Landmark based localization : Detection and update of landmarks with uncertainty analysis

Xiaozhi Qu

UNIVERSITÉ PARIS-EST

École Doctorale MSTIC

Sciences et Technologies de l'Information Géographique

Xiaozhi QU

Localisation basée amers visuels: détection et mise à jour d'amers avec gestion des incertitudes

Présentée pour obtenir le grade de docteur
de Université PARIS-EST
Octobre, 2016

Jury de Thèse

This thesis is supported by MATIS (Méthodes d'Analyses pour le Traitement d'Images et la Stéréorestitution) team of the Research Department in IGN (Institut National de l'Information Géographique et Forestière).

Cette thèse s'est déroulée au l'équipe MATIS (Méthodes d'Analyses pour le Traitement d'Images et la Stéréorestitution) du Service de la Recherche de l'Institut National de l'Information Géographique et Forestière (IGN).

BibTex entry / BibTex fiche :

```
@phdthesis{QU:PhD,
    title       =   {Landmark based localization: detection, matching
                     and update of landmark with uncertainty analysis},
    author      =   {Xiaozhi QU},
    school      =   {Université Paris-Est},
    year        =   {2016},
    address     =   {Marne-la-Vallée, France},
}
```

Abstract

Mobile mapping is the process of collecting geospatial data with a moving vehicle. These vehicles are often equipped with two types of sensors: remote sensing (cameras, lidar, radar) and geo-localization (GNSS, IMU, odometer). Precise and robust georeferencing has been a major challenge for the implementation of mobile mapping systems. Indeed, in dense urban environments, the masks of signals and multipath errors corrupt the measurements and lead to big positioning errors. High precision IMUs enable to bridge the gaps of positioning and ensure a drift low enough to fulfil the requirements of mapping in terms of accuracy. Nowadays, the hybrid positioning systems (GNSS / IMU / Odometer) are mature enough to provide reliable industrial solutions for the collection of geo-referenced data. National and private mapping agencies have started to collect the required row data for building geospatial repositories at very large scales. However, the very high cost of positioning systems incorporating high precisions IMUs restricts their use to the establishment of geospatial reference data and more affordable positioning solutions are needed for map updating purpose.

The objective of this thesis is to provide a low cost positioning solution that can be used on a large number of map updating vehicles.

We propose to use one or more cameras on a vehicle as a georeferencing system. Indeed, the vehicle's trajectory can be estimated using visual odometry techniques. To limit the drift of the trajectory due to the accumulation of errors, we propose a registration on a set of visual landmarks that are precisely georeferenced. These landmarks are reconstructed using the reference data generated by precise and expensive mapping systems. Natural road features such as road markings and traffic signs were chosen as visual landmarks.

A local bundle adjustment algorithm has been adapted to estimate the pose of the vehicle using a sequence of images acquired by one or more embedded cameras. A rigorous approach that takes into account the uncertainties enables to tune automatically the weights of every constraint in the equation system of the adjustment and to estimate the uncertainties of the parameters. They are used in a propagation based matching algorithm that accelerates the process of tracking the interest points between the images and eliminate many false matches. This significantly reduces the drift of the visual odometry by reducing the sources of errors. The remaining part of the drift is removed using georeferenced visual landmarks. The process of matching the image sequence with the landmarks is guided by the uncertainty of the poses. It adds a set of absolute constraints in the equation system of bundle adjustment. The drift is drastically reduced. Each

step of the algorithm is evaluated on real image sequences with ground truths.

Keywords : *Localization, Landmark, Local Bundle Adjustment, Uncertainty analysis, Ground Control Points, Error propagation, Traffic signs, Road markings, Pose estimation.*

Résumé

En utilisant des amers visuels géoréférencés. Le processus d'appariement de flux d'images avec ces amers est guidé par les incertitudes des poses. On ajoute des contraintes absolues dans le système d'équations de l'ajustement de faisceaux. La dérive de la trajectoire du véhicule de cartographie est très fortement réduite. Chaque étape de l'algorithme est évaluée sur des séquences d'images réelles avec une vérité terrain. Le véhicule de cartographie est le processus de collecte des données géo-spatiales. Ces véhicules sont souvent équipés de deux types de capteurs : télédétection (caméras, Lidar, Radar) et géolocalisation (GNSS, IMU, Odomètre). Le géoréférencement des données précis et robuste constitue un enjeu majeur pour la mise en œuvre des systèmes de cartographie mobile. En effet, en milieux urbains denses, les phénomènes de masquages GNSS et de trajets multiples pertubent les mesures de GNSS et conduisent à des erreurs de localisations importantes. Les centrales inertielles de grandes précisions permettent de combler les manques de localisation GNSS. Elles garantissent une dérive de position suffisamment faible pour obtenir la qualité de géoréférencement nécessaire pour la numérisation à des fins cartographiques. Aujourd'hui, la maturité des systèmes de géolocalisation hybride (GNSS/IMU/Odomètre) offre des solutions industrielles fiables pour la collecte de données géoréférencées. Les agences de cartographie nationales et privées ont commencées à faire des acquisitions de données nécessaires à la constitution de données géo-spatiales à très grande échelle. Cependant, le coût très onéreux des systèmes de géolocalisation intégrant des centrales inertielles de grandes précisions restreint leur utilisation à la constitution de données géoréférencées. Une solution plus abordable est nécessaire pour équiper les véhicules employés pour les mises à jour régulières de ces données.

L'objectif de cette thèse est de proposer une solution abordable de géolocalisation utilisable sur un grand nombre de véhicules pouvant être mobilisés pour la mise à jour de données géoréférencées.

Nous proposons d'utiliser une ou plusieurs caméras sur un véhicule comme un système de géoréférencement. En effet la trajectoire du véhicule peut être estimée par une technique d'odométrie visuelle. Pour limiter la dérive de la trajectoire due à l'accumulation des erreurs, nous proposons de le recaler sur un ensemble d'amers visuels précisément géoréférencés. Ces amers sont reconstruits en utilisant les données géoréferencées générées par des systèmes de cartographies précis et onéreux. Les caractéristiques de route telles que les signalisations horizontales et verticales ont été choisi en tant que amers visuels.

Un algorithme d'ajustement de faisceaux local a été adapté pour estimer la pose des caméras en utilisant un flux d'images acquis par un ou plusieurs caméras embarquées sur celui-ci. Une méthode rigoureuse de prise en compte des incertitudes permet de pondérer de manière automatique les différents types de contraintes dans le système d'équations de l'ajustement et d'estimer les incertitudes des paramètres. Ces dernières sont utilisées dans une approche appelée appariement par propagation qui permet d'accélérer le processus de suivi des points d'intérêt entre les images et d'éliminer un grand nombre de faux appariements. Cela réduit très fortement la dérive du véhicule en diminuant les sources des erreurs. Chaque étape de l'algorithme est évaluée sur des séquences d'images réelles avec des vérités terrains.

Mots Clés : *localisation, amers visuels, ajustement de faisceaux local, analyse des l'incertitudes, les points de contrôle au sol, propagation d'erreur, panneaux de signalisation, marquages routiers, estimation de pose*

Contents

Abstract iii

Résumé v

Tables of content x

1 Introduction 1
- 1.1 Context ... 1
- 1.2 STEREOPOLIS mobile mapping system ... 2
  - 1.2.1 Localization system ... 3
  - 1.2.2 Cameras ... 3
  - 1.2.3 LiDAR ... 4
- 1.3 Examples of applications ... 5
  - 1.3.1 Detection and reconstruction of road feature ... 5
  - 1.3.2 Reconstruction of building ... 6
  - 1.3.3 Detection of individual object ... 7
  - 1.3.4 Integrated 3D city model ... 7
- 1.4 The motivation of our research ... 8
- 1.5 Thesis outline ... 9

2 Related work 11
- 2.1 Global localization ... 12
  - 2.1.1 Global Navigation Satellite System ... 13
  - 2.1.2 Localization based on beacons ... 13
  - 2.1.3 Place registration ... 15
- 2.2 Position tracking ... 16
  - 2.2.1 Dead reckoning ... 16
  - 2.2.2 SLAM ... 17
- 2.3 Combing system ... 21
  - 2.3.1 GNSS/IMU system ... 21
  - 2.3.2 Cooperative Localization ... 21
  - 2.3.3 Maps constrained localization ... 22

2.4 Methodologies for localization 23
2.4.1 Probabilistic filter 24
2.4.2 Bundle adjustment 26
2.5 Camera configuration for localization 27
2.5.1 Monocular camera 27
2.5.2 Stereo cameras 28
2.5.3 Omnidirectional camera 28
2.5.4 Multi-camera system 29
2.6 Integration of external data 30
2.6.1 GNSS data 31
2.6.2 Low level maps 32
2.6.3 GIS data 33
2.6.4 Semantic features 34
2.7 Our strategy 38

3 Localization using vision based system 41
3.1 Camera model and tie point 42
3.1.1 Camera projection model 42
3.1.2 Tie point structure 44
3.2 Feature extraction, matching and tracking 45
3.2.1 Feature extraction 45
3.2.2 Matching between images 46
3.2.3 Tracking tie points 47
3.3 Single camera based approach 49
3.3.1 Initialization 49
3.3.2 Initial estimation of poses and tie points 50
3.3.3 Key frames selection 51
3.3.4 Refinement with LBA 52
3.3.5 Experiments using monocular images 58
3.4 Multi-camera based localization 61
3.4.1 Rigorous projection model 61
3.4.2 Parameters initialization 63
3.4.3 LBA for multi-cameras based localization 65
3.4.4 Experiment for different camera configurations 68
3.5 Conclusion of vision based localization 74

4 Propagation based matching and tracking 75
4.1 Overview 76
4.1.1 Problem statement 76
4.1.2 Our solution 77

4.2 Predict and update 79
4.2.1 Motion model 80
4.2.2 Motion prediction 80
4.2.3 Motion model update 81
4.2.4 Uncertainty propagation 82
4.3 Guided matching 82
4.3.1 Generation of searching window 83
4.3.2 Similarity measurement 85
4.3.3 Sub-pixel matching 86
4.4 Generation of new tie points 87
4.4.1 Interest points detection 87
4.4.2 Matching for monocular images 89
4.4.3 Matching for stereo images 92
4.4.4 Matching for multi-camera images 95
4.5 Experiments of new tracking methods 96
4.5.1 Relative Errors 96
4.5.2 Evaluation on training datasets 97
4.5.3 Evaluation on test datasets 100
4.5.4 Absolute errors and trajectories 101
4.5.5 Efficiency analysis 104
4.6 Conclusion 104

5 Geo-referenced landmarks based localization 105
5.1 Overview 105
5.1.1 Compared with classical GCPs 106
5.1.2 GCPs and tie points 107
5.2 Geo-referenced landmarks 107
5.2.1 Geo-referenced traffic signs 108
5.2.2 Geo-referenced road markings 109
5.3 Generation of GCPs from geo-referenced landmarks 113
5.3.1 Selection of geo-referenced candidates 113
5.3.2 Uncertainty propagation for landmark registration 115
5.3.3 Landmarks correspondences with images 116
5.4 Integration of GCPs with LBA 120
5.4.1 Formulations for monocular sequences 121
5.4.2 Integration of GCPs for multi-camera system 122
5.5 Experiments 123
5.5.1 Experiment of using traffic signs 123

5.5.2 Experiment of using road markings 131
5.6 Conclusion and perspectives 135

6 Summary **137**
6.1 Contributions 138
6.1.1 Multi-camera based localization 138
6.1.2 Integration of geo-referenced landmarks 139
6.1.3 Propagation based matching and tracking 140
6.1.4 Uncertainty analysis 141
6.1.5 Evaluation 142
6.1.6 Integration of the methods in THINGS2DO 143
6.2 Perspectives 143
6.2.1 Landmark based localization using top-down approach 143
6.2.2 Integration of low-cost sensors 144
6.2.3 Integrating multi-level landmarks 146
6.2.4 Integrating image segmentation with localization 147
6.2.5 Network design 148
6.3 Conclusion 148

Bibliography **169**

List of Figures

1.1 Same area observed by aerial image and street view image captured by mobile mapping system. 2

1.2 The mobile mapping system in MATIS. 3

1.3 Camera positions and images on STEREOPOLIS. (a) Positions of the 14 camera on STEREOPOLIS. (b) A set of images captured by cameras on STEREOPOLIS. Images in the middle compose the panorama view and two stereo rigs are mounted for forward and backward looking. 4

1.4 LiDAR sensors and point clouds. (a) RIEGL VQ-250. (b) Point clouds acquired by RIEGL VQ-250 on STEREOPOLIS II. (c) VELODYNE HDL-64E. (d) Point clouds of one scan obtained by Velodyne HDL-64E. 4

1.5 (a) Detecting traffic signs from geo-referenced images [Soheilian et al., 2013a]. (b) Extracting road markings from point clouds [Hervieu et al., 2015]. (c) Road side detection based on point clouds [Hervieu and Soheilian, 2013]. 6

1.6 Some examples about build facades. (a) Facade segmentation from single image [Burochin et al., 2009]. (b) Build facade reconstruction based on stereo images [Pénard et al., 2005]. (c) Mesh generation using point clouds [Demantké et al., 2013]. 6

1.7 Individual objects detection based on the point clouds acquired by STEREOPOLIS. (a) Tree detection [Monnier et al., 2012]. (b) Vehicle detection [Xiao et al., 2016]. (c) Moving object detection [Vallet et al., 2015] 7

1.8 Integrated 3D model [Soheilian et al., 2013b]. (a) 3D city model contains high resolution textured buildings, road markings, traffic signs and a subset of geo-referenced high resolution images. The free space on road is drawn in green and obstacles are in red. (b) Top: image captured by image. Down: virtual image generated from integrated 3D city model. 8

2.1 Overview of localization. 11

2.2 (a) Computing the position of $\boldsymbol{X}$ by measuring distances from $\boldsymbol{X}$ to at least three beacons, (b) Localization based on the measurement of angles and distance [Hightower and Borriello, 2001]; the distance between two stations is known, the angles can be obtained by measuring the phase shift of signal. 14

2.3 (a) Localization based on visual street map [Wong et al., 2014]. (b) Localization based on vocabulary tree using 3D data reconstructed with SFM [Irschara et al., 2009] 16

2.4 (a) SLAM problem depicted as Bayesian network graph. (b) The graph for *full SLAM*problem. SLAM as Markov Random Field without representing the measurements explicitly. (c) Online SLAM. (d) Key-frame based BA [Strasdat et al., 2010a] 24

2.5 Some examples of omnidirectional cameras [Scaramuzza, 2014]. 29

2.6 Examples of different external resources. (a) GPS receiver[Lhuillier, 2011]. (b) Digital map from OpenStreetMap [Floros et al., 2013]. (c) Textured 3D city model[Caron et al., 2014]. (d) Road extracted from 3D point clouds[Levinson et al., 2007]. 30

2.7 The most related research about localization with our method. 38

2.8 The proposed flowchart of our localization approach. 39

3.1 Pipeline of vision based localization. 41

3.2 Pinhole projection from 3D to 2D [Hartley and Zisserman, 2003]. 42

3.3 An example of tie point structure. X is the tie point, $I_{i-3}, I_{i-2}, I_{i-1}, I_i$ are four consecutive images. The correspondences(m_j, m_l, m_n, m_k) are the observations of X in images. 44

3.4 Matching graphs for monocular and binocular image sequences between current frame (pair) and latest three key frames (pairs) 47

3.5 Tracking across pair-wise matches. 48

3.6 X_1 represents the existing tie point and X_2 is the new tie point generated from image I_i and I_{i-1}. 49

3.7 Schematic of the LBA processing procedure. The zoom-up digram at left-bottom presents the different parameters in second step, marked with green dotted rectangle. 53

3.8 LBA graph. 56

3.9 The structure of Jacobian and normal matrix in LBA as shown in figure 3.8. The dark boxes represent the non-zero blocks in matrix. 57

3.10 Example of feature extraction and matching. (a) Image acquired by STEREOPOLIS. (b)SIFT feature points. (c) Pair-wise matches with previous frame. 58

3.11 Trajectory estimated by vision based localization and the ground truth. 59

3.12 The lateral and depth errors of localization. 60

3.13 Uncertainty propagation. Each error ellipsoid is exaggerated three times. 60

3.14 General concept of viewpoint. The blue coordinate system expresses local system of viewpoint in world. The red arrows present the offsets and rotation from every camera to the coordinate system of viewpoint. 62

3.15 Estimation of camera pose. The blue XYZ frame is the world coordinate system, $\boldsymbol{C}_t$ presents the position and $\boldsymbol{R}_t$ is the orientation of viewpoint at time t. The pose of camera i can be computed by rigid transformation from view point. 63

3.16 Left: the P3P problem for perspective projection. Right: NP3P problem, solving the pose of three arbitrary rays emanating from a generalized camera geometry and meeting three world points. The camera model can be any arbitrary projection [Nistér and Stewénius, 2007]. 64

3.17 The positions of the four cameras in STEREOPOLIS and the camera coordinate systems. 68

3.18 Design of camera configuration. **F**: Forward looking. **B**: Backward looking. (a) **Mono**: monocular camera. (b)**F_F**: two front cameras. (c) **F_B**: one front and one back camera. (d) **F_F_B_B**: using four cameras. 69

3.19 Comparing the estimated paths with ground truth in map. GT: Ground Truth ... 70

3.20 Accuracy of viewpoint position. 71

3.21 Absolute errors and volumes of error ellipsoid of viewpoint positions. (a) Absolute errors of locations. (b) Volume of error ellipsoid of viewpoint position. . 72

3.22 Uncertainties of position for **Mono** (light ellipsoids) and **F_F_B_B** (dark ellipsoids). Blue: trajectory of **Mono**. Red: trajectory of **F_F_B_B**. Green: ground truth. 72

4.1 Matching and tracking for localization. 75

4.2 Problem of pairwise matching for image on high-speed road. (a) 413 SIFT feature points. (b) 364 SIFT feature points. (c) pairwise matches between the points in (a) and (b). 76

4.3 (a) epipolar lines between images 4.2(a) and 4.2(b). (b) pixel flow between corresponding points 77

4.4 The location and orientation of frames. 77

4.5 Flowchart of propagation based matching and tracking. 78

4.6 Prediction and uncertainty propagation for new frame. A motion model is built using priorly estimated poses to predict the pose of new frame I_t. The uncertainty of the prediction is propagated from previous poses via motion model. 79

4.7 Generation of searching area for every tie point in new frame. The blue circles ($X_1, ...X_j, ...X_n$) are tie points in world coordinate system. The dotted ellipse in I_t is the error ellipse of the projection of X_j. The red rectangle is the bounding box of error ellipse which is also the searching area used for matching with image point in I_{t-1}. 83

4.8 The NCC values in a 10×10 neighborhood window around the location of maximum coefficient value. 86

4.9 Corner with sub-pixel accuracy. $\mathbf{p}$ is the location of corner. The red lines are the tangent lines of the corner in given window. 88

4.10 An example of selection of new interest points. 89

4.11 Matching for monocular case. Red cross in I_t represents the new interest point, the yellow rectangle is the searching area we expect to find the correspondence of the new interest point. 90

4.12 Existing correspond points between I_t and I_{t-1}. Red circle: the locations of image points in I_{t-1}. Blue circle: Matches of achieved by guided matching in section 4.3. Green lines: displacements of pixels. 91

4.13 Discrete values of pixel displacement. (a) Displacement in x axis.(b) Displacement in y axis. 91

4.14 Circle matching for consecutive image pairs. 92

4.15 Matching of a stereo pair. The top image is the captured by left camera and the bottom one is the right image. The start points and searching scope in epipolar line for the interest points in left image are drawn with same color. 95

4.16 Accuracy of translation for KITTI training datasets. 98

4.17 Accuracy of rotation errors. 99

4.18 (a) Absolute errors of positions. (b) Trajectories of ground truth, estimated using original method and new approach for sequence 00. 102

4.19 (a) Absolute errors of image positions. (b) Trajectories of ground truth, estimated using original method and new approach for sequence 01. 103

5.1 Flowchart of integration of geo-referenced landmarks with localization. 106

5.2 Reconstruction of traffic signs from geo-referenced images. 108

5.3 Generation of intensity ortho-image from point clouds. (a) Point clouds. (b) intensity ortho-image of the points. 110

5.4 Library of road marking template patterns ($GSD = 2cm$). 110

5.5 The object i with parameters (ℓ_i =`bike`,$x_i, y_i, \theta_i, \lambda_i$). 111

5.6 Simulated annealing-coupled RJ-MCMC optimization[Hervieu et al., 2015]. (a) initial configuration (b) RJ-MCMC optimization. (c) final results. 113

5.7 The relations between current image and geo-referenced landmarks. $\boldsymbol{C}_t$ is the position of image center in geo-referenced frame for image t, $\boldsymbol{V}_c^t$ is the depth direction of camera and $\boldsymbol{V}_X^i$ is the normal vector of traffic sign. 114

5.8 The image pose and geo-referenced road sign are provided with their uncertainties. The road sign can be projected into image plane and the uncertainty for every vertex of the projected shape is estimated with error propagation. All the ellipses determine the region for detection. 117

5.9 An example of complex situations for road markings. 118

5.10 Error propagation and searching space generation for road markings. 119

5.11 Example of MCMC optimization. 120

5.12 Experiment data 124

5.13 Trajectories of Ground Truth (GT), vision based localization (LBA) and integration of Traffic Signs (TS) for localization (LBA+TS). Black crosses stand for the locations of traffic signs. 125

5.14 **Blue**: the errors of localization using LBA. **Red**: the errors for the case of traffic signs integration. Black cross: containing GCP in the image. 125

5.15 **Red**: LBA+TS based localization. **Blue**: LBA based localization. **Cyan**: error ellipsoids of LBA based localization. **Grey**: error ellipsoids using LBA+TS. **Close-up windows**: accuracy improvement of tie points. All the error ellipsoids are ten times larger than their original size for visualization. 126

5.16 Comparing estimated trajectories with ground truth. 127

5.17 Absolute errors of localization. 128

5.18 **Red**: estimated trajectory using stereo rig and traffic signs. **Blue**: estimated trajectory using only stereo rig. **Cyan**: error ellipsoids of pose estimation using only stereo rig (LBA). **Grey**: error ellipsoids of pose estimation using stereo rig integrated with traffic signs (LBA+RS). **Close-up windows**: tie points in the selected areas. The size of all the error ellipsoids are exaggerated ten times for visualization. 129

5.19 Traffic signs detection. Yellow rectangle: searching area for each traffic sign. Green polygon: detected traffic signs. 130

5.20 (a) Testing area. (b) Geo-referenced road marking objects after manually editing. (c) An example of image acquired by the monocular camera. 131

5.21 The uncertainties of localization at the first $20m$. The error ellipsoids are exaggerated ten times and the ground truth trajectory is drawn in green. 132

5.22 Trajectories of localization without and with road marks, compared with ground truth. Error ellipsoids are exaggerated 10 times. Ground-truth trajectory is drawn in green. 133

5.23 The lateral and depth errors of localization. The blue line presents the errors for localization using LBA. The red line shows the errors of localization with the integration of road markings. 134

5.24 Absolute errors for localization without and with road marks. 134

6.1 The framework of our localization. 138

6.2 Integration of landmarks with localization. 139

6.3 Propagation based matching and tracking for tie points. 140

6.4 Integrating simulated GPS points with monocular based localization. Green: ground truth. Blue: LBA based localization. Red: GPS data integration with LBA based localization. 145

6.5 Projecting the 3D model into image with the approximate image pose. 146

6.6 Semantic segmentation based on deep learning. 147

6.7 Visual salient map (white parts). 148

List of Tables

2.1 Different types of ranging and optical sensors. 17
2.2 Summary of some vision based localization systems 36

3.1 Total time spent in each part of localization. 73

4.1 Comparison of relative errors on translation and rotation. 97
4.2 Accuracy of state-of-the-art visual odometry methods. 100
4.3 Processing time for the original and new tracking and matching methods. 104

5.1 Extra computation caused by GCPs. 130
5.2 Experiments summary. 135

Chapter 1

Introduction

1.1 Context

The up-to-date information about civil infrastructure is demanded for various applications such as city planning [Yeh, 1999], pavement management [Miraliakbari et al., 2014], self-driving [Guizzo, 2011] and virtual city [Dodge et al., 1998]. This therefore motivates the establishment of Geographic Information Systems (GIS) database for urban environment. To produce the data, the conventional methods are based on satellite remote sensing or aerial photogrammetry. They extract the features like road network [Ruskoné, 1996] or reconstruct the 3D buildings [Flamanc et al., 2003] from aerial or satellite images. These methods can produce large scale GIS data and update the database in high rate with reasonable cost, but the accuracy and the level of details are not sufficient for many applications (only with centimetric to decimetric resolution for aerial images and much lower resolution for satellite images). Besides, the vertical view images provide only a part of required information. Some features such as the build facades are missing or observed under very restricted angle. Recently, oblique aerial photography is used to reconstruct all sides of buildings [Karbo and Schroth, 2009] which is a great complementary for traditional photogrammtetry. Nevertheless, some important ground level targets such as traffic signs and sometimes road markings can not be captured completely because of view occlusion.

Figure 1.1 shows aerial and street-view images for the same road. The road markings can be partly observed in figure 1.1(a), but some road markings are sheltered by the trees. In addition, the traffic signs (vertical objects) in street are too difficult to be observed by the aerial images because of the vertical view, small size and occlusion. However, mapping these features is mandatory for many applications such as traffic simulation [Wilkie et al., 2012]. The street level imagery (optical and laser) can provide raw data for mapping these objects (*cf.* Fig. 1.1(b)).

Mobile Mapping Systems (MMSs) simplify the acquisition of street level data [El-Sheimy, 1996]. The camera or Light Detection And Ranging (LiDAR) are usually mounted on a vehicle. A set of features can be extracted from the captured images and reconstructed with photogrammetry or computer vision methods. The LiDAR scans the environment and generates dense

(a) Aerial photo (b) Street view

Figure 1.1: Same area observed by aerial image and street view image captured by mobile mapping system.

point clouds directly. Direct geo-referencing devices (GNSS/INS/odometer) are also integrated on MMS. The Global Navigation Satellite System (GNSS) is used for global positioning and the Inertial Navigation System (INS) can provide instant translation and orientation of vehicle. Together with the distance measured by odometer, the localization is achieved by fusing all the data. In order to capture detailed spatial data of street infrastructure (e.g. building facades, pavements, traffic signs, road markings, street furniture, etc.), the research about street-based mobile mapping has been conducted many years in MATIS research group of IGN [1].

1.2 STEREOPOLIS mobile mapping system

The first generation of MMS (STEREOPOLIS) in MATIS, was developed based on stereo vision system (*cf.* Fig 1.2(a)). The STEREOPOLIS consisted of two stereo rigs (4 cameras). One back-looking stereo rig (horizontal baseline) allows acquiring images of street feature and one side-Looking stereo rigs (vertical baselines) capture the images of building facades on both sides of the street. Only GPS is used for geo-referencing in STEREOPOLIS. Whereas the quality of localization may be altered in urban area due to GPS masks and multi-paths. The images were used to assist the localization of STEREOPOLIS. Bentrah et al. [2004] estimated the pose by registering the dense 3D point sets generated by vertical stereo matching, considering the constraints of finding the vertical and horizontal vanishing points of stereo images. Furthermore, the road markings which are detected and reconstructed from areal images, are taken into account as Ground Control Objects (GCOs) for sub-decimeter geo-referencing of MMS in urban areas [Tournaire et al., 2006b].

Then, a more advanced MMS STEREOPOLIS II was developed for high precision 3D city mapping [Paparoditis et al., 2012] (*cf.* Fig 1.2(b)). Compared with the first generation STEREOPOLIS, more types of street data can be acquired by STEREOPOLIS II. First, a precise di-

[1] Institut national de l'information géographique et forestière

(a) STEREOPOLIS (b) STEREOPOLIS II

Figure 1.2: The mobile mapping system in MATIS.

rect geo-referencing system (GNSS/INS/odometer) is used instead of using only GPS. Second, more high resolution cameras are mounted to capture the images of street at different directions. Third, high quality LiDAR is applied to scan the street and generate dense point clouds. The following sections will introduce the main instruments on STEREOPOLIS II.

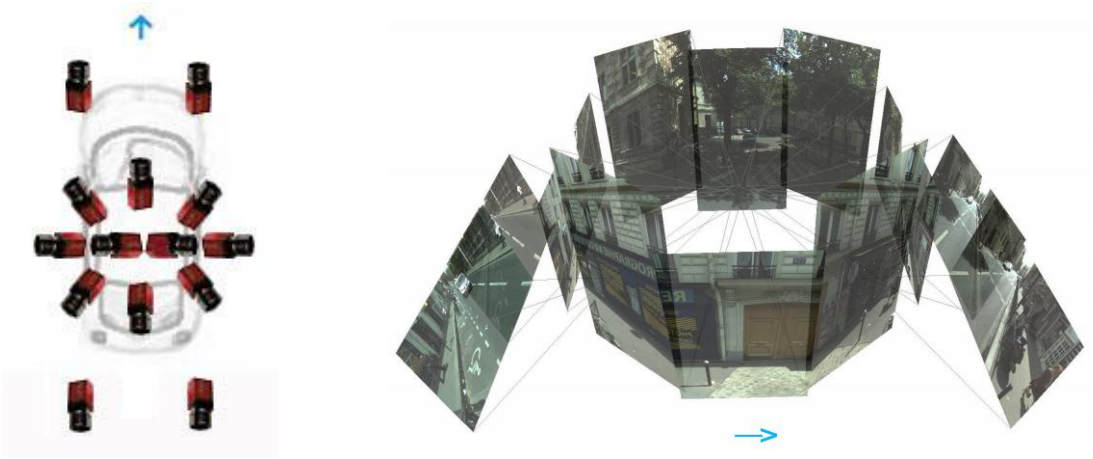
1.2.1 Localization system

An applanix POS-LV220 georeferencing system combining GNSS, an inertial unit and an odometer, is composed for absolute localization of STEREOPOLIS II. Reliable position and orientation are provided directly, even despite GNSS signals being blocked or effected (multi-path) in urban canyons. However, the GNSS masks induce the drifts that can reach one meter for a 2 minutes mask. Continuous high rate (up to 200 Hz) localization can be provided by POS LV system, which is important for accurate geo-referencing of high rate data perception such as point clouds.

1.2.2 Cameras

Multiple high resolution cameras (14 cameras) are mounted on STEREOPOLIS II. Four of those cameras constitute one forward looking stereo rig and one backward looking stereo rig. Ten cameras are mounted at the middle and used to generate panorama images (*cf.* Fig 1.3(a)).

All the cameras are well synchronized and all of them are perspective cameras. The focal length of every camera is $10\ mm$, and the image size is 1920×1024 pixels. The Field Of View (FOV) of the each camera is $70°$ in horizontal and $42°$ in vertical. STEREOPOLIS II captures images every 3 or 4 meters to limit the volume of data. Two strategies: off-line and on-line, were developed to calibrate the rigid parameters to system and intrinsic parameters of each camera [Cannelle et al., 2012]. The first strategy is based on targets network which is built



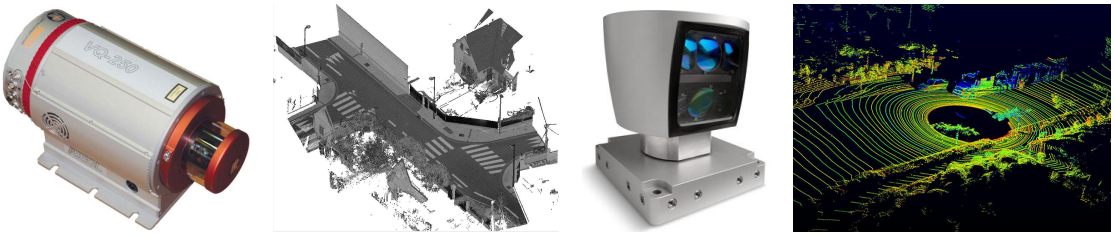
(a) (b)

Figure 1.3: Camera positions and images on STEREOPOLIS. (a) Positions of the 14 camera on STEREOPOLIS. (b) A set of images captured by cameras on STEREOPOLIS. Images in the middle compose the panorama view and two stereo rigs are mounted for forward and backward looking.

and measured preferably in outdoor environment. The second one is a self-calibration approach which is suitable for images captured in urban environment.

1.2.3 LiDAR

At first stage of STEREOPOLIS II, two Riegl scanners were placed at left and right sides of vehicle to scan the building facades and a Velodyne ((*cf.* Fig 1.4(c))) is integrated to acquire the point clouds of bottom side such as the road surface. In the latest version of STEREOPOLIS II a high-performance Riegl was mounted at the rear of vehicle to scan both road surface and building facades (*cf.* Fig 1.4(a)).



(a) (b) (c) (d)

Figure 1.4: LiDAR sensors and point clouds. (a) RIEGL VQ-250. (b) Point clouds acquired by RIEGL VQ-250 on STEREOPOLIS II. (c) VELODYNE HDL-64E. (d) Point clouds of one scan obtained by Velodyne HDL-64E.

High-performance RIEGL A Riegl VQ-250 is placed transversally in order to scan a plane orthogonal to the trajectory. RIEGL VQ-250 works with single laser firing through a rotating mirror. The scanning rate of the laser scanner is up to 100 scans per second. It scans the objects from $1.5m$ to $500m$ with angular resolution up to 0.001° in 360° FOV. High-accuracy ranging is achieved which is better than $10mm$. For every measurement, not only the range and scan angle which are used to compute (x y z) coordinates in sensor space, are captured, but also extra informations for each pulse (e.g. time of emission, echo amplitude) are recorded. The amplitude is dependent on the range and it is corrected into a relative reflectance. In visual representation, the reflectance allows for assigning a brightness for each point with respect to the target reflectivity. The scans are very anisotropic. The scanned points are very dense along scan-lines (a few mm on the road below the sensor), but the density of points along the trajectory depends on the vehicle speed (5cm at a typical acquisition speed of 5m/s = 18km/h).

Velodyne A Velodyne (HDL-64E) was mounted at the top of vehicle to scan dense point clouds of street. Instead of a single laser firing through a rotating mirror, HDL-64E contains 64 lasers which are mounted on upper and lower blocks and the entire unit spins. This design allows for 64 separate lasers to scan thousands of times per second, shown in figure 1.4(d) (the scan of each laser is marked with different color [2]). The unit scans a 360 ° horizontal FOV and a 26.8° vertical FOV. The measuring distance is up to $120m$ and 1.3 million points can be acquired per second.

1.3 Examples of applications

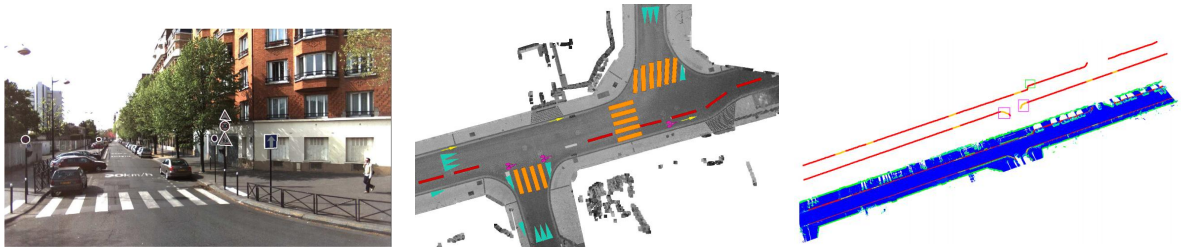
With geo-referenced imagery and dense point clouds, different street features have been extracted and reconstructed in MATIS.

1.3.1 Detection and reconstruction of road feature

Both ground based images and point clouds acquired by STEREOPOLIS are used for road modeling. Soheilian et al. [2010] extracted and reconstructed the road markings (zebra crossings and dashed lines) using stereo images. The method for traffic sign detection and reconstruction from geo-referenced images was thereafter proposed [Soheilian et al., 2013a]. The traffic signs are detected in individual images over sequences, and then they are matched to generate 3D traffic signs (*cf.* Fig 1.5(a)).

The LiDAR mounted on STEREOPOLIS II can provide dense point clouds as well as their reflectance information of urban street. Taking benefit from both dense 3D points and their reflectance, the grayscale road orthophoto and road Digital Terrain Models (DTM) can be gener-

[2] http://velodynelidar.com/hdl-64e.html



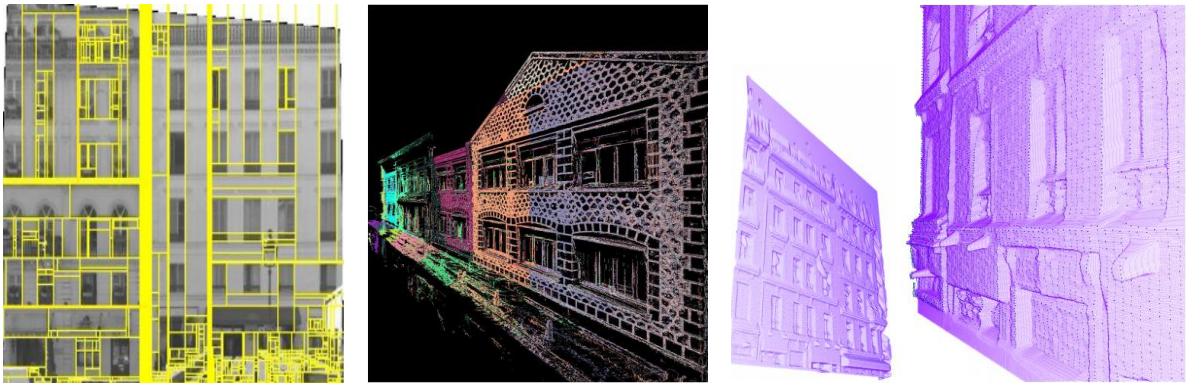
(a) (b) (c)

Figure 1.5: (a) Detecting traffic signs from geo-referenced images [Soheilian et al., 2013a]. (b) Extracting road markings from point clouds [Hervieu et al., 2015]. (c) Road side detection based on point clouds [Hervieu and Soheilian, 2013].

ated from Mobile Laser Scanning (MLS), which can produce higher accuracy and density than aerial products [Vallet and Papelard, 2015]. Based on the high quality grayscale orthophoto, road markings can be extracted from the road orthophoto (*cf.* Fig 1.5(b)). The road markings are searched by minimizing an energy function solving with RJMCMC (Reversible-Jump Markov Chain Monte Carlo) [Hervieu et al., 2015]. Besides, a semi-automatic approach was proposed for curbs and curb ramps recognition and reconstruction using the 3D point clouds and users only need to click some control points (*cf.* Fig 1.5(c)) [Hervieu and Soheilian, 2013].

1.3.2 Reconstruction of building

With the street level images captured by STEREOPOLIS, the building facades are described from a single calibrated street image based on an unsupervised segmentation method (*cf.* Fig 1.6(a)) [Burochin et al., 2009]. In fact, more research about facade reconstruction have been



(a) (b) (c)

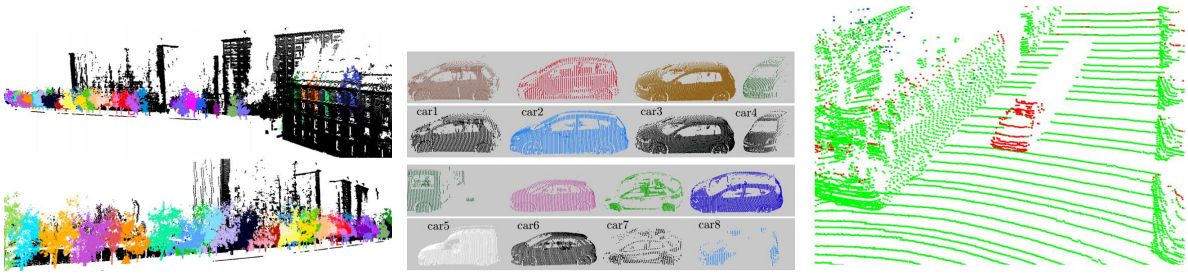
Figure 1.6: Some examples about build facades. (a) Facade segmentation from single image [Burochin et al., 2009]. (b) Build facade reconstruction based on stereo images [Pénard et al., 2005]. (c) Mesh generation using point clouds [Demantké et al., 2013].

conducted since the first generation of STEREOPOLIS. Using the images captured by a vertical

stereo rig, textured meshes of building facades are generated based on the 3D point clouds reconstructed by dense matching (*cf.* Fig 1.6(b)) [Pénard et al., 2005]. In STEREOPOLIS II, the point clouds of building facades are scanned by LiDAR. Hence, Demantké et al. [2013] proposed a framework to estimate the facade surface with a deformable 2.5d grid in a sensor-oriented coordinate system (*cf.* Fig 1.6(c)) while Caraffa et al. [2015] created watertight mesh using point clouds and textured the mesh according to regularized reflectance for each point.

1.3.3 Detection of individual object

Based on dense point clouds, various individual objects are detected in street. Monnier et al. [2012] detected trees from point clouds based on local geometric descriptors for the shape of objects (*cf.* figure 1.7(a)) while Weinmann et al. [2016] proposed a method for individual tree segmentation and localization of individual based on point-wise classification. Xiao et al. [2016] detected street side vehicles and fit them with vehicle model, then the type of the vehicle is recognized for each detected object (*cf.* figure 1.7(b)). Apart from this, moving objects



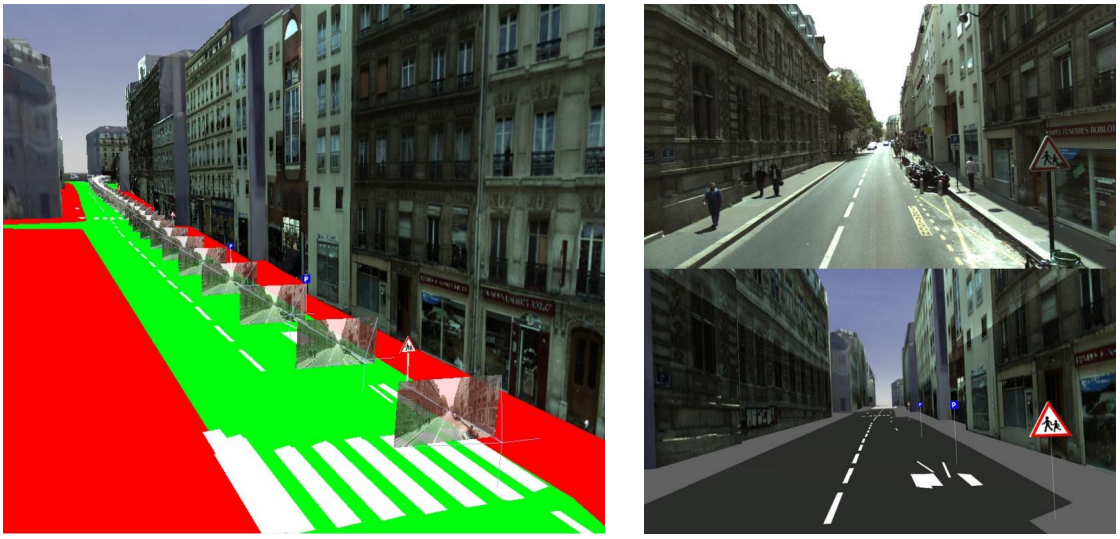
(a) (b) (c)

Figure 1.7: Individual objects detection based on the point clouds acquired by STEREOPOLIS. (a) Tree detection [Monnier et al., 2012]. (b) Vehicle detection [Xiao et al., 2016]. (c) Moving object detection [Vallet et al., 2015]

of simultaneous laser acquisition with Velodyne are detected [Vallet et al., 2015] (*cf.* figure 1.7(c)). Meanwhile, Xiao et al. [2015] studied the change detection of point clouds for the revisited areas. Thus, the static point clouds can be distinguished.

1.3.4 Integrated 3D city model

Combing data from aerial images and ground based MMS, an integrated 3D city model can be generated [Soheilian et al., 2013b]. The 3D surface of roads and buildings are reconstructed using aerial images while high resolution street-view images are used to enhance the texture of 3D build model. Semantic features of street environment (road markings and traffic signs) are generated by means of data acquired by STEREOPOLIS. Moreover, the free space and obstacle areas can be established (*cf.* figure 1.8(a)). These kinds of informations are very important



(a) (b)

Figure 1.8: Integrated 3D model [Soheilian et al., 2013b]. (a) 3D city model contains high resolution textured buildings, road markings, traffic signs and a subset of geo-referenced high resolution images. The free space on road is drawn in green and obstacles are in red. (b) Top: image captured by image. Down: virtual image generated from integrated 3D city model.

for autonomous navigation, which needs to distinguish the permanent obstacles and free space. Meanwhile, the integrated 3D city model with visual landmarks can provide the reference for localization and navigation missions in dense urban areas [Soheilian et al., 2013b].

1.4 The motivation of our research

Precise street based map is desired for many applications. Indeed, the precise MMS like STEREOPOLIS can be used to produce precise map. However, street infrastructure would change regularly due to the factors such as maintenance, reconstruction or city planning. In order to achieve up-to-date maps, large number of MMSs should be equipped for mapping and update. However, it is too expensive to afford many precise MMSs like STEREOPOLIS for one city in practice. A possible solution is to develop low cost but precise MMSs for map updating and change detection. The idea is that each city is mapped with a high cost MMSs like STEREOPOLIS to generate detailed 3D maps at first. Then, low cost MMSs are used to update the maps considering the constraints from the old maps.

The accuracy of maps is highly related to the precision of localization, but the most used combining navigation system is usually very expensive for high precision. Thus, the basic problem for low cost MMS is to develop a cheap but precise localization system. This is the goal of our research. The precise semantic landmarks generated by STEREOPOLIS give us good reference data that can be used to enhance the localization. The uncertainties of the landmarks are taken

into account when we integrate them into the localization process. We provide the six DoF (degree of freedom) poses as well as their uncertainties at the same time. Apart from this, it is well known that GNSS suffers from mask or multi-path problem in urban canyons. This can lead to low accuracy and even outage of the localization. An important purpose of this thesis is to provide a robust solution for the localization in GNSS denied areas.

