic 3D occupancy grid framework. The same landmark can be seen at different consecutive poses and can be associated to update the predicted vehicle pose. Stereovision or the fusion of stereovision and LRF are also envisaged in the future. Building plane features can also be extracted if we use historical information of the left or right building scans.
- The map based method proposed in this thesis is based on the assumption that the facades of the buildings are vertical. Quality of GIS maps can largely affect the precision of pose correction step. In order to update the newly generated map, we can choose the SLAM strategy to update both the vehicle pose and the map when the vehicle goes through the same place. Or we can repeat the trajectory in the same area during different dates to extract the facades from this large set of points, then complete the newly generated building map like the road map update strategy used in OpenStreetMap project. So the work on how to manage the sensor observations and accurately update the map with new observation should be completed.
- Many methods have been proposed to solve the problem of road matching when cross-roads appear. The local perception of the environment provided by the camera or LRF may help to reduce the ambiguity problem of road map-matching. For example, with the lane features and building facades extracted from the image, a road/building combination pattern can be obtained, this pattern can be compared with the existing road network/building footprint maps to reduce the ambiguity of vehicle position on road.
- Other attribute information provided by the GIS database might also be used, such as the slope of ground. If multiple vehicles are roving in the same area and each vehicle sends its own local map to the computation center (or to the other vehicles), the information should also be possible to be shared for making a more complete map. We can take use of the GIS map to dynamically manage the objects in urban environments, which are extracted from the horizontal and vertical LRFs, like static objects, and moving objects.

BIBLIOGRAPHIE

- [1] M. Agrawal and M. R. B. Kurt Konolige. Censure : Center surround extremas for realtime feature detection and matching. In *The 10th European Conference on Computer Vision*, pages 102–115, 2008. 52
- [2] C. Andrea, R. Alessandro, and M. Agostino. Distributed information filters for mav cooperative localization. In *10th International Symposium on Distributed Autonomous Robotics Systems (DARS)*, 2010. 91
- [3] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(5) :698–700, May 1987. 23, 59
- [4] D. Attia, C. Meurie, Y. Ruichek, and J. Marais. Counting of satellites with direct gnss signals using fisheye camera. a comparison of clustering algorithms. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2011. 40
- [5] R. Baczyk, A. Kasinski, and P. Skrzypczynski. Vision-based mobile robot localization with simple artificial landmarks. In *7th IFAC Symposium on Robot Control*, pages 217–222, 2003. 32
- [6] K. Baek and H. Bang. Vision-based target motion estimation of multiple air vehicles using unscented information filter. In *International Conference on Control Automation and Systems (ICCAS)*, pages 1567–1571, 2010. 91
- [7] L. Bai and Y. Wang. Fusing image, gps and gis for road tracking using multiple condensation particle filters. In *Intelligent Vehicle Symposium*, pages 162–167, 2008. 34
- [8] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping (slam) : Part ii. *IEEE Robotics & Automation Magazine*, 13(3) :108–117, 2006. 37, 139
- [9] T. Bailey, E. M. Nebot, J. K. Rosenblatt, and H. Durrant-whyte. Data association for mobile robot navigation : a graph theoretic approach. In *IEEE International Conference on Robotics and Automation*, pages 2512–2517, 2000. 33, 123, 130
- [10] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation : Theory Algorithms and Software*. Wiley, 2004. 95
- [11] C. Barrios and Y. Motai. Improving estimation of vehicles trajectory using the latest global positioning system with kalman filtering. *IEEE Transaction on Instrumentation and Measurements*, 60 :3747–3755, 2011. 88
- [12] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool. Surf : Speeded up robust features. *Computer Vision and Image Understanding*, 110 :346–359, 2008. 52, 53
- [13] P. J. Besl and H. D. Mckay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2) :239–256, 1992. 24, 67
- [14] D. Betaille, A. Chapelon, B. Lusetti, M. Kais, and D. Millescamps. High integrity reference trajectory for benchmarking land navigation data fusion methods. In *IEEE Intelligent Vehicles Symposium*, pages 346–351, 2007. 100

- [15] N. Bioret, G. Moreau, and M. Servières. Urban localization based on correspondences between street photographs and 2d building gis layer. In *CORESA, session spéciale Géolocalisation*, Toulouse, France, 2009. 34
- [16] N. Bioret, G. Moreau, and M. Servières. Towards outdoor localization from gis data and 3d content extracted from videos. In *IEEE ISIE special session on Emerging Technologies*, Bari, Italy, 2010. 34, 35
- [17] G. Blewitt. Basics of the gps technique : Observation equations. *Geodetic Applications of GPS*, pages 10–54, 1997. 28
- [18] Y. Bok, Y. Hwang, and I. Kweon. Accurate motion estimation and high-precision 3d reconstruction by sensor fusion. In *IEEE International Conference on Robotics and Automation*, 2007. 70
- [19] J. Borenstein, H. R. Everett, L. Feng, and D. Wehe. Mobile robot positioning sensors and techniques, 1997. 32
- [20] G. A. Borges. A split-and-merge segmentation algorithm for line extraction in 2-d range images. In *International Conference on Pattern Recognition*, pages 441–444, Washington, DC, USA, 2000. IEEE Computer Society. 123
- [21] C. Boucher, R. Lherbier, and J. C. Noyer. Multisensor unscented filtering for gps-based navigation systems. In *IEEE Instrumentation and Measurement Technology Conference*, pages 1–6, 2007. 40
- [22] C. Boucher and J. C. Noyer. A hybrid particle approach for gnss applications with partial gps outages. *IEEE Transactions on Instrumentation and Measurement*, 59(3) :498–505, 2010. 39
- [23] C. Boucher and J. C. Noyer. Gnss-based unscented filtering for road map databases management. In *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1398–1403, 2011. 121
- [24] C. Boucher and J. C. Noyer. Automatic detection of topological changes for digital road map updating. *IEEE Transactions on Instrumentation and Measurement*, 61(11) :3094–3102, 2012. 121
- [25] J. Y. Bouguet. Camera calibration toolbox for matlab, 2008. 48, 50
- [26] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro. Vehicular ad hoc networks : A new challenge for localization-based systems. *Computer Communications*, 31(12) :2838–2849, July 2008. 15, 37
- [27] D. C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8) :855–866, 1971. 48
- [28] C. Cappelle. *Localisation de véhicules et détection d'obstacles apport d'un modèle virtuel 3D urbain*. PhD thesis, Université des Sciences et Technologies de Lille, 2008. 34, 35
- [29] C. Cappelle, M. E. El Najjar, F. Charpillet, and D. Pomorski. Virtual 3D City Model for Navigation in Urban Areas. *Journal of Intelligent and Robotic Systems*, June 2011. vient d'être accepté. 34, 40
- [30] C. Cappelle, M. E. El Najjar, D. Pomorski, and F. Charpillet. Intelligent geolocalisation in urban areas using gps, 3d-gis and vision. *Journal of Intelligent Transportation Systems*, 14 :3–12, 2010. 34

- [31] S. Carreno, P. Ridao, P. Wilson, and Y. Petillot. A survey on terrain based navigation for auvs. In *International Federation of Automatic Control Conference on Control Systems in Marine Applications*, pages 684–690, Rostock, Germany, 2010. 33
- [32] M. Cazorla, D. Viejo, A. Hernandez, J. Nieto, and E. Nebot. Large scale egomotion and error analysis with visual features. *Journal of Physical Agents*, 4 :19–24, 2010. 63
- [33] A. Censi. An accurate closed-form estimate of icps covariance. In *IEEE International Conference on Robotics and Automation*, 2007. 75
- [34] T. Cham, A. Ciptadi, W. Tan, M. Pham, and L. Chia. Estimating camera pose from a single urban ground-view omnidirectional image and a 2d building outline map, 2010. 34
- [35] C. Chen, J. Ibanez-Guzman, and O. Le-Marchant. Pattern recognition for loosely-coupled gps/odometer fusion. In *IEEE International Conference on Robotics and Automation (ICRA)*, Sept 2008. 39
- [36] Z. Chen and S. T. Birchfield. Qualitative vision-based mobile robot navigation. In *IEEE International Conference on Robotics and Automation*, pages 2686–2692, 2006. 36
- [37] Y. Cheng, M. Maimone, and L. Matthies. Visual odometry on the mars exploration rovers. *IEEE Robotics and Automation Magazine*, 13 :54–62, 2006. 19, 20, 43, 56, 64
- [38] S. Cho and W. Choi. Robust positioning technique in low-cost dr/gps for land navigation. *IEEE Transaction on Instrumentation and Measurements*, 55 :1132–1142, 2006. 88
- [39] A. K. Choy. *Global Positioning System theory and design : geometric dilution of precision*. Cornell University, 1996. 29
- [40] B. Coen and K. Joep. Algorithm 457 : finding all cliques of an undirected graph. *Communications of the ACM*, 16(9) :575–577, 1973. 133
- [41] A. I. Comport, E. Malis, and P. Rives. Real-time quadrifocal visual odometry. *International Journal of Robotics Research*, 29(2-3) :245–266, February 2010. 19, 22, 43
- [42] G. N. DeSouza and A. C. Kak. Vision for mobile robot navigation : A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2) :237–267, 2002. 33
- [43] A. Diosi and L. Kleeman. Uncertainty of line segments extracted from static sick pls laser scans. In *Australasian Conference on Robotics and Automation*, 2003. 74
- [44] M. Donoser and H. Bischof. Efficient maximally stable extremal region (mser) tracking. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1, CVPR '06*, pages 553–560, Washington, DC, USA, 2006. IEEE Computer Society. 52
- [45] L. Douadi, M.J. Aldon, and A. Crosnie. Pair-wise registration of 3d/color data sets with icp. In *IEEE International Conference on Intelligent Robots and Systems*, pages 663–668, 2006. 72
- [46] Arnaud Doucet, Simon Godsill, and Christophe Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *STATISTICS AND COMPUTING*, 10(3) :197–208, 2000. 93

- [47] N. M. Drawil and O. Basir. Intervehicle-communication-assisted localization. *IEEE Transactions on Intelligent Transportation Systems*, 11(3) :678–691, September 2010. 38
- [48] N. M. Drawil and O. Basir. In *Global Navigation Satellite Systems : Signal, Theory and Applications*, chapter Emerging New Trends in Hybrid Vehicle Localization Systems, pages 279–298. InTech, 2012. 38
- [49] S. Druon, M. J. Aldon, and A. Crosnier. Color constrained icp for registration of large unstructured 3d color data sets. In *IEEE International Conference on Information Acquisition*, volume 1-2, pages 249–255, 2006. 72
- [50] G. Dubbelman and F. C. A. Groen. Bias reduction for stereo based motion estimation with applications to large scale visual odometry. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 23
- [51] H. Durrant-Whyte. Introduction to decentralised data fusion, 2002. 91, 92
- [52] D. W. Eggert, A. Lorusso, and R. B. Fisher. Estimating 3-d rigid body transformations : a comparison of four major algorithms. *Mach. Vision Appl.*, 9(5-6) :272–290, March 1997. 21, 23
- [53] A. El-Rabbany. *Introduction to GPS : the Global Positioning System*. Artech House mobile communications series. Artech House, 2006. 27, 29
- [54] O. G. Favrot and M. Parent. Laser scanner based slam in real road and traffic environment. In *ICRA09 Workshop on Safe navigation in open and dynamic environments Application to autonomous vehicles*, 2009. 24, 67
- [55] M. A. Fischler and R. C. Bolles. Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6) :381–395, June 1981. 60
- [56] D. A. Forsyth and J. Ponce. *Computer vision. a modern approach*. Prentice Hall, Pearson Education International, 2003. 48, 55
- [57] B. Fortin, R. Lherbier, and J. C. Noyer. Feature extraction in scanning laser range data using invariant parameters : Application to vehicle detection. *IEEE Transactions on Vehicular Technology*, 61(9) :3838–3850, 2012. 123
- [58] F. Fraundorfer and D. Scaramuzza. Visual odometry : Part ii - matching, robustness, and applications. *IEEE Robotics and Automation Magazine*, 19(1), 2012. 52
- [59] D. Garcia, J.J. Orteu, and M. Devy. Accurate calibration of a stereovision sensor : Comparison of different approaches. In *Conference on Vision Modeling and Visualization*, pages 25–32. Aka GmbH, 2000. 50
- [60] A. Georgiev and P. K. Allen. Localization methods for a mobile robot in urban environments. *IEEE Transactions on Robotics*, 20 :851–864, 2004. 37
- [61] G. Godin, D. Laurendeau, and R. Bergevin. A method for the registration of attributed range images. In *Proceedings of Third Int. Conference on 3D Digital Imaging and Modeling*, pages 179–186, 2001. 72
- [62] M. C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs*. Academic Press, New York, 1980. 131
- [63] J. S. Greenfeld. Matching gps observations to locations on a digital map. In *81st Annual Meeting of the Transportation Research Board*, Washington D.C, January 2002. 40, 117