Our research also provide an alternative solution for localization in applications such as Advanced Driver Assistance Systems (ADAS), self-driving car and Augmented Reality (AR). For those location based applications, the performance is related to the accuracy of localization. In order to achieve higher accuracy, data from multiple sensors are often fused for localization, this would make the system be high cost, heavy and complex. Our method which is proposed based on vision system would be a cost-effective solution. Only low-cost sensors are needed and the system is portable. In particular, the proposed localization system can work robustly in both indoor and outdoor environment that can be used for different scenarios.

1.5 Thesis outline

This thesis is formed with six chapters.

We introduce the state-of-the-art of localization methods in chapter 2. The proposed localization methods are summarized from different phases and then we explore the most relevant research to our research. At last, a pipeline of our localization method is presented.

According to the workflow presented in chapter 2, we introduce vision based localization in chapter 3 which includes the methods about feature extraction, matching, pose estimation and refinement. The detailed equation system is derived for pose estimation and optimization with Local Bundle Adjustment (LBA). The uncertainty propagation of poses are considered over the sequence. We introduce the localization method using monocular and multi-camera system separately. The experiments are presented after the introduction of theory for each case. For multi-camera case, the constraints between the rigid cameras are taken into account.

In order to improve the performance of matching and tracking used in chapter 3, a new approach which is propagation based matching and tracking method, is proposed in chapter 4. A motion model learned over frames, is used to predict the pose of new frame and the uncertainty propagation is considered to guide the tracking of feature over sequence. A comparing experiment is conducted to show the improvement of new matching and tracking method.

Although robust matching and tracking is obtained, the drift of localization is still accumulated over time unless some external data is integrated into the process. To approach this, chapter 5 presents a scheme for landmark integration. It explains how to query geo-referenced landmarks from database and introduces the methods to search the corresponding landmarks in images. At last, a set of constraints are generated for LBA. In this chapter, two types of landmark: traffic sign and road markings are used for experiments and two different strategies for landmark

detection in image are introduced. Our methods are validated using the datasets acquired by STEREOPOLIS II and the results are presented in the experiment section in this chapter. Chapter 6 is a summary of this thesis.

Chapter 2

Related work

Localization of mobile robot has been widely investigated and we summarize the proposed localization methods as shown in figure 2.1.

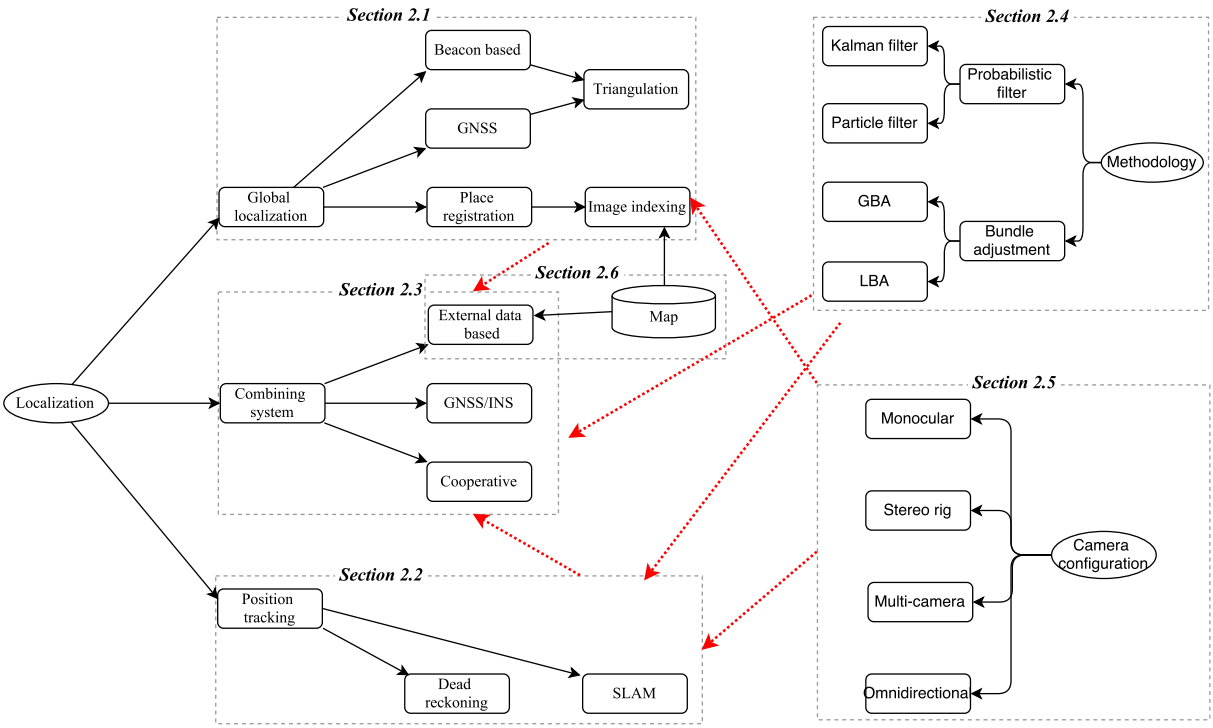


Figure 2.1: Overview of localization.

According to the type of knowledge which is known initially, the localization can be characterized as *global localization* and *position tracking* [Thrun et al., 2005]. For *global localization*, the robot does not have any knowledge about where it is in the environment. The related research about *global localization* will be introduced in section 2.1. The most popular global localization is GNSS. Recently, some alternative solutions for GNSS based localization such as wireless network, are also proposed for the localization in GNSS-denied environment. We call them as *beacon based localization* in this thesis. Meanwhile, some image based methods are

also proposed for localization, that apply the image indexing technique to recognize the image in an absolute visual database.

In the cases of knowing initial pose of a robot in advance, the localization becomes tracking the pose of the robot relative to the initial pose. This kind of methods are *position tracking* [Thrun et al., 2005]. Two types of methods are subsumed for *position tracking* which are *dead-reckoning* and *simultaneous localization and mapping (SLAM)* The relevant content will be presented in section 2.2.

In order to enhance localization performance, the *global localization* and *position tracking* methods are often integrated together, such as the GNSS/INS navigation system. These kinds of methods are introduced in section 2.3 in this thesis (*combining system*). The red dashed arrows in diagram 2.1 stand for the connections between different phases. The *cooperative localization* refers to a robot shares its own positioning information with other robots, thus more measurements are obtained and a set of constraints can be generated for localization. Quite often, the map is also integrated for localization. Different with *place recognition* which only depends on the maps, the maps here are used to generate some constraints for global localization or position tracking.

For most localization methods, the core of the solutions are probabilistic filter or bundle adjustment. These two techniques are explained in section 2.4, and several popular methods are presented (e.g. Extended Kalman Filter, particle filter and local bundle adjustment). These methods are usually applied in *position tracking* and *combining system* to improve the accuracy and robustness of localization. Although many sensors (e.g. IMU, laser scanner, camera etc.) can be used for position tracking, the vision system may be the most cheap and easy to afford. Section 2.5 analyzes the performance of different camera configuration for localization, which provide a reference for the setup of cameras in our research. The goal of our research is to integrate the geo-referenced landmarks for precise localization, the details of the related research about the integration of external data are presented in section 2.6. As last, some conclusions are made according to the previous comparison and analysis of the state-or-the-art methods. Then, a brief introduction of our strategy for localization is presented.

2.1 Global localization

The problem of global localization refers to answering a question "where am I". To estimate the absolute positions, some knowledges should be known in advance, which can be prior positions of the beacons, satellite orbital parameters and maps. The localization operation is established by measuring the relative relations between the entity and the reference points through transmitting signal or visual features.

2.1.1 Global Navigation Satellite System

The most popular way for global localization in outdoor environment is the Global Navigation Satellite System (GNSS). It allows an electronic receiver to determine its location via parsing signals received from satellites. At present, only Global Positioning System (GPS) and GLONASS, can provide operational service. Meanwhile, the Galileo from Europe and BeiDou satellite navigation system from China will be fully operational in coming years.

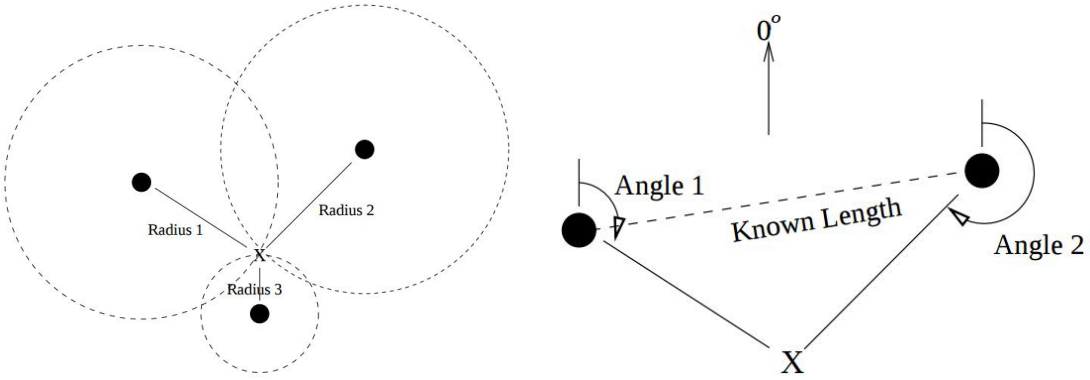
For GNSS based localization, the position of a receiver lies in a sphere surface whose center is the satellite location and radius is the distance from satellite to GNSS receiver. So if the distances can be measured from three or more satellites, the receiver location can be intersected uniquely [Hofmann-Wellenhof et al., 1992]. The orbit data (position) of each satellites are known and can be embedded into the transmitted signals. Thus, the issue is to measure the distance precisely from satellite to receiver. This can be solved through measuring the time-of-flight. The precise time when the signal broadcasts, is recorded by the atomic clock in satellite, while the receiving time of this signal is timed by the clock in receiver, thus the signal transmitting time can be calculated. However, the low cost clock in receiver can not be as precise as the satellite clock. So the measurement of time-of-flight is not precise enough. This can cause inaccurate distance measurement. In order to overcome this problem, a practical solution is to add a correction for each measuring time. In this case, there are four unknowns (three for position and one for time correction). This is the reason why at least four satellites must be observed [Hofmann-Wellenhof et al., 1992] using GPS for positioning. For civilian application, the accuracy of GPS point positioning is in tens meters that can only be used in low accuracy localization [Shaw et al., 2000]. In vehicle navigation, more accurate methods such as RTK (Real Time Kinematic) and DGPS (differential GPS) are used. They correct the mobile GPS units using the information broadcast from a base station or a network of fixed reference stations. These methods provide from centimeter to meter positioning accuracy. However, they increase the cost of localization system.

The GNSS requires the signals being received from at least four satellites directly, but it is difficult in some scenarios. For instance, the GNSS usually suffers from signal obstruction and multi-path because of the street canyon and high trees in dense urban areas [Beck, 1986]. In these situations, the localization cannot be reliable and they even cause the outage of localization.

2.1.2 Localization based on beacons

To localize the entities such as vehicle, robot, smart phone in street canyons or indoor environment, some methods are developed using the signals from wireless local area network (WLAN), Radio Frequency(RF) and bluetooth. For these methods, the absolute positions of beacons are measured beforehand, then the position of the entity can be estimated based on these fixed

beacons. Figure 2.2 illustrates the two different ways for localization based on fixed beacons.



(a) (b)

Figure 2.2: (a) Computing the position of $\boldsymbol{X}$ by measuring distances from $\boldsymbol{X}$ to at least three beacons, (b) Localization based on the measurement of angles and distance [Hightower and Borriello, 2001]; the distance between two stations is known, the angles can be obtained by measuring the phase shift of signal.

In figure 2.2(a), the position of $\boldsymbol{X}$ is computed by measuring the distance from three non-collinear beacons to $\boldsymbol{X}$, which is the intersection of the three circles. For the 3D position of $\boldsymbol{X}$, it becomes the intersection of spheres. In general, it needs at least four spheres to determine an unique 3D position. However, if all the beacons are above $\boldsymbol{X}$ or below $\boldsymbol{X}$, the 3D position can be solved by three spheres [Hightower and Borriello, 2001]. That is why the absolute 3D position on earth can be measured using three satellites if the distance can be measured precisely.

To determine the position of $\boldsymbol{X}$, the key technique is to measure the distance from beacons to entity precisely. The radio based solution is often used based on *time-of-flight* of the radio. A typical application is the GNSS as we presented in previous section. In indoor environment, each beacon can generate messages with single identifier and then the location of entity can be estimated by collecting messages from more than three beacons [Priyantha et al., 2000; Harter et al., 2002; Bahl and Padmanabhan, 2000].

Figure 2.2(b) shows another way to position entity by measuring two angles and one distance between two beacons in 2D. In 3D space, the azimuth should be defined firstly, then the 3D position can be computed using the two angle measurements and one distance measurement [Hightower and Borriello, 2001]. One classical application is the Very High Frequency(VHF) Omni-directional Ranging (VOR) aircraft navigation system [Kayton and Fried, 1997]. A VOR ground station sends out a master signal including station's identity, and a highly directional signal which rotates clockwise in space with a time stamp. The angle from station to aircraft can be computed by measuring the phase shift. If another signal is received, then the position of aircraft can be computed.

According to the type of signal, the beacon based localization can be summarized as infrared

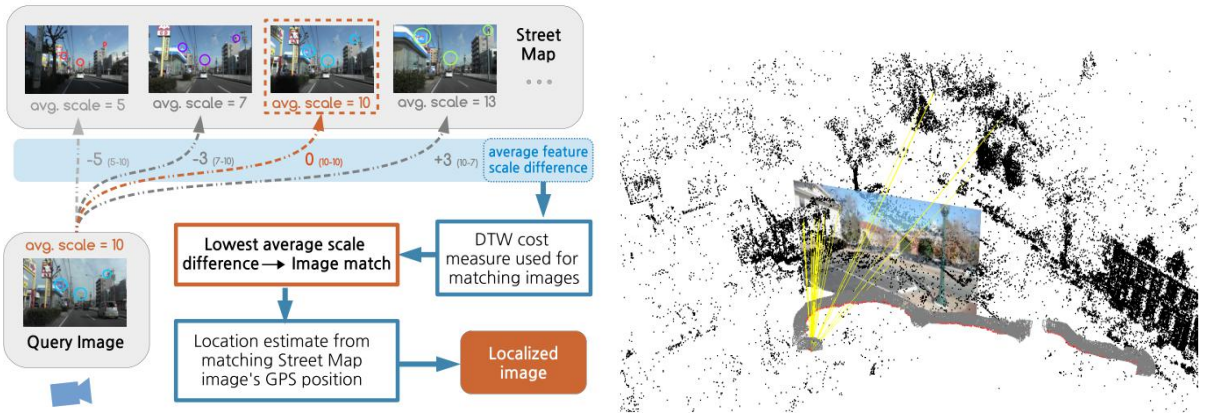
positioning system, ultrasonic positioning system and Radio-Frequency (RF) based system [Koyuncu and Yang, 2010]. Different types of signal are used to generate the unique code identifier transmitting from the beacons to receivers. The absolute position of entity is still computed with the triangulation principles introduced in previous paragraph. Koyuncu and Yang [2010] and Bessho et al. [2009] evaluated the performance of beacon based indoor localization methods and the accuracy of those methods vary from centimeters to meters for different system. The downsides are the short measuring range and signal blockage or obstruction. In additional, it would be expensive to build up the beacon network in a large and complex environment.

2.1.3 Place registration

We can also answer the question "where am I ?" by means of maps. The pose can be estimated relative to a pre-built map. We call it as *place registration*. In fact, the GNSS and beacon based localization can be regarded as *place registration* as well. The orbit parameters and the beacon networks are special maps.

In order to estimate the instant poses, the robot needs to collect the information from a mobile platform and then registers with the map. In robotic and computer vision, camera is quite often used due to the rich visual features in images. The localization of robot relaid on the recognizing results in the map produced beforehand. There are two types of maps: metric and topological. Metric maps maintain accurate information about environment details (e.g. distances, measures or sizes), which are usually referenced in a global coordinate system [Garcia-Fidalgo and Ortiz, 2015]. However, topological maps represent the environment by means of a graph, where nodes represent distinctive places and arcs model the relations. In this case, the maps are simple and compact. Many papers have proposed the localization based on topological map, but only rough position of the images can be determined using the topological maps [Ulrich and Nourbakhsh, 2000; Wu et al., 2009]. Our goal is to estimate the precise positions, thus metric maps should be used for place recognition.

The key technique for place registration is to retrieve the images in maps (topological or metric). According to the describing method for environment, the methods can be classified as: global descriptors based method (e.g. histograms, line segments, frequency analysis, etc.), local features based method (e.g. SIFT, SURF, etc.) and Bag-Of-Words (BoW) based method [Garcia-Fidalgo and Ortiz, 2015]. To achieve the absolute localization, the images captured by on-board cameras are matched with geo-referenced data. Lindsten et al. [2010] proposed a vision based localization system for UAV, making use of environmental classification and rotation invariant template registration. Furthermore, Wan et al. [2016] proposed a more robust and efficient way that uses phase correlation for image localization. These methods can determine positions of images based on metric map with global features. Wong et al. [2014] estimated the pose by registering an image with the geo-referenced image sets using some local features such as SURF descriptor (*cf.* Fig 2.3(a)). Moreover, combining the local image features, the visual



(a) (b)

Figure 2.3: (a) Localization based on visual street map [Wong et al., 2014]. (b) Localization based on vocabulary tree using 3D data reconstructed with SFM [Irschara et al., 2009]

words are applied to recognize the location of an image from a visual point clouds database generated by Structure from Motion(SFM) (cf. Fig 2.3(b)) [Irschara et al., 2009; Sattler et al., 2011]. Recently, Kendall et al. [2015] applied *deep learning* to estimate the most approximate image pose from database reconstructed using SFM.

2.2 Position tracking

If the initial pose of robot is known, the localization can be achieved by tracking the robot relative to the initial pose. This is a problem of *position tracking* [Thrun et al., 2005]. In contrast to global localization, *position tracking* is a localization in local space. Two most popular relative localization solutions are Dead-reckoning and SLAM. The dead-reckoning estimate the pose with aid of measured values such as velocity, acceleration and speed, while the SLAM trends to localize the robot and build the map of environment at the same time. The SLAM provides the poses relative to map.

2.2.1 Dead reckoning

Assuming a vehicle moves straight, the position of the vehicle relative to the start point can be calculated with a wheel encoder which counts the rotary number of the wheel. A more precise way is to measure the speed with the speedometer or acceleration with accelerometer over time. Then the displacement can be calculated. However, this is the ideal situation, the moving direction would change from time to time. In practice, the angular velocities are measured with the units like gyroscopes and compass. Together with the information such as location, speed and acceleration, the current localization can be estimated using kinematics equations.

One compact dead reckoning solution is inertial navigation system (INS) that integrate com-

puter, motion and rotation sensors together. The key component in INS is the inertial measurement unit (IMU) that contains linear accelerometers and gyroscopes to measure the moving accelerations and angular accelerations [Eshbach et al., 1990]. To calculate the position and orientation in 3D space, an IMU needs at least three accelerometers for motion (X, Y, Z) and three gyroscopes for rotation (pitch, yaw and roll) measurements. The computer is employed to resolve the 6 DoF motion combining measurements from accelerometers and gyroscopes. There are two main steps. First, it computes the current velocity using the acceleration values over time. Then, the current position and orientation are estimated using kinematics equations considering the velocity and time interval.

Unfortunately, the dead reckoning is prone to error cumulation over time which is caused by the errors of the measurements of displacement and rotation. The first kind of errors affect the accuracy of position directly while the second produces the errors of orientation immediately and influences the precision of position indirectly. Because the estimates of position are conducted by the rotation, thereby the errors of rotation are propagated to the position. Moreover, current position estimated using dead-reckoning is always related to the previous one. Thus, the errors are accumulated and the drift of localization grows over time.

2.2.2 SLAM

SLAM aims at generating the map of environment and simultaneously localizing the robot relative to the map. Different sensors are usually used for SLAM such as cameras and LiDAR. Depending on the types of data perception, SLAM can be specified vision based SLAM, ranging system based SLAM, RGB-D based SLAM and hybrid system based SLAM. Table 2.1 shows some sensors that are often used for SLAM. According to the way for data perception, the sensors can be categorized as active and passive.

Table 2.1: Different types of ranging and optical sensors.

active			passive		
					
LiDAR	RGB-D	Radar	Camera	Stereo	Omnidirectional

2.2.2.1 Vision based SLAM

The vision based SLAM refers to SLAM associated by vision system. The general SLAM computes joint posteriors over the path with full covariance [Davison, 2003], so the computation

increase quickly with the growing of images. An alternative solution which is called *visual odometry (VO)* [Nister et al., 2004], aims at estimating the egomotion of an agent (e.g. vehicle, robot, human). The VO recovers the path of agent incrementally and optimizes only last N frames in local [Scaramuzza and Fraundorfer, 2011]. The VO is often used for large scale localization.

For most of the existing vision based SLAM approaches, the procedures can be summarized as four steps: **(1) Image acquisition**. Various camera configurations are used such as monocular [Nister et al., 2004; Davison, 2003; Mouragnon et al., 2006], stereo [Se et al., 2002; Olson et al., 2000] and multi-camera rigs[Kneip et al., 2013]. **(2) Feature extraction**. Most of the SLAM methods detect the salient points (interest points) and track them through sequential images for pose estimation. To detect interest points, the detectors such as Harris[Harris and Stephens, 1988], Shi and Tomasi detector[Shi and Tomasi, 1994], scale-invariant feature transform (SIFT)[Lowe, 2004], Speeded up robust features (SURF) [Bay et al., 2006], Features from Accelerated Segment Test (FAST) [Rosten and Drummond, 2006], are usually applied. Furthermore, some methods searching the corresponding points according to the feature descriptors, the methods like SIFT, SURF, Binary Robust Independent Elementary Features(BRIEF)[Calonder et al., 2010], etc are often used for feature description. **(3) Matching and tracking**. The correspondences between images can be established via feature based matching (e.g. SIFT, SURF, etc.) or area based matching such as normalized cross correlation(NCC) [Nister et al., 2004] and phase correlation [Barnada et al., 2015]. Then the tracks over sequential images can be fund in these pair-wise matches. **(4) Pose estimation and optimization**. There are two main resolutions for this step, the filter based methods(e.g. Extend Kalman filter (EKF) [Davison, 2003; Durrant-Whyte and Bailey, 2006], particle filter [Montemerlo et al., 2002; Sim et al., 2005]), and the solutions inspired by structure from motion [Nister et al., 2004; Mouragnon et al., 2006; Eudes et al., 2010].

Apart from this, some special vision systems such as infrared camera [Wang and Chen, 2010] and multispectral system [Mouats et al., 2015] are used which can overcome the difficulties for general cameras in poor lighting conditions like the night-time localization.

The cameras used for visual odometry are often calibrated beforehand to acquire the intrinsic parameters of the cameras. Nevertheless, the visual odometry has the similar problem with dead-reckoning, that is the drift growing over time because of the errors of the interest points localization, matching and the camera calibration. All these errors are accumulated unless some drift-free constraints are taken into account.

2.2.2.2 Ranging system based SLAM

The ranging sensors can measure the real distance to objects which is widely used in obstacles detection and avoidance. With the reducing of price, size and weight, they are becoming popular for the autonomous navigation [Zhang and Singh, 2014]. One popular sensor is LiDAR

which can be categorized as 2D LiDAR and 3D LiDAR. The 2D LiDAR contains only one scan plane while the 3D LiDAR usually contains multiple layers and allow continuous scanning of environment.

Lu and Milios [1997b] proposed pose estimation with single 2D laser scanner by aligning the scans over time. To improve the accuracy, several 2D laser scanners are combined for localization [Zhang and Singh, 2014; Vosselman, 2014]. Meanwhile, most LiDAR based applications are usually using 3D laser scanner [Nüchter et al., 2007; Moosmann and Fraichard, 2010; Moosmann and Stiller, 2011]. To estimate the pose, the Iterative Closest Points(ICP) algorithm [Lu and Milios, 1997a] is employed to register scans and estimate the motion for 2D or 3D LiDAR. More precise SLAM can be conducted using pyramid grid-map which does not need to establish correspondence between feature and landmark [Xie et al., 2010]. The measurements acquired by LiDAR can be very precise, but the points on moving objects can influence the accuracy of pose estimation. We always want to register the static points. To overcome this problem, the moving object detection methods were proposed to select the static scene for motion estimation [Miyasaka et al., 2009; Schlichting and Brenner, 2016].

The sonar that measures the range via ultrasonic wave, is also used for localization [Tardós et al., 2002; Ribas et al., 2008]. The principle about localization using this kind of sensor is similar with LiDAR, but they can be used in some special environment such as underwater. Radar is another popular ranging sensor which is usually used for collision avoidance, target detection in ADAS. It is rarely used for localization because of the noisy points compared with LiDAR. Recently the radar based localization was developed according to signal clustering and particle filter [Schuster et al., 2016].

Compared with vision systems, ranging sensors are active. They can measure the 3D geometric information directly and rarely influenced by some factors such as whether and illumination change. However, the ranging signal may be disturbed by environmental noise and the ranging sensors like LiDAR are usually expensive.

2.2.2.3 RGB-D based SLAM

In recent years, a new type sensor that is RGB-D camera is more and more popular for SLAM or visual odometry. It refers to a camera that can capture 2D color image and record the depth information relative to camera like Microsoft Kinect. Here, "RGB" refers to normal color image and "D" means the depth information for every pixel in image. Thus, RGB-D sensors can get benefit from both vision and ranging sensors.

The advantage of RGB-D based localization is that the motion of moving agent can be estimated by registering the dense depth structure using ICP algorithms mentioned in previous section, then the estimated motion is refined by minimizing the back-projection error in image space [Steinbrücker et al., 2011; Endres et al., 2012]. It is able to work in the cases where very few feature points are detected in image space. The images and depth information can enhance the

localization. For instance, an initial pose can be estimated using visual pose estimation to guide 3D depth structure registration if rich features are detected. This can improve both robustness and efficiency [Heredia et al., 2015]. The RGB-D camera typically generate voluminous data. Biswas and Veloso [2012] addressed this challenge task by sampling the depth map to planes. Thus, the depth structure can be expressed with several plane parameters, so the volume of depth data was reduced significantly.

The RGB-D camera can also be used for dense reconstruction of indoor or outdoor scene [Henry et al., 2012; Steinbrucker et al., 2013]. Taking benefiting from both texture and depth information, the scene segmentation [Holz et al., 2011] and obstacle detection [Santos et al., 2015] can be accomplished at the same time.

2.2.2.4 Hybrid system based SLAM

To achieve robust and continuous pose computation, the data from two or more aforementioned sensors are usually fused for localization. In this thesis, we summarize the proposed methods as following categories.

Vision system + IMU The challenges of visual odometry are the drift due to error propagation and ill-conditioned pose estimation which is caused by insufficient correspondences or bad distribution of the correspondences. As we introduced in section 2.2.1, the relative poses can be obtained using IMU. Although we can't remove the drift, the growing rate of drift can be reduced promisingly with the consideration of IMU data. The IMU data is usually integrated into pose estimation scheme by means of Kalman filter [Armesto et al., 2007], the least squares [Lategahn et al., 2013] or RANSAC [Kneip et al., 2011a]. Moreover, the matching and tracking can be guided using the relative poses from IMU [Roumeliotis et al., 2002], which could improve the matching precision and efficiency.

LiDAR + IMU To assist the motion estimation using IMU, the lines are detected from the point clouds acquired by LiDAR to make constraints by aligning the line segments [Zhao and Farrell, 2013]. A more tightly solution was to integrate the two different types of data considering their uncertainties [Hesch et al., 2010; Li et al., 2014].

Vision system + LiDAR Combing vision system with LiDAR which is called visual-LiDAR [Zhang and Singh, 2015], has the similar feature with RGB-D camera, but the 3D points are not as voluminous as depth map. However, these points are more precise. A general solution for this case is to estimate the frame to frame pose using images and to determine the scale of translation and refine the pose by registering the scans acquired by LiDAR [Zhang and Singh, 2015; Balazadegan Sarvrood et al., 2016]. A more precise solution is to fuse the image and LiDAR data using integrated bundle adjustment to obtain optimal trajectory for moving platform [Liebold and Maas, 2014].

However, the hybrid systems are costly, heavy and consume more electrical energy. With the

joining of new sensors, the expense, weight and power is definitely increased. In practice, a trade-off between cost and accuracy should be found.

2.3 Combing system

In order to improve the performance of localization, some methods are proposed to combine the data from different sources for localization.

2.3.1 GNSS/IMU system

GNSS can provide absolute positions, but 6 DoF poses (position and orientation) are needed for navigation in most of the applications. In practice, IMU is often used to measure the orientation and it is often combined with GNSS tightly to provide absolute pose directly.

In general, the GNSS can provide drift-free positioning results with 1 *HZ* sampling rate [Farrell, 2008] while IMU has higher rate (200-1000 Hz) [Mostafa and Hutton, 2001]. Therefore, it is natural to combine them together to provide smoother poses. On the one hand, the IMU can fill in the gaps between two GNSS sampling points to provide higher rate localization, on the other hand, the drift caused by IMU error accumulation can be compensated with the absolute measurements from GNSS positioning [Abuhadrous et al., 2003]. The data from GNSS and IMU fusion is a nonlinear filtering problem, which is commonly solved using the Kalman filter (KF) [Wong et al., 1988]. Recently, both GPS and GLONASS are integrated with IMU to reduce the effect of multi-path problem [Angrisano, 2010].

The accuracy of localization is related to the precision of GNSS measurements and the quality of IMU. It can achieve the horizontal and vertical accuracy of positioning in meters for current state of the art commercial products (POS-LV [1], SPAN[2]) without post-processing. The positioning accuracy can reach few centimeters after post processing using the data from ground-based reference stations. The combined GNSS/IMU system can overcome the multi-path or mask in a short time ($< 60s$), that the locations can be measured depending on IMU dead reckoning. But the system would be outage for long period multi-path or signal blockage of GNSS. Furthermore, the accuracy of localization is affected by the quality of IMU, but a precise IMU is too expensive to be widely afforded .

2.3.2 Cooperative Localization

In cooperative localization, a group of entities (robots or vehicles) are viewed as a system for localization. The entities can communicate with each other. The task of cooperative localization

[1] http://www.applanix.com/products/poslv.htm

[2] http://www.novatel.com/products/span-gnss-inertial-systems/

is to incorporate relative sensor measurements into a Kalman filter framework to estimate the poses [Roumeliotis and Bekey, 2000]. It has been investigated a lot in robotic applications that require robots work in collaboration to perform a certain task. Lots of solutions have been proposed [Wang et al., 2008; Tully et al., 2010; Kia et al., 2015], EKF or particle filter is often used to cope with the relative measurements. The covariance intersection filter which yields consistent estimates for fusing both sensor data of the ego-vehicle and the estimates sent from other vehicles [Li and Nashashibi, 2013].

The large scale applications often desire absolute positions. GNSS based methods are used usually, but they may suffer multi-path or signal-denied problems in urban canyon. The cooperative localization provides a solution to obtain precise positions. Each entity can measure its individual positions and share them with others [Karam et al., 2006], then the measurements from all the entities are integrated together considering their uncertainty for localization. The sub-meter localization can be achieved in comparison to meters or more using only one GNSS receiver [Ekambaram and Ramchandran, 2010; Goodliss et al., 2011]. In particular, this strategy can reduce the impact of multi-path benefiting from the increasing of redundancy for localization. Meanwhile, the relative measurements captured by in-vehicle sensors such as radar can also be shared with nearby vehicles, as well as the absolute positions obtained by GPS [Fujii et al., 2011]. To enhance the localization accuracy, a more tightly cooperative localization that shares GNSS pseudo-range corrections in vehicles to reduce the biases of pseudo-ranges measurements from satellites [Lassoued et al., 2016]. Besides, the road maps can also be used, together with the knowledge about the position of surrounding vehicles, to infer where is the ego vehicle [Svensson and Sörstedt, 2016].

2.3.3 Maps constrained localization

The *place registration* only relies on the map, but the *maps constrained localization* means that the data from maps are integrated with other sensors to enhance the localization. GNSS based methods are the basic solution for localization, but the accuracy of localization is related to the status of satellite signals. Inadequate accuracy results would be provided in dense urban areas due to frequent masks and multi-path. In order to enhance the localization, the map (external data) is usually integrated. An important technique which is *map matching*, deals with the GNSS based localization errors using the reference of spatial road network [Quddus et al., 2007; Brakatsoulas et al., 2005]. The general purpose of a *map matching* algorithm is to identify the correct road segment on which the vehicle is moving and to determine the position on the segment [Greenfeld, 2002; Li et al., 2010], thus, both physical location of the vehicle and accuracy of position coordinates can be improved if the spatial map has high precision. In the other words, the improvement in term of accuracy is limited by the quality and Level of Detail (LoD) of the maps [Grush, 2008]. More detailed maps such as the 3D urban models, are used for localization to overcome the problem in urban area [Drevelle and Bonnifait, 2011; Betaille et al., 2012]. Getting benefits from precise 3D maps, few centimeters accuracy can be obtained

reliably with GPS/IMU systems for localization, specifically in urban [Levinson et al., 2007]. More details about map based localization will be presented in section 2.6.

2.4 Methodologies for localization

For global localization, the GNSS and beacon based methods use triangulation to estimate the position while the core of place registration is image indexing. In this section, we focus on the key solutions of position tracking. Depending on whether the feature correspondences are specified in 2D or 3D, the pose estimation schemes can be classified into three different types [Scaramuzza and Fraundorfer, 2011]:

- **2D-2D** This kind of methods are often used for vision based localization. They estimate the relative pose between two views using a set of corresponding image point obtained by image matching. For calibrated images, the pose can be decomposed from *essential matrix*, noted as $\boldsymbol{E}$, and $E = [t]_{\times}R$ [Hartley and Zisserman, 2003]. But the scale of translation $[t]_{\times}$ is unknown, which need to be estimated independently.

- **3D-3D** For stereo vision based approach [Milella and Siegwart, 2006] or LiDAR based localization [Nüchter et al., 2007]. The pose from frame to frame can be expressed by rigid transformation which can be estimated by registering the point clouds generated at current time with the point clouds generated before. The Iterative Closest Points(ICP) algorithm [Besl and McKay, 1992] is usually used to solve the rigid parameters.

- **3D-2D** The projection from world (3D) to image (2D) can be expressed using a matrix $\mathbf{P}_{3\times4}$ which is a 3×4 homogeneous matrix [Hartley and Zisserman, 2003]. It contains both intrinsic and extrinsic parameters of the camera. If three or more 3D-2D correspondences are obtained, the $\mathbf{P}_{3\times4}$ can be resolved. For calibrated camera, the pose estimation is a problem called Perspective-n-Point (*PnP*) for perspective camera [Horaud et al., 1989; Quan and Lan, 1999].

In a more general manner, the methodologies of localization are divided into three groups: (1) based on probabilistic filters (e.g. EKF, particle, etc.), which are classical solutions and the system maintains a probabilistic representation of both the pose of the robot and the landmarks in the environment, (2) based on incremental Structure From Motion (SFM), and (3) the methods inspired by biology [Fuentes-Pacheco et al., 2015]. In this thesis, we focus on the methods based on probabilistic filters and SFM. In particular, we investigate the methods according to the core of each solution, which are probabilistic filters and Bundle Adjustment (BA).

We note P as pose, X as landmark and m as measurement. The problem of SLAM can be modeled as Bayesian network graph (*cf.* Fig 2.4(a)). The landmarks (map) and poses are linked by the measurements. Representing the Bayesian network in a Markov Random Field (MRF)

without showing the measurements explicitly, the SLAM problem involves finding the maximum likelihood solution of graph in figure 2.4(b) [Strasdat et al., 2010a]. The graph represents a *full SLAM* problem [Thrun et al., 2005]. Each edge stated in figure 2.4(b) is a constraint. Optimizing all the constraints in graph yields the optimal solution for both poses and landmarks. This usually forms a nonlinear least squares problem and the method is called *GraphSLAM*. However, it is clear that the graph for full SLAM grows at every time step, so that the computational cost increases quickly and is out of hand for large scale workspace. In order to improve the efficiency, one strategy only involves the estimation of momentary pose along with map, called *Online SLAM* [Thrun et al., 2005]. It marginalizes all the historic poses and retains all the landmarks (*cf.* Fig 2.4(c)). New links between landmarks are built with the elimination of poses, but this makes the graph become fully inter-connected. Another solution is key frame bundle adjustment (BA). In this case, a subset of poses (keyframes) are selected for SLAM (*cf.* Fig 2.4(d)).

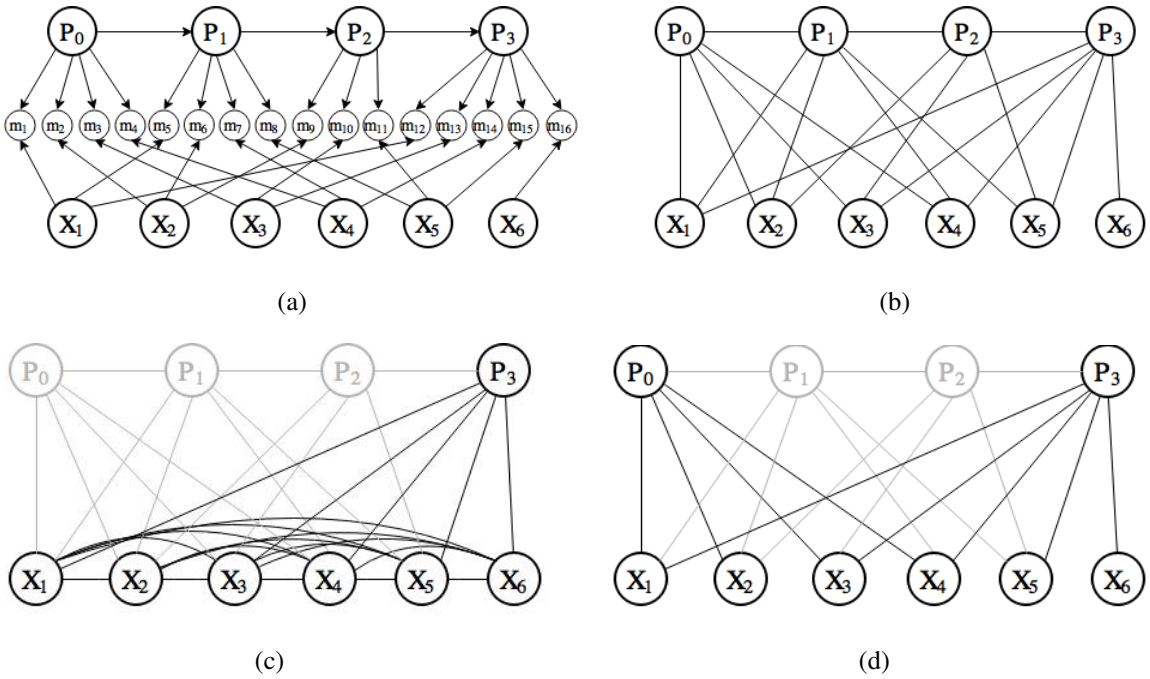


Figure 2.4: (a) SLAM problem depicted as Bayesian network graph. (b) The graph for *full SLAM*problem. SLAM as Markov Random Field without representing the measurements explicitly. (c) Online SLAM. (d) Key-frame based BA [Strasdat et al., 2010a]

According to the core of the solution for SLAM, the proposed methods can be divided into probabilistic filter based approaches and BA based approaches.

2.4.1 Probabilistic filter

The filtering problem consists of estimating the state of a dynamical system from partial and noisy observations with respect to conditional probability [Jazwinski, 1970]. In localization

process, the filter based algorithms often have two main steps: predicting and updating [Thrun et al., 2005]. For prediction stage, the pose of current time step is predicted with the condition of previous state and the control data reflected the changes of state. In updating procedure, the predicted state is corrected with association of the measurements acquired by the sensors. The knowledge of the state is represented through conditional probability distribution and the most generic algorithm to calculate is to use Bayes filter [Thrun et al., 2005]. The Bayes filter is principal algorithm which is recursive and difficult to implement in particle. Therefore, many approximated algorithms are derived from Bayes filter.

2.4.1.1 Kalman Filter

One most studied implementation of Bayes filter is Kalman Filter, invented at 1950s for linear problem. However, the model of SLAM is nonlinear. For instance, the constraint between pose and landmarks (edges between P and X in figure 2.4(b)) for vision based SLAM is usually modeled by perspective projection which is a nonlinear constraint. To resolve this problem, Extended Kalman Filter (EKF) was proposed for nonlinear model. SLAM with EKF is a standard way for *Online SLAM* [Thrun et al., 2005]. EKF is an approximate solution where the nonlinear model is linearized via first order Taylor expansion. The EKF assumes that the noises underlie a Gaussian distribution. It can estimate optimal mean and covariance. EKF is the most widely used method, but a more robust and more accurate method is Unscented Kalman filter (UKF), which applies unscented transform for liberalization [Julier and Uhlmann, 2004]. As stated in figure 2.4(c), the graph is quickly full inter-connected with growing of images. After some time, the covariance matrix becomes fully correlated. Regarding the quadratic update time of EKF, it is difficult for large amounts of landmarks.

2.4.1.2 Particle Filter

It is well known that KF can only deal with Gaussian model, but this is not always the case for the noise distribution in practice. In order to cope with the arbitrary distribution model, a Monte Carlo based technique is used to approximate the target distribution using multiple samples (particles) [Dellaert et al., 1999]. This family of methods are called *particle filter* [Gordon et al., 1993; Doucet and Johansen, 2009]. With the increasing of particles number toward infinity, the sampling converges to the actual situation. The estimates are computed in each particle individually using a proposal distribution in contrast to EKF which uses a single Gaussian for pose and landmarks. The SLAM with particle filter is *FastSLAM*, which computes the pose and related landmarks in a separate particle and maintains the full path posteriors [Montemerlo et al., 2002; Thrun et al., 2005]. In fact, FastSLAM is the algorithm that fit both *Online SLAM* and *full SLAM*. Because it computes one pose every time so it is *Online SLAM*, but it calculates the full path posteriors, thus it is *full SLAM* at the same time. The particle filter has been used in many applications [Montemerlo et al., 2002; Törnqvist et al., 2009; Ji et al.,

2015].

Compared with EKF, particle filter performs better in large number of gross errors cases [Ji and Yuan, 2016]. It means that particle filter is more robust than EKF. The precision of SLAM with particle filter is related to the number of particles. The more particles are sampled, the higher precision can be obtained, but the computational cost is increased with the growing of particles. One advantage of EKF is that it can achieve higher accuracy than particle filter according to the comparing experiments in the literatures [Ben-Afia et al., 2014; Ji and Yuan, 2016].

2.4.2 Bundle adjustment

As we introduced in previous section, the conventional solution for *full SLAM* problem is *GraphSLAM* which is based on nonlinear least squares. Let's observe graph of *full SLAM* problem in figure 2.4(b). Actually, the *full SLAM* problem can be solved by *bundle adjustment*, considering the constraints between successive poses. If there are no constraints, the SLAM problem becomes classic bundle adjustment similar as the techniques in key-frame BA (*cf.* Fig 2.4(d)). Compared with *Online SLAM*, BA based approach contains more elements in the graph since it retains the historic poses.

Bundle adjustment is a technique that aims to optimize the parameters from both structure and motion of images [Triggs et al., 2000]. It has been widely investigated in the field of photogrammetry [McGlone et al., 2004; Triggs et al., 2000]. For vision based SLAM, the unknowns are the images poses and the 3D objects points, the measurements are the interest points corresponding to the 3D object points. One way for BA is to accumulate the information into the graph and resolve the poses and landmarks of full path finally. This is the classic bundle adjustment which is off-line processing. We call it *global bundle adjustment* in this thesis. In some applications, the BA is conducted in a sliding window of most recent poses and related landmarks, which optimize the poses and local maps immediately. We call this type of BA as Local Bundle Adjustment (LBA).

2.4.2.1 Global bundle adjustment

The Global Bundle Adjustment (GBA) refers to optimize all the parameters in graph associated with all the measurements by minimizing the squared sum of residuals. This style of bundle adjustment are often used for off-line SFM with large number of unordered images [Snavely et al., 2006; Moulon et al., 2012; Wu, 2013]. Thousands of images can be processed at the same time, but it is not the case for real-time localization because of the high computing cost. The complexity of computation for GBA is $O(n^3)$ with respect to the number of parameters, growing with the number of images [Triggs et al., 2000; Engels et al., 2006]. For SFM using successive images, only key-frames [Thormählen et al., 2004; Klein and Murray, 2007] are selected from sequential images. This can reduce the number of images involved in GBA, but

the number of parameters is still increased over time.

2.4.2.2 Local bundle adjustment

The Local Bundle Adjustment (LBA) is a strategy to improve the efficiency of bundle adjustment by limiting the number of images involved in bundle adjustment each time step. The idea is that only limited number of images is considered each time. In hierarchical SFM, the whole image set is divided into several hierarchies, bundle adjustment is performed in each hierarchy to generate some local maps, then the local maps are merged to produce a global map [Zhang and Shan, 2001; Farenzena et al., 2009]. More research about LBA is about real-time localization or reconstruction. Each processing step, only fixed number of the latest images are optimized with bundle adjustment, as well as the related 3D points. Then move the processing window over sequence [Mouragnon et al., 2006; Eudes et al., 2010; Persson et al., 2015].

Compared with SLAM with EKF or particle filter, BA based approaches can achieve better accuracy for localization [Strasdat et al., 2010a]. To improve the efficiency of BA for localization, the LBA was proposed, Whereafter, Eudes and Lhuillier [2009] studied the error propagation of poses from step to step and estimated the uncertainty for every frame. The experiments indicated that uncertainty of image poses increase over time.

2.5 Camera configuration for localization

Our research aims at developing a low-cost localization system. Many sensors can be used, but some of them have limitations (e.g. GNSS, beacon based) and some of them are too expensive (LiDAR, IMU). Thus, it is sensible to investigate the vision systems, which have been widely used for localization. In the last decades, various camera configurations were proposed for localization in robotic and computer vision. We summarize them as monocular camera, stereo rig and omnidirectional camera and multi-camera system.

2.5.1 Monocular camera

Davison [2003] was the first to develop a capable Extended Kalman Filter (EKF) based SLAM using a monocular camera. Sparse interest points are detected and tracked along the sequential images to reconstruct the 3D landmark map and estimate the pose of the moving camera. But Davison's methods can only work in a small environment due to the quickly increasing of computational complexity with the growing number of landmarks [Fuentes-Pacheco et al., 2015]. In the next stage, more efficient solutions for monocular camera based localization were proposed. Klein and Murray [2007] processed the tracking and mapping procedures into two parallel threads separately for SLAM . Meanwhile, lots of methods improved the efficiency only

considering the key frames in sequence [Mouragnon et al., 2006; Strasdat et al., 2010b; Engel et al., 2014], thus, the number of images are reduced significantly

The advantage of monocular camera based solutions is that they are low-cost. But one common problem for monocular visual odometry or SLAM is the scale. It is hard to get the metric results, unless some prior knowledge about the scale factor is set or other sensors such as GPS, odometors, are integrated.

2.5.2 Stereo cameras

The stereo cameras might be the most used for vision based localization. The intrinsic parameters and extrinsic parameters of the two cameras are often calibrated beforehand, so that the metric scale can be determined according to baseline between two cameras in stereo rig.

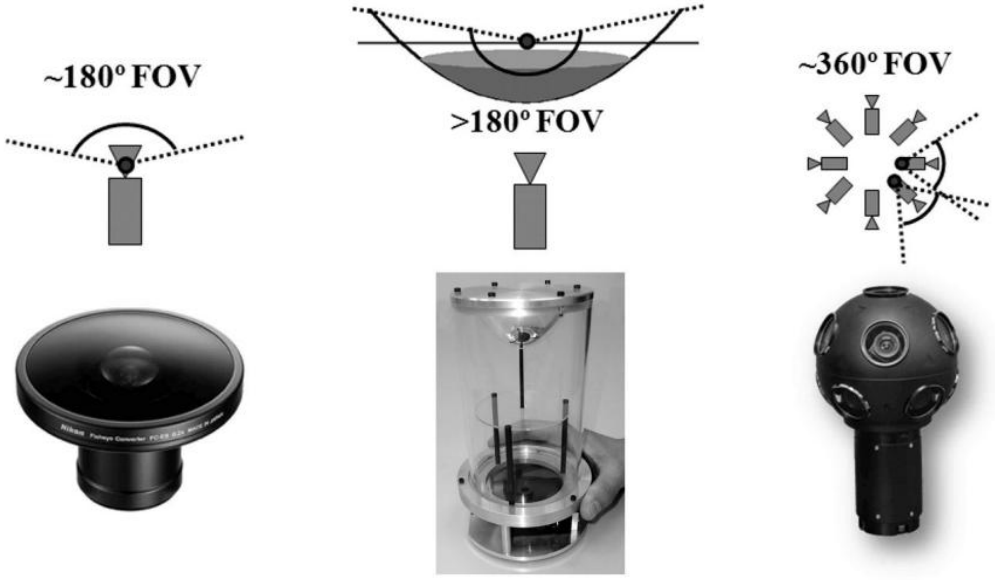
The first stereo based localization was proposed by Olson et al. [2000] for the navigation of robot in long distance. A more accurate solution is proposed by Nister et al. [2004]. The interest points based stereo matching is applied to find some corresponding image points at the beginning. Then some 3D landmarks are reconstructed with triangulation, the poses of the new coming image pairs are estimated with the association of landmarks tracked in the new images. The stereo vision system was also widely used for the localization for autonomous navigation [Milella and Siegwart, 2006; Geiger et al., 2011; Engel et al., 2015]. Compared with monocular camera, there are two advantages for stereo. First, the stereo rig can provide the metric scale directly. Second, the stereo vision system can work more robust and precise. Because more tracks can be searched along sequences and the fixed extrinsic parameters of stereo rig provide inner constraints to maintain accuracy. However, the stereo cameras on the other hand increase the cost on both price and computation. In addition, it also increases the complexity of calibration which might need larger field and more control points to resolve both intrinsic and extrinsic parameters precisely.

2.5.3 Omnidirectional camera

Large Field of View(FOV) is always desirable for vision based localization, because it can provide observations in a larger scene. Thus, the image points can be tracked in longer period which is better for pose estimation. Besides, larger FOV makes it easier to observe informative scenes in some particular cases such as untextured road, wall areas, moving objects, which are challenging for localization. Therefore, some researcher use omnidirectional camera for localization.

An omnidirectional camera refers a camera with 360°FOV in horizontal plane, or with a visual field that covers a hemisphere or entire sphere [Scaramuzza, 2014]. Figure 2.5 shows some solutions for omnidirectional camera. Figure 2.5(a) demonstrates a dioptric lens. With this kind of lens (fish-eye), the image FOV can reach up to 180°. Some researchers often use fisheye or

wide angle for localization [Hansen et al., 2009; Caruso et al., 2015]. Another special lens is a shaped mirror (e.g. parabolic, hyperbolic, elliptical mirror) which can provide larger FOV images (*cf.* Fig 2.5(b)). Some examples of localization using catadioptric lens were also proposed [Lhuillier, 2005; Scaramuzza et al., 2006; Lhuillier, 2008]. However, the images obtained by there two lenses are not real omnidirectional images. Nowadays, the only real omnidirectional camera is composed by integrating multiple cameras together to obtain 360°image (*cf.* Fig 2.5(c)). Therefore, some articles reported the localization using panoramic images [Silpa-Anan et al., 2005; Scaramuzza and Siegwart, 2008; Litvinov et al., 2013]. For omnidirectional camera, we usually do not known about the physical imaging process, thus, a Generic Camera Model (GCM) was developed to represent the projection from object to image [Luhmann et al., 2016].



(a) Dioptric (fisheye) (b) Catadioptric (c) Panoramic

Figure 2.5: Some examples of omnidirectional cameras [Scaramuzza, 2014].

Indeed, the images captured using dioptric or catadioptric lens have large FOV, but the angular resolution of the images is lower. For large scale localization, high angular resolution is desired [Zhang et al., 2016]. The panoramic images can have high resolution, but the projection model from world to image is not rigorous. Because the panoramic images are generated by stitching images captured by different cameras together.

2.5.4 Multi-camera system

In order to obtain images with large FOV, high angular resolution and rigorous sensor model. The multi-camera rig system is introduced. In fact, the multi-camera system is a combination of multiple cameras. The cameras in system are mounted rigidly. The rigid parameters are

calibrated beforehand, which are known in the operation of localization. The multi-camera rig has been used in many applications [Carrera et al., 2011; Shi et al., 2012; Kneip et al., 2013]. Each camera in multi-camera rig can be placed at different directions, thus, large FOV can be obtained. Perspective cameras are often used in multi-camera system and the image captured by each camera is used directly and gives contribution for localization. Thus, the angular resolution is high and the rigorous projection model can be used for each image in localization. The only problem would be the growing of computation due to the increasing images.

Although different camera configurations can be used and we optimize the poses with bundle adjustment for localization, the drift is still accumulated over time. This could lead to poor pose estimation for long term localization. Although the loop closure technique is usually employed to reduce the drift in SLAM, it is time consuming for large environment, on the other hand there is no loop in many situations for the localization of intelligent vehicle [Fuentes-Pacheco et al., 2015]. In order to reduce the drift accumulation for vision based localization, the external data (maps) need to be integrated. The external data is drift-free and contains the absolute information. The types of external data and the methods about integration with vision based localization will be introduced in following section.

2.6 Integration of external data

We have mentioned the maps constrained localization in section 2.3.3, but more details about integrating maps with localization are introduced in this section. Different with the methods proposed for *place registration* which estimate the pose depending on the maps, the integration based methods mean that the external data from maps are associated with other sensors to enhance localization. In this thesis, the external data refers to the absolute position measured by GNSS or the pre-built database such as the maps, 3D landmarks and geo-referenced semantic features, as shown in flowing figures.

(a) (b) (c) (d)

Figure 2.6: Examples of different external resources. (a) GPS receiver[Lhuillier, 2011]. (b) Digital map from OpenStreetMap [Floros et al., 2013]. (c) Textured 3D city model[Caron et al., 2014]. (d) Road extracted from 3D point clouds[Levinson et al., 2007].

With these external data, the localization can be regarded to perform in a well or partly known environment, thus the strategy of the integration top-down processing [Aynaud et al., 2014]. We

categories the methods according to the types of data.

2.6.1 GNSS data

As we introduced at the beginning of this chapter, GNSS can provide the drift-free position in global. Thus, it is of a solution that combines vision based localization with positions measured by GNSS. As the GNSS can provide position directly, one simple way just needs to estimate orientations with known position [Carceroni et al., 2006]. However, the GNSS measurements may jump from time to time. So a better solution is to fuse the positions obtained by GNSS with vision based localization considering their accuracy [Kume et al., 2010; Shi et al., 2012].

Loosely coupled Agrawal and Konolige [2006] maintained global consistency using an inexpensive GPS in a Kalman filter while a solution to fuse a GNSS localization with an Evidential SLAM using a particle filter [Trehard et al., 2015]. The drift was corrected with the north and east positions from GPS since the height measurements are unreliable. A similar approach was studied [Wei et al., 2011; Geng et al., 2015], but both planimetric and height measurements were considered with their uncertainty. A more complex strategy was proposed by Schleicher et al. [2009] that separated the integration of GPS with SLAM as low level and high level steps. The absolute positions were used in EKF to compensate drift for simultaneous localization in a low level while the high level integration considered the topological constraints of the trajectories in a map to improve the global accuracy. Besides, the GNSS measurements are also combined with LBA [Michot et al., 2010; Lhuillier, 2011]. The fusing methods proposed in [Michot et al., 2010; Kume et al., 2010] need to be given proper weights for GNSS measurements. However, this is difficult in practice. An alternative solution was proposed in Lhuillier [2011, 2012], which enforced an upper bound for the back projection error and constrained LBA is used for integration.

Tightly coupled All above-mentioned methods use the positions from GNSS. A tighter solution is to incorporate pseudo-range measurements from GNSS directly. This has been investigated in photogrammetry that integrated raw GPS data with block bundle adjustment [Ellum, 2006]. Recently, some methods were proposed to integrate the pseudo-range measurements directly with vision based localization in EKF [Aumayer et al., 2014] or particle filter [Schreiber et al., 2016]. The tightly coupled manner enables to reject the single satellite's measurement which has large errors, so it achieves higher precision than loosely manner [Aumayer et al., 2014].

The GNSS data is easy to be integrated and the drift can be compensated, but the issue is that the accuracy is high related to the quality of GNSS measurements. Even though the uncertainty of those measurements are considered for most of the proposed integrating method, it is still a problem in GNSS denied environment or multi-path situations. Therefore, some external data

from maps are integrated within localization.

2.6.2 Low level maps

Different types of map are generated, we introduce them according to the features of the maps. In this thesis, the maps contain visual features or point clouds are called as *low level maps* .

2.6.2.1 Visual features

The visual feature based maps are usually generated by means of vision based system. For dense visual feature database, a popular application is *teach and repeat* [Furgale and Barfoot, 2010]. In teaching step, a map of the environment is built by a mobile mapping system [Konolige and Bowman, 2009] or SFM [Royer et al., 2007; Charmette et al., 2010]. In repeating step, the robot tries to follow the same route by registering the image with the visual features in database simultaneously [Konolige and Bowman, 2009; Royer et al., 2007; Charmette et al., 2010]. Although the images are usually captured successively, but the poses are estimated individually for each frame. Usually, some prior knowledge should be known in advance. For instance, the start point of the path is given to reduce the searching scope in database [Royer et al., 2007].

Besides, the geo-referenced images (e.g. satellite images, aerial images) were also applied to provide absolute constraints for vision based localization. The rich texture information of these data make it possible to register the features from images with these referenced image. Leung et al. [2008] presented a monocular vision based localization using the aerial ortho-imagery as the reference map in particle filter. To match the images in street view with the aerial orthoimagery, the building boundaries were extracted from reference images and the edges detected from locating images are analyzed with vanishing theory. Ji et al. [2015] introduced a method to generate Ground Control Points(GCPs) from orthoimagery for vision based localization using panoramic images. The GCPs are generated by matching the ortho-rectified panoramic image patches with reference images. A similar strategy was proposed by Kume et al. [2015] who aims to process perspective images based localization. The bundle adjustment was applied to refine the parameters.

2.6.2.2 Point clouds

A high-resolution environment maps can be generated using LiDAR data acquired by a mobile mapping system . A reliable localization with accuracy in the 10 cm was achieved with the help of these high-resolution maps [Levinson et al., 2007]. But a GPS/INS navigation system and a LiDAR was mounted in the vehicle. Bodensteiner et al. [2011] proposed to use geo-referenced LiDAR data to optimize the trajectories estimated by monocular camera. An intensity based

map was generated using the LiDAR point clouds. The image poses were corrected by matching the captured images with the intensity map using MI. In localization, we want to use the static information such as road topology, building shape, white line, curb, traffic light etc. A method to realize these features for localization was proposed in [Yoneda et al., 2014] while Schlichting and Brenner [2016] kept the static scene by removing the moving objects according to the results of change detection. Maddern et al. [2014] made use of a so-called illumination invariant color space to minimize the variations caused by viewpoint and illumination conditions and estimated the image poses by matching with the point clouds acquired by mobile mapping system.

The precision of level maps could be high, but there are some drawbacks for them. First, high storage volume is required in the embedded system due to the large number of features. Second, they suffer from ambiguity problems for matching which are caused by the repeatable or low informative features.

2.6.3 GIS data

In fact, more research about external data integration usually high level maps such as urban road network, OpenStreetMap[3](OSM) and 3D city model.

2.6.3.1 2D maps

The 2D maps contain rich spatial information including road segments, building boundary, locations and attributes of objects etc. As vehicle moving on road, so the relative relations between vehicle and road segments can be regarded as constraints. The issue is to know the correspondences between vehicle and road segments. To do this, a cheap GPS was applied to provide the initial position of visual odometry [Alonso et al., 2012; Brubaker et al., 2013], then curve-to-curve map-matching was performed to correct the drift. Nedevschi et al. [2013] aligned the lane markings extracted from images captured by an on-board camera, with digital maps to estimate position of vehicle simultaneously. All these three methods need GPS to provide initial position for map-matching. Recently, a graph matching based method was proposed to align the trajectory estimated by visual odometry with GIS databases like OpenStreetMap[4](OSM) [Gupta et al., 2016] directly. Meanwhile, some high level maps are also integrated to obtain precise localization for autonomous driving. For instance, the lane markings recognized from images are used to correct bias of GPS by aligning them with road maps Jo et al. [2013]. The lane marking maps are also used to correct the errors of GPS/IMU by shape registration [Cui et al., 2016].

[3]http://wiki.openstreetmap.org/wiki
[4]http://wiki.openstreetmap.org/wiki

2.6.3.2 High dimensional maps

The GIS data often contains multiple layers and the 2D maps is only one of the layers. Some higher dimension data such as Digital Elevation Model(DEM), 3D city model, point clouds, are often included. The DEM is a kind of 2.5D data, the city model and point clouds are in 3D. These kinds of data are massively produced and easy to be afforded nowadays, so many articles presented the methods to integrate these data with vision based localization.

Arth et al. [2015a] proposed the pose estimation of individual image using 2D untextured city model (OSM with building height) based on semantic image segmentation, this kind of methods can be used for the initialization of SLAM. For localization of moving vehicle, the relative height from camera to road can be assumed to be fixed. If we know the elevation of road at each time step, the altitude of camera can be calculated easily by adding a fixed value to the road altitude so that the six DoF pose can be reduced to five DoF. This altitude constraints can provide new parameterization of pose model in bundle adjustment [Larnaout et al., 2012]. An improved strategy was proposed that combined constraints from GPS and DEM using constrained bundle adjustment [Larnaout et al., 2013]. The problem of DEM is the accuracy that is in meters for elevation. The 3D city model is also used that registers the coarse structure generated by vision based localization with 3D model using ICP algorithm, then correct the drift in bundle adjustment associated by the 3D points [Lothe et al., 2009].

The GIS data is easy to store and manage so that it can be used on-line and saved in a local computer. However, the absolute precision of the GIS data such as 3D city model, DEM and 2D maps is still in meters nowadays.

2.6.4 Semantic features

In urban area, there are rich features like road markings, traffic signs, buildings and other man-made landmarks which are static, low ambiguity and easy to be detected. In this thesis, these high level features are called *semantic landmarks*. If the geo-referenced semantic landmark database is built beforehand, the constraints can be generated for vision based localization.

In autonomous driving, one important information for vehicle is to know the relations relative to road markings. This information is important to identify vehicle state and desirable for safety driving system [Pilutti and Ulsoy, 1999]. One relevant application is the lane keeping, which estimates position of vehicle relative to road markings [Sivaraman and Trivedi, 2013; Suhr and Jung, 2015]. The lane keeping applications pay more attention on the topological relations of vehicle to road markings. Lauffenburger et al. [2008] recognized traffic signs to enhance vehicle localization based on map matching. In this thesis, we aim to integrate the landmarks with vision based localization.

The road markings were extracted in ground based imagery and matched with geo-referenced aerial images. Then these features can be used for Ground Control Objects (GCOs) [Tournaire

et al., 2006a]. The road features can also be extracted from point clouds acquired by a mobile mapping system. Brenner [2010] generated poles such as traffic signs, traffic lights, and trees from the point clouds as geo-referenced landmarks. In localization steps, the constraints were generated by matching the poles features. Then the constraints were used to reduce the drift of dead-reckoning in the least squares. With the same goal, Schlichting and Brenner [2014] trended to use pole-like and build facades extracted from point clouds while a high level road structural feature composed by a set of line segments on lane markings, curbs, poles, building edges, etc, is used by Yu et al. [2014b]. A more affordable solution is to combine the vision based localization with the semantic landmarks database (e.g. lane [Pink, 2008], road markings [Schreiber et al., 2013; Wu and Ranganathan, 2013], traffic signs [Wei et al., 2014]). Recently, the semantic landmarks are also integrated with LBA, where the heterogeneous features including points, lines, planes, vanishing points and their inner geometric constraints are jointly considered [Lu et al., 2014]. A multilayer feature graph is defined to manage the various elements in the landmark database.

Comparing with general geo-referenced data, the semantic landmarks have higher precision, lighter storage volume and lower ambiguity. In addition, they are easier to be detected and managed. To integrate this kind of landmarks with vision based localization, the main problem is able to detect and reconstruct them precisely and efficiently. Recent years, many methods have been proposed (e.g. traffic signs [Soheilian et al., 2013a], road markings [Soheilian et al., 2010; Hervieu et al., 2015]) for detection and reconstruction, that motivates us to integrate the semantic features with vision based localization.

Table 2.6.4 is a summary of recent research on vision based localization using successive images. Different camera configurations (monocular, stereo, omnidirectional and multi-camera rig) are investigated. Large FOV configuration can obtain more robust pose estimation because the visual landmarks could be tracked in longer period. Hoverer, increasing FoV with fixed resolution (fisheye, catadioptric lens) reduces the angular resolution of the image, that decrease the precision of visual points location. In general, the cameras with larger FOV perform better in indoor environments(confined environment), while smaller FoV cameras are preferable in urban canyon scenarios with the benefit of higher angular resolution [Zhang et al., 2016]. Besides, the filter based solution (EKF, particle filter) and bundle adjustment are the most popular solutions for localization. More methods about integrating GNSS data with visual odometry have been proposed for the integration of geo-referenced data, especially semantic landmarks. In this thesis, we aim to integrate geo-referenced semantic features with vision based localization adapting to different camera configurations.

Table 2.2: Summary of some vision based localization systems

Author	Camera configuration	Image features		external data	Key techniques	Type of environment
		Detector	Descriptor			
Olson et al. [2000]	Stereo	Forstner	image patch	no	3D-3D pose estimation	outdoor
Davison [2003]	monocular	Shi&Tomasi	image patch	no	EKF	indoor
Nister et al. [2004]	monocular,stereo	Harris	image patch	no	P3P+RANSCAC	outdoor
Mouragnon et al. [2006]	monocular	Harris	image patch	no	P3P+LBA	outdoor
Klein and Murray [2007]	monocular	Fast	image patch	no	Parallel Tracking + LBA + GBA	outdoor
Scaramuzza and Siegwart [2008]	omnidirectional	SIFT	image patch	no	homography-based tracker, plane based pose estimation	outdoor
Mouragnon et al. [2009]	monocular, stereo, omnidirectional	Harris	image patch	no	GCM[5] +LBA	indoor, outdoor
Lothe et al. [2009]	monocular	SURF	SURF	3D model	ICP, BA	outdoor
Eudes and Lhuillier [2009]	monocular	Harris	SURF	no	LBA, uncertainty propagation	outdoor
Kaess and Dellaert [2010]	multi-camera	Harris	image patch	no	expectation maximization, BA, loop closure	indoor
Wei et al. [2011]	stereo	SURF	SURF	GNSS	EKF, loosely coupled	outdoor
Lhuillier [2011]	monocular	Harris	image patch	GNSS	constrained-LBA	outdoor

[5]Generalized Camera Model

Bodensteiner et al. [2011]	monocular	NULL	SIFT/SURF	Point clouds	MI[6], BA	outdoor
Geiger et al. [2011]	stereo	Blob/corner	Sobel responses	no	Kalman filter	outdoor
Kazik et al. [2012]	Non-overlap stereo	AGAST[7]	BRIEF	no	LBA, loosely coupled	Indoor, outdoor
Larnaout et al. [2012]	monocular	SURF	SURF	DEM	Five DoF Pose, BA	outdoor
Larnaout et al. [2013]	monocular	SURF	SURF	DEM, GPS	Constrained BA	outdoor
Kneip et al. [2013]	multi-camera system	Simulation data		no	GPnP+LBA , tightly coupled	indoor, outdoor
Lu et al. [2014]	mono	SIFT points and LSD[8] lines		no	SIFT, vanishing point detection and matching, LBA	indoor, outdoor
Aumayer et al. [2014]	stereo	Harris	image patch	GNSS	EKF, tightly coupled	outdoor
Schneider and Förstner [2014]	omnidirectional	Shi&Tomasi	image patch	GNSS	KLT[9] tracker Incremental BA	outdoor
Persson et al. [2015]	stereo	Fast	BRIEF	no	LBA	outdoor
Desai and Lee [2016]	monocular, stereo	SURF	SYBA[10]	no	drift reduction	outdoor
Schreiber et al. [2016]	stereo	Blob/corner detector	Sobel filter responses	GNSS	EKF, tightly coupled	outdoor

[6]Mutual Information
[7]Adaptive and Generic Accelerated Segment Test
[8]Line Segment Detector
[9]Kanade–Lucas–Tomasi
[10]SYnthetic BAsis

2.7 Our strategy

The goal of our research is to develop a low-cost but precise localization system. As we discussed in section 2.1, the beacon based approaches are difficult for large scale localization while the accuracy of *place resignation* depends on the quality of maps and the results of place recognition which is far from being precise. The GNSS are the most popular method for global localization, but it suffers from mask and multi-path in urban canyons. Compared with global localization, the approaches of *position tracking* can overcome parts of problems in global localization such as multi-path and mask, but the estimated poses are relative (*cf. section* 2.2). Therefore, the combining solutions are usually taken to enhance the localization. However, the combination of multiple sensors makes the system be costly, heavy and more energy consumption (*cf. section* 2.3). In this case, we trend to the integration of maps with vision based localization. The maps are produced off-line and the vision system is low cost, portable and low power consumption. The characteristic of different camera configurations are analyzed in section 2.5. For vision based localization in large scale, high angular resolution is desired, so we use perspective camera. To enlarge the FOV, multiple cameras can be combined together for localization. Different techniques have been proposed for localization, we divide them into

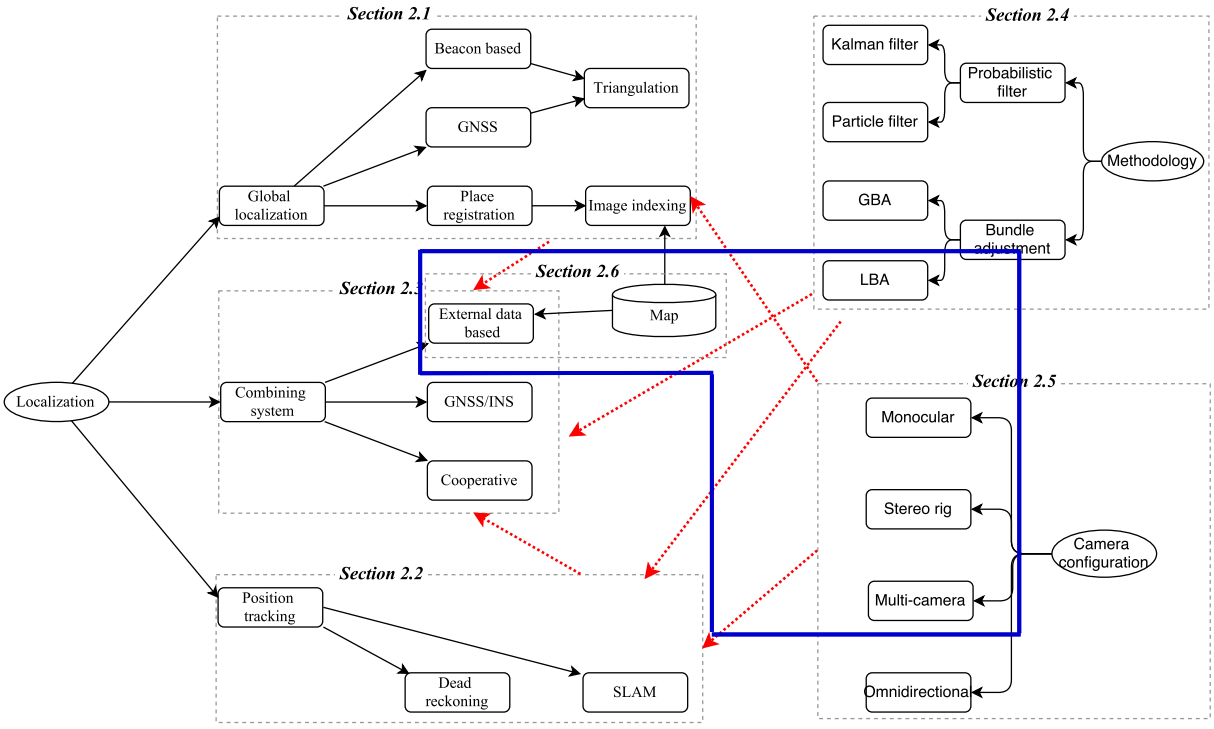


Figure 2.7: The most related research about localization with our method.

probabilistic filter based methods and bundle adjustment based methods according to the core of the solutions (*cf. section* 2.4). It is well known that *bundle adjustment* can achieve more precise results than *probabilistic filter* [Klein and Murray, 2007; Strasdat et al., 2010a]. Therefore, we integrate the maps into localization via bundle adjustment. In particular, the uncertainties are considered over the integration to reduce the impact of inaccurate data in maps. By com-

paring different types of map, the geo-referenced semantic features are most suitable for the localization in dense urban environment (*cf. section* 2.6).

The most relevant research about localization has been marked, as shown inside the blue box in figure 2.7. We intend to integrate the vision based localization with geo-referenced semantic objects to provide precise localization in urban environment. The semantic objects are taken into account as landmarks and they are integrated into LBA considering the uncertainty propagation. In our research, we aim at developing the localization method which can be easily adopted for different camera configurations, composed by perspective cameras.

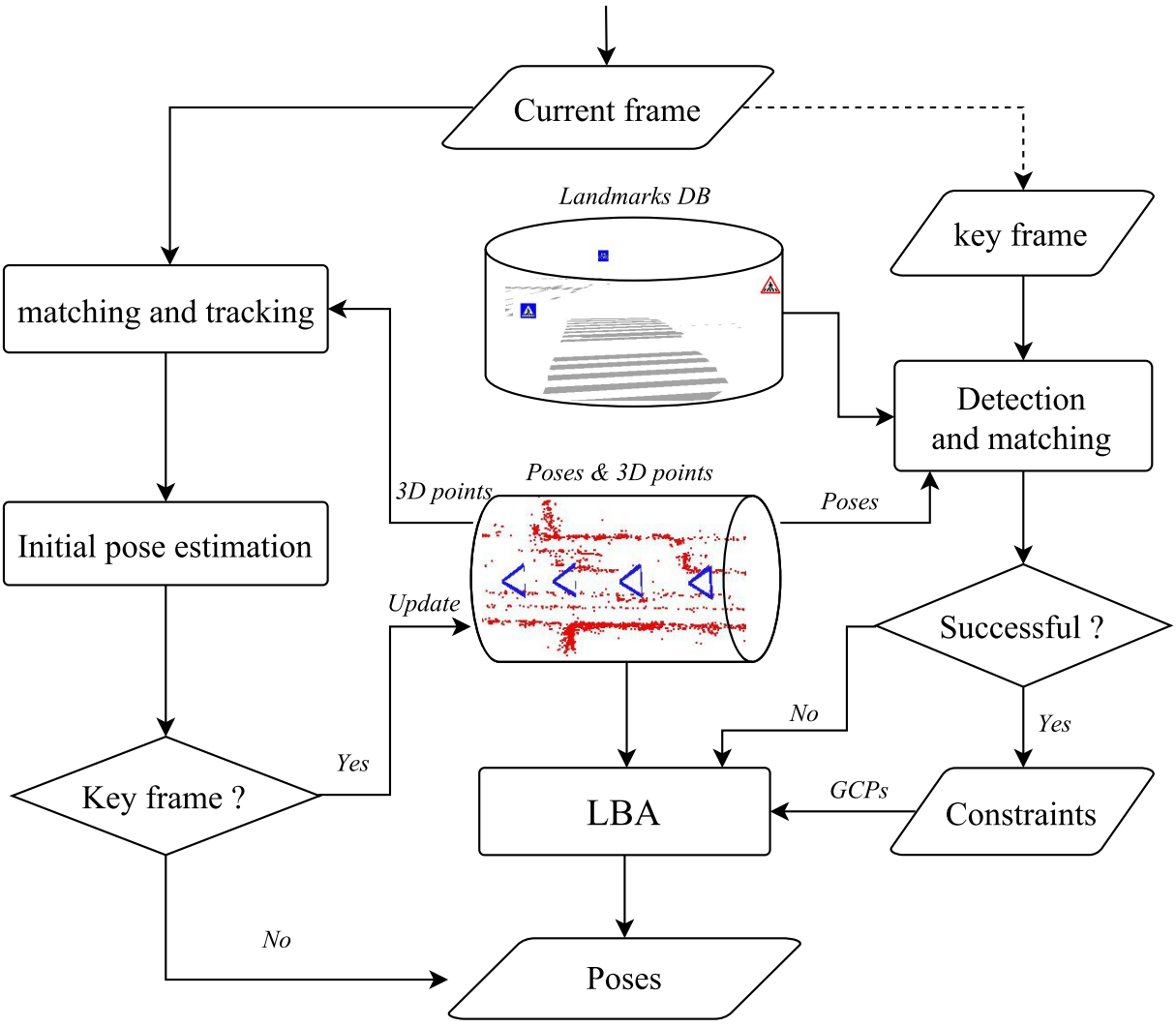


Figure 2.8: The proposed flowchart of our localization approach.

In order to provide global localization approach, we need to know at least a start point for vision based localization at beginning. In practice, the initialization of localization can be solved by fusing the data from GNSS, beacon based global localization or place registration. In this thesis, we suppose that we start from a known point. Then, the successive poses are tracked relative to the start point and optimized by integrating the geo-referenced landmarks.

The proposed flowchart is demonstrated in diagram 2.8. For each frame, we detect the interest

points in image, match and track the image points over sequences. The pose is estimated for every frame instantly, but only keyframes are selected and involved into LBA considering the constraints from landmarks integration. If key frame is identified, the semantic landmarks are detected and matched with the patterns in database to obtain a set of geo-referenced constraints (*cf.* Fig 2.8). If the constraints are generated successfully, constrained LBA is performed considering the uncertainty propagation of both image poses and geo-referenced landmarks. Otherwise, the parameters are estimated with general LBA with uncertainty propagation. In our research, we intend to model the absolute constraints via Ground Control Points (GCPs)

Chapter 3

Localization using vision based system

This chapter introduces our vision based localization using single or multi-camera system. Figure 3.1 (highlight steps) shows the pipeline of vision based localization. As we discuss in

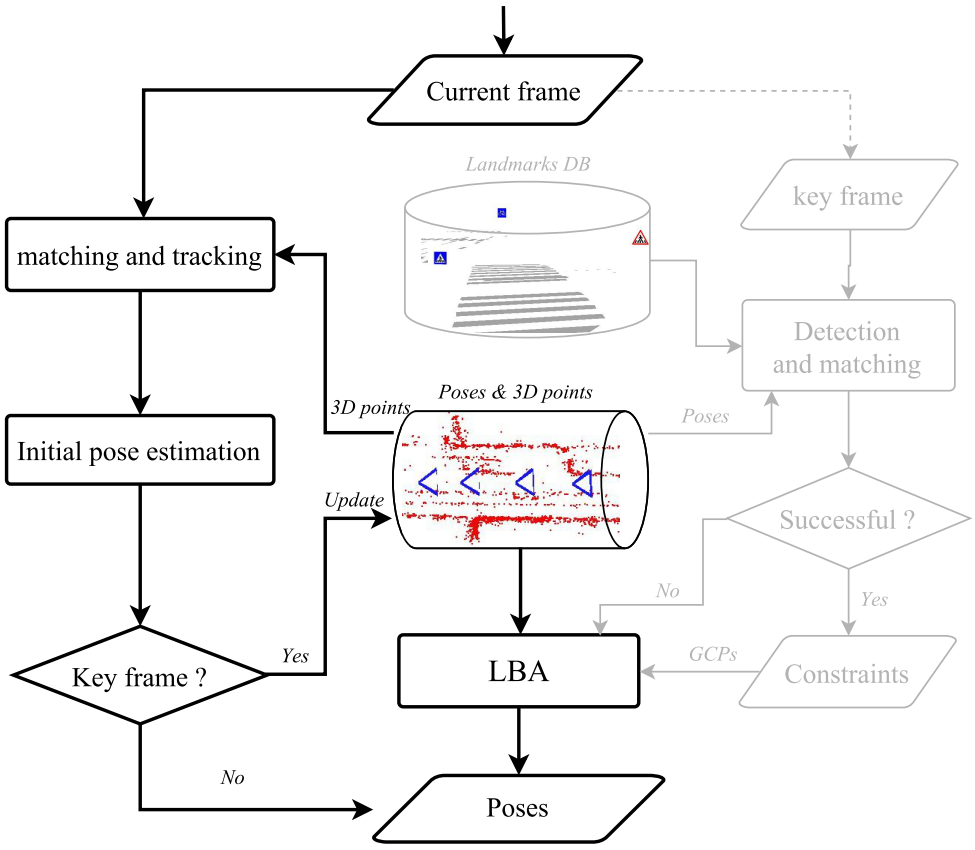


Figure 3.1: Pipeline of vision based localization.

chapter 2, the image sequences are captured by perspective camera in street. At the beginning of this chapter, we introduce the projection model from object to image for perspective camera. A *tie point* structure is defined in this thesis to express the object point and its links in image. The camera model and tie point will be introduced in 3.1. In order to estimate pose of every frame, a set of correspondences of interest points over images need to be fund, this process is

conducted by *matching and tracking* in section 3.2. Then we estimate the initial pose using matching and tracking results for every frame and select key frame as input for LBA (see *initial pose estimation* and *key frame* in figure 3.1). The keyframe poses and tie points are optimized via LBA. The work-flow is same, but different camera configurations are developed for localization in our research. Section 3.3 introduces initial pose estimation, key frame selection and LBA using monocular camera. A generic model for multi-camera system is presented in section 3.4.

3.1 Camera model and tie point

3.1.1 Camera projection model

Although both perspective and omnidirectional cameras can be used for localization, we focus on perspective cameras which are cheap. The perspective camera model assumes a pinhole projection from world to image, as shown in figure 3.2. Let $\boldsymbol{X} = [X, Y, Z]^T$ be a 3D object point in camera reference frame. The image point $\mathbf{x} = [u, v]^T$, which is the projection of $\boldsymbol{X}$ in image plane, is located at the intersection of the line $\boldsymbol{X}$ to camera center and the image plane. Point $\mathbf{p}$ is called principal point, that is the projection of the camera center on the image plane [Hartley and Zisserman, 2003].

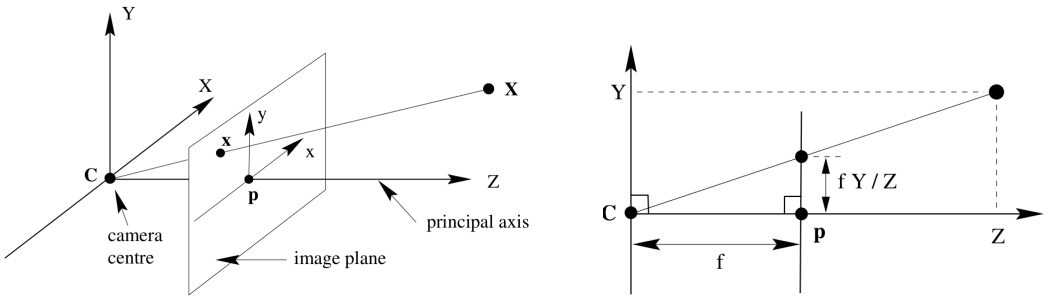


Figure 3.2: Pinhole projection from 3D to 2D [Hartley and Zisserman, 2003].

As shown in the right figure 3.2, the coordinate of 2D point $\mathbf{x}$ in Y axis in camera reference frame can be computed by similar triangles, that is fY/Z, where f is the focal length. Similarly, we can compute the value of $\mathbf{x}$ in x axis. Thus, we can easily map object point $\boldsymbol{X} = [X, Y, Z]^T$ to $[fX/Z, fY/Z, f]^T$ in image plane. If we ignore the third coordinate in $[fX/Z, fY/Z, f]^T$ and denote $[u, v]^T$ as the coordinates of image point, we see

$$[X, Y, Z]^T \longmapsto [u, v]^T = [fX/Z, fY/Z]^T,$$

that describes the central projection from a point in Euclidean 3-space to an image point in Euclidean 2-space [Hartley and Zisserman, 2003].

This is an ideal situation that principal point $\mathbf{p}$ is located at the center of image. In practice, it might have offsets to image center, noted as $[u_0, v_0]^T$. So a general expression of the mapping

from 3D to 2D is:

$$[X, Y, Z]^T \longmapsto [u + u_0, v + v_0]^T.$$

Express the coordinates in homogeneous coordinates as:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & u_0 & 0 \\ 0 & f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{1}$$

where, $\lambda = Z$, which is Z coordinate of object point.

Writing intrinsic matrix of camera as:

$$\boldsymbol{K} = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

A simple form of equation 1 is:

$$\mathbf{x} = \boldsymbol{K}[\boldsymbol{I}|\mathbf{0}]\boldsymbol{X} \tag{3}$$

The object point and image point in equation 3 are expressed in a same coordinate system. However, they are often measured in different euclidean coordinate systems in practice. For instance, the object points are often defined in an absolute system, while the image points are in a local image space. In this case, a transformation between the two coordinate systems should be considered, when we project an object point into image plane using equation 3.

Let $\boldsymbol{R}$ as rotation matrix for orientation and $\boldsymbol{C}$ as position of image in world coordinate system. Thus, the same object point in camera system, noted as $\boldsymbol{X}_{cam}$, can be estimated via:

$$\boldsymbol{X}_{cam} = \boldsymbol{R}(\boldsymbol{X} - \boldsymbol{C}).$$

. Considering the transformation into equation 1, the general form of central projection is:

$$\mathbf{x} = \boldsymbol{K}\boldsymbol{R}[\boldsymbol{I}| - \boldsymbol{C}]\boldsymbol{X} \tag{4}$$

A more compact form is:

$$\mathbf{x} = \mathbb{P}_{3\times4}\boldsymbol{X}, \tag{5}$$

where, $\mathbb{P}_{3\times4}$ is called camera projection matrix, which contains both intrinsic and extrinsic parameters of camera. Our localization is based on calibrated camera, so the distortion of interest points are rectified with pre-calibrated distortion model. The intrinsic parameters for each camera are known as well. With known intrinsic parameters, we define the pinhole projection as function $F(\boldsymbol{P}, \boldsymbol{X})$ that projects 3D points $\boldsymbol{X}$ into image with known transformation and orientation. In this case, the equation 4 can be written as:

$$\mathbf{x} = F(\boldsymbol{P}, \boldsymbol{X}) = \boldsymbol{K}\boldsymbol{R}[\boldsymbol{I}| - \boldsymbol{C}]\boldsymbol{X}. \tag{6}$$

where, $\boldsymbol{P}$ and $\boldsymbol{X}$ represent unknown parameters in back-projection function. $\boldsymbol{P}$ is image pose, composed by translation and rotation measurements, seeing equation below:

$$\boldsymbol{P} = [X, Y, Z, \psi, \theta, \phi]^T.$$

The position X, Y, Z is determined by the position of camera center $\boldsymbol{C}$. The three angles are decomposed from rotation matrix $\boldsymbol{R}$ which is composed with three sub-rotation in xyz axes [Slabaugh, 1999] :

$$\boldsymbol{R} = \begin{bmatrix} cos(\phi) & -sin(\phi) & 0 \\ sin(\phi) & cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & 1 & 0 \\ -sin(\theta) & 0 & cos(\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\psi) & -sin(\psi) \\ 0 & sin(\psi) & cos(\psi) \end{bmatrix}.$$

3.1.2 Tie point structure

In this thesis, we call a 3D object point that can be visually observed in overlap areas between two or more images as tie point (*cf.* Fig 3.3). $\boldsymbol{X}$ is tie point in 3D space, the projections of tie point in multiple images are referred to 2D measurements of tie point in image, as $(\mathbf{x}_j, \mathbf{x}_l, \mathbf{x}_n, \mathbf{x}_k)$ points (*cf.* Fig 3.3).

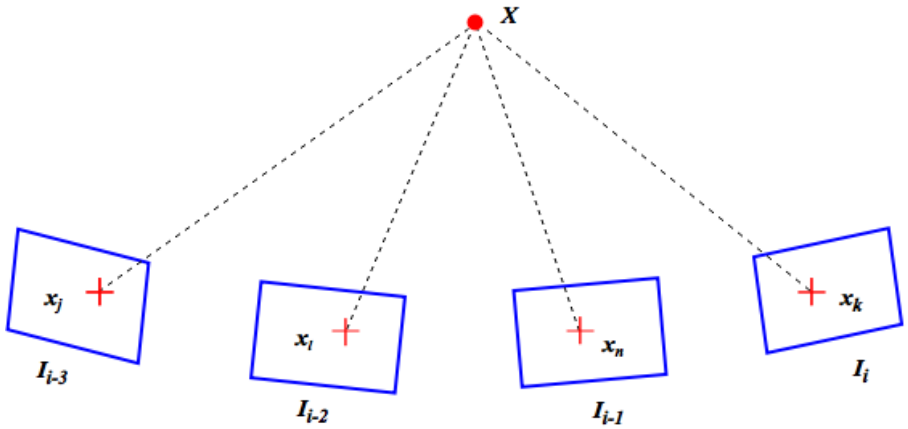


Figure 3.3: An example of tie point structure. X is the tie point, $I_{i-3}, I_{i-2}, I_{i-1}, I_i$ are four consecutive images. The correspondences(m_j, m_l, m_n, m_k) are the observations of X in images.

According to the description of tie point, the data structure of one tie point can be defined using a table like:

$TiePoint$
{
Int ID; *unique ID*
Point3D $\boldsymbol{X}$ *coordinates*
Vector $\mathbf{x_s}$ *image points*:$\langle I_i, m_k \rangle; \langle I_{i-1}, m_n \rangle; \langle I_{i-2}, m_l \rangle; ...$
}

We allocate a unique ID for each tie point. The data structure includes the coordinates of tie point and its measurements in images. In the vector of image points, each image point cell includes the image ID and coordinates. It represents the location of tie point in the specified image.

The aim of matching and tracking is to build the links between images point and tie points as shown in figure 3.3. This is one of the basic technique for pose estimation, because we need 3D-2D correspondences to solve the pose of current frame. Meanwhile, some new tie points should be reconstructed for pose estimation of coming images over time.

3.2 Feature extraction, matching and tracking

3.2.1 Feature extraction

Many methods for interest points detection and feature description have been proposed in last decades. In this thesis, we choose SIFT and SURF as candidate methods for feature extraction due to their outstanding performance. Both of SIFT and SURF are invariant to scale change, rotation and illumination [Lowe, 2004; Bay et al., 2006].

SIFT creates a scale space for image using DoG (Difference of Gaussian) and potential key-points are detected in scale space. Then the low contrast points and poorly localized points are eliminated when calculated Laplacian values are smaller than the given threshold. The precise location of each key points is interpolated based on quadratic Taylor expansion of the difference-of-Gaussian scale-space function and the descriptor is formed using a gradient histogram in a region around the point with 128 bins weighted by a Gaussian function. SIFT has been applied in vision based localization [Se et al., 2001; Yang et al., 2009]. However, the computation of SIFT feature extraction is time consuming. A more efficient method called SURF (Scale Invariant Feature Transform), was proposed by Bay et al. [2006], which built the scale space using integral images and described the feature based on Haar wavelet using a 64 dimensions descriptor. It is also widely used in vision based localization methods for feature detection [Murillo et al., 2007; Eudes and Lhuillier, 2009].