- [64] M. Grimes and Y. LeCun. Efficient off-road localization using visually corrected odometry. In *IEEE International Conference on Robotics and Automation*, pages 2649–2654, 2009. 39
- [65] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.J. Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on Signal Processing*, 50(2) :425–437, 2002. 93
- [66] M. Hachman. 'eyes' of google's self-driving car may bust crooks, April, 2012. 67
- [67] D. Hahnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose. Mapping and localization with rfid technology. In *IEEE International Conference on Robotics and Automation*, volume 1, pages 1015–1020, 2004. 25
- [68] R.M. Haralick. Propagating covariance in computer vision. In *Workshop on Performance Characteristics of Vision Algorithms*, pages 493–498, 1994. 62
- [69] C. Harris and M. Stephens. A combined corner and edge detector. In *The Fourth Alvey Vision Conference*, pages 147–151, 1988. 36, 52
- [70] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004. 22, 45, 50, 60, 63, 64
- [71] B. K. P. Horn and J. G. Harris. Rigid body motion from range image sequences. *CVGIP : Image Underst.*, 53(1) :1–13, January 1991. 23
- [72] A. Howard. Real-Time Stereo Visual Odometry for Autonomous Ground Vehicles. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008. 19, 43
- [73] P. Ignacio, S. M. Ángel, L. David F., F. Carlos, L. Ángel, H. Noelia, and G. Iván. Visual odometry and map fusion for GPS navigation assistance. In *IEEE International Symposium on Industrial Electronics (ISIE)*, pages 832–837, june 2011. 39, 117
- [74] M. Jabbour, Ph. Bonnifait, and V. Cherfaoui. Enhanced local maps in a gis for a precise localisation in urban areas. In *IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2006. 36, 112, 115
- [75] M. Jabbour, Ph. Bonnifait, and V. Cherfaoui. Integrated navigation using gis-based information. *European Journal of Navigation*, Volume 5(N1) :38–43, 2007. 36, 112
- [76] M. Jabbour, Ph. Bonnifait, and V. Cherfaoui. Autonomous navigation in urban areas using gis-managed information. *International Journal of Vehicle Autonomous Systems*, 6(1/2) :84–103, 2008. Special Issue on Advances in Autonomous Vehicles and Intelligent Transportation. 36, 40, 112
- [77] G. R. Jagadeesh, T. Srikanthan, and X. D. Zhang. A map matching method for gps based real-time vehicle location. *Journal of Navigation*, pages 429–440, 2004. 40
- [78] C. Jekeli. *Inertial Navigation Systems With Geodetic Applications*. Walter de Gruyter, 2000. 39
- [79] R. Jirawimut, S. Prakoonwit, F. Cecelja, and W. Balachandran. Visual odometer for pedestrian navigation. *IEEE Transaction on Instrumentation and Measurements*, 52 :1166–1173, 2003. 20
- [80] A. E. Johnson and S. B. Kang. Registration and integration of textured 3-d data. In *International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, 1997. 72

- [81] J. H. Joung, K. H. An, J. W. Kang, M. J. Chung, and W. Yu. 3d environment reconstruction using modified color icp algorithm by fusion of a camera and a 3d laser range finder. In *IEEE/RSJ international conference on Intelligent robots and systems*, 2009. 72
- [82] S. J. Julier and J. K. Uhlmann. A new extension of the kalman filter to nonlinear systems. In *the 11th Int. Symp. on Aerospace/Defense Sensing, Simulation and Controls*, pages 182–193, 1997. 88, 90
- [83] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME—Journal of Basic Engineering*, 82(Series D) :35–45, 1960. 88
- [84] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window : Theory and experiment. In *IEEE International Conference on Robotics and Automation (ICRA '91)*, volume 2, pages 1088–1095, April 1991. 54
- [85] K. Kidono, J. Miura, and Y. Shirai. Autonomous visual navigation of a mobile robot using a human-guided experience. *Robotics and Autonomous Systems*, 40 :2–3, 2002. 36
- [86] M. Kieffer, L. Jaulin, E. Walter, and D. Meizel. Robust autonomous robot localization using interval analysis. *Reliable Computing*, 6(3) :337–362, 2000. 94
- [87] K. Konolige, M. Agrawal, and J. Sola. Large scale visual odometry for rough terrain. In *International Symposium on Robotics Research*, 2007. 19, 23, 43
- [88] F. A. Kruse. In *Remote sensing for the earth sciences : manual of remote sensing (3rd edition)*, chapter Visible-infrared sensors and case studies, page 567611. Wiley, New York, 2000. 24
- [89] J. Laneurit, R. Chapuis, and F. Chausse. Accurate vehicle positioning onto a numerical map. *International Journal of Control, Automation and Systems*, 3 :15–31, 2005. 40
- [90] D. J. Lee. Nonlinear estimation and multiple sensor fusion using unscented information filtering. *IEEE Signal Processing Letters*, 15(3) :861–864, 2008. 91, 93
- [91] T. Lemaire, C. Berger, I. K. Jung, and S. Lacroix. Vision-based slam : Stereo and monocular approaches. Technical report, *Int. J. Compt. Vision*, 2006. 37
- [92] J. Leonard and H. Whyte-Durrant. Mobile robot localization by tracking geometric beacons. *IEEE Transactions on Robotics and Automation*, 7(3) :376–382, June 1991. 24
- [93] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. In Martin A. Fischler and Oscar Firschein, editors, *Readings in computer vision : issues, problems, principles, and paradigms*, pages 61–62. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987. 22, 50
- [94] P. Lothe, S. Bourgeois, F. Dekeyser, E. Royer, and M. Dhome. Towards geographical referencing of monocular slam reconstruction using 3d city models : Application to real-time accurate vision-based localization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, Florida, June 2009. 36
- [95] P. Lothe, S. Bourgeois, E. Royer, M. Dhome, and S. N. Collette. Real-time vehicle global localisation with a single camera in dense urban areas : Exploitation of coarse 3d city models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, California, June 2010. 36

- [96] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2) :91–110, November 2004. 52
- [97] Y. Lu, E. G. Collins, and M. F. Selekwa. Parity relation based fault detection, isolation and reconfiguration for autonomous ground vehicle localization sensors. In *24th Army Science Conference*, 2005. 95
- [98] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *7th international joint conference on Artificial intelligence*, pages 674–679, 1981. 57
- [99] R. Madhavan and H. Durrant-Whyte. Natural landmark-based autonomous vehicle navigation. *Robotics and Autonomous Systems*, 46(2) :79–95, 2004. 32, 36
- [100] P. C. Mahalanobis. On the generalised distance in statistics. In *Proceedings of the National Institute of Sciences of India*, page 4955, 1936. 54
- [101] O. Le Marchand, P. Bonnifait, J. Ibaez-Guzman, and D. Betaille. Vehicle localization integrity based on trajectory monitoring. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3453–3458, 2009. 94
- [102] L. Matthies and S. Shafer. Error modeling in stereo navigation. *IEEE Journal of Robotics and Automation*, RA-3(3) :239 – 250, June 1987. 63
- [103] J. Meguro, T. Murata, J. Takiguchi, Y. Amano, and T. Hashizume. Gps multipath mitigation for urban area using omnidirectional infrared camera. *IEEE Transactions on Intelligent Transportation Systems*, 10(1) :22–30, March 2009. 40
- [104] D. Meizel, O. Laveque, L. Jaulin, and E. Walter. Initial localization by set inversion. *IEEE Transactions on Robotics*, 18(6) :966–971, 2002. 94
- [105] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1-2) :43–72, November 2005. 52
- [106] M. Milanese. *Bounding approaches to system identification*. Plenum Press, 1996. 94
- [107] F. A. Moreno, J. L. Blanco, and J. González. An efficient closed-form solution to probabilistic 6d visual odometry for a stereo camera. In *9th international conference on Advanced concepts for intelligent vision systems, ACIVS'07*, pages 932–942, Berlin, Heidelberg, 2007. Springer-Verlag. 20, 64
- [108] F. A. Moreno, J. L. Blanco, and J. Gonzalez. A probabilistic observation model for stereo vision systems : Application to particle filter-based mapping and localization. In *Pattern Recognition and Image Analysis*, volume 4477 of *Lecture Notes in Computer Science*, pages 346–353. Springer Berlin Heidelberg, 2007. 62
- [109] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Generic and real-time structure from motion using local bundle adjustment. *Image and Vision Computing*, 27(8) :1178–1193, July 2009. 36
- [110] A. Mueller, M. Himmelsbach, T. Luettel, F. von Hundelshausen, and H. Wuensche. GIS-Based Topological Robot Localization through LIDAR Crossroad Detection. In *IEEE Conference on Intelligent Transportation Systems*, Washington, DC, USA, October 2011. IEEE. 34
- [111] M. E. El Najjar and Ph. Bonnifait. A road-matching method for precise vehicle localization using belief theory and kalman filtering. *Autonomous Robots*, 19(2) :173–191, 2005. 100, 120

- [112] Maan E. El Najjar. A road-matching method for precise vehicle localization using belief theory and kalman filtering. *Autonomous Robots*, 19 :173–191, 2005. 117
- [113] A. Ndjeng Ndjeng, D. Gruyer, S. Glaser, and A. Lambert. Low cost imu-odometer-gps ego localization for unusual maneuvers. *Information Fusion*, 12(4) :264–274, October 2011. 39
- [114] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6) :756–777, June 2004. 22, 50
- [115] D. Nistér, O. Naroditsky, and J. Bergen. Visual odometry. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1, June 2004. 20, 21, 23, 43
- [116] S. Nogueira. *Localisation de mobiles par construction de modèle 3D en utilisant la stéréovision*. PhD thesis, Université de Technologie de Belfort - Montbéliard, 2008. 37
- [117] S. Nogueira, Y. Ruichek, and F. Charpillat. *A Self Navigation Technique using Stereovision Analysis*. Stereo Vision InTech, 2008. 36
- [118] K. Ohno, T. Tsubouchi, and S. Yuta. Outdoor map building based on odometry and rtk-gps positioning fusion. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 684–690, 2004. 112
- [119] C. F. Olson, L. Matthies, M. Schoppers, and M. W. Maimone. Rover navigation using stereo ego-motion. *Robotics and Autonomous Systems*, 43, 2003. 20
- [120] Paul Ozog and Ryan M. Eustice. On the importance of modeling camera calibration uncertainty in visual slam. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3762–3769, Karlsruhe, Germany, May 2013. 66
- [121] S. Peyraud, D. Bétaille, S. Renault, M. Ortiz, F. Mougél, D. Meizel, and F. Peyret. About non-line-of-sight satellite detection and exclusion in a 3d map-aided localization algorithm. *Sensors*, 13(1) :829–847, 2013. 40
- [122] S. T. Pfister, S. I. Roumeliotis, and J. W. Burdick. Weighted line fitting algorithms for mobile robot map building and efficient data representation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 14–19, 2003. 125
- [123] O. Pink. Visual map matching and localization using a global feature map. *Computer Vision and Pattern Recognition Workshop*, 0 :1–7, 2008. 34
- [124] B. Qin, Z. J. Chong, T. Bandyopadhyay, M. H. Ang Jr., E. Frazzoli, and D. Rus. Curb-intersection feature based monte carlo localization on urban roads. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2012. 112
- [125] M. A. Quddus, W. Y. Ochieng, and R. B. Noland. Current map-matching algorithms for transport applications : State-of-the art and future research directions. *Transportation Research Part C : Emerging Technologies*, 15(5) :312 – 328, 2007. 40, 117
- [126] A. Rae and O. Basir. A framework for visual position estimation for motor vehicles. In *Workshop on Positioning, Navigation and Communication*, Hannover, Germany, 2007. 34
- [127] A. Rae and O. Basir. Fusion of gps and machine vision for absolute vehicle positioning. In *American Society of Photogrammetry and Remote Sensing Annual Conference*, Tampa, FL, USA, 2007. 34

- [128] A. Rae and O. Basir. Reducing multipath effects in vehicle localization by fusing GPS with machine vision. In *12th International Conference on Information Fusion*, pages 2099–2106. IEEE, July 2009. 40
- [129] I.M. Rekleitis. A particle filter tutorial for mobile robot localization. Technical Report TR-CIM-04-02, Centre for Intelligent Machines, McGill University, 3480 University St., Montreal, Québec, CANADA H3A 2A7, 2004. 93
- [130] A. Remazeilles, F. Chaumette, and P. Gros. Robot motion control from a visual memory, 2004. 32
- [131] S. Renault, A. Le Meur, and D. Meizel. Gps/gis localization for management of vision referenced navigation in urban environments. In *IEEE International Conference on Intelligent Transportation Systems*, pages 608 – 613, 2005. 115
- [132] B. Riisgaard. Slam for dummies, 2005. 37
- [133] F. Rodriguez, V. Frémont, and P. Bonnifait. An experiment of a 3d real-time robust visual odometry for intelligent vehicles. In *12th International IEEE Conference on Intelligent Transportation Systems*, 2007. 19, 23, 43
- [134] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, pages 430–443, 2006. 52
- [135] E. Royer, M. Lhuillier, M. Dhome, and T. Chateau. Towards an alternative gps sensor in dense urban environment from visual memory. In *British Machine Vision Conference*, 2004. 36
- [136] E. Royer, M. Lhuillier, M. Dhome, and J. Lavest. Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision*, 74(3) :237–260, 2007. 36
- [137] E. Royer, M. Lhuillier, M. Dhome, and J. M. Lavest. Monocular vision for mobile robot localization and autonomous navigation. *Int. J. Comput. Vision*, 74(3) :237–260, 2007. 20, 23, 36, 37
- [138] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *International Conference on 3D Digital Imaging and Modeling*, 2001. 68
- [139] A. Savvides, C. C. Han, and M. B. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *7th annual international conference on Mobile computing and networking*, MobiCom '01, pages 166–179, New York, NY, USA, 2001. ACM. 38
- [140] D. Scaramuzza. Performance evaluation of 1-point-ransac visual odometry. *J. Field Robot.*, 28(5) :792–811, September 2011. 20
- [141] U. Scheunert, H. Cramer, and G. Wanielik. Precise vehicle localization using multiple sensors and natural landmarks. In Per Svensson and Johan Schubert, editors, *Seventh International Conference on Information Fusion*, volume II, pages 649–656, Mountain View, CA, Jun 2004. International Society of Information Fusion. 34, 111, 112
- [142] S. Se and P. Jasiobedzki. Stereo-vision based 3d modeling for unmanned ground vehicles,. In *SPIE Unmanned Systems Technology IX*, 2007. 36
- [143] S. Se, D. Lowe, and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks, 2002. 37
- [144] J. Shi and C. Tomasi. Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94)*, pages 593 – 600, 1994. 52