Recently, some new computing techniques such as GPU (graphics processing unit) [Wu, 2007; Terriberry et al., 2008], application-specific integrated circuits (ASIC) or field-programmable gate arrays (FPGA) [Yao et al., 2009; Lee et al., 2014], have been used for implementation of SIFT or SURF, which can speed up the feature extraction. With these advanced computing techniques, both SIFT and SURF are able to be applied in real time application. In this case, the problem is to know which one is better for our application between SIFT and SURF. Although many articles have already investigated the performance of SIFT and SURF for visual odometry and feature tracking [Ballesta et al., 2007; Valgren and Lilienthal, 2007; Gauglitz et al., 2011], SIFT and SURF perform differently for every case. Therefore, we carefully design several

experiments using the sequences captured by our mobile mapping system to evaluate SIFT and SURF. Besides, the impact of interest point distribution was also studied. We evaluated the performances from four aspects: repeatability, precision, accuracy and runtime. According to the results of our experiments, SIFT was more reliable and precise for localization than SURF. So, SIFT is our choice for feature extraction. The detailed information about the experiment setup was introduced in Qu et al. [2016].

3.2.2 Matching between images

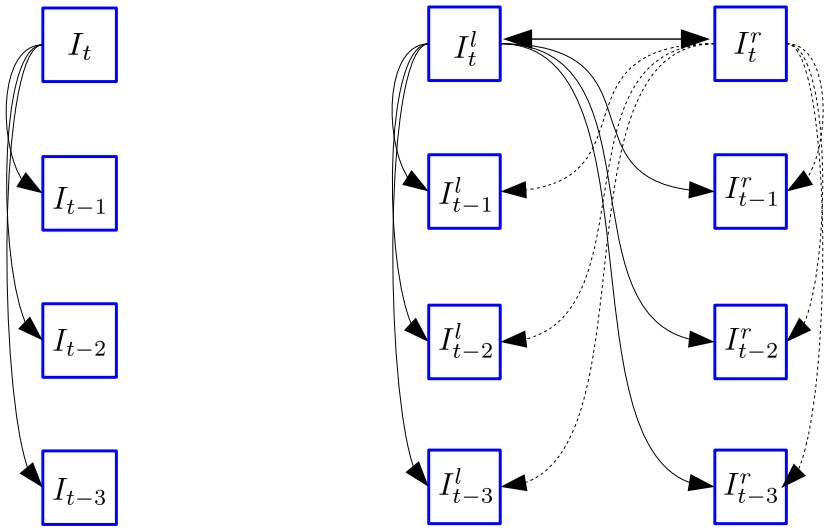
The issue at this stage is to find the correspondences of interest points (keypoints) over sequence. As the rate of video or images are often very high, so we only keep key frames, retrieved from original image sequences to maintain the trajectory. The method for key frame selection will be presented in section 3.3.3. For every current frame, we match it with a fixed number of previous keyframes.

Regarding a camera moving straightly, it is easier to match the current frame with the latest key frame than the key frames before. However, larger time interval means longer baseline between current frame and key frame, which can provide well-conditioned matches for pose estimation and point triangulation. Now, we should determine how many key frames should be matched with the current frame. We suppose sampling distance between two successive images is larger than 1m and the intersection angle, which is the angle between two rays jointing a tie point and its image points, should be above $3°$at a depth of $50m$. Then the length of the baseline is $3m$ approximately. It is well known that the intersection angle is related to the locations of image points. This assumption here only give us an idea about the relationship between baseline length and tie point depth relative to camera.

With this hypothesis, we match the current frame with three latest key frames. We denote I_t as current frame, the matching for image I_t is to search the correspondences in the subsequence $(I_{t-1}, I_{t-2}, I_{t-3})$. This procedure can be divided into three steps which corresponds to three pair-wise matching: $I_t \Leftrightarrow I_{t-1}$, $I_t \Leftrightarrow I_{t-2}$, $I_t \Leftrightarrow I_{t-3}$ (*cf.* Fig 3.4(a)). We call I_t as *reference image* and images $I_{t-1}, I_{t-2}, I_{t-3}$ as *target images*.

The matching process for binocular images is similar as monocular sequence. The current pair is matched with three latest pairs (see figure 3.4(b)), but there are more pair-wise matching units compared with monocular case. For multi-camera system, the matching can be still divided into several pair-wise matching units.

To search correspondence for each key point in *reference image* I_t into *target image* , we compare the similarity between SIFT descriptors. The matches are found by searching its nearest neighbor which is defined as the minimum Euclidean distance between two SIFT descriptors. In our research, FLANN (Fast Library for Approximate Nearest Neighbors) [Muja and Lowe, 2009] is employed for the nearest neighbor searching and two best matches are kept as candidates for each key point. To filter the false matches, two principles are taken into account.



(a) Matching for monocular images. (b) Matching graph for stereo pairs.

Figure 3.4: Matching graphs for monocular and binocular image sequences between current frame (pair) and latest three key frames (pairs) .

First, if the distance ratio between first minimum distance and the second minimum distance is less than 0.8, we accept the matches. This measure filters most of the false matches. The distance calculated for correct matches should be significantly smaller than the incorrect matches [Lowe, 2004]. Second, the rest outliers of matches are rejected according to epipolar constraint. An AC-RANSAC (A Contrario RANSAC) [Moulon et al., 2012] based algorithm is applied to estimate fundamental matrix. Reject the matches whose distances from image points to their corresponding epipolar lines are larger than 2.0 pixels.

In practice, relative pose between two images can be estimated with a set of 2D correspondences [Nistér, 2004], but metric translation is unknown. In vision based localization using successive images, we desire consistent absolute scale over time. It is well known that the absolute pose can be estimated using 3D-2D correspondences [Ganapathy, 1984], that are the correspondences between tie points and image points in thesis. Hence, the issue is to track the tie points over sequence for pose estimation.

3.2.3 Tracking tie points

3.2.3.1 Tracking from pair-wise matches

Matching image I_i with previous three key frames $I_{i-3}, I_{i-2}, I_{i-1}$ in figure 3.3, we can obtain three pair-wise matches: $x_k \Leftrightarrow x_l$, $x_k \Leftrightarrow x_n$, $x_k \Leftrightarrow x_j$. Yet obtaining those pair-wise matches is not our purpose, we need image points chain $x_k \Leftrightarrow x_l \Leftrightarrow x_n \Leftrightarrow x_j$, which links a tie point with its images points. This is a problem of how to merge two-view matches into a consistent image points chain. The solution is so called *tracking* and the aim is illustrated in figure 3.5.

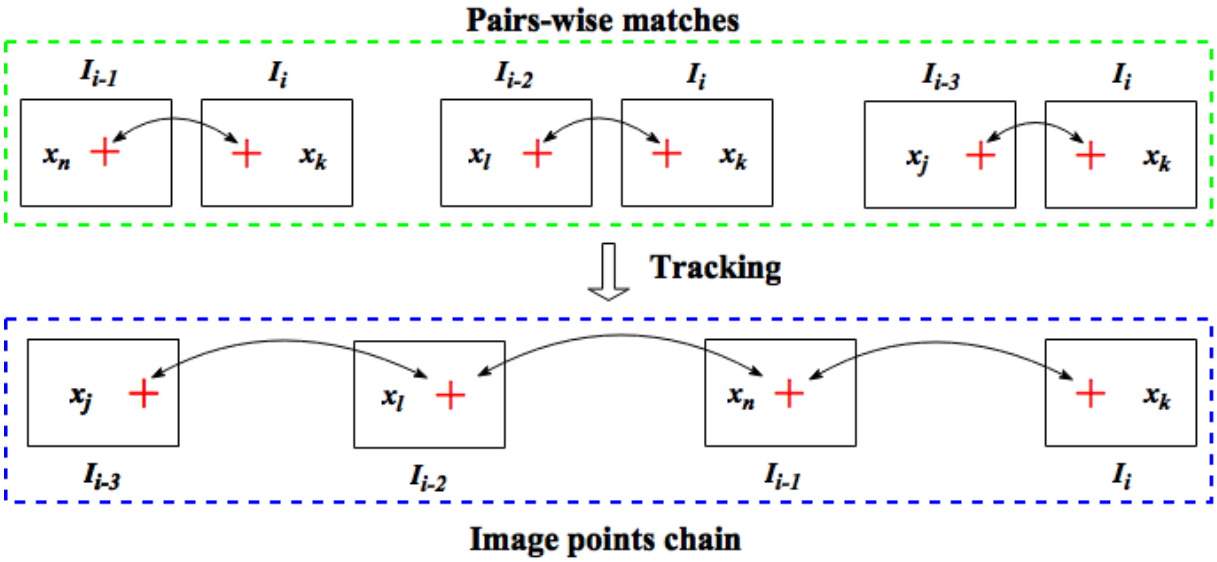


Figure 3.5: Tracking across pair-wise matches.

The inputs of tracking are the pairwise matches between images, the outputs are the correspondences through images, shown as the image points chain in figure 3.5. We set a unique ID for each image point. The pair-wise matches can be represented by their corresponding image points ID. Hence, if two pair-wise matches contain a same image point (same image point ID), these two matches can be merged. Then, compare other pair-wise matches with the same method until all the pairwise matches are searched in the graph. The techniques to solve this problem are usually used in structure from motion [Snavely et al., 2006; Irschara et al., 2009]. In this thesis, a tracking method proposed by Moulon and Monasse [2012] is applied. This tracking method is based on Union-Find algorithm [Galler and Fisher, 1964]. The *Union* refers to pairwise connection and the *Find* aims to search the connections over all the pairwise connections. The key is to build the join function for Union-Find algorithm, which merges the correspondence subsets. The outputs are a vector of image points ID lists. Each list contains all image point IDs for one tie point.

For multi-camera cases, this tracking method can still be used to merge the pair-wise matches. The inputs of the tracking method is only pairwise matches represented by image point IDs, which is not affected by camera configuration.

3.2.3.2 Tie points merging

After tracking, the data structure for each tie point, presented in section 3.1.2, can be built. However, the issue is that some of these points may have been reconstructed in previous steps through image $I_{t-1}, I_{t-2}, I_{t-3}$. Thus, the newly tracked tie points can be divided into two categories: *existing tie points* and *new tie points*. Figure 3.6 illustrates the difference between new tie point (green point X_2) and existing tie point (red point X_1). The X_1 is an existing tie point which has been estimated using previous images, and X_2 is a new tie point. In this case, the

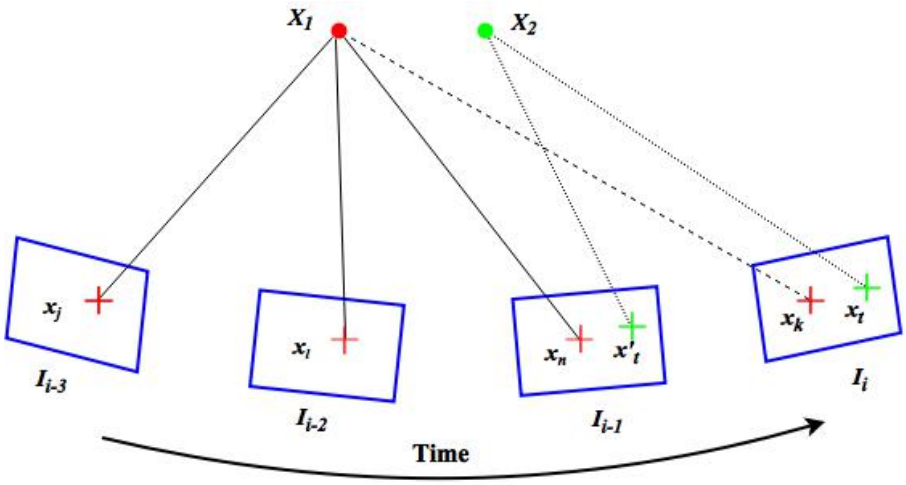


Figure 3.6: X_1 represents the existing tie point and X_2 is the new tie point generated from image I_i and I_{i-1}.

3D coordinates of X_1 is known while X_2 is an unknown. To find the *existing tie points*, we compare the image point IDs between newly tracked tie points and inherited tie points from previous steps. If one or more same image points are fund, the newly tracked tie point belongs to *existing tie points*.

3.3 Single camera based approach

As shown in figure 3.1 at the beginning of this chapter, the initial pose is estimated using the results of matching and tracking. Then identify the current frame whether it is key frame. If current frame is key frame, it is inputted into LBA buffer to optimize the pose and tie points. In order to obtain poses in absolute system, at least the start point and absolute scale should be known for monocular localization. In this section, we start with the initialization of localization which introduces how to select start point and set absolute scale for localization (see section 3.3.1). Then, our localization is relative to the start point. Section 3.3.2 presents the methods for initial pose and tie points estimation. We estimate pose for every frame, but only keyframes are selected for LBA. The strategy for key frame selection is proposed in section 3.3.3. With the initial values, the keyframe poses and tie points are optimized with LBA (see section 3.3.4). At last, we evaluate the proposed approach using real datasets (section 3.3.5).

3.3.1 Initialization

To obtain metric localization in the geo-referenced system, a reference point need to be given in an absolute system for mono based localization. In practice, geo-referenced map [Gupta et al., 2016] can be used to provide initial information. Absolute scale can also be provided using a wheel encoder or the height of camera relative to ground [Zhou et al., 2016]. The absolute

orientation at start point can be measured by compass or low-cost IMU. A more relevant approach for initialization is *place registration* introduced in chapter 2. Both absolute positions and orientations can be determined by indexing several images into geo-referenced database. In our current implementation, the absolute position is provided by GPS, the distance is also calculated from two GPS points. The initial orientation is measured using IMU.

We assume that the pose of first frame is $\boldsymbol{P}_0$ which is known with uncertainty. The position and orientation of first frame are noted as $\boldsymbol{C}_0$ and $\boldsymbol{R}_0$. $\boldsymbol{R}_0$ is rotation matrix. The absolute scale is determined by giving the distance D from first to second key frame. At this stage, the pose of second key frame is still not known. To obtain it, we first estimate the relative pose of second frame $(\boldsymbol{R}, \boldsymbol{\nu})$, relative to first frame. The Nister's 5-point algorithm [Nistér, 2004] with RANSAC scheme is applied for relative pose estimation, using the correspondences between first two images. The absolute pose of second frame is defined as $\boldsymbol{P}_1$. We note camera position as $\boldsymbol{C}_1$ and rotation matrix as $\boldsymbol{R}_1$ for second frame. The absolute pose of second frame can be obtained using the equations below:

$$\begin{cases} \boldsymbol{R}_1 = \boldsymbol{R}\boldsymbol{R}_0 \\ \boldsymbol{C}_1 = -D \cdot \boldsymbol{R}_1\boldsymbol{\nu} \end{cases} \tag{7}$$

With matches and poses of first and second images, the initial tie points can be reconstructed by triangulation. From third frame, the pose is estimated using 3D-2D correspondences derived from matching and tracking between the current frame and previous key frames. With the newly estimated image pose, more 3D points can be reconstructed. The same procedure will be repeated for all the coming new frames. The initial pose estimation and tie point reconstruction from third image is introduced in following section.

3.3.2 Initial estimation of poses and tie points

The methods for the estimation of pose and tie points are applied for images started from third image.

3.3.2.1 Pose estimation

The matching and tracking of current frame is done with the strategies proposed in section 3.2. Tracking *existing tie points* in current image, a set of *3D-to-2D* correspondences can be obtained to estimate the pose of current frame. For perspective camera, the pose estimation using 3D-to-2D correspondences, is *PnP* problem. The common solution for this problem is Direct Linear Transformation (DLT) [Sutherland, 1974; Ganapathy, 1984; McGlone et al., 2004], that solved the projection matrix $\mathbb{P}_{3\times4}$ using at least six 3D-2D correspondences. The DLT is usually used for camera calibration, because both intrinsic and extrinsic parameters of camera can be decomposed from the projection matrix $\mathbb{P}_{3\times4}$ at the same time. However, our

localization method is based on calibrated camera, hence only the poses of the moving camera $[R, C]$ need to be estimated.

In order to solve the six independent parameters of image pose, the minimal number of 3D-to-2D correspondences is three. This particular case is named *P3P* [Yuan, 1989; Wolfe et al., 1991; Yang, 1998; Gao et al., 2003] which is the minimal solution for *PnP* problem. The *P3P* algorithm can obtain up to four solutions, so we need a fourth object point to remove the ambiguity [Quan and Lan, 1999; Kneip et al., 2011b]. The correct solution of four solutions is the one in which makes the back projection of the fourth object point lie into image plane. Although other methods to solve the *PnP* problem have been presented by [Quan and Lan, 1999], *P3P* is the most efficient one. Recently, a more efficient solution of *P3P* was proposed by Kneip et al. [2011b], that reduced some intermediate derivation and leaded to a comparable accuracy at a lower computing cost in comparison to the classical methods such as the method proposed by Gao et al. [2003]. This method can estimate the absolute transformation in a single step, while most of existing solutions attempt to transform the 3D point into camera reference frame at first, then compute the position and orientation of the camera in world frame by aligning the two point sets. In this thesis, we use Kenip's implementation of *P3P* for pose estimation and a robust estimation is obtained by RANSAC scheme.

3.3.2.2 Triangulation

The triangulation refers to reconstruction of an tie point in 3D space, given its projections across two, or more images [Hartley and Zisserman, 2003]. In triangulation, we suppose that the projection matrix $\mathbb{P}_{3\times4}$ for every image has been estimated and the corresponding image points as shown in figure 3.5 have been obtained by matching and tracking, the problem is to compute the coordinates of tie point $\boldsymbol{X}$ in figure 3.3.

Each image point and camera center conduct a line through the object point in 3D space (*cf.* Fig 3.2), the equation can be written as shown in equation 5. If a set of corresponding image points in two, or more images can be found, they should intersect at a 3D point $\boldsymbol{X}$. Then we can obtain a list of linear equations as $\mathbf{x}_j = \boldsymbol{P}_j\boldsymbol{X}$. The $\boldsymbol{X}$ is resolved with the solution known as linear least squares [Hartley and Zisserman, 2003].

3.3.3 Key frames selection

In order to improve efficiency and accuracy of LBA, we select key frames over sequence. If current frame is not keyframe, the procedure will output the pose estimated using *P3P* and skip to the process of new coming frame. This kind of strategy, on the one hand, we can reduce the number of images which are needed to be optimized with LBA. On the other hand, larger baseline between neighbor images are kept which can obtain better results for triangulation of tie points. Many papers about key frame selection have been proposed for SFM or SLAM.

One popular method is Geometric Robust Information Criterion (GRIC) [Torr et al., 1999], which is often used for key frames selection in video based SFM [Pollefeys et al., 2002; Ahmed et al., 2010]. A score which is related to the goodness of fitting and the parsimony of the model between current image and latest key frame. The epipolar geometry (F-matrix) and homography (H-matrix) are usually used. If the score of F-GRIC is smaller than H-GRIC, it is a key frame [Pollefeys et al., 2002; Gibson et al., 2002; Thormählen et al., 2004]. However, theses kind of methods need to compute the H-matrix and F-matrix for every frame.

Mouragnon et al. [2006] proposed a simple way to choose the key frames, which is based on the number of matches between current frame and the first (M) and second ($M^{'}$) latest key frame. If M and $M^{'}$ are less than the given thresholds, a keyframe is added ($Th_M = 400, Th_{M'} = 300$). However, the threshold would change dynamically for different experiments. Seo et al. [2008] proposed the correspondence ratio between the number of frame-to-frame matches and the number of feature points. The ratio decreases with the moving of camera [Ahmed et al., 2010].

In this thesis, a new ratio τ, representing the percentage of *existing tie points* (Φ_0) in the tie points set (Φ) contained by current image :

$$\tau = \frac{\Phi_0}{\Phi}.$$

Where, $0 \leq \tau \leq 1$. With the moving of camera, τ becomes smaller and smaller. We set the threshold for the ratio as 0.3 in our experiments. Apart from this, two additional criterions are considered as well: the moving distance and the rotating angle relative to the latest key frame. We set the minimal distance between two keyframes as $1.5m$ and the minimal rotating angle as $10°$. If current frame meets one of the conditions, the frame is added as key frame.

3.3.4 Refinement with LBA

Many methods have been proposed such as EKF, Particle Filter, bundle adjustment etc, but bundle adjustment can obtain the most accurate results [Strasdat et al., 2010a; Ji and Yuan, 2016]. However, the classical bundle adjustment has high computing complexity, increasing quickly with the growing of images. In order to overcome this problem, Local Bundle Adjustment (LBA) was proposed [Zhang and Shan, 2001; Mouragnon et al., 2006; Eudes and Lhuillier, 2009].

In general, LBA process a sliding window of N frames in which the latest n frames ($n < N$) are newly estimated and the other $N - n$ frames are inherited from previous steps. Figure 3.7 depicts the procedure of the incremental approach in LBA, where $N = 5$, $n = 2$. In current step, there are $N - n$ frames that have prior knowledges of their poses and covariance matrix since they have been resolved by LBA in the previous steps. For instance, the poses of frames $2, 3, 4$ in figure 3.7 have been estimated at first step. When they are used in second step, they are regarded as parameters with prior knowledges, which will provide constrained equations in

cost function. Whereas the poses of frames 5 and 6 are unknowns without any constraints.

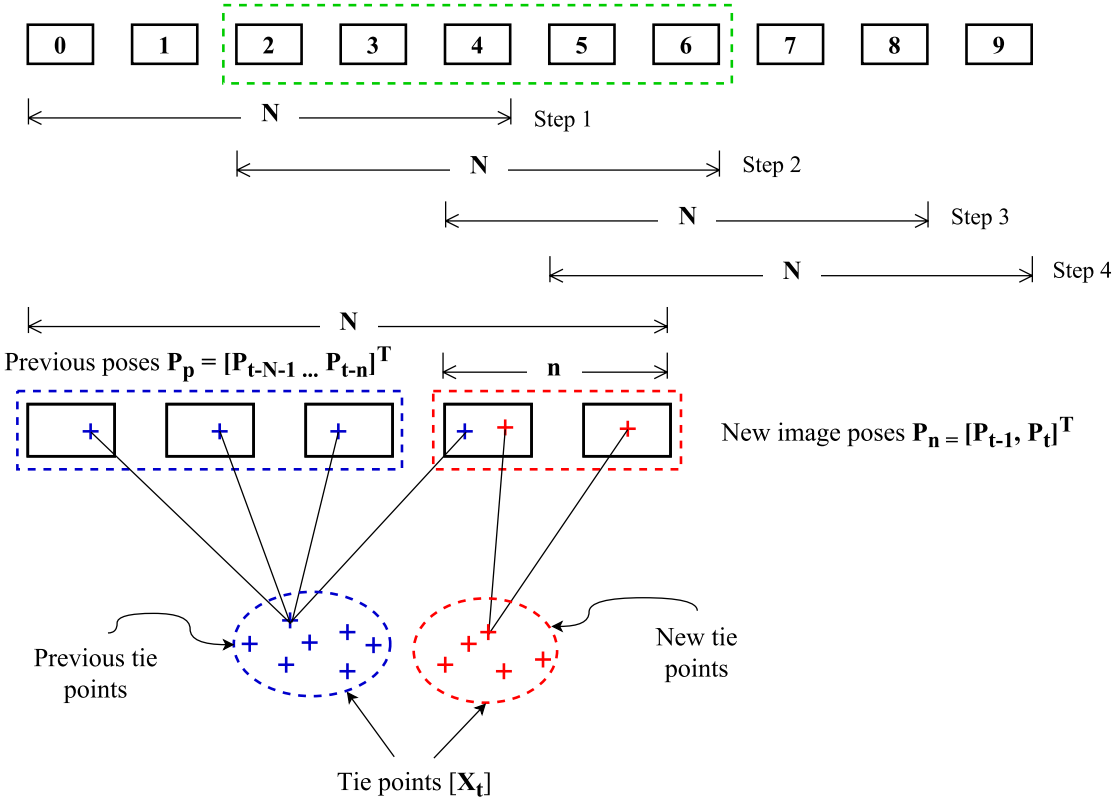


Figure 3.7: Schematic of the LBA processing procedure. The zoom-up digram at left-bottom presents the different parameters in second step, marked with green dotted rectangle.

3.3.4.1 Notations and definition

For better presentation, we define some basic notations.

Basic notations As we denoted before, t is time and the index of frame. $\boldsymbol{P}$ is pose of frame. $\boldsymbol{X}$ is 3D tie point. Some basic arguments for LBA are defined as:

- $\boldsymbol{P}_n$ is the vector of new image poses in current step.
- $\boldsymbol{P}_p$ consists of the poses inherited from previous steps.
- $\boldsymbol{X}_t$ is the vector of 3D tie points.
- $\boldsymbol{v}_t$ is the vector of back projection error.
- $\boldsymbol{\Sigma}_t$ is covariance matrix of the measurements of image points.
- $\boldsymbol{v}_p$ is the vector of residuals for the poses with respect to $\boldsymbol{P}_p$.
- $\boldsymbol{P}_p^0$ consists of the prior estimates of $\boldsymbol{P}_p$.

- $\boldsymbol{\Sigma}_p$ is the covariance matrix of $\boldsymbol{P}_p^0$.

Notations of parameters In LBA, $\boldsymbol{P}_p, \boldsymbol{P}_n, \boldsymbol{X}_t$ are parameters and they can be presented as expression 8, where N is the size of sliding window and n is the number of new frames, the elements in each parameter vector are like:

$$\begin{aligned} \boldsymbol{P}_p &= [\boldsymbol{P}_{t-N-1} \quad ... \quad \boldsymbol{P}_{t-n}]^T \\ \boldsymbol{P}_n &= [\boldsymbol{P}_{t-n+1} \quad ... \quad \boldsymbol{P}_t]^T \\ \boldsymbol{X}_t &= [... \quad \boldsymbol{X}_i \quad ...]^T \end{aligned} \tag{8}$$

In this thesis, $\boldsymbol{\beta}$ is defined as the symbol of parameters in LBA, which is a combination of $\boldsymbol{P}_p, \boldsymbol{P}_n, \boldsymbol{X}_t$, that is:

$$\boldsymbol{\beta} = [\boldsymbol{P}_p, \boldsymbol{P}_n, \boldsymbol{X}_t]^T$$

3.3.4.2 Mathematics of LBA

Special LBA at first step The special case in LBA is the first step (*cf.* step 1 in Fig 3.7). All images in processing window are optimized with LBA first time, so there are no extra constraints. The back projection errors are computed by:

$$\boldsymbol{v}_t = F(\boldsymbol{P}, \boldsymbol{X}_t) - \mathbf{x}_t,$$

in terms of equation 6, where $\mathbf{x}_t$ is the vector of measurements corresponding to image position of points X_i. Our aim in this case is to minimize the weighted sum of squared back projections, thus the cost function becomes:

$$f(\boldsymbol{\beta}) = \frac{1}{2}\boldsymbol{v}_t^T \boldsymbol{\Sigma}_t^{-1} \boldsymbol{v}_t,$$

where,

$$\Sigma_t = \sigma_t^2 \boldsymbol{I}$$

and

σ_t : standard deviation of 2D interest point detection

This is conventional bundle adjustment and nonlinear least square is employed to solve the solution [Triggs et al., 2000].

To deal with the outliers in bundle adjustment, a loss function is usually applied to reduce the influence. In this case, the cost function is :

$$f(\boldsymbol{\beta}) = \frac{1}{2}\varphi_t \boldsymbol{v}_t^T \boldsymbol{\Sigma}_t^{-1} \boldsymbol{v}_t$$

φ_t is a loss function which is a scalar function used to reduce the influence of outliers on the solution of nonlinear least square problems. The cost for large residuals therefore is reduced using a loss function. This leads to outlier terms getting down-weighted so they do not overly affect the final solution [Agarwal et al.].

General LBA In order to separate the new poses and inherited poses in LBA, we rewrite $F(\boldsymbol{P}, \boldsymbol{X})$ as $F(\boldsymbol{P}_p, \boldsymbol{P}_n, \boldsymbol{X})$, so the back projection errors for all the tie points $\boldsymbol{v}_t$ is noted as:

$$\boldsymbol{v}_t = F(\boldsymbol{P}_p, \boldsymbol{P}_n, \boldsymbol{X}_t) - \mathbf{x}_t \tag{9}$$

In particular, we consider the constraints for $\boldsymbol{P}_p$, which are generated depending on their uncertainty estimated in previous steps. To integrate the constraints, a set of linear error equations are defined for cost function together with the back projection errors, weighted according to their covariance matrix Σ_p. The error equation array is:

$$\boldsymbol{v}_p = \boldsymbol{P}_p - \boldsymbol{P}_p^0, \tag{10}$$

where, $\boldsymbol{P}_p^0$ is the estimated poses from previous steps.

Supposing that there is no covariance between image residuals and the previously estimated poses, the parameters are resolved by minimize the following cost function:

$$f(\boldsymbol{\beta}) = \frac{1}{2}(\boldsymbol{v}_t^T \boldsymbol{\Sigma}_t^{-1} \boldsymbol{v}_t + \boldsymbol{v}_p^T \boldsymbol{\Sigma}_p^{-1} \boldsymbol{v}_p) \tag{11}$$

There is no closed-form solution to a non-linear least squares problem. Instead, numerical algorithms are used to find the value of the parameters that minimize the cost function. Most algorithms need initial values for the parameters. Then, the parameters are refined iteratively. In the most used algorithms, the model is linearized by approximation to a first-order Taylor series expansion at each iteration. The Taylor expansion is employed at point $[\overline{\boldsymbol{P}}_p, \overline{\boldsymbol{P}}_n, \overline{\boldsymbol{X}}_t]$. Thus, the linear equations of 9 and 10 can be obtained:

$$\begin{bmatrix} \boldsymbol{v}_p \\ \boldsymbol{v}_t \end{bmatrix} = \underbrace{\begin{bmatrix} \overline{\boldsymbol{P}}_p - \boldsymbol{P}_p^0 \\ F(\overline{\boldsymbol{P}}_p, \overline{\boldsymbol{P}}_n, \overline{\boldsymbol{X}}_t) - \mathbf{x}_t \end{bmatrix}}_{\boldsymbol{y}} + \underbrace{\begin{bmatrix} I & 0 & 0 \\ \frac{\partial F}{\partial \boldsymbol{P}_p} & \frac{\partial F}{\partial \boldsymbol{P}_n} & \frac{\partial F}{\partial \boldsymbol{X}_t} \end{bmatrix}}_{\boldsymbol{J}} \underbrace{\begin{bmatrix} \delta_{\boldsymbol{P}_p} \\ \delta_{\boldsymbol{P}_n} \\ \delta_{\boldsymbol{X}_t} \end{bmatrix}}_{\boldsymbol{\delta}_\beta} \tag{12}$$

Where:

$\overline{\boldsymbol{P}}_p, \overline{\boldsymbol{P}}_n, \overline{\boldsymbol{X}}_t$: approximate estimation of the parameters

$\delta_{\boldsymbol{P}_p}, \delta_{\boldsymbol{P}_n}, \delta_{\boldsymbol{X}_t}$: corrections to current values of parameters.

$\boldsymbol{y}$: differences between observations and the predicts estimated using initial parameters

$\boldsymbol{J}$: Jacobian matrix

$\boldsymbol{\delta}_\beta$: corrections of the parameters

(13)

In each iterative step, the estimates of the parameter corrections, noted as $\hat{\boldsymbol{\delta}}_\beta$, are resolved from normal equation below:

$$\underbrace{(\boldsymbol{J}^T \boldsymbol{W} \boldsymbol{J})}_{\boldsymbol{H}} \hat{\boldsymbol{\delta}}_\beta = -\boldsymbol{J}^T \boldsymbol{W} \boldsymbol{y} \tag{14}$$

with:

$$\boldsymbol{W} = diag(\boldsymbol{\Sigma}_p^{-1}, \boldsymbol{\Sigma}_t^{-1}) \tag{15}$$

where, $\boldsymbol{H}$ is normal matrix. If we denote the corrections at current step k as $\hat{\boldsymbol{\delta}}_{\beta}^{k}$, the parameters are updated for next step as:

$$\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^{k} + \hat{\boldsymbol{\delta}}_{\beta}^{k}.$$

The iteration is performed until the thresholds are reached.

To explore the features of LBA, a LBA graph about the images and tie points are simulated in figure 3.8. There are five images with ten tie points in this network graph. The poses of three images are estimated beforehand with their uncertainty (marked with blue squares) and two new images (marked with gray squares) in this step. The graph in figure 3.8 illustrates the relations between the tie points and images. Figure 3.9 shows the structure of Jacobian and normal matrix for the LBA graph in figure 3.8. Each block of rows in the Jacobian matrix contains the contributions of each observation for relevant parameter blocks[Triggs et al., 2000]. For conventional bundle adjustment, each row in Jacobian is related to one pose and one tie point. However, some blocks in Jacobian matrix are only linked to poses in the case of LBA, seeing the top-left block in figure 3.9(a). This is caused by the additional error equations about $\boldsymbol{P}_p$. In Jacobian matrix, each sub-matrix in top-left block is a 6×6 identity matrix. The interesting thing is that, the additional equations don't change the structure of normal matrix (Hessian matrix), as shown in figure 3.9(b). However, the uncertainties of $\boldsymbol{P}_p$ have been integrated into the corresponding parameter blocks about $\boldsymbol{P}_p$ via the weight matrix $\boldsymbol{W}$.

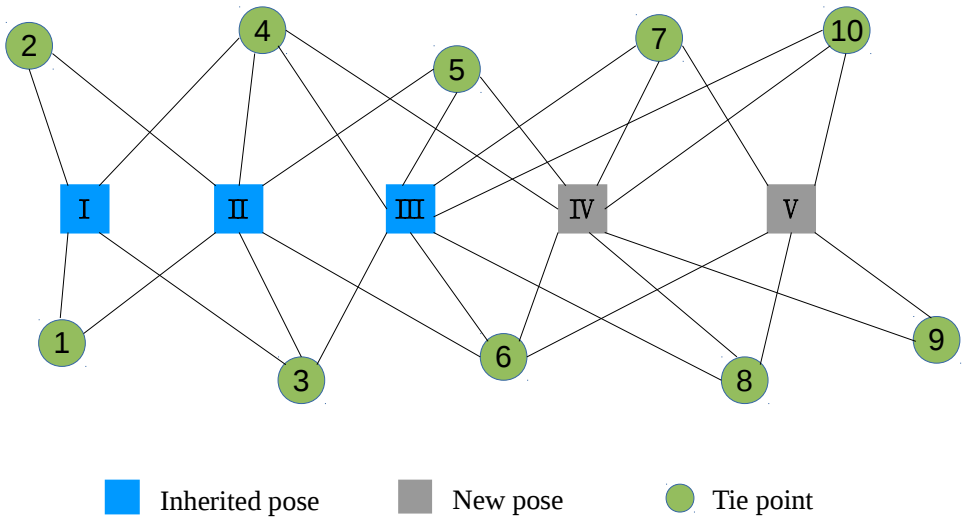
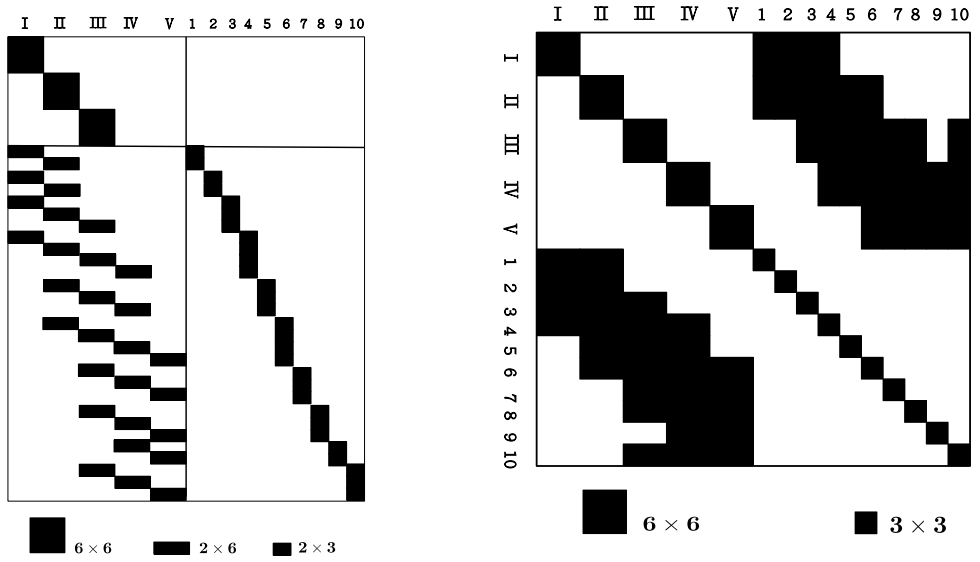


Figure 3.8: LBA graph.

We attempt to update uncertainty of $\boldsymbol{P}_p$ and estimate the uncertainty of new poses $\boldsymbol{P}_n$ after LBA. It is known that the covariance matrix of the parameters can be obtained from normal matrix $\boldsymbol{H}$, which is $\boldsymbol{H}^{-1}$. In bundle adjustment, the dimension of the normal matrix is dominated by the number of tie points. As shown in figure 3.9(b), the dimension of normal matrix can be calculated by $6 \times N + 3 \times M$, where N is the number of images and M is the number of tie



(a) Jacobian matrix in LBA.

(b) Normal matrix in LBA.

Figure 3.9: The structure of Jacobian and normal matrix in LBA as shown in figure 3.8. The dark boxes represent the non-zero blocks in matrix.

points. In LBA, N is fixed, thus, the computation of covariance matrix is dominated by M and $M \gg N$ in practice. According to the fact that the covariance matrix of image poses is the top-left block of $\boldsymbol{H}^{-1}$, it is natural to calculate the top-left part of $\boldsymbol{H}^{-1}$. The Schur complement is used to estimate the pose covariance matrix [Triggs et al., 2000].

3.3.4.3 Variance Component Estimation (VCE)

We assume that all feature points have same precision. The standard deviation of measurements is σ_t. Thus, the covariance matrix for all interest points used in LBA, is $\Sigma_t = \sigma_t^2 \boldsymbol{I}$. The existing LBA methods regard σ_t as one pixel Eudes and Lhuillier [2009]. However, the precision of SIFT detector might be better than one pixel [Lowe, 2004]. Although the variation of σ_t does not influence the estimated values of parameters, it affects the scale of error ellipsoids estimated for poses. The incorrectly scaled uncertainties would generate inaccurate searching areas for matching, tracking and landmark matching which will be introduced in chapter 4 & 5. In general, it is difficult to obtain value of σ_t in advance. With the approach proposed by Luxen [2003], we estimate the variance scale using posterior variance component estimation. As we discussed at the beginning of this section, the first step of LBA is conventional bundle adjustment. We set Σ_t as identity matrix at first. After bundle adjustment at first processing window, the variance of interest points can computed by following equation :

$$\hat{\sigma_t}^2 = \frac{\hat{\boldsymbol{v}_t}^T \hat{\boldsymbol{v}_t}}{r} \tag{16}$$

$\hat{v_t}$: residual vector after adjustment

r : the number of redundant observations.

Then we use the estimate of σ_t for other LBA steps.

3.3.5 Experiments using monocular images

To test the proposed localization method with single camera, the data acquired by STEREOPOLIS [Paparoditis et al., 2012] are used for experiment. The ground truth is measured by a precise navigation system, whose accuracy is up to centimeters. Images are captured by a calibrated front looking camera. The focal length of the camera is $10\ mm$, the image size is 1920×1024 pixels. The FOV of the camera is $70°$ in horizontal and $42°$ in vertical.

(a) (b) (c)

Figure 3.10: Example of feature extraction and matching. (a) Image acquired by STEREOPOLIS. (b)SIFT feature points. (c) Pair-wise matches with previous frame.

The feature extraction, matching and tracking for monocular sequence is done with the methods presented in section 3.2. Figure 3.10(b) shows the interest point detected by SIFT and figure 3.10(c) depicts the pairwise matches between the current image with the latest key frame.

There are 270 key frames in a trajectory of $750m$. Figure 3.11 demonstrates the vertical view of the trajectory recovered using the proposed method and the ground truth acquired by combining navigation system. We start the operation from the left-top in the path marked with black box in figure 3.11. The localization is very accurate at beginning and the drift is pretty small. However, it is growing over time.

In order to analyze the change of drift over time, we divide the drift into errors at depth direction and lateral errors. The lateral error is defined in a plane that is orthogonal to depth direction and parallel with image plane. To compute depth error and lateral error, we transform image position from world coordinate system to camera space:

$$\boldsymbol{\nu}_t = -\boldsymbol{R}_t \boldsymbol{C}_t, \tag{17}$$

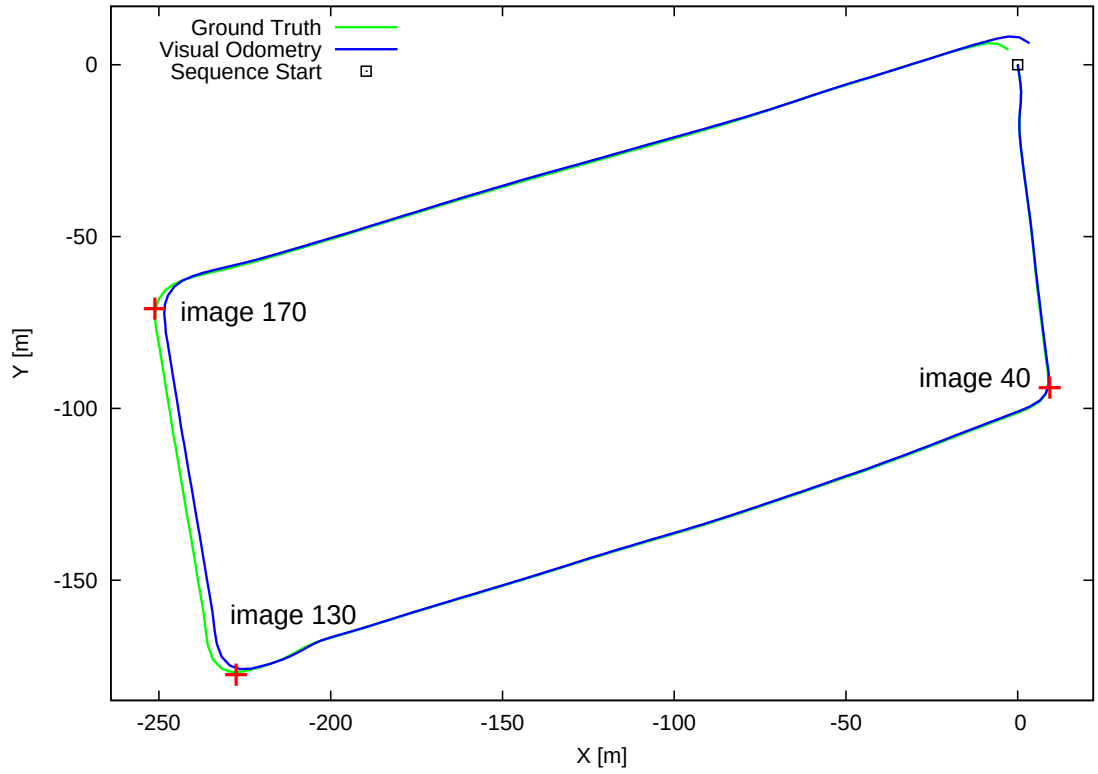


Figure 3.11: Trajectory estimated by vision based localization and the ground truth.

where, $\boldsymbol{R}_t$ are rotation matrix from world to camera space and $\boldsymbol{C}_t$ is the position of camera center in world. Then we define *lateral error* and *depth error*:

- **Lateral error.** error in xy plane in camera space.
- **Depth error.** error at z axis in camera space.

In general, when the moving direction of vehicle is consistent with the depth direction of camera, the errors along depth direction dominate the drift because of the depth uncertainty. The errors are shown in diagram 3.12, where the red line represents the lateral errors and the blue line presents the errors in depth. Both of them are very small at the beginning and grow over time, but errors in depth grow faster than the lateral errors. However, this is not always the case, the errors in depth reduces quickly around image 130 and then increase after image 170, while the change of errors in image plane is inverse, (*cf.* Fig 3.12). By combining the results shown in figure 3.11 and 3.12, we find the image 130 and 170 are around the second and third turnings from start point and the change of camera direction between key frames are heavy. In this case, the difference between depth direction and moving direction is large and they are not consistent any more, so the drift at depth direction is transited to lateral direction. The similar situation occurs at the first turning, but the change of depth errors is not as big as the second turning due to the flat turning.

We estimate the uncertainties for the estimated poses of image and also consider the uncertainty propagation over sequence in LBA. The error ellipsoids are estimated to present the uncer-

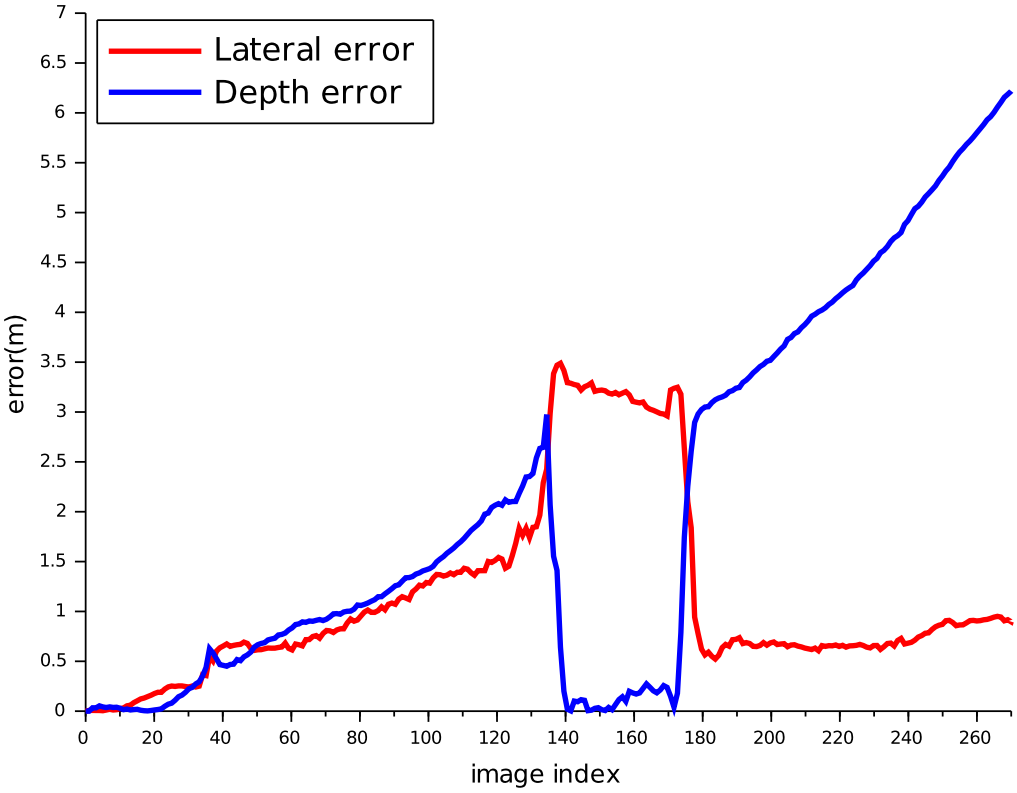


Figure 3.12: The lateral and depth errors of localization.