- [145] M. Y. Shih and B. C. Fu. Robust moving object detection on moving platforms. In *First Pacific Rim conference on Advances in Image and Video Technology, PSIVT'06*, Berlin, Heidelberg, 2006. Springer-Verlag. 61
- [146] B. Siciliano and O. Khatib. *Springer Handbook of Robotics*. Springer, 2008. 91
- [147] R. C. Smith and P. Cheeseman. On the representation and estimation of spatial uncertainty. *International Journal of Robotics Research*, 5(4) :56–68, December 1986. 65
- [148] F. Solina. *Errors in Stereo Due to Quantization*. Technical report. University of Pennsylvania, Moore School of Electrical Engineering. Department of Computer and Information Science, 1985. 57
- [149] H. L. Song and S. Hsien. Automatic vehicle location in cellular communications systems. *IEEE Transactions on Vehicular Technology*, 43(4) :902–908, 1994. 38
- [150] S. Sukkarieh, E. M. Nebot, and H. F. Durrant-Whyte. A high integrity imu/gps navigation loop for autonomous land vehicle applications. *IEEE Transactions on Robotics and Automation*, 15 :572–578, 1999. 39
- [151] N. Sünderhauf and P. Protzel. Stereo Odometry—A Review of Approaches, 2007. 23
- [152] J. P. Tardif, Y. Pavlidis, and K. Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2531–2538, 2008. 20
- [153] W. Thorsten, K. Nico, and D. Klaus. Precise ego-localization in urban areas using laserscanner and high accuracy feature maps. In *IEEE Intelligent Vehicles Symposium (IV2005)*, 2005. 32, 33
- [154] S. Thrun. Finding landmarks for mobile robot navigation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 958–963, 1998. 32
- [155] S. Thrun. *Simultaneous Mapping and Localization with Sparse Extended Information Filters : Theory and Initial Results*. Research paper. School of Computer Science, Carnegie Mellon University, 2002. 91
- [156] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents series)*. Intelligent robotics and autonomous agents. The MIT Press, August 2005. 37, 91, 93
- [157] R. Toledo-Moreo, D. Betaille, F. Peyret, and J. Laneurit. Fusing gnss, dead-reckoning, and enhanced maps for road vehicle lane-level navigation. *IEEE Journal of Selected Topics in Signal Processing*, 3(5) :798–809, 2009. 40
- [158] A. Tsalatsanis, K. P. Valavanis, A. Kandel, and A. Yalcin. Multiple sensor based ugv localization using fuzzy extended kalman filtering. In *Mediterranean Conference on Control & Automation*, pages 1–8, 2007. 91
- [159] M. Tsogas, N. Floudas, P. Lytrivis, A. Amditis, and A. Polychronopoulos. Combined lane and road attributes extraction by fusing data from digital map, laser scanner and camera. *Information Fusion*, 12(1) :28–36, 2011. 34
- [160] S. Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13 :376–380, 1991. 23, 59

- [161] O. Veksler. Fast variable window for stereo correspondence using integral images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 556–561, 2003. 54
- [162] V.Nguyen, S. Gächter, A. Martinelli, N. Tomatis, and R. Siegwart. A comparison of line extraction algorithms using 2d range data for indoor mobile robotics. *Autonomous Robots*, 23(2) :97–111, August 2007. 122
- [163] L. Wei, C. Cappelle, Y. Ruichek, and F. Zann. Intelligent vehicle localization in urban environments using ekf-based visual odometry and gps fusion. In *18th IFAC World Congress*, 2011. 39
- [164] J. Wendel and G. F. Trommer. Tightly coupled gps/ins integration for missile application. *Aerospace Science and Technology*, 8 :627–634, 2004. 39
- [165] C. White. Some map matching algorithms for personal navigation assistants. *Transportation Research Part C : Emerging Technologies*, 8(1-6) :91–108, December 2000. 40, 117
- [166] D. Wilkie, J. Sewall, and M. Lin. Transforming gis data into functional road models for large-scale traffic simulation. *IEEE Transactions on Visualization and Computer Graphics*, 18(6) :890–901, June 2012. 115
- [167] B. Williams and I. Reid. On combining visual slam and visual odometry. In *International Conference on Robotics and Automation*, 2010. 19, 37, 43
- [168] O. J. Woodman. An introduction to inertial navigation. Technical Report UCAM-CL-TR-696, University of Cambridge, Computer Laboratory, August 2007. 17
- [169] B. Zeisl, P. Georgel, F. Schweiger, E. Steinbach, and N. Navab. Estimation of location uncertainty for scale invariant feature points. In *British Machine Vision Conference (BMVC)*, 2009. 64
- [170] Q. Zhang and R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2301–2306, 2004. 70
- [171] W. Zhang and J. Kosecka. Image based localization in urban environments. *Third International Symposium on 3D Data Processing Visualization and Transmission*, pages 33–40, 2006. 36
- [172] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11) :1330–1334, November 2000. 47, 48

TABLE DES FIGURES

2.1	Different localization methods	15
2.2	Wheel encoder and vehicle motion model	16
2.3	3D orientations of vehicle movement	18
2.4	Localization algorithm of INS system	18
2.5	Visual odometry results of NASA's Mars Exploration Rovers	20
2.6	Some camera systems used for vision based localization	20
2.7	Localization by triangulation with three beacons	25
2.8	2D position estimation by trilateration with three beacons	26
2.9	Localization based on GPS trilateration	27
2.10	GPS satellites in space	28
2.11	Principle of GPS pseudo-ranges	28
2.12	Uncertainty areas of GPS localization under different satellites distributions	30
2.13	Multi-path problem of GPS signal	30
2.14	Differential GPS	31
2.15	Real time kinematic (RTK) GPS	31
2.16	Road geometry features extracted from image and road network	34
2.17	Vertical planes features extracted from images	35
2.18	Localization by monocular camera and 3D virtual model	35
2.19	Reconstructed environment model based on monocular/stereo images . . .	37
3.1	Pinhole camera model projection	44
3.2	Chessboard images used for camera calibration	48
3.3	Epipolar geometry of a stereoscopic system	49
3.4	Ideal stereoscopic configuration after stereo rectification	50
3.5	Working flow of stereovision based visual odometry method	51
3.6	Example of SURF features	53
3.7	Triangulation with corresponding image points	56
3.8	Vehicle model and reference pair selection	58
3.9	Urban environment with moving vehicles or strong sunlight	60

3.10	Uncertainty region of a reconstructed 3D point	63
3.11	Error propagation of stereovision based odometry	65
3.12	Some LRF systems used in autonomous vehicle research	66
3.13	Polar coordinates system	67
3.14	Comparison of classic ICP and OR-ICP methods	69
3.15	Calibration between a camera and a LRF system	70
3.16	One laser scan and its corresponding camera image points	71
3.17	ICP with corresponding image color constraint	72
3.18	LRF points are associated by coordinates and corresponding image de- scriptors	74
3.19	Configuration of the experimental vehicle	77
3.20	RTK-GPS with fixed GPS base	77
3.21	Landmarks from stereoscopic system overlapped on Google aerial image .	78
3.22	Comparison of the vehicle trajectories and yaw angles	79
3.23	Position error of stereoscopic system based estimation	80
3.24	Comparison of results respectively obtained by LRF and RTK-GPS receiver	80
3.25	Vehicle trajectory overlapped on aerial image (Google)	81
3.26	Vehicle trajectories obtained by different ICP methods	82
3.27	Comparison of the vehicle speed with the ground truth	84
3.28	Comparison of the angular velocity with the ground truth	85
4.1	Extended NIS between multiple sensors	96
4.2	Overview of the proposed UIF based vehicle localization method	97
4.3	Different coordinate frames of the vehicle system	97
4.4	Some results of vehicle state and covariance estimation with UIF method .	102
4.5	Comparison of fusion method with ground truth	103
4.6	Vehicle trajectory during the GPS masks : extended NIS increasing method	105
4.7	Vehicle trajectory during the GPS masks : extended NIS threshold method	106
4.8	Comparison of position error after adding GPS masks	106
4.9	Vehicle trajectory with simulated erroneous GPS positions	107
4.10	NIS changes of the relative measurements	108
4.11	Sensors validated for sensor fusion after adding simulated GPS jumps . . .	108
4.12	Comparison of position error after adding random GPS position errors . . .	109
5.1	In-car GPS navigation system and digital map provided by NAVTEQ	112
5.2	LRF features used for localization in outdoor environments	112

5.3	Two LRF systems are mounted on the roof of the vehicle : the vertical LRF scan is in blue and the horizontal LRF scan is in red	113
5.4	The proposed GIS aided localization method	114
5.5	Organization of GIS layers	115
5.6	Vehicle pose estimation with GIS road map layer	116
5.7	An example of road map based map-matching	117
5.8	Choosing the corresponding road segment	118
5.9	Track vehicle position on the previous road segment	119
5.10	Representation of the error ellipse of map-matching observations	119
5.11	Representation of vehicle positions uncertainties obtained by different methods	121
5.12	Method of vehicle pose refining with LRFs and original GIS building map	122
5.13	Line parameters in polar coordinates	123
5.14	Lines extracted from a horizontal LRF scan with constrained Hough transform	124
5.15	Lines detected in a horizontal LRF scan	126
5.16	Lines detected in a vertical LRF scan	127
5.17	Environment analysis with horizontal and vertical LRFs	128
5.18	Method of vehicle pose correction with non-parallel lines	129
5.19	Extraction of candidate building facades in the FOV of LRF-H scan	130
5.20	Line feature graph in a LRF scan	131
5.21	Line feature graph in a GIS building map layer	132
5.22	Correspondence graph and the maximum clique of the correspondence graph	133
5.23	Method of orientation correction with one line/parallel lines from LRF-H scan	135
5.24	Virtual intersections of the LRF-V scan with building facades	136
5.25	Lateral position correction with parallel lines	136
5.26	Method of lateral position correction with vertical lines in LRF-V scan	137
5.27	LRF-H in global reference before and after the map based pose correction	138
5.28	Covariance of facade landmarks in i^{th} LRF frame	140
5.29	Covariance of facade landmarks in global frame	140
5.30	Nearest building point of LRF-V on GIS building map	141
5.31	Detected independent objects by LRF-V	142
5.32	Corresponding positions of LRF-V based features on GIS building map	142
5.33	Process of vehicle localization with the new map	143
5.34	The experimental vehicle - SeTCar	145
5.35	DSP 3000 Fibre optic gyro	145

5.36 SICK <i>LMS221</i> and <i>LMS291</i>	146
5.37 Vehicle trajectory overlapped on the OpenStreetMap and Google aerial image	146
5.38 Trajectory from RTK-GPS and low precision GPS in Google Street View . .	147
5.39 Vehicle trajectories estimated by different approaches	147
5.40 The second vehicle trajectory overlapped on map	148
5.41 Vehicle positions obtained by road - building maps based methods	149
5.42 LRF-H scans before and after the vehicle pose correction	150
5.43 Comparison of vehicle localization errors with different approaches	151
5.44 Observation from LRFs and GIS map - result 1	152
5.45 Observation from LRFs and GIS map - result 2	153
5.46 New maps with extracted building facade landmarks and independent objects	154
5.47 One tree in front of the building is treated as an independent object by LRF-V	155
5.48 Localization results with new building map (with simulated GPS jumps) . . .	156
5.49 Localization results with new building map (with simulated GPS mask) . . .	156
5.50 Comparison of vehicle position errors after adding GPS masks	157
A.1 Transformation of GPS coordinates	182

LISTE DES TABLES

2.1	Motion estimation by 2D/2D method	21
2.2	Motion estimation by 3D/2D method	22
2.3	Motion estimation by 3D/3D method	24
2.4	Comparison of relative and absolute localization methods	38
3.1	Comparison of feature detectors	52
3.2	Area similarity based matching metrics	54
3.3	Comparison of vehicle speed and angular velocity	82
4.1	Parity relations between every two motion measurements	101
4.2	NIS relation between every two sensors (1)	101
4.3	Localization results with the different sensors and UKF, EIF, UIF	104
4.4	Localization results of the fusion methods in the area with GPS masks	105
5.1	Types of GIS objects	115
5.2	Organization of landmarks	143
5.3	Localization error during GPS mask (/meter)	150
5.4	Localization results with simulated GPS jumps (/meter)	155
5.5	Localization results with the different approaches (/meter)	157

TRANSFORMATION OF GPS COORDINATES

A.1/ COORDINATE SYSTEMS

Geodetic coordinate system : WGS84. World Geodetic System (WGS84) is the reference coordinate system used by the Global Positioning System, as shown in Fig. A.1(a) ¹. The WGS84 datum (reference ellipsoid and origin) surface is an ellipsoid with major radius $6378.13m$ at the equator, and minor radius $6356.75m$ at the poles.

The position of a geographic location on the earth's surface is described by the longitude, latitude and elevation in the world geodetic system. These are measures of the angles (in degrees) from the center of the earth to the location on the earth's surface. Latitude angles ϕ are measured in a north-south direction from the equator to the measured point. Longitude measures λ are based on the Prime Meridian from the North Pole through Greenwich to the South Pole and they stand for the angular distance from this Reference Meridian to the measured point.