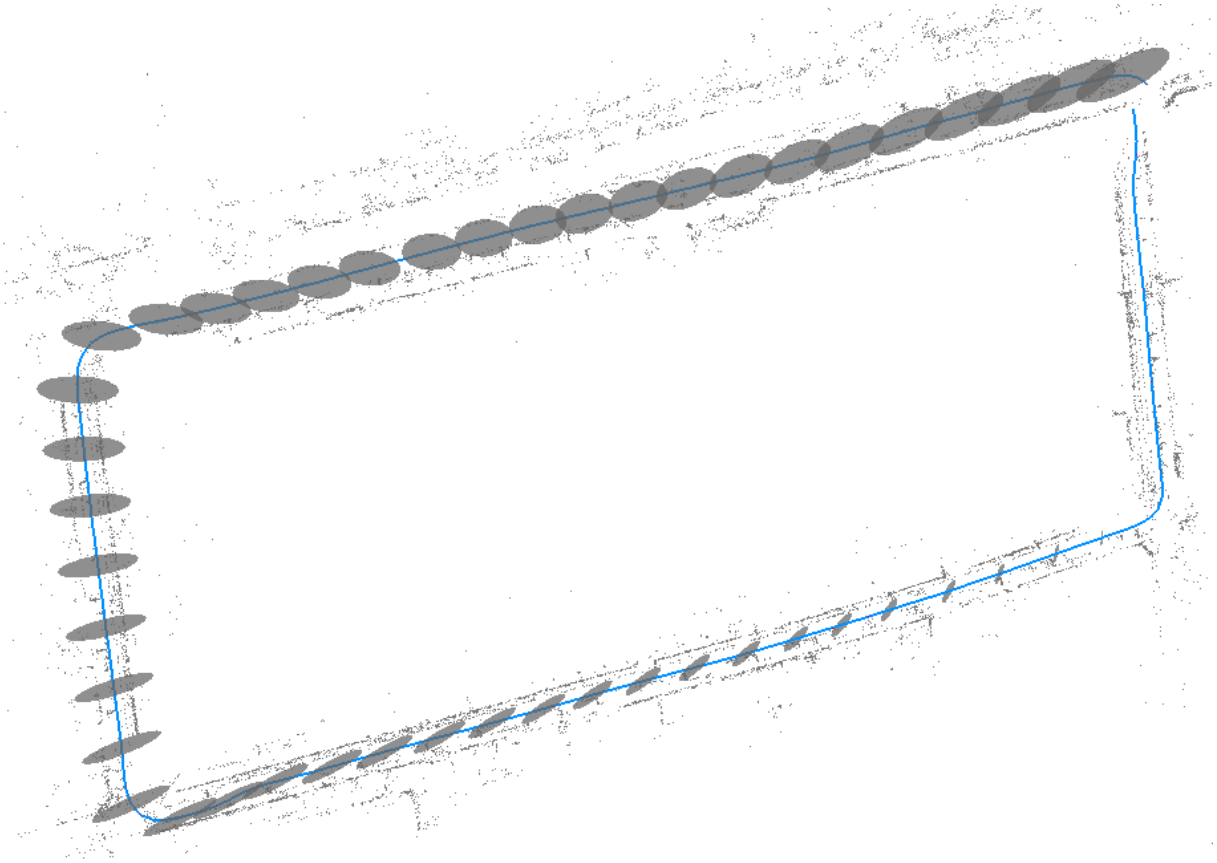


Figure 3.13: Uncertainty propagation. Each error ellipsoid is exaggerated three times.

tainties of locations, which are computed from the covariance matrix of image pose. In this experiment, the confident level of error ellipsoid is set as 99 %. The direction of the major axis of error ellipsoid represents the major uncertainty. As shown in figure 3.13, the major axis of error ellipsoid points at the depth direction except for the images between the second turning and third turning, where the major axis directions are consistent with the lateral direction. This trend corresponds the diagram presented in figure 3.12.

3.4 Multi-camera based localization

Many papers study the SLAM or visual odometry based on stereo cameras [Olson et al., 2001; Nister et al., 2004; Milella and Siegwart, 2006; Geiger et al., 2011; Engel et al., 2015]. In stereo case, the absolute scale can be given by known length of the baseline. The scale doesn't need to be determined externally in comparison to monocular scheme. Additionally, more observations can be obtained for the tie points that can improve accuracy of the triangulation, then enhance the quality of pose estimation. As we introduced in chapter 2, vision based localization can get benefits from the large Field of View (FOV) which can provide longer tracks and more informative observations of environment to improve the robustness and accuracy of pose estimation. This is the motivation of using omnidirectional cameras for localization [Scaramuzza and Siegwart, 2008; Silpa-Anan et al., 2005].

The problem of omni-directional camera is the low angular resolution which is not preferable for large scale visual odometry [Zhang et al., 2016]. In this thesis, we use multi-camera system which is combined by several perspective cameras at different directions. It can acquire images with larger FOV and have high angular resolution at the same time. The stereo rig is categorized as multi-camera rig as well. In this section, we propose our localization method using multi-camera system. First of all, the rigorous projection model of multi-camera is studied. Then we present how to estimate the initial values of image poses and tie points. At last, the LBA is extended to adapt the multi-camera cases.

3.4.1 Rigorous projection model

In the case of multi-camera based localization, the key is still to estimate the pose of the moving entity (e.g. robot, vehicle) in real time. In monocular case, if we estimate the pose of every frame, the pose of the entity can be computed by transforming the pose of image frame to the entity body rigidly. The offset and orientation of the camera to entity body coordinate system can be measured beforehand and they are fixed during localization. For multi-camera configurations, each camera has different pose at a time step, but the relative relations between these poses are fixed. Our aim is to obtain the pose of entity directly. To achieve this, we define a local coordinate system on entity body. The rotation and translation of this local system represent the pose of entity in world. Every camera can be presented in this local system relatively.

3.4.1.1 viewpoint

In this thesis, the local system is called *viewpoint*, and the position and orientation of every camera can be transformed from pose of viewpoint. Figure 3.14 shows the relationship between viewpoint and cameras. The view point can be put at anywhere in entity body. Denote $\boldsymbol{\Gamma}$ as the rigid transformation vector from view point to camera center. Γ_i expresses the transformation from viewpoint to camera i. Each $\boldsymbol{\Gamma}$ contains six parameters. The first three for translation and the rest are the rotation angles. Thus, the offset vector $\boldsymbol{T}_i$ and the rotation matrix $\boldsymbol{R}_i$ for camera i can be generated from $\boldsymbol{\Gamma}_i$.

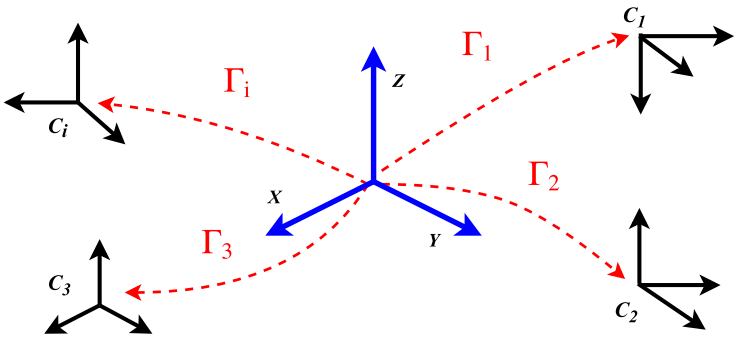


Figure 3.14: General concept of viewpoint. The blue coordinate system expresses local system of viewpoint in world. The red arrows present the offsets and rotation from every camera to the coordinate system of viewpoint.

3.4.1.2 Pose of camera

To estimate the pose of view point $\boldsymbol{P}_t$, the data perceived by cameras at time t should be associated. We note the orientation and position of viewpoint at time step t as $(\boldsymbol{R}_t, \boldsymbol{C}_t)$ (*cf.* Fig 3.15). The relations from view point to camera i is $\boldsymbol{\Gamma}_i$ which is a rigid transformation. The rotation matrix and translation of camera i are $\boldsymbol{R}_i, \boldsymbol{T}_i$.

Since we know the transformation from camera to view points, the position and orientation of camera in world can be computed by the formula below:

$$\begin{aligned} \boldsymbol{R}_t^i &= \boldsymbol{R}_i \boldsymbol{R}_t \\ \boldsymbol{C}_t^i &= \boldsymbol{R}_t^T \boldsymbol{T}_i + \boldsymbol{C}_t \end{aligned} \tag{18}$$

where, $\boldsymbol{R}_t^i$ is the rotation matrix for camera i at time t and $\boldsymbol{C}_t^i$ is the position of camera center relative to absolute system . The vector of pose of camera i, denoted as $\boldsymbol{P}_t^i$, is composed from $\boldsymbol{R}_t^i$ and $\boldsymbol{C}_t^i$.

3.4.1.3 Projection model

In our research, we focus on the perspective cameras. The camera model for each camera has been proposed in equation 4. If we know $\boldsymbol{P}_t^i$, which is the pose of camera i at time t, then

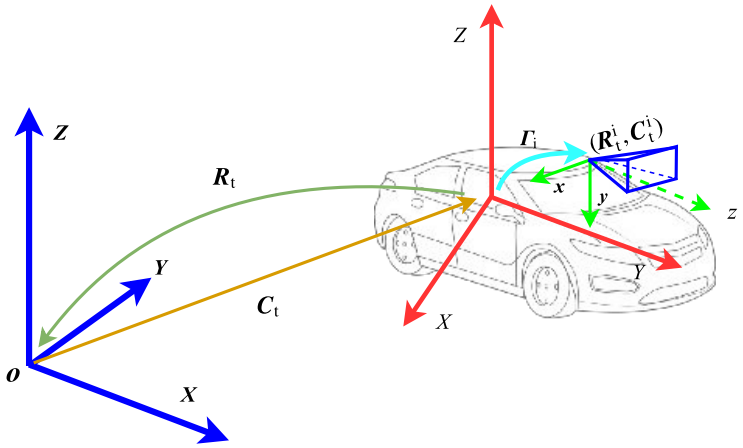


Figure 3.15: Estimation of camera pose. The blue XYZ frame is the world coordinate system, $\boldsymbol{C}_t$ presents the position and $\boldsymbol{R}_t$ is the orientation of viewpoint at time t. The pose of camera i can be computed by rigid transformation from view point.

the orientation and position of the camera, noted as $\boldsymbol{R}_t^i$ and $\boldsymbol{C}_t^i$, can be obtained. With one 3D world point $\boldsymbol{X}$, its projection in image can be calculated by :

$$\mathbf{x}_t^i = \boldsymbol{K}_i \boldsymbol{R}_t^i [\boldsymbol{I}| - \boldsymbol{C}_t^i] \boldsymbol{X},$$

where $\mathbf{x}_t^i$ is the coordinates in pixels into image and $\boldsymbol{K}_i$ is the calibration matrix of camera i. Combining equation 18, projecting the world point $\boldsymbol{X}$ into image captured by camera i at time t is obtained by:

$$\mathbf{x}_t^i = \boldsymbol{K}_i \boldsymbol{R}_i \boldsymbol{R}_t [\boldsymbol{I}| - (\boldsymbol{R}_t^T \boldsymbol{T}_i + \boldsymbol{C}_t)] \boldsymbol{X}, \tag{19}$$

where, $\boldsymbol{R}_t, \boldsymbol{C}_t$ are the rotation matrix and position of viewpoint at time t, which can be considered as the real time pose of vehicle or robot.

As the parameters for rigid transformation of camera i is $\boldsymbol{\Gamma}_i$, referencing equation 6, a simplified projection model for multi-camera model is:

$$\mathbf{x}_t^i = \boldsymbol{K}_i \boldsymbol{R}_i \boldsymbol{R}_t [\boldsymbol{I}| - (\boldsymbol{R}_t^T \boldsymbol{T}_i + \boldsymbol{C}_t)] \boldsymbol{X} = F(\boldsymbol{P}_t, \boldsymbol{\Gamma}_i, \boldsymbol{X}). \tag{20}$$

The equation 20 doesn't represent the pinhole projection any more. It integrates the rigid transformation with the perspective projection.

3.4.2 Parameters initialization

In equation 20, $\mathbf{x}_t^i$ is an image point, $\boldsymbol{\Gamma}_i$ is measured in system calibration before localization. The unknowns are viewpoint pose $\boldsymbol{P}_t$ and tie point $\boldsymbol{X}$.

3.4.2.1 Pose estimation for viewpoint

As we presented in section 3.3.2, solving the pose using a set of 2D-3D correspondences for perspective image is a PnP problem. The transformation from camera to view point is fixed and the parameters are calibrated beforehand. A solution for initial pose estimation of viewpoint is to estimate the initial pose of image using PnP algorithms. However, there are too few 2D-3D correspondences in some cases using image points of single camera due to restriction of FOV. To improve the robustness of pose estimation, we desire to use 2D-3D correspondences in a larger field which means the image points might locate in different images captured by different camera at one time step. In this case, the projection model for multi-camera system is not perspective.

To estimate the pose, a Generalized Camera Model (GCM) has been investigated by Grossberg and Nayar [2001]. The solution for pose estimation using GCM is a Non-Perspective n Points (NPnP) problem [Chen and Chang, 2004]. The minimal solution for PnP is a P3P problem using three 3D-2D correspondences. Similarly, the minimal solution for NPnP problem is also resolved with three 3D-2D correspondences, that is NP3P Chen and Chang [2004]; Nistér and Stewénius [2007]. Figure 3.16 illustrates the difference between the classical P3P and NP3P for generalized problem.

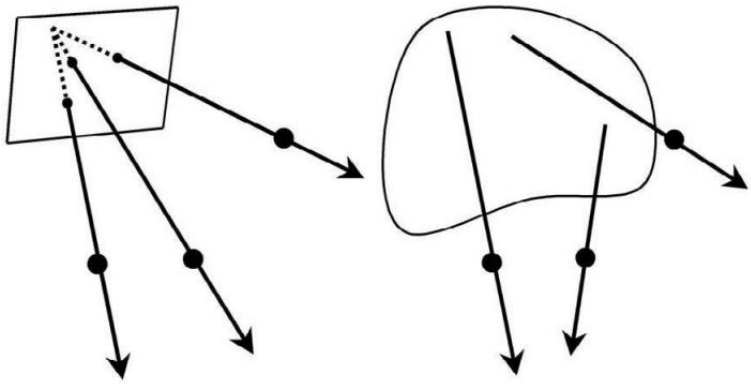


Figure 3.16: Left: the P3P problem for perspective projection. Right: NP3P problem, solving the pose of three arbitrary rays emanating from a generalized camera geometry and meeting three world points. The camera model can be any arbitrary projection [Nistér and Stewénius, 2007].

In practice, it is difficult to solve the problem of NP3P in algebra which can generate up to 8 possible solutions from a numerical 8-th order polynomial [Chen and Chang, 2004; Nistér and Stewénius, 2007; Kneip et al., 2013]. In this case, a number of methods were proposed to solve the NPnP problem using iterative solutions lied on global optimization [Chen and Chang, 2004; Tariq and Dellaert, 2004; Schweighofer and Pinz, 2008]. These methods need initial values for the pose, but it is usually difficult to obtain accurate initialization easily. In addition, these kinds of methods are computationally expensive due to the iteration. Ess et al. [2007] achieved a non-iterative linear method to solve NPnP problem, but the complexity of the algorithm is

quadratic in the number of the correspondences. Toward this direction, a more efficient solution was proposed by Kneip et al. [2013], that solve the NPnP problem with non-iterative solution with linear computing complexity.

In this thesis, we apply the Kneip's solution in a RANSAC scheme to achieve the pose of each viewpoint [Kneip et al., 2013]. The first step is still to find the existing tie points in current viewpoint using the strategy proposed in section 3.2. Then the projections of these 3D tie points in images are obtained using the matching and tracking. At last, the pose of the viewpoint is resolved with a set of 3D world points and their measurements in images.

Compared with PnP based pose estimation, the NPnP method can get benefits from larger FOV. On one hand, more informative points can be observed in a larger view. On the other hand, the tracking of tie points can be obtained in a longer period. These advantages can make the pose estimation more robust and accurate, especially in some challenging situations.

3.4.2.2 Tie points reconstruction

Some new tie points should be exploited. With the matching and tracking for multi-camera cases were proposed in section 3.2, the image points for new tie points can be obtained. In order to reconstruct the position of the tie point in world, we need to know the pose for each image. As the pose of viewpoint $\boldsymbol{P}_t$ is estimated with the method introduced in previous section, the pose of every camera can be estimated using equation 18. In this case, the new tie points can be reconstructed with triangulation proposed in section 3.3.2.2.

3.4.3 LBA for multi-cameras based localization

3.4.3.1 Back projection error

For single camera with perspective projection model the back projection error is calculated by $\boldsymbol{v}_t = F(\boldsymbol{P}_t, \boldsymbol{X}_t) - \mathbf{x}_t$, regarding the notation of pinhole projection model in equation 6. Considering the definition of projection model of multi-camera system in section 3.4.1.3, the back projection error in camera i can be formulated by :

$$\boldsymbol{v}_t^i = F(\boldsymbol{P}_t, \boldsymbol{\Gamma}_i, \boldsymbol{X}) - \mathbf{x}_t^i, \tag{21}$$

where $\boldsymbol{v}_t^i$ is the residual vector of tie point $\boldsymbol{X}$ at time t in camera i and $\mathbf{x}_t^i$ is the image point in image for $\boldsymbol{X}$. $F(\boldsymbol{P}_t, \boldsymbol{\Gamma}_i, \boldsymbol{X})$ is the projection model from 3D to images of every camera.

We extend the original LBA proposed by Mouragnon et al. [2006]; Eudes and Lhuillier [2009] to adopt multi-camera model. The procedure of LBA approach is similar with monocular case, but the minimal processing unit is viewpoint. Only the images in latest N viewpoints are considered for bundle adjustment and there are n new viewpoints out of N in each step. In each viewpoint, the unknowns are $\boldsymbol{P}_t$ and $\boldsymbol{X}$. The parameters of rigid transformation for cameras, noted as Γ^0,

are calibrated beforehand. In our approach, we consider the uncertainties of Γ^0 and $N-n$ poses estimated in previous steps.

In LBA process, we divide the poses of viewpoint into two categories: the new poses and inherited poses. As the above-mentioned notation, there are n new viewpoints and $N-n$ inherited poses in one processing window. Because the uncertainty of pose is estimated after LBA, we can know the uncertainty of the $N-n$ inherited poses. Although the inherited poses are also the parameters in LBA, they can be optimized inside their uncertainty areas. For better expression, $\boldsymbol{P}_p$ is noted as the inherited poses and $\boldsymbol{P}_n$ is the new poses, thus the error equation of back projection error can be written as:

$$\boldsymbol{v}_t^i = F(\boldsymbol{P}_p, \boldsymbol{P}_n, \boldsymbol{\Gamma}_i, \boldsymbol{X}) - \mathbf{x}_t^i \tag{22}$$

where:

$\boldsymbol{v}_t^i$: back projection error of X in camea i

$\boldsymbol{P}_p$: inherited poses of viewpoints in window

$\boldsymbol{P}_n$: new poses of the viewpoints

$\boldsymbol{\Gamma}_i$: rigid transformation parameters for camera i

$\boldsymbol{X}$: 3D tie points in world

$\mathbf{x}_t^i$: measurements of tie point in camera i

3.4.3.2 Modeling the constraints

Equation 22 models the back projection errors for each camera in multi-camera system. Now the issue is how to consider the error propagation of $\boldsymbol{P}_p$ and $\boldsymbol{\Gamma}$ in LBA approach. The constraints for inherited poses are same with monocular case. The covariance matrix of $\boldsymbol{P}_p$ is noted as $\boldsymbol{\Sigma}_p$. A linear error equation is defined (see equation 10). A similar method is employed to generate the error equations for rigid transformation. The covariance matrix for $\boldsymbol{\Gamma}$ is denoted as $\boldsymbol{\Sigma}_\Gamma$ and the pre-measured values of $\boldsymbol{\Gamma}$ is noted as $\boldsymbol{\Gamma}^0$, a linear error equation is defined as following:

$$\boldsymbol{v}_\Gamma = \boldsymbol{\Gamma} - \boldsymbol{\Gamma}^0 \tag{23}$$

3.4.3.3 Cost function

Combining the error equations 10, 23 and 22, the equation arrays are :

$$\begin{cases} \boldsymbol{v}_p = \boldsymbol{P}_p - \boldsymbol{P}_p^0 \\ \boldsymbol{v}_\Gamma = \boldsymbol{\Gamma} - \boldsymbol{\Gamma}^0 \\ \boldsymbol{v}_t^i = F(\boldsymbol{P}_p, \boldsymbol{P}_p, \boldsymbol{\Gamma}_i, \boldsymbol{X}) - \mathbf{x}_t^i \end{cases} \tag{24}$$

The aim is to obtain optimal solution for the parameters that makes the errors computed by equation 24 be minimal. We minimize the weighted sum of squared residuals which is written

as:

$$[\hat{\boldsymbol{P}}_p, \hat{\boldsymbol{P}}_n, \hat{\boldsymbol{\Gamma}}_i, \hat{\boldsymbol{X}}] = argmin\{\frac{1}{2}({\boldsymbol{v}_t}^T \boldsymbol{\Sigma}_t^{-1} \boldsymbol{v}_t + \boldsymbol{v}_p^T \boldsymbol{\Sigma}_p^{-1} \boldsymbol{v}_p + \boldsymbol{v}_\Gamma^T \boldsymbol{\Sigma}_\Gamma^{-1} \boldsymbol{v}_\Gamma)\} \tag{25}$$

where:

$[\hat{\boldsymbol{P}}_p, \hat{\boldsymbol{P}}_n, \hat{\boldsymbol{\Gamma}}_i, \hat{\boldsymbol{X}}]$: optimal solutions of parameters.

$\boldsymbol{v}_t$: residuals of back projection errors for all tie points.

$\boldsymbol{v}_p$: residuals of all inherited poses.

$\boldsymbol{v}_\Gamma$: residuals rigid transformation for all cameras

$\boldsymbol{\Sigma}_t$: covariance matrix for all images points

$\boldsymbol{\Sigma}_p$: covariance matrix for inherited poses

$\boldsymbol{\Sigma}_\Gamma$: covariance matrix for all the measured rigid transformation parameters

Equation 25 is called the cost function. The optimal results of the parameters are obtained when the cost function meets the minimal value.

3.4.3.4 Solution and error propagation

As proposed in section 3.3.4, nonlinear least squares approaches the optimal position for every parameters iteratively, starting from an initial position. In 24, the third equation is nonlinear in term of the projection from multi-camera system. In bundle adjustment, equation 24 is linearized with a first-order Taylor expansion, that is:

$$\begin{bmatrix} \boldsymbol{v}_p \\ \boldsymbol{v}_\Gamma \\ \boldsymbol{v}_t \end{bmatrix} = \underbrace{\begin{bmatrix} \overline{\boldsymbol{P}}_p - \boldsymbol{P}_p^0 \\ \overline{\boldsymbol{\Gamma}} - \boldsymbol{\Gamma}^0 \\ F(\overline{\boldsymbol{P}}_p, \overline{\boldsymbol{P}}_n, \overline{\boldsymbol{X}}_t) - \mathbf{x}_t \end{bmatrix}}_{\boldsymbol{y}} + \underbrace{\begin{bmatrix} \boldsymbol{I} & 0 & 0 & 0 \\ 0 & 0 & \boldsymbol{I} & 0 \\ \frac{\partial F}{\partial \boldsymbol{P}_p} & \frac{\partial F}{\partial \boldsymbol{P}_n} & \frac{\partial F}{\partial \boldsymbol{\Gamma}} & \frac{\partial F}{\partial \boldsymbol{X}_t} \end{bmatrix}}_{\boldsymbol{J}} \underbrace{\begin{bmatrix} \delta_{\boldsymbol{P}_p} \\ \delta_{\boldsymbol{P}_n} \\ \boldsymbol{\delta}_\Gamma \\ \delta_{\boldsymbol{X}_t} \end{bmatrix}}_{\boldsymbol{\delta}_\beta} \tag{26}$$

where, $\boldsymbol{y}$ is the vector of differences between predicts and observations, $\boldsymbol{J}$ is Jacobian matrix and $\hat{\boldsymbol{\delta}}_\beta$ is the vector of corrections of the parameters. We note $\boldsymbol{\beta}$ as the vector of unknowns in multi-camera based localization, including the pose of viewpoints $\boldsymbol{P}_t$, 3D tie points $\boldsymbol{X}_t$ and the rigid transformation parameters $\boldsymbol{\Gamma}$. We also define $\boldsymbol{\beta}^k$ as the estimates of the parameters in step k. Then the parameters will be updated by: $\boldsymbol{\beta}_{k+1} = \boldsymbol{\beta}_k + \hat{\boldsymbol{\delta}}_\beta$ in next step. The corrections are obtained by solving the normal equation:

$$\boldsymbol{J}^T \boldsymbol{W} \boldsymbol{J} \hat{\boldsymbol{\delta}}_\beta = \boldsymbol{J}^T \boldsymbol{W} \boldsymbol{y}, \tag{27}$$

where, $\boldsymbol{W}$ is the weight matrix:

$$\boldsymbol{W} = diag(\boldsymbol{\Sigma}_p, \boldsymbol{\Sigma}_\Gamma, \boldsymbol{\Sigma}_t)^{-1}.$$

. The dimension of the normal matrix is $6 \times N + 3 \times M + 6 \times L$. Here, N is the number of viewpoints in LBA window, M is still the number of tie points and L is the number of cameras in the multi-camera system. N depends on the size of LBA window which is usually small and L is fixed. Thus, the computation when we solve the normal equation is determined by M. Although we add error equations about $\boldsymbol{\Gamma}$, the structure of normal equation is not changed. The left-top is a block diagonal matrix and each block is 6×6, the right-bottom is also 3×3 block diagonal matrix about tie points. However, the size of left-top sub-matrix in normal matrix is $6 \times (N + L)$ instead of $6 \times N$ in single camera case, introducing in section 3.3.4.2. The same method (*cf.* section 3.3.4.2) is used to estimate the covariance matrix of the poses from $[\boldsymbol{J}^T \boldsymbol{W} \boldsymbol{J}]^{-1}$.

3.4.4 Experiment for different camera configurations

The data for experiment was acquired by STEREOPOLIS which is mounted with multiple cameras. In this experiment, four cameras which are two front looking and two backward looking, are selected. These four cameras are marked as camera 11, camera 12, camera 51 and camera 52 in STEREOPOLIS. Figure 3.17 shows the position of each camera in the mobile mapping system. The cameras are calibrated beforehand and all the relative rigid transformation between cameras have been measured during system calibration.

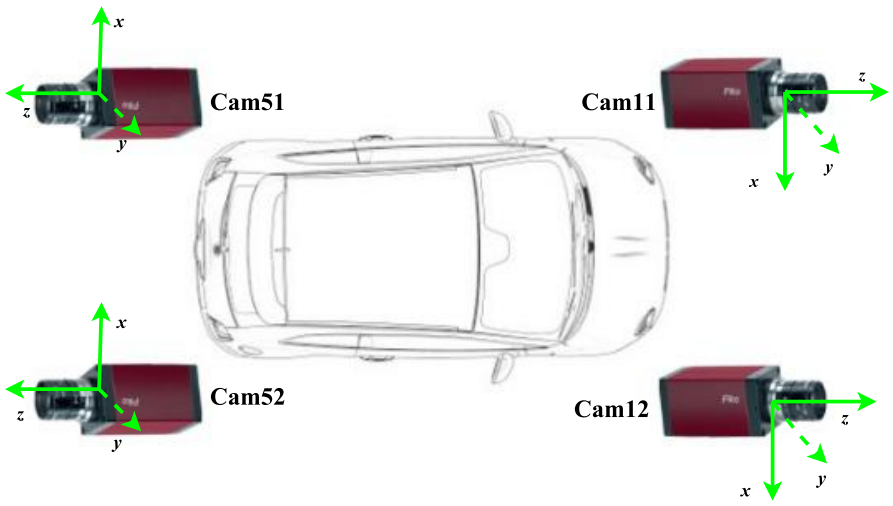
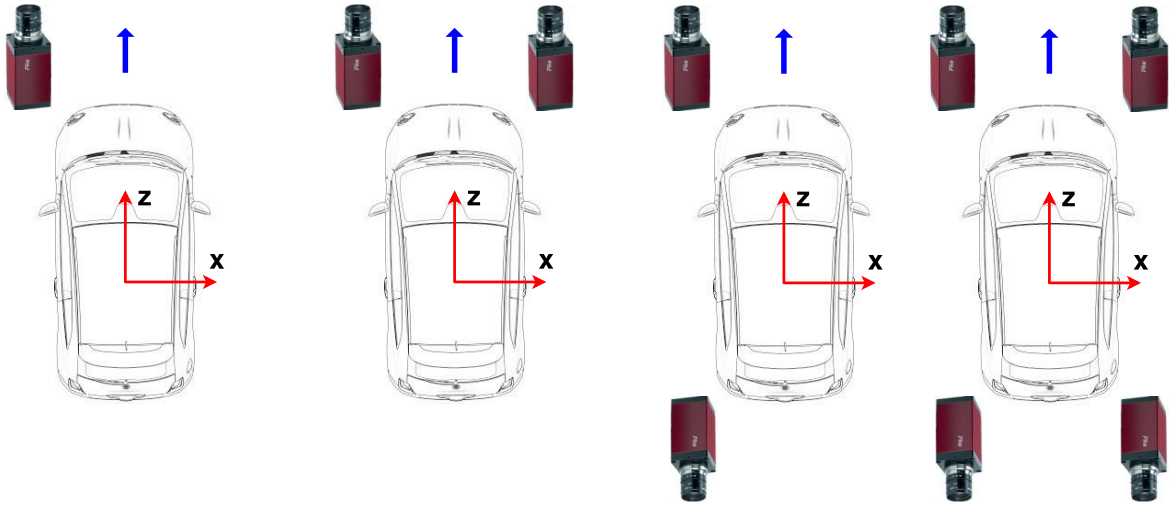


Figure 3.17: The positions of the four cameras in STEREOPOLIS and the camera coordinate systems.

3.4.4.1 Camera configuration

In this experiment, we design four different camera configurations to evaluate their performance, as shown in figure 3.18.

The four camera configurations are noted as: Mono, F_F, F_B and F_F_B_B:



(a) **Mono** (b) **F_F** (c) **F_B** (d) **F_F_B_B**

Figure 3.18: Design of camera configuration. **F**: Forward looking. **B**: Backward looking. (a) **Mono**: monocular camera. (b)**F_F**: two front cameras. (c) **F_B**: one front and one back camera. (d) **F_F_B_B**: using four cameras.

- **Mono**: One forward looking camera (camera 11) is used in this case, which has been used in the experiment in 3.3.5.
- **F_F**: Two forward looking cameras are used to compose a conventional stereo vision. They are camera 11 and camera 12 in STEREOPOLIS.
- **F_B**: One front looking camera and one backward looking camera are applied for this non-overlap stereo.
- **F_F_B_B**: All the four cameras are used to generate a camera cluster.

The mono and F_F are the most popular camera configurations used in visual odometry and SLAM. In the case of F_B, it can enlarge global FOV. Although there is no overlap between two cameras, each camera can make contributions for the pose estimation at different directions. We generate the F_B using camera 11 and 51 in experiment. It can also be composed using camera 12 and 52. For the configuration of F_F_B_B, it should be more robust than other three ones, which have larger FOV and more observations. But the expense is increased from the point view of computation.

3.4.4.2 The results of localization

The length of trajectory is 750 m which is the same area which was tested in the experiment of monocular case, it contains 270 viewpoints. For all the cases, we assume that our localization approach starts from a known point. In mono case, we set the distance from first image to second

key frame manually to determine the absolute scale. The same method is used for F_B case. Although we know the metric relations between the two cameras, there is no overlap between two cameras. Thus, it is impossible to set the absolute scale using the known transformation between two cameras. For F_F and camera cluster F_F_B_B, the absolute scale comes from the length of the baseline in stereo rig.

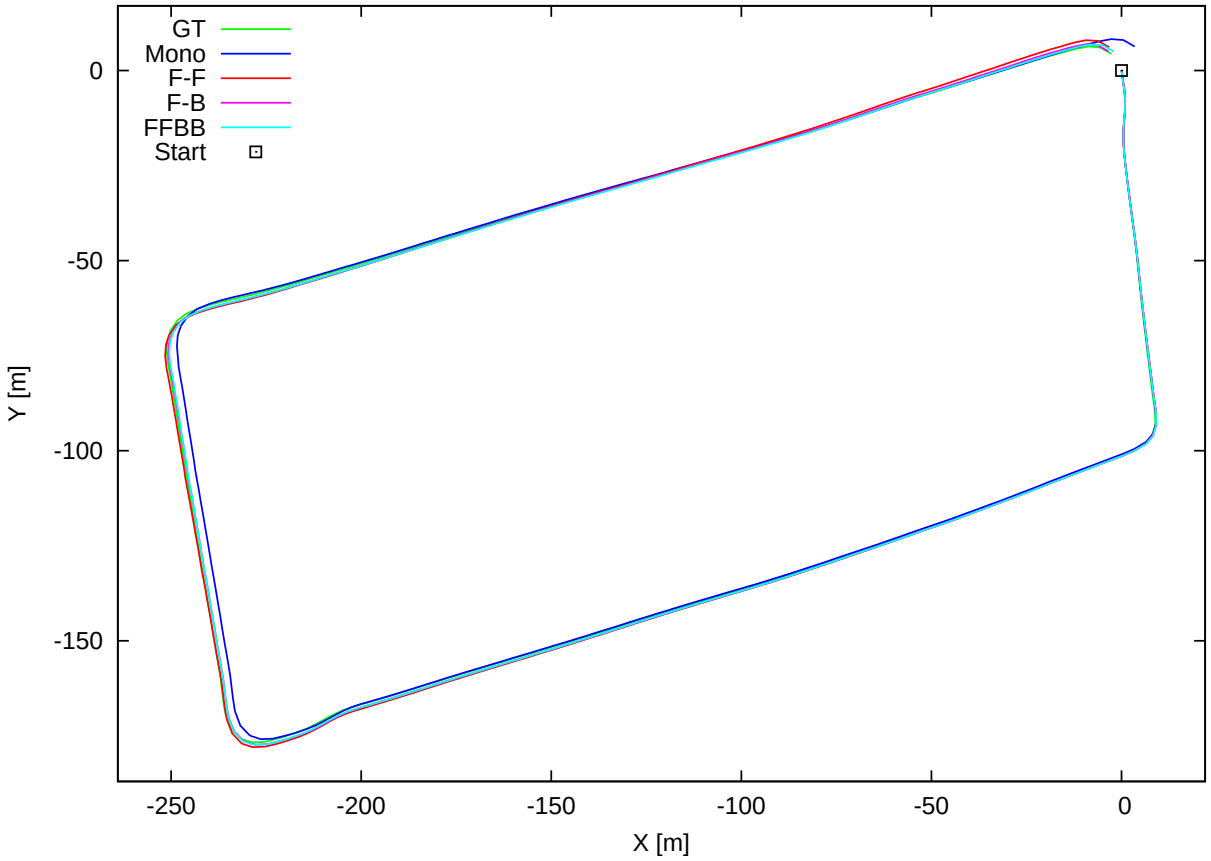
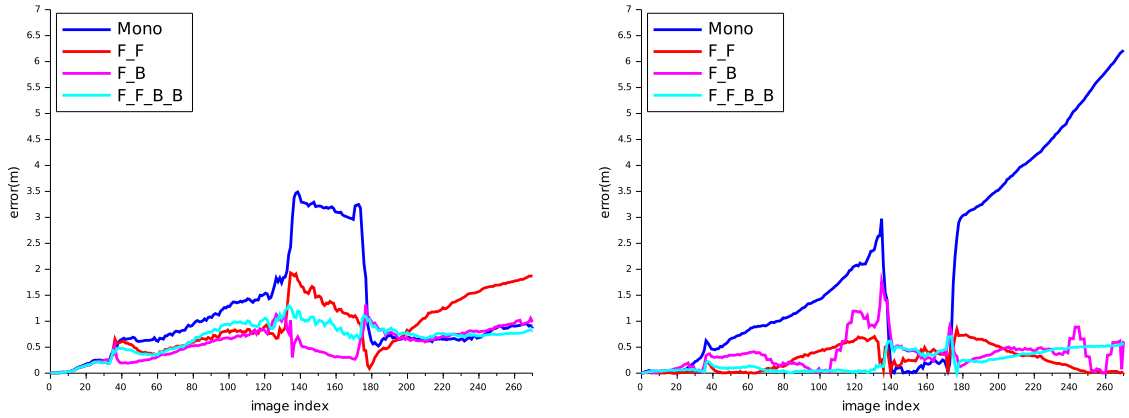


Figure 3.19: Comparing the estimated paths with ground truth in map. GT: Ground Truth

We compare the trajectories estimated using each configuration with the Ground Truth (GT) obtained by GNSS/INS/odometer system. The results are presented in figure 3.19. The mono case is the worst one while the results for other three cases are very close. F_F_B_B is slightly better than F_B and F_F.

A similar strategy that has been used for analyzing the results in experiment 3.3.5, is applied in order to analyze the accuracy of localization at depth and lateral direction separately. The errors are shown in figure 3.20. First, we analyze the errors along image plane in figure 3.20(a). One common conclusion we can learn from this diagram is that, the F_B and F_F_B_B perform better than mono and F_F. The reason might be that the configurations of F_B and F_F_B_B have larger FOV than Mono and F_F. Then, we analyze the errors at depth direction in figure 3.20(b), the F_F and F_F_B_B indicate their robustness to keep the accuracy in depth. Meanwhile, F_B also performs very well for many images, but it is not as stable as F_F and F_F_B_B in some situations such as turning where the moving direction changes quickly.



(a) lateral errors. (b) Depth errors

Figure 3.20: Accuracy of viewpoint position.

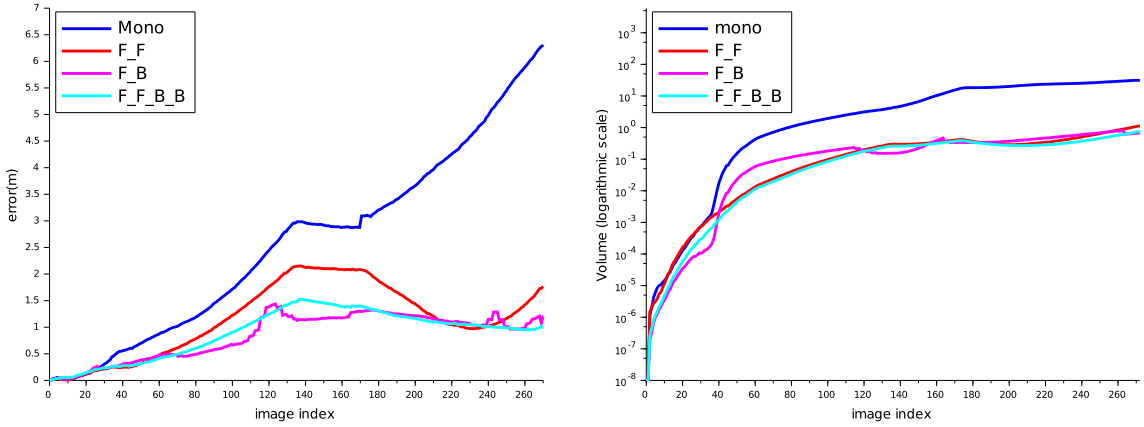
We also estimate the uncertainties of viewpoint poses that consider the error propagation of inherited poses and rigid transformation of each camera. The diagrams in figure 3.21 shows the absolute error position and the length of major axis of error ellipsoids.

The absolute errors in figure 3.21(a) is to calculate euclidean distance from estimation position to ground truth in 3D, calculated by

$$abs\_error = \sqrt{(X - X_0)^2 + (Y - Y_0)^2 + (Z - Z_0)^2},$$

where, (X, Y, Z) is the estimated position and (X_0, Y_0, Z_0) is ground truth. The error ellipsoid of location is computed from its covariance matrix, the size of ellipsoid represents the uncertainty of estimated pose. The statistics are drawn in figure 3.21, where figure 3.21(a) shows the absolute errors of localization compared with ground truth and figure 3.21(b) demonstrates the volumes of error ellipsoid. The increasing values from starting image in figure 3.21(b) represents the growing of uncertainty over time. Both the errors and uncertainties are reduced when we increase the number of cameras. In the cases of F_B and F_F, F_B can obtain more precise position than F_F(*cf.* Fig 3.21(a)), but the uncertainties of F_F are smaller, seen from figure 3.21(b). The accurate localization of F_B takes benefit from its larger FOV. For the smaller uncertainties obtained by F_F, this is caused by the increasing number of observations for each tie points.

Figure 3.22 compares error ellipsoids between Mono and F_F_B_B and the error ellipsoids are exaggerated three times bigger . Only one error ellipsoid is drawn in successive five key frames for better visualization. We can observe that the error ellipsoids of F_F_B_B are always smaller than those of Mono. This illustrates that the uncertainty is reduced by using multi-camera system.



(a) Absolute errors

(b) Error ellipsoid size

Figure 3.21: Absolute errors and volumes of error ellipsoid of viewpoint positions. (a) Absolute errors of locations. (b) Volume of error ellipsoid of viewpoint position.

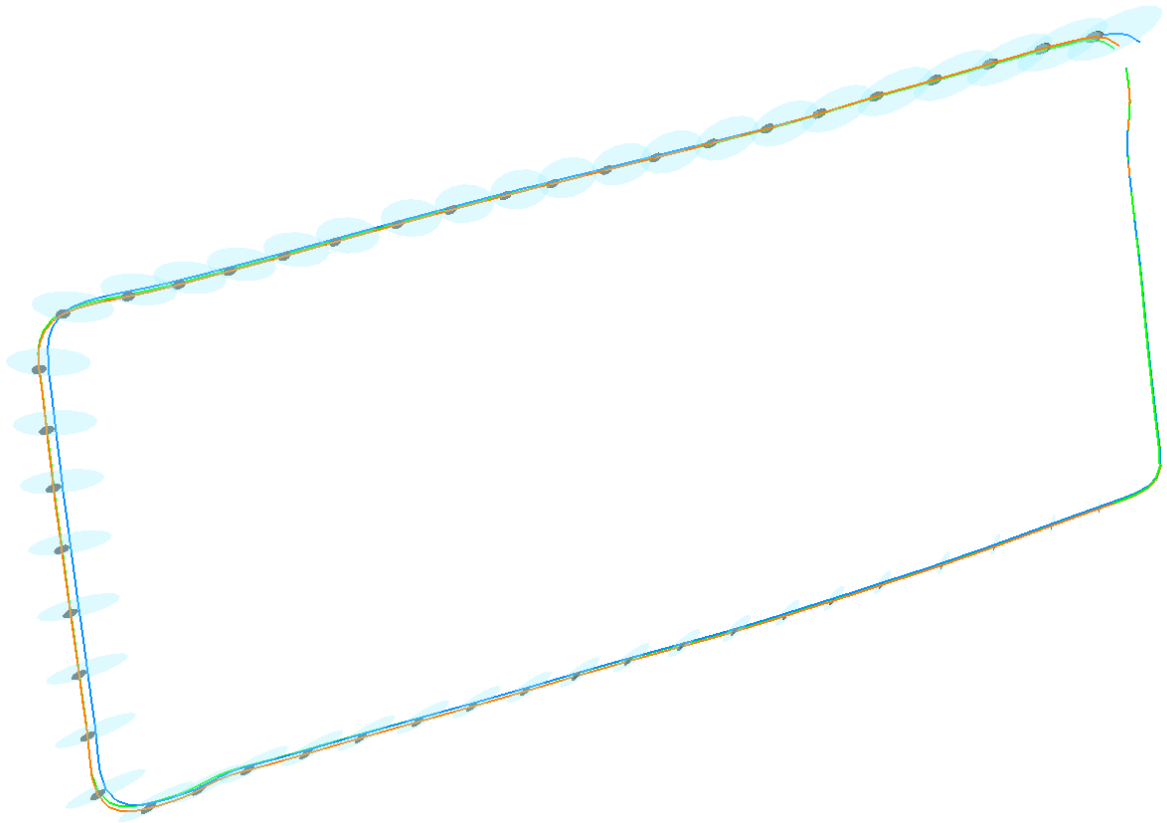


Figure 3.22: Uncertainties of position for **Mono** (light ellipsoids) and **F_F_B_B** (dark ellipsoids). Blue: trajectory of **Mono**. Red: trajectory of **F_F_B_B**. Green: ground truth.

3.4.4.3 Efficiency analysis

Accurate localization is always desired, but the efficiency is also an important issue that we need to consider. The previous experiments indicate the great advantage of F_F_B_B, but it is

obvious that the cost of computation increases with the growing number of cameras. We count the processing time spent on feature extraction, matching, tracking and parameters initialization and refinement with LBA. In the experiments, the size of LBA is ten and the only one new pose in each LBA window. It means $N = 10, n = 1$. There are 270 images for each camera. In this case, the total computing time is shown in table 3.1.

Table 3.1: Total time spent in each part of localization.

	Mono	F_F	F_B	F_F_B_B
Feature extraction (s)	232.05	453.60	460.71	913.92
Matching&tracking (s)	156.76	668.13	291.39	1203.35
LBA (s)	95.42	293.20	358.76	506.91

Time on feature extraction The computation SIFT feature extraction only depends on the number of images. This means that when we add one more camera, the time spent for feature extraction will increase one time. The time statistic in table 3.1 for feature extraction have illustrated this, where the time for two-camera cases (F_F and F_B) is similar, which is two times more than mono. Meanwhile, the four camera cluster spends almost four times more processing time than mono on feature extraction. The average number of feature points detected for each image is about 2780.

Time on matching and tracking As we introduced in section 3.2, the matching and tracking for stereo is more complicate than mono, which is four times more due to the cross matches. The time in the table from experiments trends to prove this where the time for F_F is $668.13s$ which is almost four times bigger than the matching and tracking time for mono $156.76s$. For the cases of F_B and F_F_B_B, their matching methods are a combination of the matching method for mono and stereo. In the case of F_B, the matching can be divided into two matching units for mono. The matching in F_F_B_B is composed by the matching of two stereo independently. In this case, the time spent on F_B should be two times more than mono. Meanwhile, F_F_B_B is two times more than F_F and eight times bigger than mono, observing data in second row in table 3.1. In global, the order of time complexity for all the camera configuration is

$$time(Mono) < time(F\_B) < time(F\_F) < time(F\_F\_B\_B).$$

Time on LBA The time spent on LBA relays on the number of parameters and the number of iterative steps to approach optimal estimates of parameters. The former one is influenced by the number of images, the number of tie points, while the second factor depends on the quality of initial values of the parameters and the distribution of observations. So there are many factors that affect the processing time, that makes it difficult to analyze the relations of the processing time between different camera configurations in table 3.1. However, one common conclusion

is that, the processing time increases with the growing of camera numbers. Because more new tie points are generated and more observations for tie points are measured with the growing of images.

The above experiments are performed on a Linux PC (Intel i5 CPU at 3.30GHz, 4 cores, 7.8 GB of RAM memory and 64 bit OS). The SIFT algorithm implemented in Vlfeat is applied to extract the features [Vedaldi and Fulkerson, 2008]. Our LBA algorithm is developed based on Ceres-Solver which is an open source C++ library for modeling and solving optimization problems [Agarwal et al.].

3.5 Conclusion of vision based localization

The localization methods introduced in this chapter only use cameras. A LBA based approach for localization using monocular camera is introduced at first. Then we extend the method adapting to multi-camera system, considering uncertainty propagation.

Accuracy of localization The proposed methods for localization are evaluated using real image sequences captured with precise MMS. We compare our localization results with ground truth which is acquired by a high precision GNSS/IMU/Odometer system. From our experiments, the accuracy of localization is improved from mono to multi-camera system. The maximum error is over $6m$ using monocular system while it is reduced to less than $1.5m$ by using multi-camera rig.

Uncertainty propagation of localization For localization using monocular camera, we propagate the uncertainties of image poses over sequence. For multi-camera rig based localization, we consider uncertainty propagation for both poses and relative transformation from cameras to view points. Our experiments in section 3.3 and section 3.4 indicate that uncertainties of poses are growing over time, but the increasing speed is decreased when we extend monocular camera based localization to multi-camera system based approach.

Efficiency of vision based localization In summary, it is a trade-off between accuracy and efficiency for vision based localization. But one interesting point is that F_B could be a good solution which improve the localization accuracy significantly in comparison to Mono case, while the processing time doesn't grow as much as F_F or the multi-camera system. From table 3.1, we would say that it is not a real-time localization approach for current implementation.

The proposed methods was tested on other datasets, but it suffers from insufficient matches using current matching and tracking strategy. Moreover, most of processing time is spent on feature extraction, matching and tracking (*cf.* Tab 3.1). Thus, the performance of matching and tracking need to be promoted . The relevant research will be presented in next chapter.

Chapter 4

Propagation based matching and tracking

This chapter presents a new method for detection, matching and tracking of interest points to improve the performance of localization. The related phases in the entire pipeline of localization are shown as following figure: In this chapter, section 4.1 is an overview that places the problem

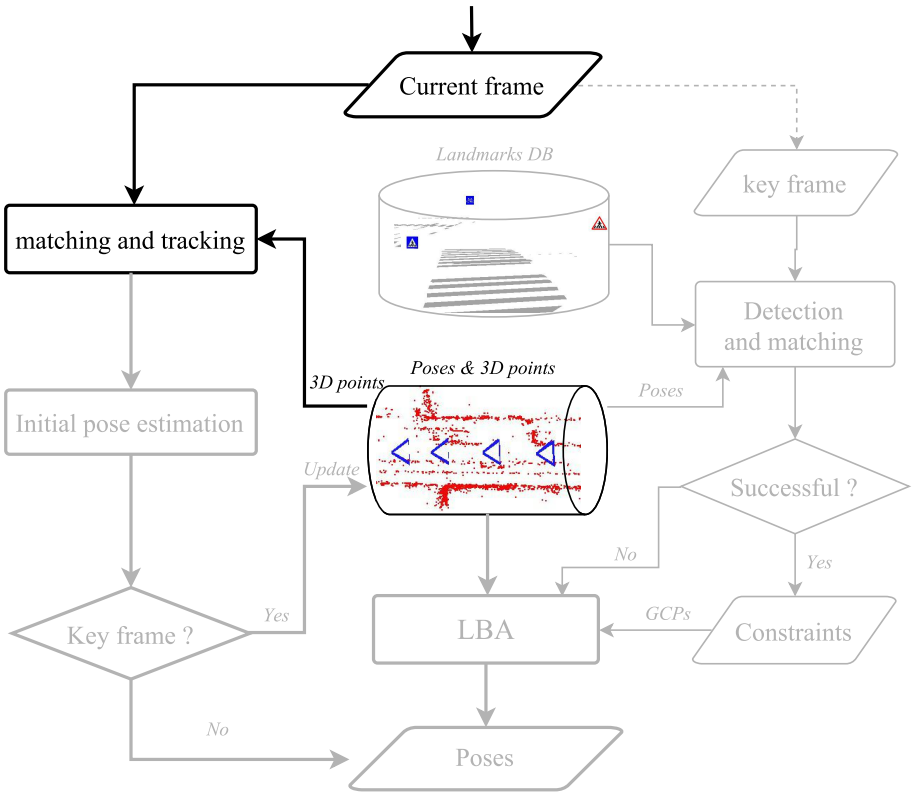


Figure 4.1: Matching and tracking for localization.

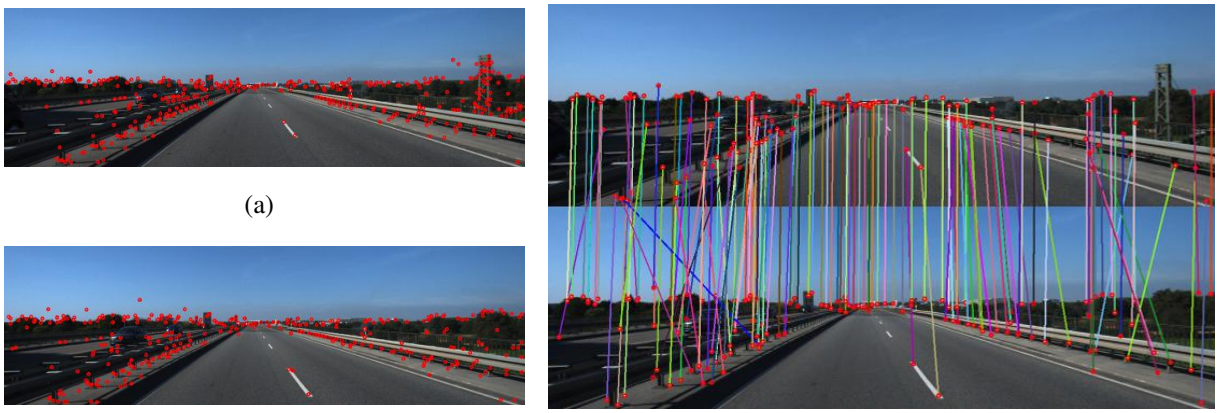
of original matching and tracking methods and presents the flowchart of new method. Section 4.2 explains pose prediction and uncertainty propagation. The guided matching is introduced in section 4.3 and section 4.4 presents the strategy to explore new tie points. As last, we test the new matching and tracking methods using real image sequences.

4.1 Overview

4.1.1 Problem statement

In chapter 3, we have introduced the methods for feature extraction, matching and tracking. We used SIFT for feature extraction and FLANN based pair-wise matching was applied. The tie points were tracked over the pairwise matches. This kind of methods are usually used in structure from motion [Pollefeys et al., 2004; Snavely et al., 2006; Moulon and Monasse, 2012], but the computation is too high for visual odometry. As we discussed in section 3.4.4.3, feature extraction and matching spend most of processing time. Thus, we need an efficient way to establish the links of tie points for localization.

Another reason that motives us to explore a new matching and tracking method is the robustness. We test our localization method proposed in chapter3 on KITTI visual odometry benchmarks [Geiger et al., 2012] and we find that the current feature tracking, matching method is difficult to be used in some scenarios. Figure 4.2 shows feature extraction and matching for one adjoint image pair on high speed road using the methods proposed in section 3.2 (*cf.* chapter 3).



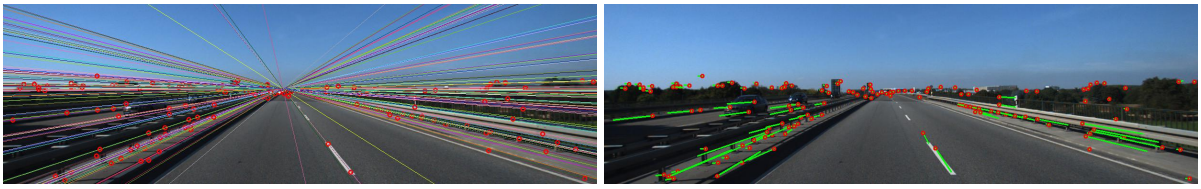
(a)

(b)

(c)

Figure 4.2: Problem of pairwise matching for image on high-speed road. (a) 413 SIFT feature points. (b) 364 SIFT feature points. (c) pairwise matches between the points in (a) and (b).

The matches in figure 4.2(c) have been refined by rejecting the outliers using epipolar constraints, but there are still many false matches due to the ambiguity of relations between epipolar lines and false matching points. As shown in figure 4.3, the directions of pixel-flow and epipolar lines are almost parallel. In this case, when the false matches are laid on or very close to epipolar lines, we can not reject them according to the distance to epipolar. The images demonstrated in previous figures were captured on high speed road where the texture in image is poor. The false matches are caused by repeatable texture over sequences and they can't be rejected according to epipolar lines. In urban area, texture information would be richer, but it is also difficult to filter the false matches on building facades in image according to epipolar lines. In this case, a robust approach for matching and tracking is desired.



(a) (b)

Figure 4.3: (a) epipolar lines between images 4.2(a) and 4.2(b). (b) pixel flow between corresponding points

.

4.1.2 Our solution

In practice, the motion of vehicle is restricted because of inertial moment for both translation and rotation. As shown in figure 4.4, these motions change smoothly over time. Thus, the overall motions can be expressed in terms of a dynamic model [Bradler et al., 2015]. It is known that the distribution of the corresponding points between two images are related to the motion. If the initial motion for the new frame can be predicted, the exploring of correspondences can be limited into small regions instead of searching in entire image.

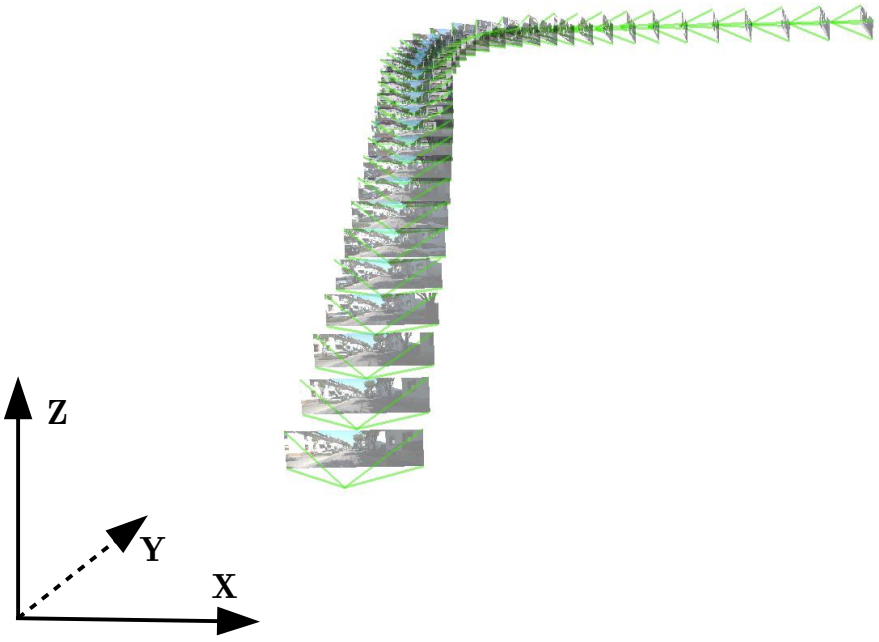


Figure 4.4: The location and orientation of frames.

In this thesis, we intend to develop a propagation based matching and tracking method for tie points exploration to improve both efficiency and robustness. The propagation based tracking is recently proposed by Nolang Fanani and Mester [2016]. In this approach, the relative rotation is estimated independently from translation using enhanced phase correlation proposed in Barnada et al. [2015], then the translation of every new frame is predicted according to the previously

estimated transformations [Bradler et al., 2015]. With the predicted pose, all the tie points from previous frames can be propagated into current frame. Knowing the epipolar geometry between two frames, the precise location of every propagated point is searched along epipolar line. We have the same goal, but a more regular dynamic model is proposed to predict the pose. Our model is obtained by learning from previous poses and predict both rotation and translation at the same time. Moreover, we consider uncertainty of prediction when the tie points are propagated into images.

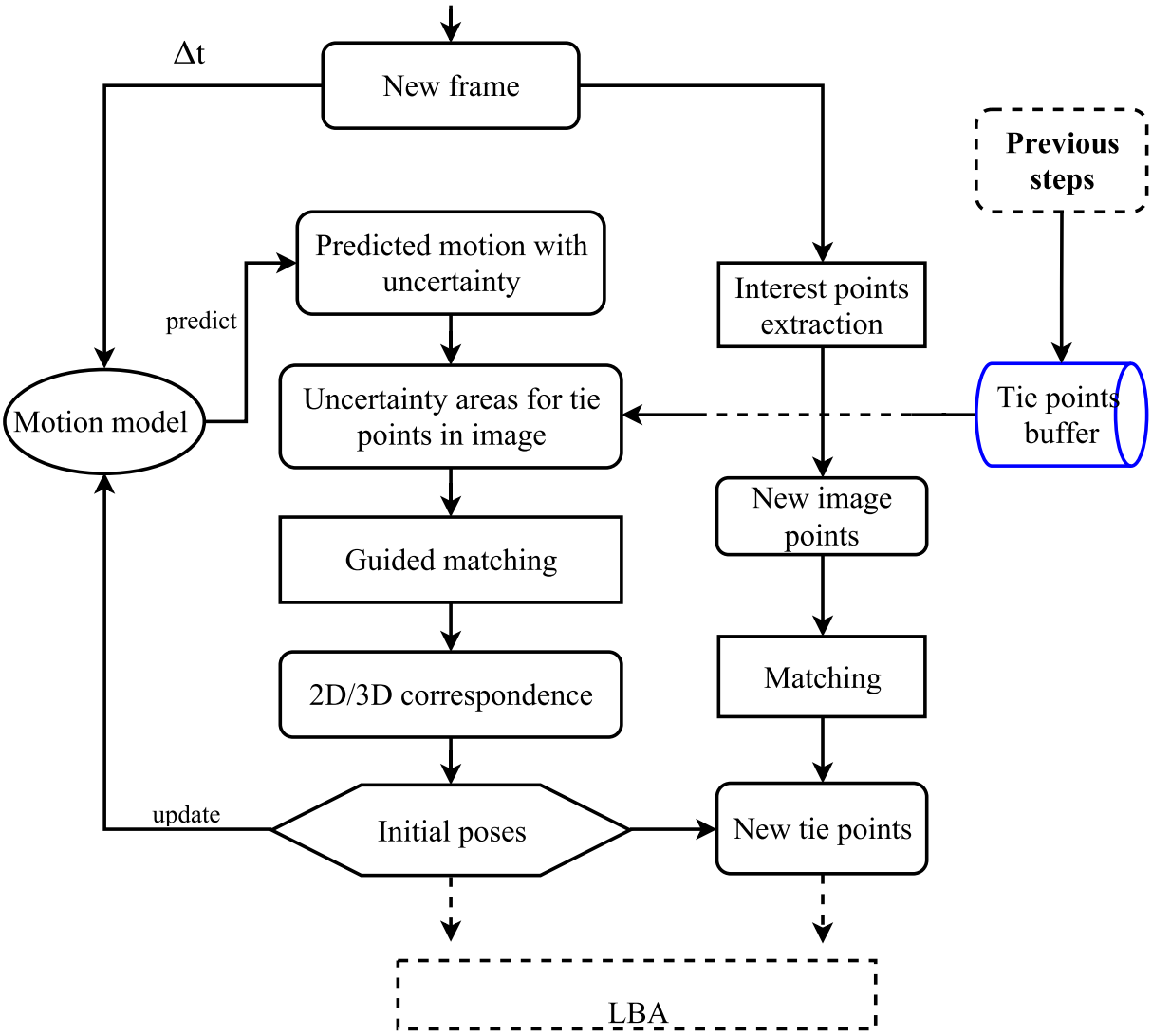


Figure 4.5: Flowchart of propagation based matching and tracking.

The workflow of our strategy is presented in figure 4.5. First, the pose of each new frame is predicted with a predefined motion model when we know the time interval δt. Then, an approximate area for every existing tie points can be generated considering the uncertainty propagation in the new image. In this case, we only search the precise locations of tie points inside the approximate areas in image. Hence, a set of 3D-2D point-to-point correspondences are obtained so that the precise image pose of image at current time step can be estimated. Then we have two pipelines (*cf.* Fig 4.5), which are update and enriching. The motion model should

be updated timely with the new pose to maintain the state over time. Another pipeline is to find new tie points and triangulate their 3D coordinates.

4.2 Predict and update

The motion prediction is well studied in kinematics associated with a variety of quantities like displacement, velocity, acceleration, and time. The knowledge of each quantity provides descriptive information about the motion of an object [Forshaw and Smith, 2014]. Wei et al. [2014] applied a constant acceleration model to present the motion of vehicle to predict the vehicle translation over time. It assumes that the acceleration of moving vehicle is constant from time step $t-1$ to t. Meanwhile, a constant velocity model was also proposed to predict the translation of vehicle [Bradler et al., 2015], which is linear and much simpler than acceleration based motion model. The velocity from time step $t-1$ to t is considered to be constant. However, these two papers only use the motion model to predict the translation. The orientations are measured by IMU [Wei et al., 2014], or estimated using phase correlation [Barnada et al., 2015]. In fact, the same strategies can also be used for rotation prediction, Persson et al. [2015] predicted the poses using a constant acceleration and constant angular acceleration model jointly in visual odometry when the pose estimation is in ill-conditions (very few tie points are tracked for new images). More related applications can be found in SLAM. For instance, Davison [2003] predicted the motion in time step using constant velocity model for both translation and rotation. It does not mean that the vehicle moves with constant speed all the time, but that the motion in a time step is on average and the undetermined accelerations is expected to occur with a Gaussian profile [Davison, 2003].

In our case, we intend to apply constant velocity motion model for prediction. As we can estimate the uncertainty for every pose from LBA, the uncertainty of the predicted pose can be obtained via motion model (*cf.* Fig 4.6).

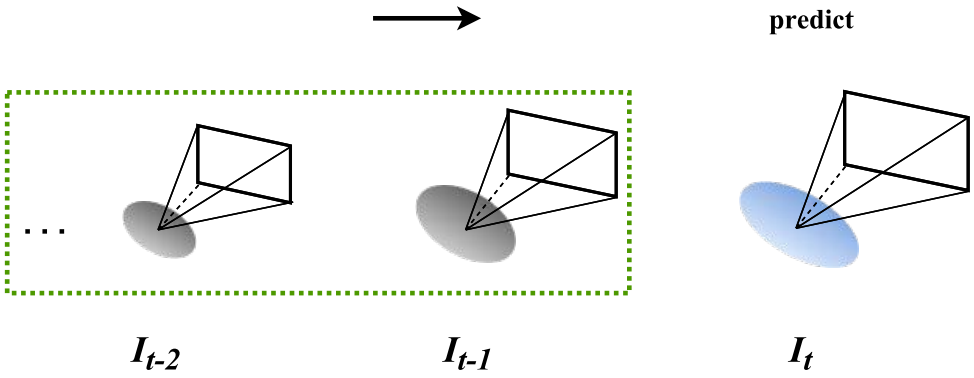


Figure 4.6: Prediction and uncertainty propagation for new frame. A motion model is built using priorly estimated poses to predict the pose of new frame I_t. The uncertainty of the prediction is propagated from previous poses via motion model.

4.2.1 Motion model

In 1-D kinematic, if we know the position at time step $t-1$ (P_{t-1}), the velocity from $t-1$ to t is V_t and the time interval is Δt. The position at time step t can be obtained: $P_t = P_{t-1} + V_t \cdot \Delta t$. The principle can be derived from multi-dimensional kinematic. In this thesis, $\boldsymbol{P}$ is defined as the vector of image pose which contains 6 parameters. The first three stands for position of perspective center and the last three values represent orientation of the image. $\boldsymbol{V}$ is noted as the vector of velocity for translation and rotation.

$$\begin{cases} \boldsymbol{P} = [X, Y, Z, \alpha, \beta, \gamma]^T \\ \boldsymbol{V} = [V_X, V_Y, V_Z, V_\alpha, V_\beta, V_\gamma]^T \end{cases} \tag{1}$$

where, V_X, V_Y, V_Z are the speed on each axis and $V_\alpha, V_\beta, V_\gamma$ are the angular velocities on every axis. The strict angular velocity should be expressed with quaternion, which is linear in a four-dimension. This model has been introduced in Davison's work [Davison, 2003]. However, the vehicle moves fast and the angular velocities are very small most of time. So it is possible to use the velocities in 3D Cartesian space to approximate the angular velocities. One advantage of using the approximate angular velocity model is to simplify the motion model that makes it more efficient to obtain the uncertainty of prediction.

We note $\boldsymbol{P}_t$ as the pose and $\boldsymbol{V}_t$ as the velocity vector at time step t. The time interval is Δt from $t-1$ to t. The pose at time t can be computed using a linear equation:

$$\boldsymbol{P}_t = \boldsymbol{P}_{t-1} + \boldsymbol{V}_t \cdot \Delta t \tag{2}$$

Thus, the pose $\boldsymbol{P}_t$ relies on the reference pose $\boldsymbol{P}_{t-1}$ and the velocity $\boldsymbol{V}_t$.

4.2.2 Motion prediction

We assume that the vehicle moves smoothly on the road. The translation velocity and angular velocity keep constant in one time step. According to the constant velocity model presented in equation 2, three quantities: $\boldsymbol{P}_{t-1}$, $\boldsymbol{V}_t$ and Δt should be known for new pose prediction. In practice, Δt can be easily obtained using a timer coupled with camera. The pose at previous time step can also be estimated precisely with methods proposed in chapter 3. Thus, the key for motion prediction is to determine the velocity vector. With the assumption of constant velocity, the velocity at time step t can be expressed using the velocity at time step $t-1$, that is:

$$\boldsymbol{V}_t^* = \boldsymbol{V}_{t-1}, \tag{3}$$

where, $\boldsymbol{V}_t^*$ is the approximate velocity at time step t. Therefore, the approximate pose at time step t, noted as $\boldsymbol{P}_t^*$, can be predicted using equation 2, associated with the approximate velocity, as shown in equation below:

$$\boldsymbol{P}_t^* = \boldsymbol{P}_{t-1} + \boldsymbol{V}_t^* \cdot \Delta t \tag{4}$$

4.2.3 Motion model update

As we discussed at the beginning of this section, the constant velocity model doesn't mean that the moving speed and angular velocity of vehicle is always constant. Actually, we only assume that the velocity is constant from previous time step to current time step, as shown in equation 3. Therefore, in order to get accurate prediction, $\boldsymbol{V}_{t-1}$ should be computed dynamically that makes it close to the real velocity at time step t. The aim of model update is to estimate the accurate velocity $\boldsymbol{V}_t$ when we have estimated the accurate pose $\boldsymbol{P}_t$ using the matching and tracking results at current time. Then, $\boldsymbol{V}_t$ will be used for prediction of a new time step $t+1$.

Regarding the terms in equation 2, the only unknown term is $\boldsymbol{V}_t$. It can be easily computed using equation below:

$$\boldsymbol{V}_t = \frac{\boldsymbol{P}_t - \boldsymbol{P}_{t-1}}{\Delta t} \tag{5}$$

where:

- $\boldsymbol{P}_t$: accurate pose at current time.
- $\boldsymbol{P}_{t-1}$: accurate pose at previous time step.
- Δt: time interval from step $t-1$ to t.

In order to improve the robustness of prediction and reduce the impact of erroneous pose estimation, we compute the velocity for time step t using the latest three poses that are P_t, P_{t-1}, P_{t-2}. The time intervals are noted as $\Delta t, \Delta t_1$. The Δt_1 is the time interval from t to $t-2$. In this case, the velocity vector is computed by:

$$\boldsymbol{V}_t = \frac{(\boldsymbol{P}_t - \boldsymbol{P}_{t-1}) + (\boldsymbol{P}_t - \boldsymbol{P}_{t-2})}{\Delta t + \Delta t_1} \tag{6}$$

To obtain a simplified expression, we define:

$$\Delta T = \Delta t + \Delta t_1,$$

thus the equation 6 can be written as:

$$\boldsymbol{V}_t = \frac{2\boldsymbol{P}_t - \boldsymbol{P}_{t-1} - \boldsymbol{P}_{t-2}}{\Delta T} \tag{7}$$

The special case is the beginning of localization. We start from a known point, there is no way to compute velocity vector in advance. So we suppose that the velocity vector is zero, thus the poses of second frame is

$$\boldsymbol{P}_1^* = \boldsymbol{P}_0$$

according to equation 4. Then, we use equation 5 to compute the initial velocity for the prediction of third frame.

4.2.4 Uncertainty propagation

The estimation of uncertainty for every key frame in LBA enables us to estimate the uncertainty of the predicted pose. We write equation 7 as matrix style:

$$\boldsymbol{V}_t = \underbrace{\begin{bmatrix} \frac{2}{\Delta T}\boldsymbol{I}_{6\times6} & -\frac{1}{\Delta T}\boldsymbol{I}_{6\times6} & -\frac{1}{\Delta T}\boldsymbol{I}_{6\times6} \end{bmatrix}}_{\boldsymbol{A}} \begin{bmatrix} \boldsymbol{P}_t \\ \boldsymbol{P}_{t-1} \\ \boldsymbol{P}_{t-2} \end{bmatrix}, \tag{8}$$

where, $\boldsymbol{I}_{6\times6}$ is a 6×6 identity matrix and A is a 6×18 matrix in terms of time interval. It is obvious that equation 8 is linear. Considering the principles of covariance propagation, the covariance matrix of $\boldsymbol{V}_t$ is:

$$\boldsymbol{\Sigma}_{V_t} = \boldsymbol{A}\boldsymbol{\Sigma}_P\boldsymbol{A}^T \tag{9}$$

where, $\boldsymbol{\Sigma}_{V_t}$ is the covariance matrix of $\boldsymbol{V}_t$, $\boldsymbol{\Sigma}_P$ is the covariance matrix of $\boldsymbol{P}_t, \boldsymbol{P}_{t-1}$ and $\boldsymbol{P}_{t-2}$. In the prediction of next frame $t+1$, we suppose:

$$\boldsymbol{V}^*_{t+1} = \boldsymbol{V}_t$$

, hence the covariance of the velocity:

$$\boldsymbol{\Sigma}^*_{V_{t+1}} = \boldsymbol{\Sigma}_{V_t}$$

. The pose at time $t+1$ can be obtained by equation 4. The covariance of the predicted pose $\boldsymbol{P}^*_{t+1}$, noted as $\boldsymbol{\Sigma}^*_{P_{t+1}}$, can be estimated by considering the uncertainty propagation from $\boldsymbol{P}_t$ and $\boldsymbol{V}^*_{t+1}$. We define the time interval from t to $t+1$ as $\Delta t+1$ and assume that P_t is independent to $\boldsymbol{V}^*_{t+1}$. The covariance of $\boldsymbol{P}^*_{t+1}$ can be obtained by below equation:

$$\begin{aligned} \boldsymbol{\Sigma}^*_{P_{t+1}} &= \begin{bmatrix} \boldsymbol{I}_{6\times6} & \Delta_{t+1}\boldsymbol{I}_{6\times6} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_{P_t} & 0 \\ 0 & \boldsymbol{\Sigma}_{V_t} \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_{6\times6} \\ \Delta_{t+1}\boldsymbol{I}_{6\times6} \end{bmatrix} \\ &= \boldsymbol{\Sigma}_{P_t} + \Delta^2_{t+1}\boldsymbol{\Sigma}_{V_t} \end{aligned} \tag{10}$$

where, $\boldsymbol{\Sigma}^*_{P_{t+1}}$ is the covariance matrix of our prediction for the pose of new frame and $\boldsymbol{\Sigma}_{V_t}$ is covariance matrix of velocity.

4.3 Guided matching

The section 4.2 introduces how to predict motion and propagate the uncertainty for new frame. In this section, we will introduce how to use those predictions for guided matching. A searching area for every tie point will be generated using the predicted pose and its uncertainty, instead of searching every existing tie point in entire image. Our aim is to decrease false matches and

speed up the matching processing. There are two major issues for guided matching: 1) how to generate the searching area for each 3D tie point in image and 2) which algorithm should be employed to measure the similarity for matching in the searching area.

4.3.1 Generation of searching window

With the predicted pose of current frame, every existing tie point can be projected into current image plane using the projection equation (*cf.* equation 6 in section 3.1) proposed in chapter 3. In order to find precise locations, we need to search into a surrounded area around the approximate back-projection. The issue is how to determine searching scope. The simplest way is to set a fixed size for all the points, but it has two restrictions in our applications. First, the error propagation will influence the precision of prediction for new pose over time. Second, the precision for back projections of a tie point depends on its depth, where smaller depth means larger back-projection errors though the tie points have same precision. In this case, small searching areas for far tie points and large areas for close tie points should be set. Therefore, we desire a solution that can determine the size of searching area dynamically considering the precision of prediction and depth uncertainty.

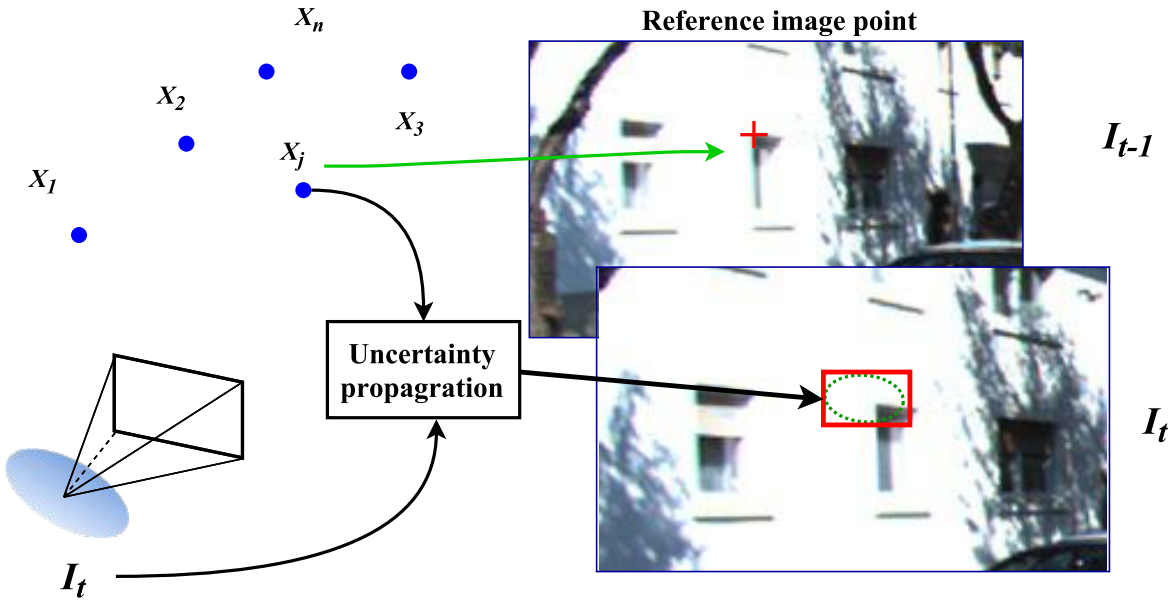


Figure 4.7: Generation of searching area for every tie point in new frame. The blue circles $(X_1, ...X_j, ...X_n)$ are tie points in world coordinate system. The dotted ellipse in I_t is the error ellipse of the projection of X_j. The red rectangle is the bounding box of error ellipse which is also the searching area used for matching with image point in I_{t-1}.

Ochoa and Belongie [2006] proposed a solution to generate the searching boundary for every reference feature point according to uncertainty propagation, the transformation between two images is modeled by homography and the matching is guided by the uncertain region of every mapped image points in target image. Our solution for this problem is also derived from uncer-

tainty for guided matching, but the boundary of searching area is determined by the uncertainty propagated from the predicted pose and tie points. Figure 4.7 shows our proposed method. The uncertainty of image pose is estimated using the method proposed in section 4.2.4. The existing tie points were computed by previous steps which are considered having same precision at this stage. We propagate the uncertainty and estimate the error ellipse for each back-projected location in image (*cf.* Fig 4.7). The searching area is determined by the bounding box of error ellipse, shown with red rectangle in figure 4.7. To search the precise image point location for each existing tie point, we match the reference image point for each tie point into the searching area. The reference image point is chosen the nearest image point in the observation chain of tie point over image sequence (*cf.* red cross in image I_{t-1} in Fig 4.7).

4.3.1.1 Covariance of predicted locations

Considering equation 6 (*cf.* chapter 3), the back-projection of each existing tie point can be computed via:

$$\mathbf{x}_i^* = F(\boldsymbol{P}_t^*, \boldsymbol{X}_i), \tag{11}$$

where, $\boldsymbol{P}_t^*$ is predicted pose, $\boldsymbol{X}_i$ is one of existing tie points and $\mathbf{x}_i^*$ is the predicted location of tie point in image.

F is a nonlinear function. To obtain the covariance of $\mathbf{x}_i^*$, we propagate the covariance from predicted pose and tie points to image plane using first order approximation of the F through Taylor expansion. We denote the covariance matrix for existing tie points as $\boldsymbol{\Sigma}_X$. At time t, the covariance of predicted pose $\boldsymbol{P}_t^*$ is written as $\boldsymbol{\Sigma}_{P_t}^*$, estimated by equation 10 in this chapter. In this thesis, we suppose that $\boldsymbol{P}_t^*$ and $\boldsymbol{X}$ are independent. According to the principle of error propagation, the covariances of predicted points in image can be computed by:

$$\boldsymbol{\Sigma}_{\mathbf{x}_i} = \begin{bmatrix} \frac{\partial F}{\partial \boldsymbol{P}_t^*} & \frac{\partial F}{\partial \boldsymbol{X}_i} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_{P_t^*} & 0 \\ 0 & \boldsymbol{\Sigma}_{X_i} \end{bmatrix} \begin{bmatrix} \frac{\partial F}{\partial \boldsymbol{P}_t^*} \\ \frac{\partial F}{\partial \boldsymbol{X}_i} \end{bmatrix} \tag{12}$$

where, $\boldsymbol{\Sigma}_{\mathbf{x}_t}$ is the covariance matrix of predicted point.

4.3.1.2 Error ellipse

The error ellipse of the projected point in image can be calculated from its covariance matrix [Draper and Smith, 1981; Fan, 1997]. The lengths of error ellipse axes are determined by the eigenvalues, the eigenvectors represent the direction of the two major axis of error ellipse. We note the eigenvalues as λ_1 and λ_2, the eigenvectors as $\boldsymbol{v}_1$, $\boldsymbol{v}_2$. The direction of major axis is the direction in which the errors increase the most.

The summed squared Gaussian data is underlying chi-square distribution in terms of DoF. Thus, the scale of error ellipse s can be determined by giving a confidence level. For instance, 99%

confidence level corresponds to $s = 9.210$. We define a as semi major axis and b as semi minor axis of error ellipse, then

$$\begin{cases} a = \sqrt{\lambda_1 \cdot s} \\ b = \sqrt{\lambda_2 \cdot s} \end{cases}$$

As $\boldsymbol{v}_1$ represents the direction of major axis, the orientation of error ellipse can be calculated via:

$$\alpha = atan(\frac{\boldsymbol{v}_1(1)}{\boldsymbol{v}_1(0)})$$

where, α is the angle relative to x axis for major axis of error ellipse.

4.3.2 Similarity measurement

In our approach, we measure the similarity by comparing two image patches. Then we move the reference image patch over the searching area to find the most similar location. This becomes the problem of template matching. Many methods have been proposed for template matching technique such as Sum of Absolute Difference (SAD), the Sum of Squared Differences (SSD), Normalized Cross Correlation (NCC) [Gonzalez and Woods, 1992; Li et al., 1994; Lewis, 1995; Alsaade, 2012] . The NCC is proved to be the most robust one in the three proposed methods [Zitova and Flusser, 2003].

NCC measures the similarity of two patches by computing discrete 2D correlation. We denote ρ as the correlation coefficient. Given two image patches, the normalized correlation coefficient is computed by below equation:

$$\rho = \frac{\sum_{x,y}[I(x,y) - \overline{I}][T(x,y) - \overline{T}]}{\sqrt{\sum_{x,y}[I(x,y) - \overline{I}]^2 \sum_{x,y}[T(x,y) - \overline{T}]^2}} \tag{13}$$

where, $\overline{I}$ and $\overline{T}$ refer to the mean intensity value of patches in current and reference image. Then we slide the patch T to compute the correlation coefficient for every pixel until the entire area is covered. Thus, a correlation coefficient map can be generated. The value of ρ varies between -1.0 and 1.0. The value of ρ is -1.0 when the texture of two patches inverses of each other and $\rho = 1.0$ means the two patches are exactly same. To guarantee the matching quality, we set a threshold for ρ, the matching result is accepted only if the maximum correlation coefficient is bigger than the given threshold.

NCC is invariant to affine change in image radiometry [Liu and Moore, 1990; Faugeras et al., 1993], whereas it is not invariant to image deformation. In our case, the images are acquired with high sampling rate, so the deformation between adjacent image would not be heavy. So the NCC can still be used for measurement.

4.3.3 Sub-pixel matching

The location of maximum NCC value corresponds to an integer pixel coordinates. Figure 4.8 shows the real correlation coefficient values around the peak, the locations achieved by previous template matching method would locate in one of the pixel near the real peak. To reach the real peak with sub-pixel accuracy of surface conducted by NCC values, either interpolation or fitting can be conducted. As shown in figure 4.8, the NCC values around the peak can be regarded as lying on a smooth surface. Hence, an analytical function would be defined to represent the surface and then the location of peak could be estimated from the function.

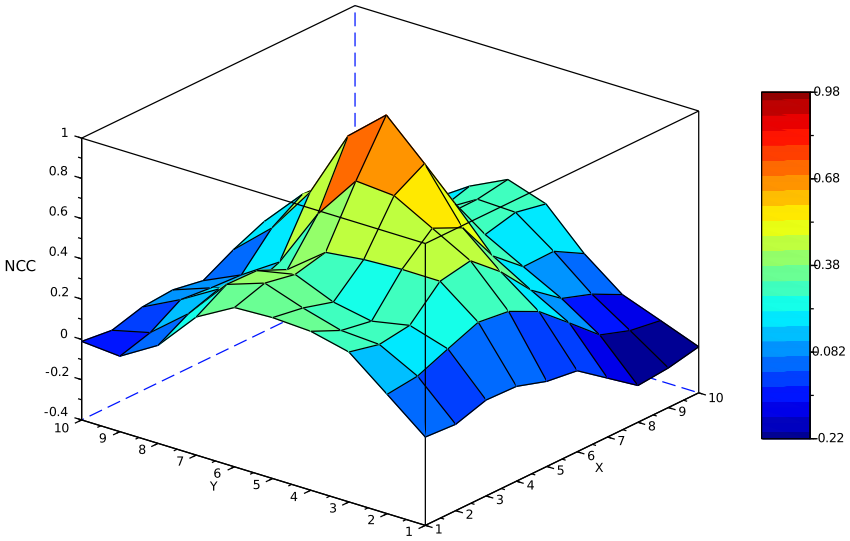


Figure 4.8: The NCC values in a 10×10 neighborhood window around the location of maximum coefficient value.

For continuous version, the surface of correlation coefficient around the peak often forms a bell shape (*cf.* figure 4.8). Thus, two orthogonal parabolic curves can be used to fit the shape in x and y axes profiles. Then the peak is computed independently by fitting one dimensional quadratic function [Naidu and Fisher, 1991; Debella-Gilo and Kääb, 2011]. Let's note location of the peak as $(x_0 + \delta x, y_0 + \delta y)$, where (x_0, y_0) is integer position which has the maximum correlation coefficient and $(\delta x, \delta y)$ stands for the offsets from integer position (x_0, y_0) to the peak.

We define a parabolic curve that connects the adjacent three points at each axis and estimate the position where the curve reach its maximum. It means that three points $(x_0-1, y_0), (x_0, y_0), (x_0+1, y_0)$ are used to fit the parabolic curve at x axis direction and $(x_0, y_0-1), (x_0, y_0), (x_0, y_0+1)$ are used to fit the curve at y axis. The offsets at x axis and y axis can be estimated using

following equation [Naidu and Fisher, 1991; Debella-Gilo and Kääb, 2011]:

$$\begin{cases} \delta x = \dfrac{\rho(x_0 - 1, y_0) - \rho(x_0 + 1, y_0)}{2\rho(x_0 - 1, y_0) - 4\rho(x_0, y_0) + 2\rho(x_0 + 1, y_0)} \\ \delta y = \dfrac{\rho(x_0, y_0 - 1) - \rho(x_0, y_0 + 1)}{2\rho(x_0, y_0 - 1) - \rho(x_0, y_0) + 2\rho(x_0, y_0 + 1)} \end{cases} \tag{14}$$

4.4 Generation of new tie points

In order to find new tie points, the first step is to detect the new image points which are different with existing image points obtained by guided matching. In this thesis, we call them as *new interest points*. Then we generate new tie points from *new interest points* by matching.

4.4.1 Interest points detection

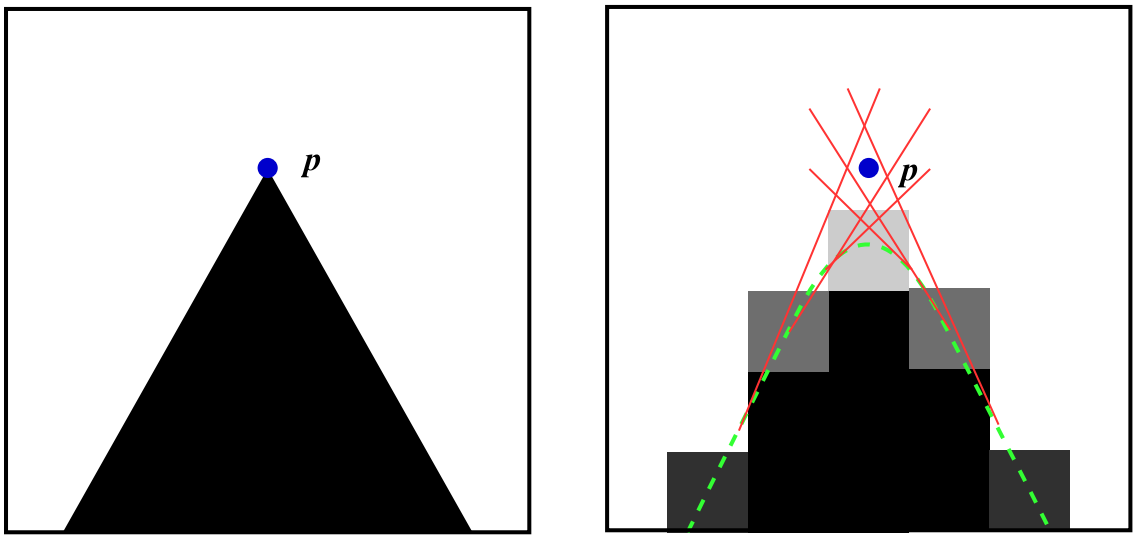
To determine the interest points in image, many detectors can be applied. The algorithms such as SIFT [Lowe, 2004] and SURF [Bay et al., 2006] yield good feature points that are invariant to the change of scale and rotation. However, they are too intensive for real-time application because of the generation of scale space. So the efficient algorithms such as Harris[Harris and Stephens, 1988], Shi and Tomasi detector(good feature for tracking detector)[Shi and Tomasi, 1994], are often chosen for the interest points detector in visual odometry and SLAM [Davison, 2003; Nister et al., 2004; Mouragnon et al., 2006]. But their efficiency is still not high enough in some high rate operation[Rosten et al., 2010], because they need to estimate the eigenvalues and eigenvectors from the matrix computed from image derivatives. In our research, we choose an efficient algorithm that is Features from Accelerated Segment Test (FAST) [Rosten and Drummond, 2006; Rosten et al., 2010] to detect the interest points for the entire image and then select the suitable interest points according to the distance to existing points in image.

4.4.1.1 Selection of new interest points

The FAST detector extracts lots of interest points for the new frame. However, some interest points have already existed in current frame obtained by guided matching. We define two principles for the new interest points. 1) The new interest points should keep away from the existing points. We set the minimal distance to its nearest existing image point as 20 pixels. If the minimal distance to an existing point is smaller than 20 pixels, it will not be selected as new interest point. 2) The distribution of the interest points should be as uniform as possible over image. A similar strategy as non-maximal suppression in FAST detector is taken to filter interest points, but we apply a larger mask which is a 20×20 window over the image. Only one key point is kept in one window which has the largest response.