Earth-centered earth-fixed coordinate system. The position represented by the longitude, latitude and elevation can be transformed to coordinates (X, Y, Z) in an earth-centered and earth-fixed (ECEF) 3D Cartesian coordinates system, as shown in Fig. A.1(b). The origin of this system $(0, 0, 0)$ is defined as the center of mass of the Earth. The X axis of this system passes through the equator at the prime meridian. The Z axis passes through the north pole. The X axis can be determined by the right-hand rule to pass through the equator at 90° longitude.

Local North-East-Up coordinate system. In order to use these information on the local surface of the earth, we need to convert the coordinates into a local surface plane XY , which is tangent to the Earth's surface fixed to a specific local location, and the Z axis is upward (as shown in Fig. A.1(c)). In order to project the coordinates from earth's spherical surface onto a two-dimensional local Cartesian coordinate plane, different projections could be chosen according to the location of the measured point. For example, in Belfort, France where we did the experiments, the extended Lambert II projection model is generally chosen, as shown in Fig. A.1(d).

Steps for converting the WGS84 coordinates to the extended Lambert II system² are

1. http://webhelp.esri.com/arcgisdesktop/9.2/index.cfm?TopicName=Georeferencing_and_coordinate_systems

2. <http://geodesie.ign.fr/contenu/fichiers/documentation/pedagogiques/transfo.pdf>

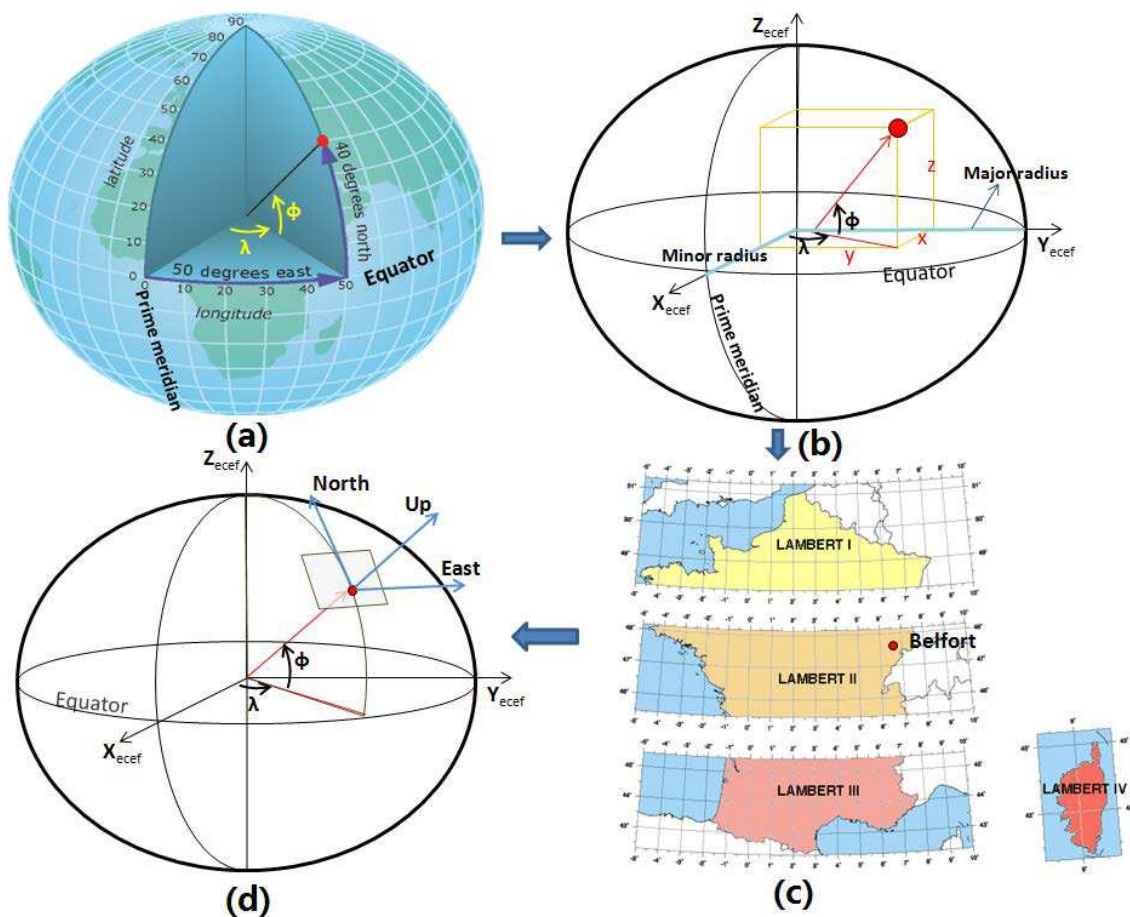


FIGURE A.1 – a) WGS-84 reference ellipsoid for GPS, a location is represented by Geographic latitude ϕ and longitude λ ; b) Earth-centered earth-fixed coordinate system (ECEF) position : (x, y, z) ; c) Lambert projection from geographic coordinates to Cartesian coordinates in France. Different projections could be chosen according to the location; d) Local North-East-Up coordinate system

detailed in the following part.

A.2/ TRANSFORMATION FROM WGS84 TO EXTENDED LAMBERT II

Data from GPS receiver are in the form of NMEA sentences, in which the longitude and the latitude are the most important information. These information are in the geodetic system of GPS : WGS84. In order to use these information on the local surface of the earth, we need to transform these data by the projection of the longitude and latitude information onto the earth surface. The WGS84 data are converted into the extended Lambert II system in several steps shown in Algorithm 5.

Algorithme 5: Transformation from WGS 84 to Extended Lambert II

Input : Geographic latitude and longitude $(\phi_{w0}, \lambda_{w0})$ from GPS receiver

Output : Coordinates in extended Lambert II system (x^g, y^g)

0. Transform the default latitude and longitude $(\phi_{w0}, \lambda_{w0})$ from decimal degrees format to Degrees/radians format (ϕ_w, λ_w) :

$$\text{Latitude : } \underbrace{\times \times}_{\text{Degrees}} \underbrace{\times \times . \times \times \times \times}_{\text{Minutes}} N/S, \text{Longitude : } \underbrace{\times \times \times}_{\text{Degrees}} \underbrace{\times \times . \times \times \times \times}_{\text{Minutes}} W/E \quad (\text{A.1})$$

for example, if $(\phi_{w0}, \lambda_{w0}) = (4235.5632, N, 00623.3453, W)$, then :

$$\phi_w = (42 + 35.5632/60) \times \pi/180, \lambda_w = (6 + 23.3453/60) \times \pi/180$$

1. Transform (ϕ_w, λ_w) to Cartesian coordinates (x_w, y_w, z_w) : $e_w = \frac{a_w^2 - b_w^2}{a_w^2}$, $N = \frac{a_w}{\sqrt{1 - e_w \sin^2(\phi_w)}}$, then :

$$\begin{cases} x_w = N \cos \phi_w \cos \lambda_w \\ y_w = N \cos \phi_w \sin \lambda_w \\ z_w = N(1 - e_w) \sin \phi_w \end{cases} \quad (\text{A.2})$$

with : $a_w = 6378137$ and $b_w = 6356752.314$.

2. Transform (x_w, y_w, z_w) to Cartesian coordinates NTF (Nouvelle Triangulation de la France) (x_n, y_n, z_n) by the translation vector $(168, 60, -320)$, with :

$$x_n = x_w + 168, y_n = y_w + 60, z_n = z_w - 320.$$

3. Transform (x_n, y_n, z_n) to geographic coordinates NTF (ϕ_n, λ_n) :

$$e_n = \frac{a_n^2 - b_n^2}{a_n^2} \quad (\text{A.3})$$

$$p_0 = \text{atan}(d_n z_n (1 - (a_n e_n) / (\sqrt{x_n^2 + y_n^2 + z_n^2}))) \quad (\text{A.4})$$

$$p_1 = \text{atan}(d_n z_n / (1 - (a_n e_n \cos(p_0)) / (\sqrt{(x_n^2 + y_n^2)(1 - e_n \sin^2(p_0))}))) \quad (\text{A.5})$$

with : $a_n = 6378249.2$, $b_n = 6356515$ and $d_n = 1 / \sqrt{x_n^2 + y_n^2}$.

3.1 **While** $|p_1 - p_0| > 1e^{-10}$, do

$$p_0 = p_1 ;$$

Re-calculate p_1 with Eq.A.5 ;

End

3.2 Then, $\phi_n = p_1$ and $\lambda_n = \text{atan}(y_n/x_n)$.

4. Transform (ϕ_n, λ_n) to the coordinates in the extended Lambert II system (x^g, y^g) :

$$\begin{cases} x^g = x_s + c e^{-nL} \sin(n(\lambda_n - \lambda_0)) \\ y^g = y_s - c e^{-nL} \cos(n(\lambda_n - \lambda_0)) \end{cases} \quad (\text{A.6})$$

with : $n = 0.7289686274$, $c = 11745793.39$, $x_s = 600000$, $y_s = 8199695.768$,

$$\lambda_0 = 0.04079234433198 \text{ and } L = \log\left(\tan\left(\frac{p_1}{4} + \frac{\phi_n}{2}\right)\left(\frac{1 - \sqrt{e_n} \sin \phi_n}{1 + \sqrt{e_n} \sin \phi_n}\right)\right) \sqrt{e_n}/2.$$

Résumé :

Afin d'améliorer la précision des systèmes de navigation ainsi que de garantir la sécurité et la continuité du service, il est essentiel de connaître la position et l'orientation du véhicule en tout temps. La localisation absolue utilisant des systèmes satellitaires tels que le GPS est souvent utilisée à cette fin. Cependant, en environnement urbain, la localisation à l'aide d'un récepteur GPS peut s'avérer peu précise voire même indisponible à cause des phénomènes de réflexion des signaux, de multi-trajet ou de la faible visibilité satellitaire. Afin d'assurer une estimation précise et robuste du positionnement, d'autres capteurs et méthodes doivent compléter la mesure. Dans cette thèse, des méthodes de localisation de véhicules sont proposées afin d'améliorer l'estimation de la pose en prenant en compte la redondance et la complémentarité des informations du système multi-capteurs utilisé. Tout d'abord, les mesures GPS sont fusionnées avec des estimations de la localisation relative du véhicule obtenues à l'aide d'un capteur proprioceptif (gyromètre), d'un système stéréoscopique (odométrie visuelle) et d'un télémètre laser (recalage de scans télémétriques). Une étape de sélection des capteurs est intégrée pour valider la cohérence des observations provenant des différents capteurs. Seules les informations validées sont combinées dans un formalisme de couplage lâche avec un filtre informationnel. Si l'information GPS est indisponible pendant une longue période, la trajectoire estimée par uniquement les approches relatives tend à diverger, en raison de l'accumulation de l'erreur. Pour ces raisons, les informations d'une carte numérique (route + bâtiment) ont été intégrées et couplées aux mesures télémétriques de deux télémètres laser montés sur le toit du véhicule (l'un horizontalement, l'autre verticalement). Les façades des immeubles détectées par les télémètres laser sont associées avec les informations « bâtiment » de la carte afin de corriger la position du véhicule.

Les approches proposées sont testées et évaluées sur des données réelles. Les résultats expérimentaux obtenus montrent que la fusion du système stéréoscopique et du télémètre laser avec le GPS permet d'assurer le service de localisation lors des courtes absences de mesures GPS et de corriger les erreurs GPS de type saut. Par ailleurs, la prise en compte des informations de la carte numérique routière permet d'obtenir une approximation de la position du véhicule en projetant la position du véhicule sur le tronçon de route correspondant et enfin l'intégration de la carte numérique des bâtiments couplée aux données télémétriques permet d'affiner cette estimation, en particulier la position latérale.

Mots-clés : Localisation de véhicule, fusion multi-capteur, véhicule intelligent, télémètre laser, vision stéréoscopique, système d'information géographique

Abstract:

In order to improve the accuracy of assisted navigation systems so as to guarantee driving security on road, it is essential to know the absolute/relative vehicle positions and orientations at all times. In some dense urban environments (e.g., a street with tall buildings around), vehicle localization results provided by Global Positioning System (GPS) receiver might not be accurate or even unavailable due to signal reflection (multi-path) or poor satellite visibility. Two vehicle localization methods were proposed in this thesis to assist vehicle localization in urban environments by taking use of the redundancy and complementarity of multiple sources. 1) GPS localization method is complemented by a gyroscope, stereovision based visual odometry, horizontal laser range finder (LRF) based scan alignment to provide a coarse vehicle pose estimation. A sensor selection step is applied to validate the coherence of different observations: only information provided by the validated sensors are combined under a loosely coupled probabilistic framework. 2) If GPS signals are lost for long period, the accumulated localization error of DR-only method are bounded by a road network map and a building footprint map, together with two LRF systems (a horizontal LRF and a vertical LRF) mounted on the roof of the vehicle. The extracted facade landmarks from the vertical LRF scan are stored in a new GIS map layer. The proposed approach is tested and evaluated with real data sequences. Experimental results with real data show that fusion of the stereoscopic system and LRF can continue to localize the vehicle during GPS outages in short period and to correct the GPS positioning error such as GPS jumps; the road map can help to obtain an approximate estimation of the vehicle position; and the integration of the building information can help to refine the initial pose estimation when GPS signals are lost for long time, especially for the lateral position.

Keywords: Vehicle localization, multi-sensor fusion, intelligent vehicle, laser range finder, stereovision, geographic information system