4.4.1.2 Precise localization for the interest points

The position of $\mathbf{p}$ in figure 4.9(b) is the precise corner where we aim to approach, but the image point extracted via FAST detector might locate at an arbitrary pixel around $\mathbf{p}$.



(a) Ideal corner (b) Real corner

Figure 4.9: Corner with sub-pixel accuracy. $\mathbf{p}$ is the location of corner. The red lines are the tangent lines of the corner in given window.

Förstner and Gülch [1987] proposed an approximate solution to compute the sub-pixel accuracy corner. He supposed that the real corner is closest to all the tangent lines of the pixels in a neighborhood window. The optimal location of $\mathbf{p}$ was solved based on least-square. As shown in figure 4.9(b), the red lines are the tangent lines through one neighbor pixel in window. The solution is the point of intersection of the tangent lines [Belongie, 2000].

We note $\mathbf{q}_i$ as an arbitrary pixel location in window and $\bigtriangledown_{q_i}$ as the gradient at $\mathbf{q}_i$. For every tangent line, the vector from the $\mathbf{p}$ to a point $\mathbf{q}_i$ is orthogonal to the vector of image gradient at $\mathbf{q}_i$. Thus, the equation of tangent line through $\mathbf{q}_i$ can be expressed as:

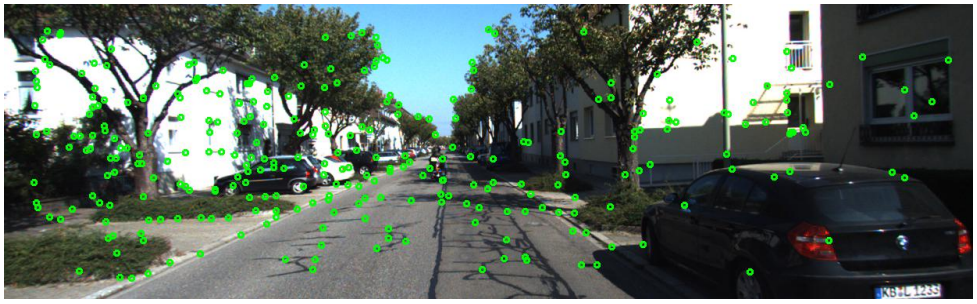
$$D_i(\mathbf{p}) = \bigtriangledown_{q_i}{}^T \cdot (\mathbf{p} - \mathbf{q}_i) = 0 \tag{15}$$

Every observation within the window can obtain one equation like equation 15. Our goal is to find the optimal $\hat{\mathbf{p}}$ point which can achieve the minimum perpendicular distance to all the tangent lines:

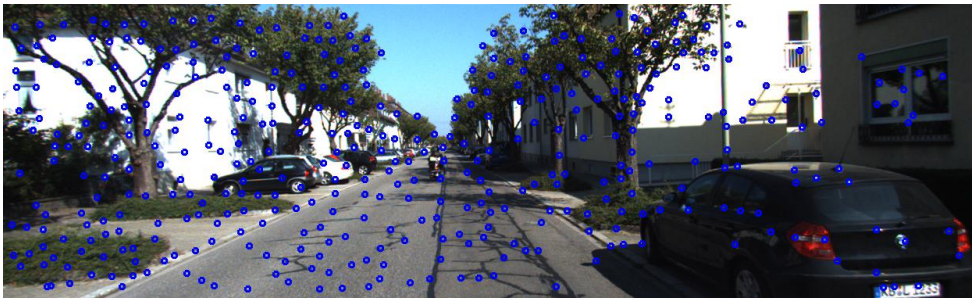
$$\hat{\mathbf{p}} = \underset{\mathbf{p}\in\mathbb{R}^2}{\operatorname{argmin}} \sum_{\mathbf{q}_i\in\mathcal{N}} D_i(\mathbf{p})^2$$

where $\mathcal{N}$ represents the window of neighborhood around the corner. This formulation express the sum of squares of all the distances from $\mathbf{p}$ to all lines in $\mathcal{N}$.

Figure 4.10 demonstrates the results of interest points detection in each step. The points in figure 4.10(a) are found by guided matching. Figure 4.10(b) shows the interest points detected



(a) Existing interest point obtained by guided matching.



(b) The interest points detected with the proposed methods.

(c) New interest points used for enriching of tie points.

Figure 4.10: An example of selection of new interest points.

by FAST algorithm and filtered with a 20×20 pixels grid. The new interest points in figure 4.10(c) are a subset of points in figure 4.10(b), which are selected with the method proposed in section 4.4.1.1.

With the detected interest points, the following steps are to search the correspondences for these interest points to generate new tie points. We will discuss the matching for different cases: monocular sequences, binocular sequences and multi-camera rigs.

4.4.2 Matching for monocular images

After guided matching for existing tie points, some correspondences between the new image and previous images have been built. These corresponding points can be used as prior knowledge for the matching of new interest points as shown in figure 4.10(c). We note current image as I_t and match the new interest points into the nearest key frame I_{t-1}. Our aim is illustrated in

figure 4.11.

Figure 4.11: Matching for monocular case. Red cross in I_t represents the new interest point, the yellow rectangle is the searching area we expect to find the correspondence of the new interest point.

We have a new interest point (locating at the center of red cross) in new image I_t, the goal is to search its correspondence in a limited area in image I_{t-1} (see the yellow rectangle in figure 4.11). Our solution is to fit a mathematical model using the existing correspondences between image I_t and I_{t-1} obtained by guided matching as shown in figure 4.12.

In computer vision, the transformation between two images is usually homography. For instance, we assume one image point $\mathbf{x}$ in I_t, it can be mapped into image I_{t-1} by

$$\mathbf{x}^{'} = H\mathbf{x}$$

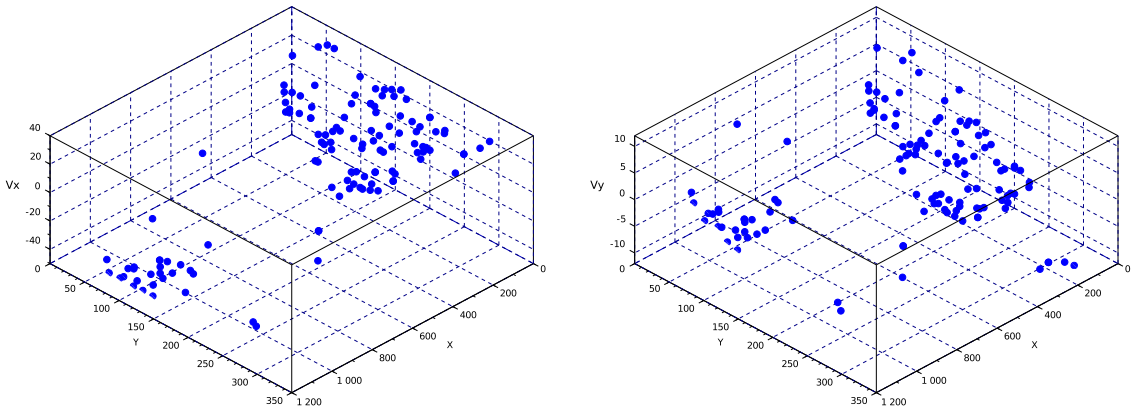
where $\mathbf{x}^{'}$ is its correspondence in I_{t-1} and H is a nonsingular 3×3 matrix [Hartley and Zisserman, 2003]. However, the homography transformation is true only if the all image points are coplanar in 3D space. Unfortunately, it is not the case for images captured using a forward looking camera in urban field, because the points on road and on the facades of buildings are not coplanar. The homography model could be used unless the objects in images such as road, facades, are segmented and then the homography can be used to map the image points for each object separately. But this is not the case in this thesis. We aim to generate the searching area using one function for all points.

Let's define $\boldsymbol{v}$ as pixel displacement for corresponding points from image I_t to I_{t-1}. We note $\mathbf{x}_i = [x_i, y_i]^T$ as an interest point in I_t and $\mathbf{x}_i^{'} = [x_i^{'}, y_i^{'}]^T$ as its correspondence in I_{t-1}. Then:

$$\boldsymbol{v}_i = \mathbf{x}_i^{'} - \mathbf{x}_i,$$

which is the vector from red point to blue point as shown in figure 4.12. Consider the displacements at x and y axes individually. Figure 4.13 illustrates the displacement of existing matches in a space defined by the image plane and the displacements. Each discrete point in figure 4.13(a) is expressed using $[x_i, y_i, v_i^x]$ while each discrete point in figure 4.13(b) is expressed like $[x_i, y_i, v_i^y]$

Figure 4.12: Existing correspond points between I_t and I_{t-1}. Red circle: the locations of image points in I_{t-1}. Blue circle: Matches of achieved by guided matching in section 4.3. Green lines: displacements of pixels.



(a) (b)

Figure 4.13: Discrete values of pixel displacement. (a) Displacement in x axis.(b) Displacement in y axis.

The discrete points drawn in figure 4.13(a) and figure 4.13(b) can be expressed using an unknown function $\Psi(x, y)$ with respect to the location of $\mathbf{x}$, written as:

$$\upsilon = \Psi(x, y).$$

υ can be either the displacement in x or y directions.

For new interest points in I_t, the issue is that we don't know their correspondences in image I_{t-1}. This problem can be solved if we can estimate the displacements for the new interest points. To approach this, we need to know the equation of $\Psi(x, y)$. In this thesis, we suppose that the pixel displacements from image I_{t-1} to I_t varies continuously. Thus, $\Psi(x, y)$ can be fitted using a high-order surface equation. We use bi-cubic interpolation to predict the displacement at x and

y directions separately:

$$\begin{cases} \boldsymbol{v}^x = \sum_{i=0}^{3}\sum_{j=0}^{3} a_{ij}x^i y^j \\ \boldsymbol{v}^y = \sum_{i=0}^{3}\sum_{j=0}^{3} b_{ij}x^i y^j \end{cases} \tag{16}$$

The coefficients $a_{ij}, \{i = 0, ..., 3; j = 0, ..., 3\}$ and $b_{ij}, \{i = 0, ..., 3; j = 0, ..., 3\}$ of the function can be estimated using the existing pairwise image points and at least sixteen pairwise points are needed.

With the bi-cubic equation, the displacements for new interest points can be predicted, then the approximate correspondences of new interest points can be obtained. However, this is only an approximation, we need to find the precise matches for new interest points in image I_{t-1}. To do this, we generate a searching area around the approximate locations. In this thesis, the size of searching area is fixed as a 30×30 pixel window. Then the precise matches of new interest points are searched in their corresponding searching areas.

4.4.3 Matching for stereo images

For binocular image sequences, the matching is not only between images along moving direction, but also includes the cross matching between left and right images. To find the new tie points for stereo case, we consider a circle approach which matches between left, right and two consecutive images [Geiger et al., 2011; Cvišić and Petrović, 2015]. The setup of the circle matching is shown in figure 4.14 which presents the four steps to obtain the correspondences for one new interest point in left image.

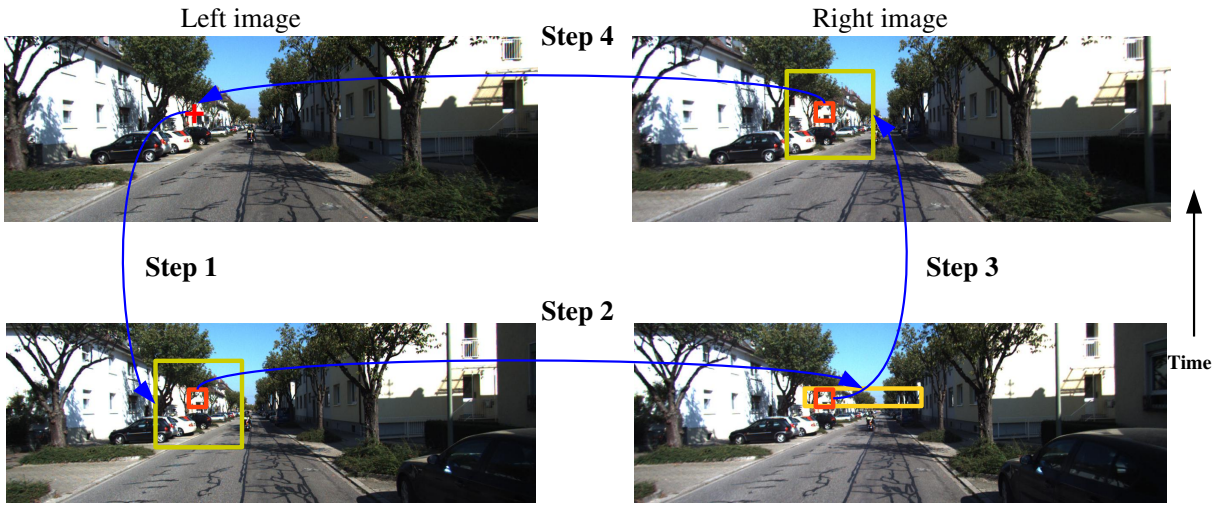


Figure 4.14: Circle matching for consecutive image pairs.

Starting from new interest points in left image (red cross), the first step is to find its corresponding point in previous left image with the matching method proposed for monocular case, the

correspondence is shown as the red square. The second step is to search the correspondence of the new matching point in previous left image in previous right image (*cf.* step 2 in figure 4.14). The third step is to match the point in previous right image with current right image with the matching method for monocular case. Finally, we check the circle matching results by matching the point in current right with the starting interest point in current left image. If the coefficient calculated by NCC is larger than the given threshold, the circle matching is successful. Otherwise, we remove the matches. The same steps are applied for all the new interest point in current left image. For new interest points in current right image, the same strategy of circle matching is employed, but the order of the steps is inverse. It starts from current right and is matched with previous right image. Then precess the previous left image and do the matching with current left image. The final step is to check the circle matching between current left and right images.

In the original circle matching method proposed by Geiger et al. [2011], the centers of searching area in matching step 1 and step 3 are determined only using the coordinates of new interest point, that is $\mathbf{x}^{'} = \mathbf{x}$. Then searching the precise matches in a fixed window. This method is simple but it is only suitable for correspondences having small displacement. However, the displacement is large for small depth objects. In this thesis, the centers of searching areas are predicted using a polynomial function in matching of monocular case. So our method can have a more precise position for matching that enables us to reduce the searching area and improve the matching robustness. For the cross matching between left and right images as shown in step 2 in figure 4.14, we also employ the epipolar constraints. But the difference is that we learn the searching direction and distance based on the existing pairwise points. The following paragraphs will introduce how to implement our new epipolar constraints.

4.4.3.1 Epipolar geometry

In stereo rig, the relative translation and rotation between left and right cameras are fixed in the operation of localization, thus the fundamental matrix can be estimated with the first image pair at the beginning, then it can be used for all the other image pairs. We denote $\mathbb{F}_{lr}$ as the fundamental matrix from left to right and $\mathbb{F}_{rl}$ represents the matrix from right camera to left. The $\mathbb{F}_{lr}$ and $\mathbb{F}_{rl}$ are 3×3 matrix. In epipolar geometry, the corresponding homogeneous image points $\mathbf{x}_j, \mathbf{x}_j^{'}$ in image I and $I^{'}$ hold the formula:

$$\mathbf{x}_j^{'T} \mathbb{F} \mathbf{x}_j = 0 \tag{17}$$

where $\mathbb{F}\mathbf{x}_j$ presents the epipolar line in image $I^{'}$, which passes through point $\mathbf{x}_j^{'}$. Therefore, if we know the fundamental matrix between two images, the searching of matching point can be limited on an epipolar line. The fundamental matrix can be estimated with a set of pairwise image points using the methods proposed in [Hartley and Zisserman, 2003; Armangué and Salvi, 2003; Luong and Faugeras, 1996]. In our implementation, a robust scheme is applied based on RANSAC.

4.4.3.2 Matching with the constraints of epipolar geometry

We know that the good matches of new interest points lie on their corresponding epipolar lines in matching image. The general solution is to search all the pixels in entire epipolar line to find the best matches. Our idea is to find the matches in a short segment in epipolar line. To do this, we should solve the following three issues: 1) where is the start point for searching on epipolar line, 2) which direction should be searched along epipolar line, 3) how long should be searched.

Start point for searching Define the general equation of epipolar line in image as :

$$ax + by + c = 0,$$

which is computed from fundamental matrix. In this thesis, we focus on horizontal stereo rig and the images are not rectified to epipolar geometry, thus, for one new interest point $\mathbf{x}_i = [x_i, y_i]^T$ in image, the start point on epipolar line is:

$$\begin{cases} x_0^{'} = x_i \\ y_0^{'} = -\dfrac{ax_i + c}{b} \end{cases} \tag{18}$$

This means that the location of the start point in target image is related to the position of the candidates in reference image.

Searching direction There are only two possible directions for searching at the start point in epipolar line. We intend to learn the direction from the guided matching according to displacement vector $\boldsymbol{v}$. In fact, if we know the direction of $\boldsymbol{v}$ in either x or y axis, the searching direction can be determined. For horizontal stereo rig, the displacements in x axis are bigger than those in y axis most of time, so we only consider the status of v^x. We collect all the values of v^x from existing matches, the principle for direction learning is :

$$\overrightarrow{\mathcal{D}} = \begin{cases} left & \sum_i^M v_i^x > 0 \\ right & \sum_i^M v_i^x < 0 \end{cases} \tag{19}$$

where, $\overrightarrow{\mathcal{D}}$ represents the searching direction in epipolar line and M is the number of existing matches. We should mention that this method is only suitable for the horizontal stereo rig. For vertical stereo rig, we can consider the displacement in y axis.

Searching scope In our implementation, we set the searching distance as 1.5 times of maximum displacement, written as:

$$\delta_d = 1.5 \cdot max\{\|\boldsymbol{v}_1\|, \|\boldsymbol{v}_2\|, ..., \|\boldsymbol{v}_M\|\} \tag{20}$$

where, δ_d is the length for searching, $\|.\|$ represents the norm of displacement vector.

Figure 4.15 is an example of matching for a stereo pair. The purpose is to match the interest points in left image with those in right image. Instead of searching whole epipolar, we only search a small part of the epipolar line to find the best matches. The epipolar line could have some errors, so the matches on epipolar lines would not very accurate. In order to overcome these problems, we refine the matches by finding the locations in a small window around the matched location in epipolar line. In our experiment, we set the size of the searching window as 1.5 times larger than the image template for each interest point.

Figure 4.15: Matching of a stereo pair. The top image is the captured by left camera and the bottom one is the right image. The start points and searching scope in epipolar line for the interest points in left image are drawn with same color.

4.4.4 Matching for multi-camera images

Previous two sections introduce the matching of new interest points in monocular and binocular cases. For the other camera configurations such as four cameras cluster (two forward looking and two backward looking) and non-overlap stereo (one forward looking and one backward looking), the matching for new interest points is explained in this section.

The matching for non-overlap stereo can be considered as two mono cameras separately, that is to start the matching of forward looking camera with the strategy proposed in section 4.4.2 at first. Then the same way can be applied for the backward looking camera.

Similarly, the four camera case can be divided into two stereo cases that are one forward looking

and one backward looking stereo. For each stereo case, the method introduced in section 4.4.3 can be applied for matching.

For other multi-camera configurations, no matter how many cameras they have, it can be always divided into a combination of stereo and monocular cases. Then the strategies introduced in sections 4.4.2 and 4.4.3 are used for matching for each case.

4.5 Experiments of new tracking methods

As we presented at the beginning of this chapter, our previous matching and tracking strategy suffers from problems when we test our localization method proposed in chapter 3 on KITTI benchmark for the high speed road case. With the propagation based tracking and matching method, we test our method using the same datasets again.

The images used for visual odometry are captured by a forward looking stereo rig and the provided images are rectified with the calibration parameters. There are eleven stereo sequences for training and accurate ground truth (<5cm) is provided by a GPS/IMU system with RTK corrections enabled for each sequence [Geiger et al., 2012, 2013] .

4.5.1 Relative Errors

To compare the accuracy of localization for each sequence, a measure that operates the relative geometric relations between poses along the trajectory, is used to evaluate the performance of visual odometry. In previous chapter, we compute absolute errors of every estimated poses with respect to the ground truth to evaluate the localization accuracy. However, this kind of methods could mislead the comparison of localization performances, because this measure strongly depends on the point in time step where the error has been made [Geiger et al., 2012]. For instance, the translation errors could increase with growing of trajectory length, or the rotational errors earlier in the sequence lead to larger end-point errors.

Kümmerle et al. [2009] proposed a new method to evaluate the accuracy of localization which aimed at computing a relative measure in a fixed distance over sequence. The measure mixed the performance of rotation and translation errors together to express the performance of localization. This method was extended by Geiger et al. [2012], that considered the performance of translation and rotation separately. The measures for translation and rotation are defined as:

$$\begin{cases} e(\mathcal{L})_r = \dfrac{1}{|\mathcal{L}|} \displaystyle\sum_{(i,j)\subset\mathcal{L}} \angle[(\hat{\mathcal{P}}_j \ominus \hat{\mathcal{P}}_i) \ominus (\mathcal{P}_j \ominus \mathcal{P}_i)] \\ e(\mathcal{L})_t = \dfrac{1}{|\mathcal{L}|} \displaystyle\sum_{(i,j)\subset\mathcal{L}} \|(\hat{\mathcal{P}}_j \ominus \hat{\mathcal{P}}_i) \ominus (\mathcal{P}_j \ominus \mathcal{P}_i)\| \end{cases} \tag{21}$$

where, $\mathcal{L}$ stands for frame group and (i, j) is one image pair in the group. $|\mathcal{L}|$ is the trajectory length from one frame to another in each image pair. $e(\mathcal{L})_r$ and $e(\mathcal{L})_t$ represent the relative

errors for rotation and translation. $\mathcal{P}$ is a 4×4 matrix which contains the rotation and translation:

$$\mathcal{P} = \begin{bmatrix} \boldsymbol{R}_{3\times3} & \boldsymbol{\nu}_{3\times1} \\ 0 & 1 \end{bmatrix}$$

$\boldsymbol{R}_{3\times3}$ is rotation matrix and $\boldsymbol{\nu}_{3\times1}$ is translation vector of camera center. $\ominus$ stands for the inverse composition operator of image motion [Geiger et al., 2012].

4.5.2 Evaluation on training datasets

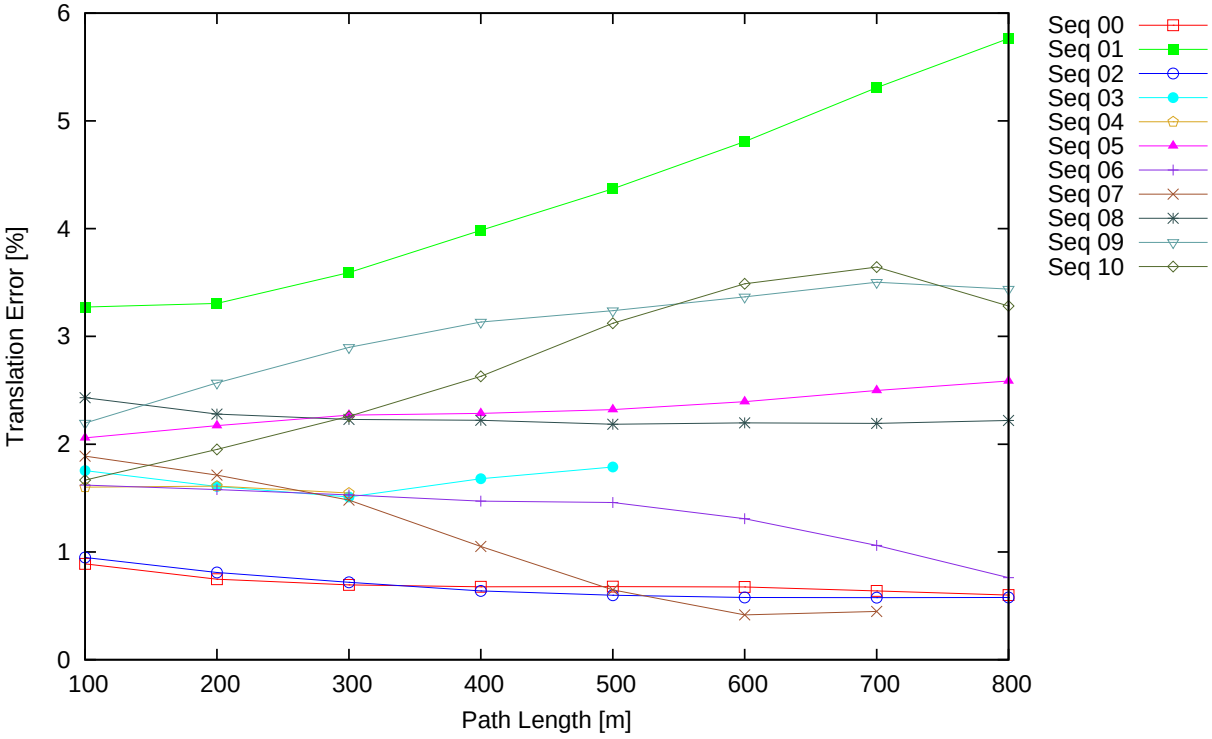
The relative rotation and translation errors are estimated on the images with fixed distances. The distance list is filled with 100m, 200m, 300m, 400m, 500m, 600m, 700m and 800m. It means that the relative errors are computed using equation 21 every 100m, 200m, 300m, 400m, 500m, 600m, 700m and 800m from the beginning of trajectory. Finally, the average values are computed on each fixed distance value. Eleven sequences captured in different scenarios with ground truth are used to evaluate the approach. We test vision based localization method with matching and tracking method proposed in chapter 3 and the localization with propagation based matching and tracking method proposed in this chapter. The average translation and rotation errors on all the eleven sequences are computed in table 4.1. The accuracy on both translation and rotation is improved a lot by using propagation based matching and tracking method for localization. The translation error is reduced from 1.62% to 0.95% and the rotation errors is reduced from 0.0045 deg/m to 0.0034 deg/m

Table 4.1: Comparison of relative errors on translation and rotation.

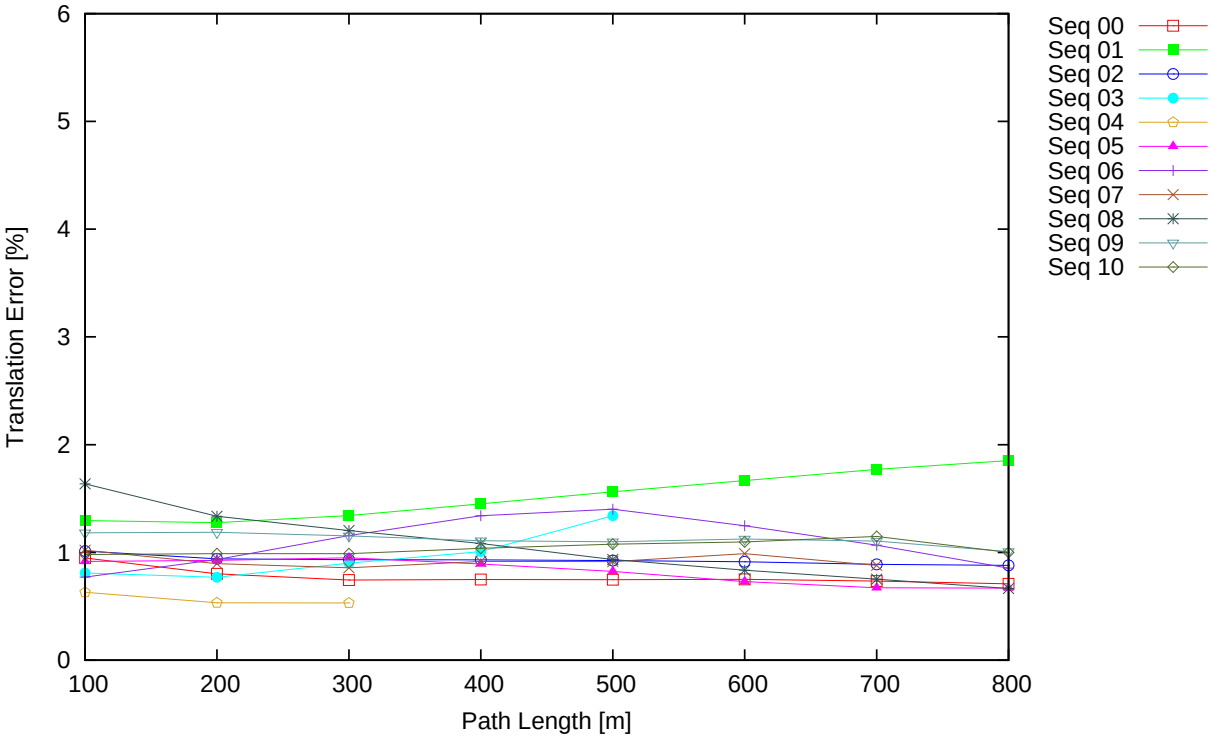
	Translation error (%)	Rotation error (deg/m)
Original	1.62	0.0045
New method	0.95	0.0034

The translation errors and rotational errors on training datasets for each sequence are shown in figure 4.16 and figure 4.17. Horizontal axis stands for the length trajectory and vertical axis presents the average relative errors.

Table 4.1 represents the average errors for both translation and rotation in figure 4.16 and figure 4.17. Figure 4.16(b) illustrates that all the translation errors are less than 2.0% and most of them are less than 1.2% for propagation based matching and tracking while the translation errors obtained by original method are larger than 1.0% for most of sequences. For sequence 01 which is the most challenging sequence in training datasets, the translation error is more than 3.0% at the beginning and increases to almost 6.0% at the end using original method, but the translation errors are between 1.2% and 1.8%, which has great improvement by applying propagation based matching and tracking method. The accuracy of rotation is also improved significantly by comparing the results in figure 4.17(a) and figure 4.17(b).

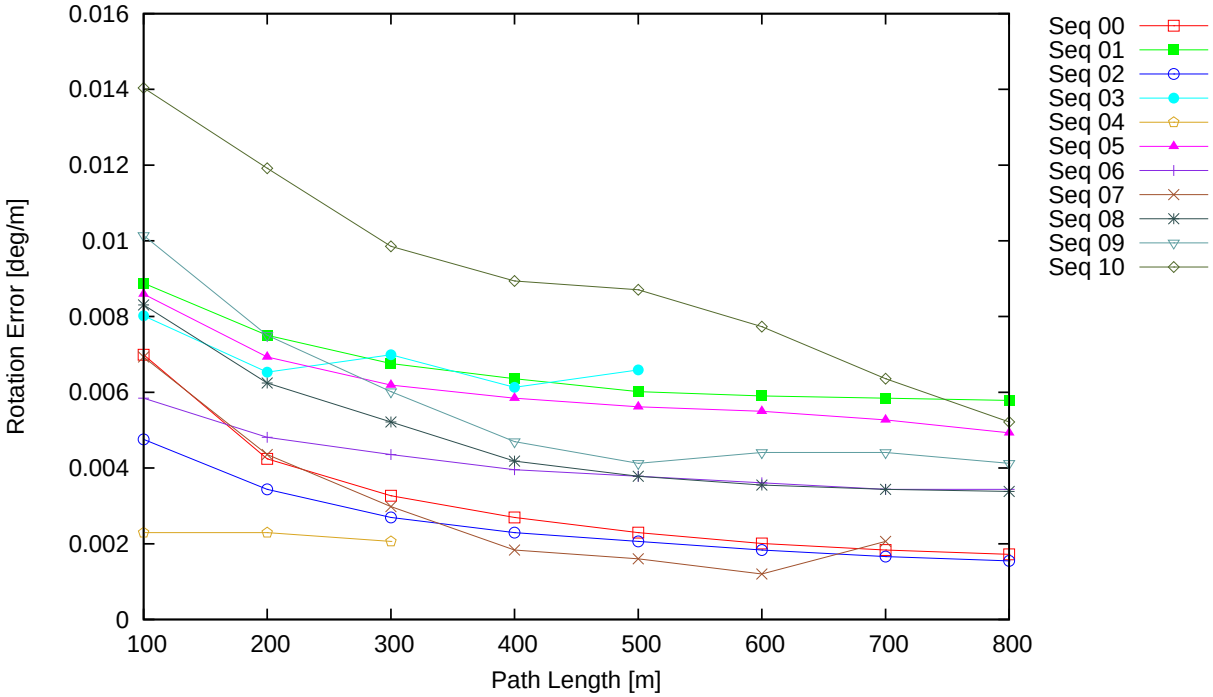


(a) Original matching and tracking.

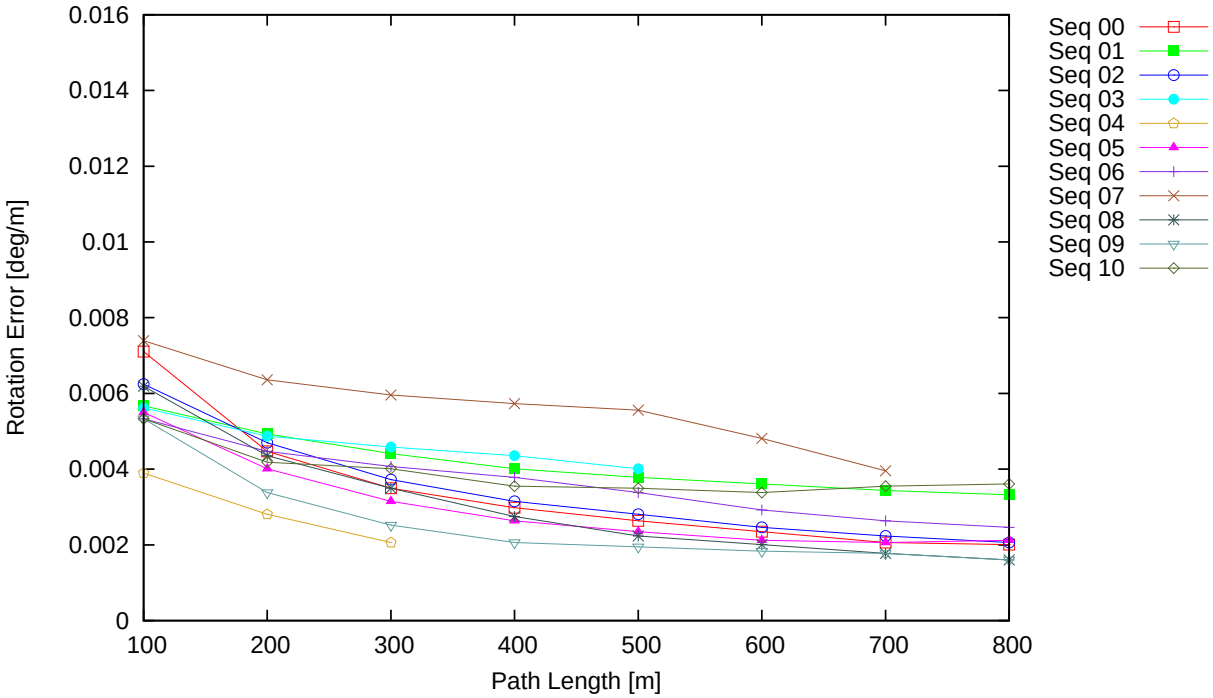


(b) Propagation based matching and tracking.

Figure 4.16: Accuracy of translation for KITTI training datasets.



(a) Original matching and tracking.



(b) Propagation based matching and tracking.

Figure 4.17: Accuracy of rotation errors.

4.5.3 Evaluation on test datasets

There are 22 sequences in KITTI benchmark website for odometry. The ground truth of poses for first 11 sequences (00-10 sequences) are known for us that we used to evaluate our approach in previous section. At the same time, we can summit the estimated poses of the rest (11-21) sequences on KITTI site to compare the method with the state-of-the-art methods. We call 11-21 sequences as test datasets.

The average translation error for test datasets is 1.25% and the average rotation error is 0.0041 deg/m using our method, which is called SLUP on the website [1]. Comparing with the average errors for training datasets (*cf.* Tab 4.1), the accuracy of test datasets is not as good as training datasets which is 0.95% for translation error and 0.0034 deg/m for rotation error. By analyzing the sequences, we found that there are three sequences (12, 20, 21) out of test datasets captured on highway which have poor texture. For those sequences, many points of interest would be detected and tracked on moving vehicles on road which could lead to poor pose estimation during localization. In fact, we improved the accuracy of localization significantly for sequence 01 in training datasets by using propagation based matching and tracking method (comparing results in figure 4.17(a) and figure 4.17(b)). However, the translation error for sequence 01 is still bigger than other sequences in training datasets (*cf.* figure 4.17(b)). This is caused by the reason that the ratio of moving interest points among all the detected interest point for sequence 01 is higher than others sequences in training datasets. For the same situation, the accuracy of localization using sequences 12, 20, 21 would not so good that influences the average translation error and rotation error in test datasets.

Table 4.2 presents the performance of three state-of-the-art visual odometry methods and our method published on KITTI benchmark suite. The best translation accuracy can approach

Table 4.2: Accuracy of state-of-the-art visual odometry methods.

Method	Translation error	Rotation error (deg/m)	Description
SOFT[Cvišić and Petrović, 2015]	0.88%	0.0022	Feature selection
RotRocc[Buczko and Willert, 2016a]	0.88%	0.0025	Normalized Reprojection Error
ROCC[Buczko and Willert, 2016b]	0.98%	0.0028	feature-adaptive scaling
SLUP	1.25 %	0.0041	our method

0.88% which is achieved by SOFT [Cvišić and Petrović, 2015] and RotRocc [Buczko and Willert, 2016a]. The SOFT estimates the poses using selected tracks along sequences according to the length of tracks, because the longer tracking of interest points amongst multiple frames

[1] http://www.cvlibs.net/datasets/kitti/eval_odometry.php

are found, the better quality is obtained for pose estimation. RotRocc can get same level of translation accuracy (0.88%), but the rotation accuracy is slightly worse than SOFT. RotRocc normalized the back-projection errors that removes the impact of different depth of 3D object points for the back projections. The similar method is used in ROCC [Buczko and Willert, 2016b] that applies feature-adaptive scaling for back-projection errors to make them almost invariance to the 3D position of each feature. It provides a good criterion to remove outliers that can reject the inaccurate matches. Comparing our results with these methods, we should improve our method in future work. The algorithms such as feature selection, back projection scaling proposed by previous algorithms, give us some new ideas to improve the accuracy.

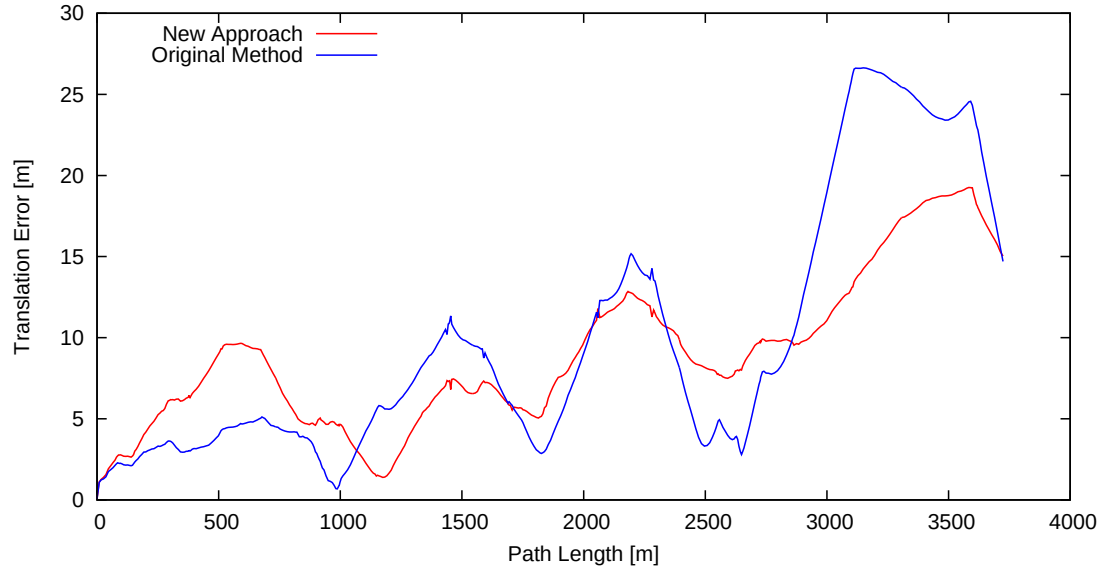
4.5.4 Absolute errors and trajectories

Previous sections presents the errors in relative. Now, let's select some typical cases to show the absolute errors of localization. In this thesis, we choose two cases which are the best and the worst cases achieved by original matching and tracking method. The best accuracy is approached in sequence 00 while the most difficult case is sequence 01. We compute the absolute errors compared with ground truth for both methods and draw the trajectories. The results are shown in figure 4.18 and figure 4.19.

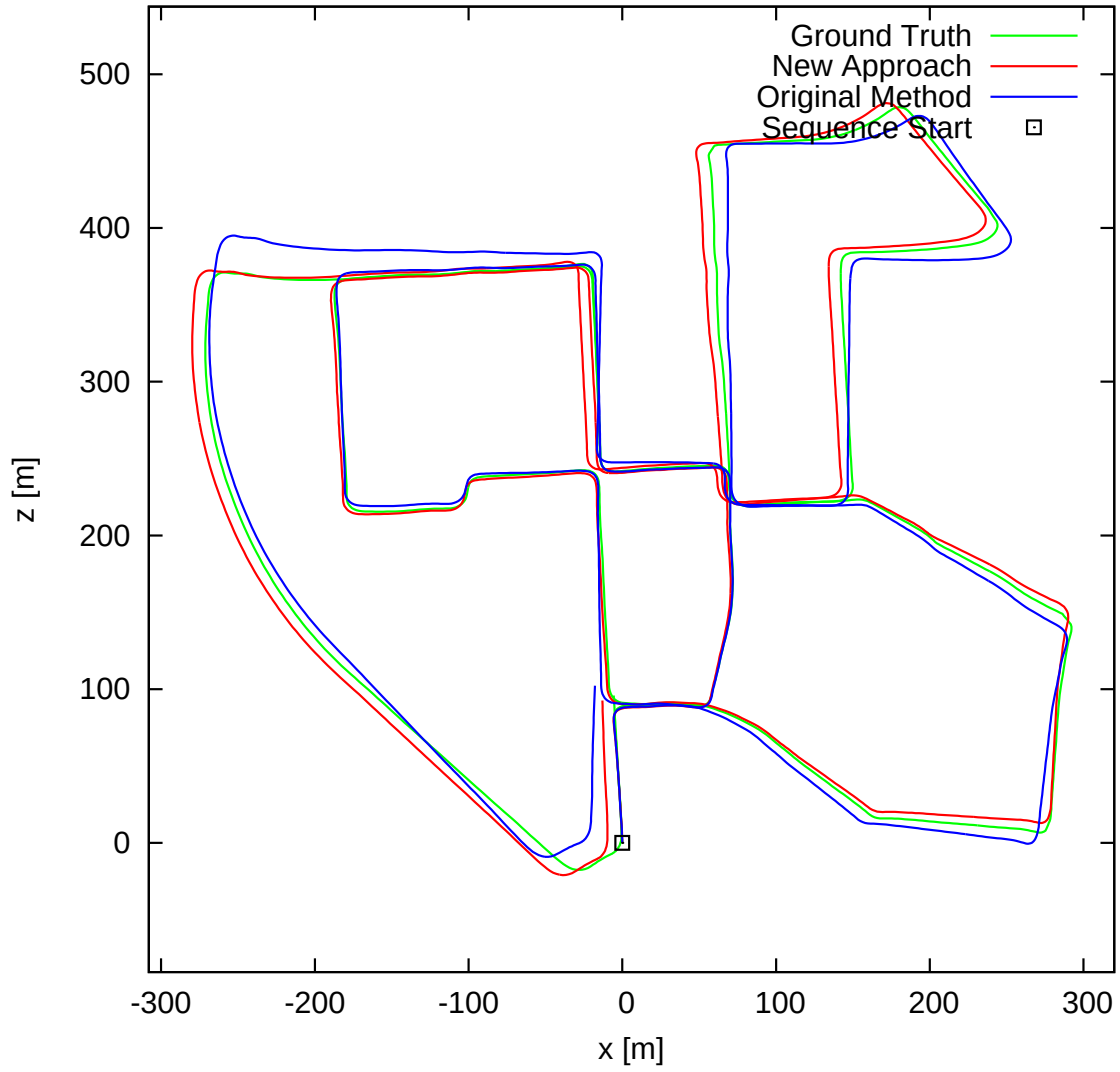
Let's analyze the absolute errors first (cf. Fig 4.18(a)). At the beginning of trajectory, the original matching and tracking performs better than propagation based method, then their performances become similar, but the drift increases quickly after $2.5km$ trajectory for original method while the increasing speed of propagation based approach is quite slow. We draw the trajectories estimated using two different matching and tracking strategies, as shown in figure 4.18(b). We can see the drift at the end of trajectory using original method, which is larger than the propagation based tracking and matching method.

The most challenging case for original matching and tracking method is the sequence 01 in KITTI datasets. The original matching and tracking method suffers from the matching ambiguity problem as we presented at the beginning of this chapter because of the repeatable texture on road and roadsides. There are lots of false matches, which cause very poor pose estimation, thus the absolute errors increase very quickly (cf. Fig 4.19(a)). Figure 4.19(b) illustrates the trajectories. The blue trajectory shows the path estimated using original method which turns to a wrong direction. The red line shows the results estimated using propagation based approach. We can see the accuracy is improved a lot with the new approach.

From the results drawn in figure 4.18 and figure 4.19, we can say that the robustness of using propagation based matching and tracking method for visual odometry is improved. In some cases, the original matching and tracking method using SIFT feature obtains more accurate position than the new method, this could get benefit from massive matches along sequences. For the new matching and tracking method, the number interest points are limited, which is less than 400 per image.

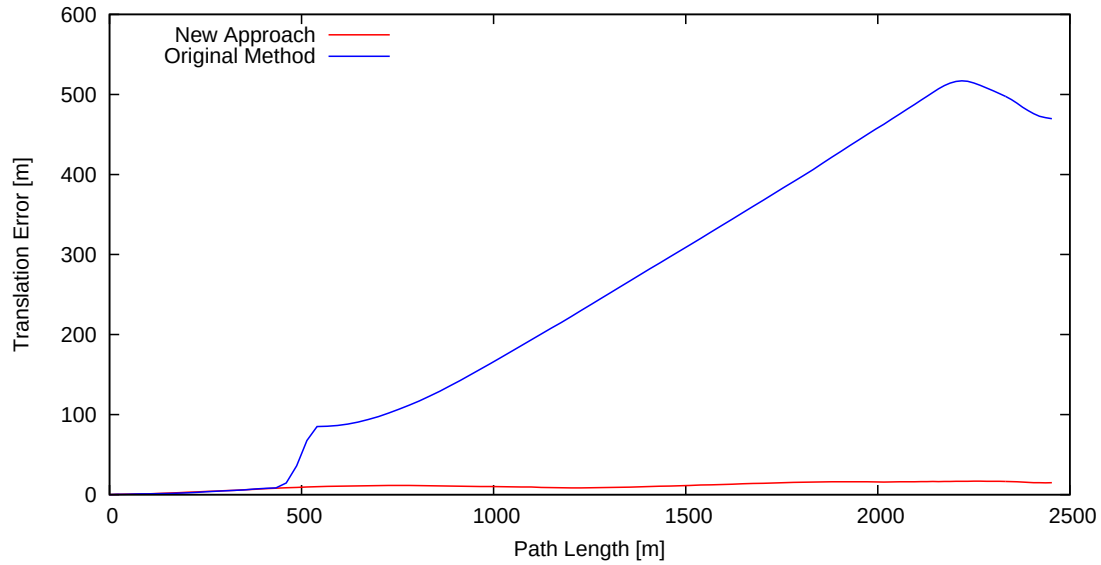


(a) Absolute errors

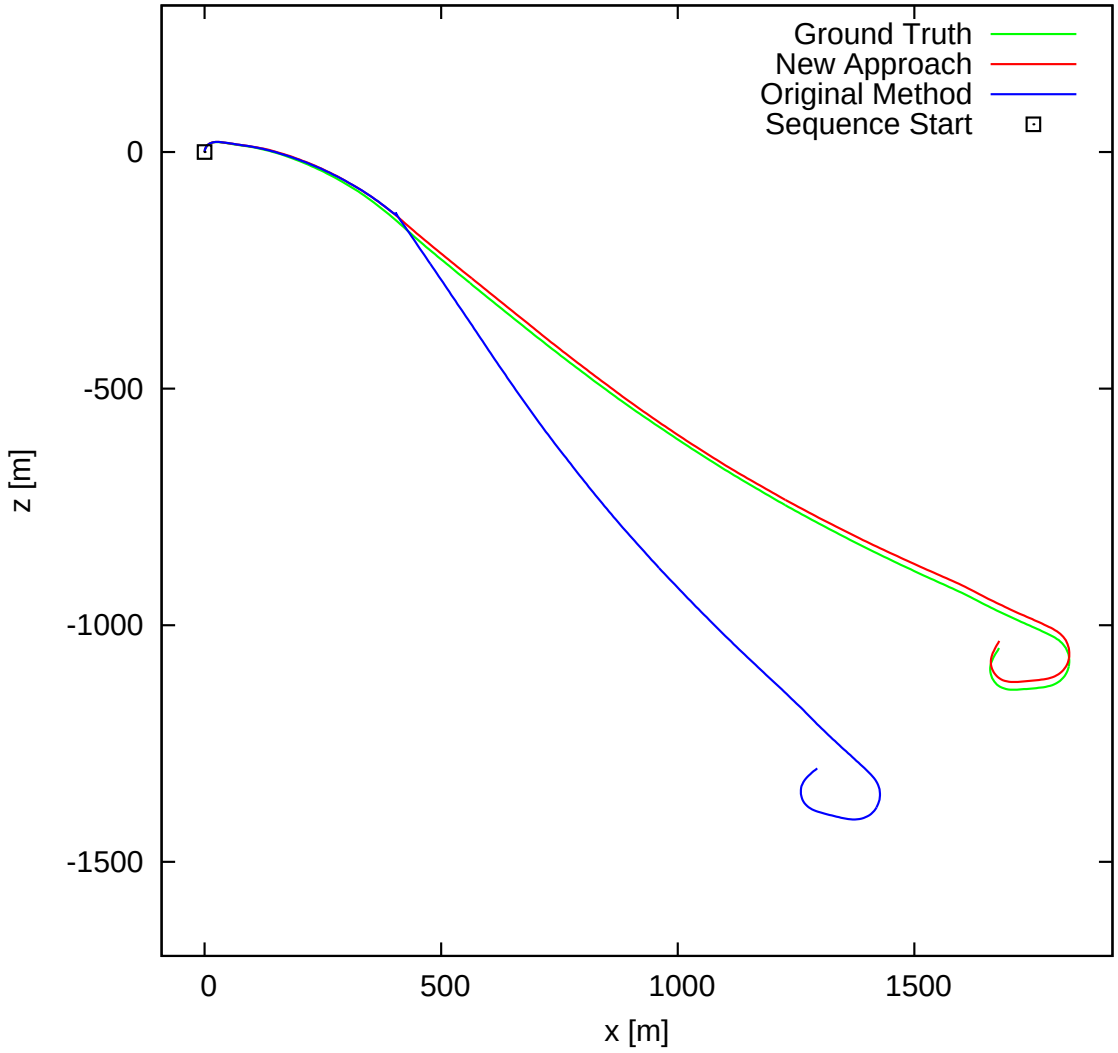


(b) Trajectories

Figure 4.18: (a) Absolute errors of positions. (b) Trajectories of ground truth, estimated using original method and new approach for sequence 00.



(a) Absolute errors



(b) Trajectories

Figure 4.19: (a) Absolute errors of image positions. (b) Trajectories of ground truth, estimated using original method and new approach for sequence 01.

4.5.5 Efficiency analysis

From the analysis of efficiency in chapter 3, we found feature extraction, matching and tracking are the most time-consuming parts. This is also a reason that motivates us to develop propagation based matching and tracking. Table 4.3 gives us a global idea about the efficiency improvement using the new method, which presents the average time spent on each image pair.

Table 4.3: Processing time for the original and new tracking and matching methods.

	Feature extraction	matching and tracking
Original	0.64s	0.4s
New	0.04s	0.17s

From the previous table, we can see that FAST detector is much faster (more than ten times) than SIFT. It is a real-time detector. The efficiency of matching and tracking is also improved about two times with the new approach, compared with the FLANN based matching and graph based tracking. However, the efficiency of propagation based tracking and matching should be improved further for real-time application. In fact, most of the time spent in this part, is caused by the computation of NCC. In current implementation, we compute NCC between patches one by one. This can be implemented with parallel computing techniques in the future to speed up the process.

4.6 Conclusion

This chapter presented a propagation based matching and tracking method to improve the performance of localization. The pose of new frame was predicted by a motion model and the uncertainty was estimated for the prediction. With the prediction, the searching of existing tie points were guided considering uncertainty propagation. We used FAST algorithm to detect points of interest in image and a constrained matching method was proposed for matching of new tie poitns. The proposed matching and tracking method was tested using datasets for visual odometry in KITTI benchmark sites. From the experiments results, both accuracy and efficiency were improved by using propagation based matching and tracking method. We also submitted our results to the benchmark and average translation error is 1.25%. In our future work, we can improve the localization performance by filter the interest point detected on moving objects in images and also taken into account the advantages of the state-of-the-art methods published on KITTI site.

Chapter 5

Geo-referenced landmarks based localization

5.1 Overview

With propagation based matching and tracking (*cf.* chapter 4), robust correspondences can be obtained for pose estimation. Then, poses, as well as tie points, are optimized with LBA. However, the drift of localization is still cumulated over time, because LBA only concerns the local accuracy of the trajectory [Scaramuzza and Fraundorfer, 2011]. To achieve accurate localization in global, one popular solution is loop closure. It implies that the robot should remember its visited path and keep tracks of all the previous history of images. In practice, the loop closure is usually used in visual SLAM for small workspace, but it has difficulties in large scale environment. On the one hand, the quickly increasing tracks in vision based localization makes it heavy for memory and computation. On the other hand, many trajectories do not have any loops. In this case, to reduce the drift of vision based localization, the external data such as GNSS, maps and geo-referenced landmarks, should be taken into account.

As we discussed in chapter 2, different types of external data (e.g. GNSS [Agrawal and Konolige, 2006; Lhuillier, 2012; Shi et al., 2012], maps [Alonso et al., 2012; Gupta et al., 2016], 3D building models [Larnaout et al., 2012; Arth et al., 2015b] and ortho-photos [Jaud et al., 2013; Ji et al., 2015]) have been integrated with vision based localization. In this thesis, we aim at integrating geo-referenced semantic objects (e.g. traffic signs and road markings), which have specific geometry and texture information. Those features make them being detected precisely and matched easily. We take traffic signs and road markings as landmarks in our method. Only keyframes are taken into account for integration. Figure 5.1 shows the pipeline of the integration (highlight forms). Our strategy is inspired by the knowledge about ground control in photogrammetry. The Ground Control Points (GCPs) are measured in world coordinate system and the corresponding image control points are measured in images [Malmström, 1986]. Then, these GCPs and image control points are combined with bock bundle adjustment to improve

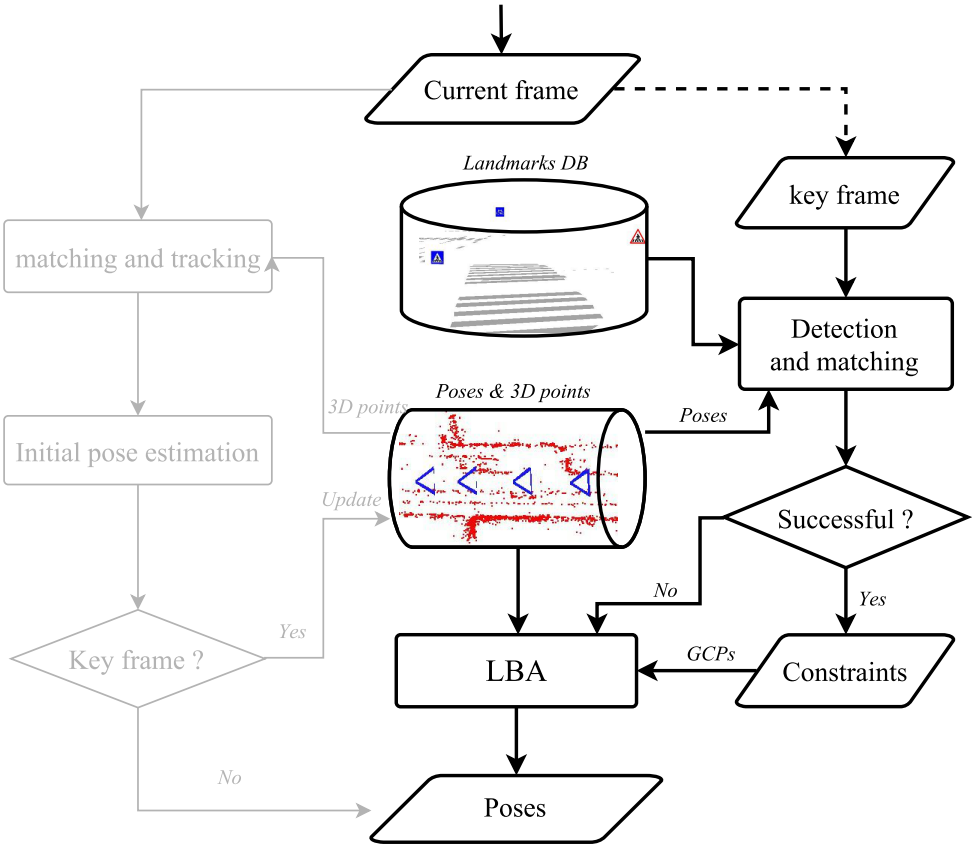


Figure 5.1: Flowchart of integration of geo-referenced landmarks with localization.

the accuracy. In our research, we combine GCPs with LBA. The GCPs are generated from geo-referenced landmarks. We aim to develop the methods to match and measure the image control points automatically. Section 5.4 will introduce our equation system for GCPs integration. An automatic strategy of image control points detection will be explained in section 5.3.

In this thesis, the GCP and image control point are noted as:

- $\boldsymbol{X}_g$: GCP in geo-referenced frame.
- $\mathbf{x}_g$: image control point.

5.1.1 Compared with classical GCPs

In photogrammetry, the GCPs are usually used to establish the relations between the local coordinate system and absolute coordinate system. It is modeled in a rigid transformation with seven parameters (rotation, translation and scale) [Malmström, 1986; Schenk, 2005]. Thus, at least three GCPs (two X, Y, Z GCPs and one vertical Z GCP) and their correspondences in local coordinates system should be measured to resolve the transformation parameters. In practice, more GCPs are usually used to improve the accuracy of mapping.

In our approach, the localization starts from a known point in geo-referenced system, so we do

not need to register two different coordinate systems. Our objective is to correct the drift caused by error accumulation over time. In this case, we can generate constraints for LBA even if only one GCP is available.

5.1.2 GCPs and tie points

In our method, each GCP is in terms of X, Y, Z coordinates in absolute system. Each image control point contains u, v coordinates in image plane. For any GCP $\boldsymbol{X}_g^i$, we denote its image control point in j^{th} image as ${}^i\mathbf{x}_g^j$, so the data structure about GCPs for LBA is expressed as:

$$
\begin{array}{l}
data\_association \\
\{ \\
\quad \text{Point3D} \quad \boldsymbol{X}_g^i \qquad i^{th} \textit{ 3D GCP} \\
\quad \text{Mat3} \quad \boldsymbol{\Sigma}_{X_g^i} \qquad \textit{covariance matrix of } i^{th} \textit{ 3D GCP} \\
\quad \text{Point2D} \quad {}^i\mathbf{x}_g^j \qquad \textit{image control point in } j^{th} \textit{ image} \\
\quad \text{Mat2} \quad \boldsymbol{\Sigma}_{{}^i\mathbf{x}_g^j} \qquad \textit{covariance matrix of } {}^i\mathbf{x}_g^j \\
\quad \text{Point2D} \quad {}^i\mathbf{x}_g^k \qquad \textit{image control point in } k^{th} \textit{ image} \\
\quad \text{Mat2} \quad \boldsymbol{\Sigma}_{{}^i\mathbf{x}_g^k} \qquad \textit{covariance matrix of } {}^i\mathbf{x}_g^k \\
\quad \ldots \\
\}
\end{array}
$$

At first glance, the above-mentioned structure is similar as that of tie point proposed in section 3.1.2. Each 3D point corresponds a set of image points. Actually, we would say that they have the same structure, but the GCP $\boldsymbol{X}_g^i$ is known while tie point $\boldsymbol{X}$ is unknowns in LBA.

5.2 Geo-referenced landmarks

In this thesis, we regard geo-referenced traffic signs and road markings as landmarks for localization. These two types landmarks are reconstructed with the data acquired by STEREOPOLIS [Paparoditis et al., 2012]. Each landmark contains the following informations:

- Position in geo-referenced coordinate system.
- Category (e.g. prohibition, warning, obligation or road marking).
- Precision of reconstruction.
- Normal direction.

5.2.1 Geo-referenced traffic signs

The 3D traffic signs are generated using geo-referenced color images. The pipeline is presented in figure 5.2. There are two main techniques: extraction and reconstruction. The details of the methods are presented by Soheilian et al. [2013a]

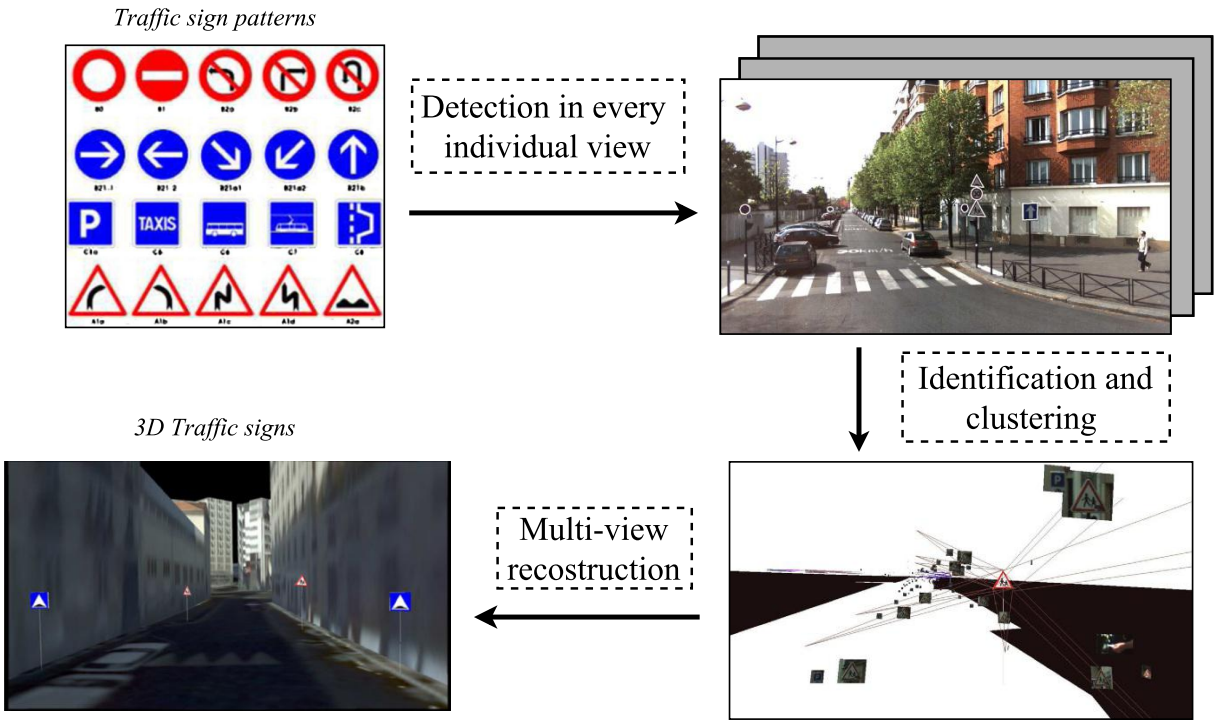


Figure 5.2: Reconstruction of traffic signs from geo-referenced images.

5.2.1.1 Traffic sign detection

During extraction procedure, the traffic signs are detected in every individual image. The first step for traffic sign detection is to determine the Region of Interests (ROIs). The ROIs are generated based on color segmentation. Because most of the traffic signs are blue or red in urban environment. Then, three simple shapes (ellipse, quadrilateral, triangle) are detected by Soheilian et al. [2013a] in ROIs. Because the geometric forms of traffic signs are circular (obligation and prohibition), quadrilateral(indication) or triangular (warning) in real world. Sub-pixel edge points are extracted for every detected signs, then the shape is estimated precisely with RANSAC scheme.

To identity the detected candidates, every local image patch of detected sign is rectified. The quadrilateral is rectified to rectangle with perspective transformation while ellipse and triangle are rectified to circle and equilateral triangle using affine. Then, the Zero-mean Normalized Cross Correlation (ZNCC) based template matching is performed to compare the rectified image patch with reference patterns. The traffic signs are identified according to the maximum ZNCC score. When the maximum score is larger than a given threshold, the candidate is accepted.

5.2.1.2 Traffic sign reconstruction

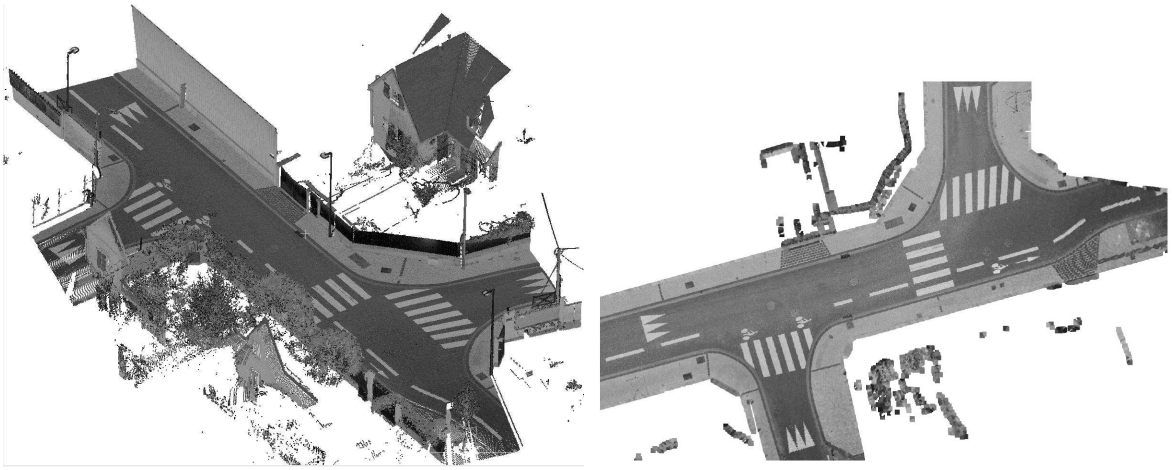
The traffic sign detection method proposed in previous section aim at detecting traffic signs in individual images. The images used for traffic sign reconstruction are captured by mobile mapping system, thus the geo-referenced pose for each image is known. In order to reconstruct the traffic signs, the same signs in images observed at different locations should be clustered (*cf.* the third step in Fig 5.2). To do this, a hypothesis generation and verification based approach was proposed [Soheilian et al., 2013a]. For one same traffic sign, its observations in images should have same identifications obtained by traffic sign detection in image. Meanwhile, the relationships between two corresponding traffic signs in two different images can be described by epipolar geometry which can be generated according to known poses of image poses. Using previous two principles, the traffic signs detected in individual images can be grouped. Then, the 2D traffic signs with known poses in same group are used to reconstruct one 3D traffic sign using a multi-view algorithm, integrating the priori constraints about traffic sign. The triangular and rectangular signs have been reconstructed in the work presented in [Soheilian et al., 2013a]. Thereafter, the method for circular sign reconstruction was proposed, where the uncertainty propagation of 2D ellipses is taken into account [Soheilian and Brédif, 2014].

5.2.2 Geo-referenced road markings

Many methods have been proposed for road marking detection and reconstruction. Some research detected the road markings from high resolution aerial images [Tournaire et al., 2006c; Kim et al., 2006]. With the development of Mobile Mapping System (MMS), more methods are proposed based on the data acquired by MMS. The road markings can be detected and reconstructed from stereo image [Soheilian et al., 2010] or point clouds [Yu et al., 2014a; Guan et al., 2014]. Most of the methods use a bottom-up strategy. They start from low level feature, then provide higher level objects with grouping algorithms. Some top-down strategies are also applied [Tournaire and Paparoditis, 2009; Yu et al., 2014a; Hervieu et al., 2015]. In contrast to bottom-up strategies, the top-down methods are more generic and can get benefit from the prior knowledge about road markings, but this kind of methods are slower than bottom-up based methods.

5.2.2.1 Intensity ortho-image

With the high rate LiDAR on STEREOPOLIS, dense point clouds on road surface can be obtained. In this case, we can build the road marking database from point clouds. A top-down approach is taken for road marking extraction from point clouds [Hervieu et al., 2015]. In this approach, the road markings are extracted from ortho-image of road, which is generated by the vertical projection of the points (*cf.* figure 5.3). The ortho-images with two channels (intensity and height) undergo a hole-filling filter to cope with the irregular LiDAR sampling.



(a) (b)

Figure 5.3: Generation of intensity ortho-image from point clouds. (a) Point clouds. (b) intensity ortho-image of the points.

The road marking map is generated using an extension of the method proposed by Hervieu et al. [2015]. We first summarize here this approach, then detail the proposed extensions, and finally lift the 2D extractions as a 3D road marking database.

5.2.2.2 Original approach

Within the intensity ortho-image, road markings are then searched for as occurrences of a translated/rotated/scaled rectangular road marking template instanced from a library of road markings (figure 5.4). This search space is modeled as a set of road marking types and for each

Figure 5.4: Library of road marking template patterns ($GSD = 2cm$).

type a fixed aspect ratio, an interval of scale and a template vector pattern delineating the white road marking area against a dark background. Thus the extraction of road markings boils down to finding a set of road markings $\mathcal{X} = (\ell_i, x_i, y_i, \theta_i, \lambda_i)$ parameterized by a type ℓ, a translation (x, y), a rotation by θ and a scaling λ (figure 5.5). The marking type defines a pattern I_ℓ that may be rasterized into the intensity orthophoto geometry using the affine transform $T_{x,y,\theta,\lambda}$ (denoted $T_{\mathcal{X}_i}$ for short).

Hervieu et al. [2015] formulate the road marking extraction as an energy minimization problem over the varying-dimension search space defined above, with an energy defined over a set of road markings $\mathcal{X} = (\mathcal{X}_i)_{i=1...n}$ as :

$$U(\mathcal{X}) = \sum_{i=1}^{n} u_1(\mathcal{X}_i) + \sum_{i<j} u_2(\mathcal{X}_i, \mathcal{X}_j) \quad (1)$$

$$u_1(\mathcal{X}_i) = f^0 - \max(0, ZMNC(I_{\ell_i}, T_{\mathcal{X}_i}^{-1}(I))) \quad (2)$$

$$u_2(\mathcal{X}_i, \mathcal{X}_j) = \beta \frac{\mid S(\mathcal{X}_i) \cap S(\mathcal{X}_j) \mid}{min(\mid S(\mathcal{X}_i) \mid, \mid S(\mathcal{X}_j) \mid)} \quad (3)$$

where f^0 is the cost of adding an object. A low value of f^0 (0.35 as used in Hervieu et al. [2015]) enables the optimization to add objects with lower correlation scores at lower costs. A high value, in contrast penalizes the objects with low correlation scores. It should provide a trade-off between the number of over-detections and under-detections. In the present work we chose a higher value $f^0 = 0.55$ in order to reduce the number of over-detections which comes at a cost of higher number of under-detection. $ZMNC(I, I')$ denotes the zero-mean normalized correlation between images I and I' and $S(\mathcal{X}) = T_{\mathcal{X}}(I_{\ell_{\mathcal{X}}})$ is the resampled image of the pattern and $\mid \cdot \mid$ and $\cap$ denote respectively the area and intersection of white pixels. The coefficient β tunes the tradeoff between the energy terms u_1 and u_2 ($\beta = 100$ in Hervieu et al. [2015]).

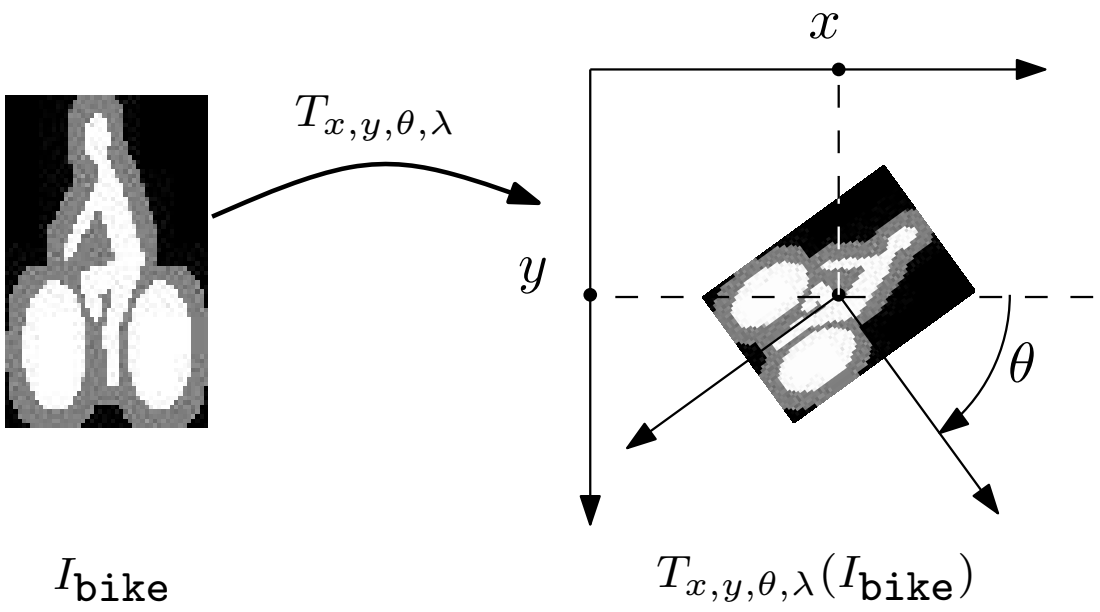


Figure 5.5: The object i with parameters (ℓ_i =bike,$x_i, y_i, \theta_i, \lambda_i$).

This energy is minimized using a Reversible-Jump Markov Chain Monte Carlo (RJMCMC) sampler coupled with a simulated annealing, which may cope with search spaces of varying dimensions (the number of road markings to extract itself being unknown) and arbitrary energy functions (cf. Fig. **??**). Hervieu et al. [2015] further discusses both standard and more advanced RJMCMC kernels which may be used to bias the random sampling toward good solutions, thereby improving the convergence rate.

5.2.2.3 Proposed extensions

- New patterns have been introduced, leveraging the extensibility of the original paper (figure 5.4).
- The data energy term has been scaled by the road-marking perimeter in order to reduce over-detections.

$$u'_1(\mathcal{X}_i) = u_1(\mathcal{X}_i)\,\mathrm{perimeter}(\mathcal{X}_i) \tag{4}$$

  It enables to favour larger objects that could be replaced by many smaller objects using previous data energy.
- A new binary orientation energy u'_{orient} has been introduced in order to penalize incompatible orientations of neighboring road markings. Road markings follow usually a same direction and are nearly parallel except in the intersections where perpendicular markings are observed. This energy term is computed for neighboring objects that are situated at a distance lower than $5m$.
- The raster-based intersection energy proved to be very time consuming as it required the resampling of the template pattern and pixel-by-pixel raster comparisons to get the raster area of intersection. This energy has been replaced by a simplified version u'_{inter}, penalizing the intersection of the road marking oriented bounding boxes (OBB) instead. This drastically reduced computing times while the approximation is very reasonable as road markings are very rarely sufficiently close that their oriented bounding boxes intersect.

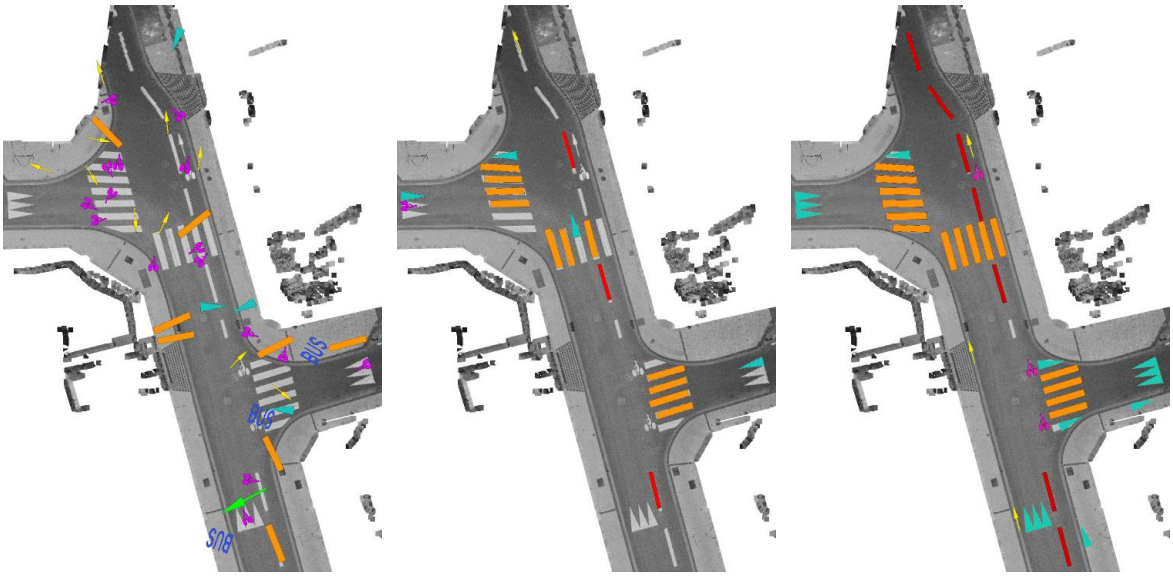
$$\begin{aligned} u'_2(\mathcal{X}_i, \mathcal{X}_j) &= u'_{orient}(\mathcal{X}_i, \mathcal{X}_j) + u'_{inter}(\mathcal{X}_i, \mathcal{X}_j) \\ u'_{orient}(\mathcal{X}_i, \mathcal{X}_j) &= \max(0, -cos4\Delta_\theta) \\ u'_{inter}(\mathcal{X}_i, \mathcal{X}_j) &= u_{inter}(\mathrm{OBB}(\mathcal{X}_i), \mathrm{OBB}(\mathcal{X}_j)) \end{aligned} \tag{5}$$

- Road markings tend to follow a regular layout, thus we added a birth/death in a neighborhood kernel which gives the sampler the opportunity to explore more efficiently the possibility that some road marking might exist next to an already detected one. The inclusion of this kernel also resulted in a significant performance boost **?**.
- Finally, another kernel was added to enable a uniform type switch, which proved to be necessary in order to help the sampler find the right road marking type.

5.2.2.4 3D road marking database

Once the 2D rectangles labelled with a road marking type have been extracted, they are lifted in 3D using the digital terrain model (DTM) encoded in the height channel of the Lidar orthophoto. A simple height lookup enables the lifting of these 2D rectangles as a 4-sided 3D polygon.

Dictated by the targeted application, and due to the abundance of road markings in street view images, the detection tradeoff has been tuned to minimize false detections at the cost of under-detecting some road markings. This results in an extraction with some under-detection but very limited over-detection. In order to ensure the accuracy of this database the extracted road markings may be validated interactively in order to remove the remaining few false positives. Note that this manual intervention is optional and very limited as the extracted road markings may be sorted using their data attachment term u'_1 such that the operator only has to review the few extracted road markings that have the worst data evidence.



(a) (b) (c)

Figure 5.6: Simulated annealing-coupled RJ-MCMC optimization[Hervieu et al., 2015]. (a) initial configuration (b) RJ-MCMC optimization. (c) final results.

5.3 Generation of GCPs from geo-referenced landmarks

The challenge for the integration of geo-referenced landmarks with vision based localization is to find the most relevant geo-referenced landmarks and detect the relevant landmarks in images. In this section, we propose our strategy for GCPs generation and image control points detection from key frames over sequence. The proposed method can work on both monocular or multi-camera images.

5.3.1 Selection of geo-referenced candidates

There might be thousands of landmarks in geo-referenced database, but only few of them are the relevant landmarks for one key frame. Thus, we need to define conditions to select the geo-referenced landmarks which are most possible being relevant landmarks. In this thesis, we

call them as *landmark candidates*. To obtain them, we assume that we know the approximate pose for each key frame. In our method, the approximate poses are estimated using LBA based approach in chapter 3.

Figure 5.7 illustrates our purposes for landmark candidates selection for image $\boldsymbol{I}_t$ from database. We note the approximate pose of $\boldsymbol{I}_t$ as $\boldsymbol{P}_t^{'}$. So $\boldsymbol{C}_t^{'}$ is the approximate position of image center and $\boldsymbol{R}_t^{'}$ is the approximate rotation matrix.

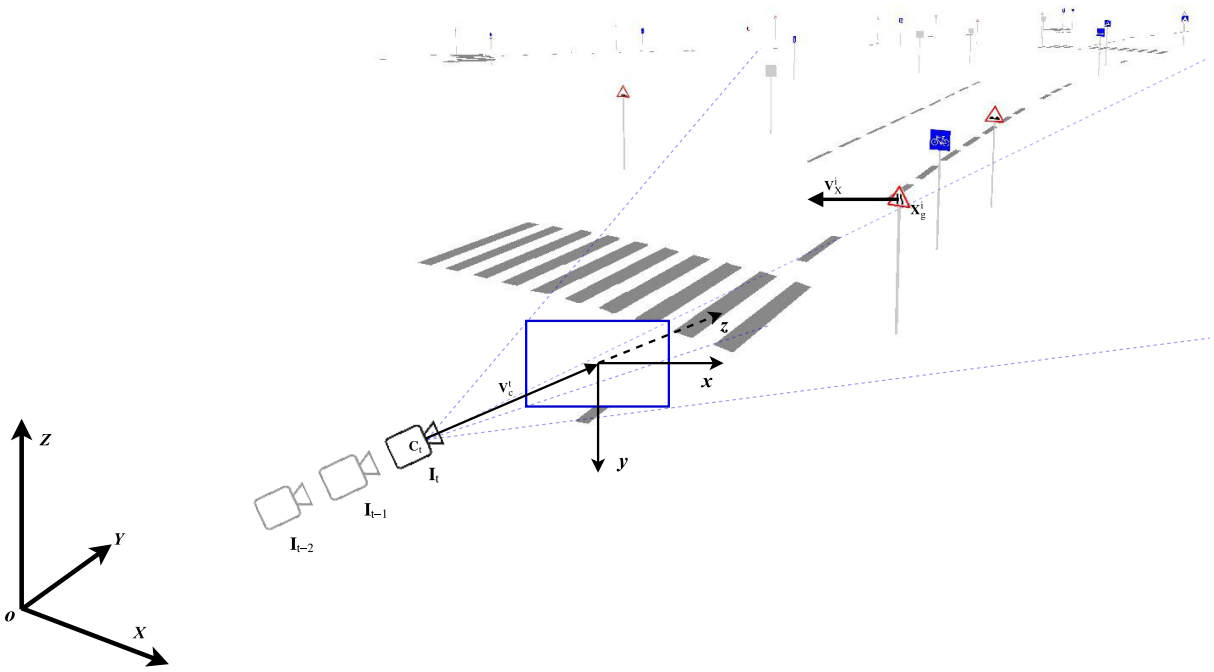


Figure 5.7: The relations between current image and geo-referenced landmarks. $\boldsymbol{C}_t$ is the position of image center in geo-referenced frame for image t, $\boldsymbol{V}_c^t$ is the depth direction of camera and $\boldsymbol{V}_X^i$ is the normal vector of traffic sign.

To select the landmark candidates from database, we divide our strategy into three steps:

(1) Determining searching space in 3D. The distance between landmark and image can be a relevant condition. With the approximate pose and searching distance, we can define a searching space in 3D database. Only the landmarks whose distances to image are less than a given threshold TH_s, are kept. The criterion for this step is:

$$\|\boldsymbol{X}_g^i - \boldsymbol{C}_t^{'}\| < TH_s,$$

where $\boldsymbol{X}_g^i$ is the position of 3D landmark center.

(2) Rejecting invisible landmarks After the first step, we keep the landmarks being inside a sphere space determined by center $\boldsymbol{C}_t^{'}$ and radius length Th_s. However, only the landmarks in front of image are relevant. To judge if one landmark is in front of an image. We define the

view direction of camera: pointing from camera center to image plane along optical axis, as:

$$\boldsymbol{V}_c^t = [\boldsymbol{R}_t^{'}]^T \begin{bmatrix} 0 \\ 0 \\ f \end{bmatrix}, \tag{6}$$

where f is focal length.

Let us define another vector which points from camera center to landmark center. Considering the approximate position of camera center $\boldsymbol{C}_t^{'}$, this vector can be expressed as:

$$\boldsymbol{V}_{cx}^t = \boldsymbol{X}_g^i - \boldsymbol{C}_t^{'}, \tag{7}$$

where $\boldsymbol{V}_{cx}^t$ is the vector from camera center to landmark in absolute coordinate system. If the angle between $\boldsymbol{V}_{cx}^t$ and $\boldsymbol{V}_c^t$ is smaller than $90°$, we consider that the landmark is in front of current image. This condition can be expressed by equation below:

$$\boldsymbol{V}_{cx}^t \bullet \boldsymbol{V}_c^t > 0 \tag{8}$$

For some landmarks, even they fulfill the condition expressed by equation 8, they are not visible for current image. As shown in figure 5.7, some landmarks are inside the image field of view (gray traffic signs), but they are the signs for the vehicle moving at opposite direction. In this case, we need extra conditions that considers the normal direction of landmarks. We note the normal vector of a landmark as $\boldsymbol{V}_X^i$, thus, the intersection angle between $\boldsymbol{V}_X^i$ and $\boldsymbol{V}_c^t$ should be larger than $90°$ (*cf.* figure 5.7). The extra condition can be expressed using dot product of two vectors:

$$\boldsymbol{V}_c^t \bullet \boldsymbol{V}_X^i < 0. \tag{9}$$

(3) Relations to the image view There is a simple truth that the 3D candidates should lie inside the field of camera view. To check the remaining landmarks after previous two steps, we project them using equation 6 in chapter 3:

$$\mathbf{x}_{prj} = F(\boldsymbol{P}_t^{'}, \boldsymbol{X}_g) \tag{10}$$

If the projection $\mathbf{x}_{prj}$ is inside the image plane, the 3D landmark is a candidate.

5.3.2 Uncertainty propagation for landmark registration

The above-mentioned three steps present our principles to select landmark candidates for each image, but we do not know their locations in image yet. With the approximate image pose, an initial location of each landmark candidate can be predicted in image. But the quality of initial location of 2D landmark depends on the accuracy of image pose. In this thesis, we consider the uncertainty propagation to determine the searching region in image for every landmark candidate.

We estimate uncertainty of the projections of traffic signs in image propagated from poses and the 3D road signs. For the landmarks that can be presented with polygons, we project their vertexes into image. For some complicate case, such as circular traffic signs, we need to project circles from 3D to 2D image plane and consider the error propagation. The solution is proposed in [Soheilian and Brédif, 2014]. In this thesis, we focus on the polygon based landmarks such as rectangle, square and triangle.

As defined in chapter 3, a 3D vertex $\boldsymbol{X}_j$ can be projected into image plane using pinhole projection function for single camera case and non-projective projection for multi-camera system. Wo derive the formulations of uncertainty estimation for single camera case first. For every vertex X_j of a 3D traffic sign, its predicted coordinates in image is obtained as

$$\mathbf{x}_j = F(\boldsymbol{P}_t, \boldsymbol{X}_j).$$

The covariance matrix of image pose $\boldsymbol{\Sigma}_{\boldsymbol{P}_t}$ is estimated by LBA and the covariances of 3D road signs $\boldsymbol{\Sigma}_{\boldsymbol{X}_j}$ are obtained during the traffic sign reconstruction. The covariance matrix of $\boldsymbol{\Sigma}_{\mathbf{x}_j}$ can be estimated using error propagation principle for nonlinear equations:

$$\boldsymbol{\Sigma}_{\mathbf{x}_j} = \begin{bmatrix} \frac{\partial F}{\partial \boldsymbol{P}} & \frac{\partial F}{\partial \boldsymbol{X}_j} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_{\boldsymbol{P}_t} & 0 \\ 0 & \boldsymbol{\Sigma}_{\boldsymbol{X}_j} \end{bmatrix} \begin{bmatrix} \frac{\partial F}{\partial \boldsymbol{P}} \\ \frac{\partial F}{\partial \boldsymbol{X}_j} \end{bmatrix} \tag{11}$$

For multi-camera case, the back projection of one vertex of geo-referenced traffic signs can be estimated by

$$\mathbf{x}_j^i = F(\boldsymbol{P}_t, \boldsymbol{\Gamma}_i, \boldsymbol{X}_j),$$

where, $\mathbf{x}_j^i$ is the projection of traffic sign in camera i and $\boldsymbol{\Gamma}_i$ is the transformation parameters from view point to camera. To estimate the covariance matrix of $\mathbf{x}_j^i$, we consider the uncertainty of $\boldsymbol{\Gamma}_i$, the equation is:

$$\boldsymbol{\Sigma}_{\mathbf{x}_j^i} = \begin{bmatrix} \frac{\partial F}{\partial \boldsymbol{P}} & \frac{\partial F}{\partial \boldsymbol{\Gamma}_i} & \frac{\partial F}{\partial \boldsymbol{X}_j} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_{\boldsymbol{P}_t} & 0 & 0 \\ 0 & \boldsymbol{\Sigma}_{\boldsymbol{\Gamma}_i} & 0 \\ 0 & 0 & \boldsymbol{\Sigma}_{\boldsymbol{X}_j} \end{bmatrix} \begin{bmatrix} \frac{\partial F}{\partial \boldsymbol{P}} \\ \frac{\partial F}{\partial \boldsymbol{\Gamma}_i} \\ \frac{\partial F}{\partial \boldsymbol{X}_j} \end{bmatrix} \tag{12}$$

where, $\boldsymbol{\Sigma}_{\boldsymbol{\Gamma}_i}$ is the covariance of $\boldsymbol{\Gamma}_i$.

5.3.3 Landmarks correspondences with images

In order to generate the constraints for localization, we need to register the 3D landmarks within image to obtain the 2D correspondences of 3D landmarks. Considering the uncertainty propagation introduced in previous section, a searching space can be generated for each 3D landmark in image. In this case, we search the 2D correspondences of landmarks inside the searching space. Two different approaches are developed for registering the landmarks in search space in images:

- Registering by object detection. This approach aims at detecting and recognizing a certain object inside the searching area in image generated 3D landmarks, because the attributes of 3D landmarks are known for us.
- Registering by template matching. This strategy is to match the pattern of a 3D landmark into searching area in image to find the location of 2D correspondences of the landmark.

5.3.3.1 Registering by traffic signs detection

In this thesis, this method is based on the same detector for traffic signs database reconstruction introduced in this chapter, but the detection is conducted in searching are for each landmark. Thus, the detector is accelerated by limiting the ROI in image. We determine the searching region for traffic sign extraction according to approximate locations of landmarks in image and their uncertainties. Figure 5.8 shows the procedure of searching region generation. For every vertex of the 3D road sign, its error ellipse is computed from the covariance matrix estimated with the method proposed in previous section. Then, the searching region for traffic sign is determined according to the bounding box of all the error ellipses of the sign vertexes (shown as the red rectangle) in figure.

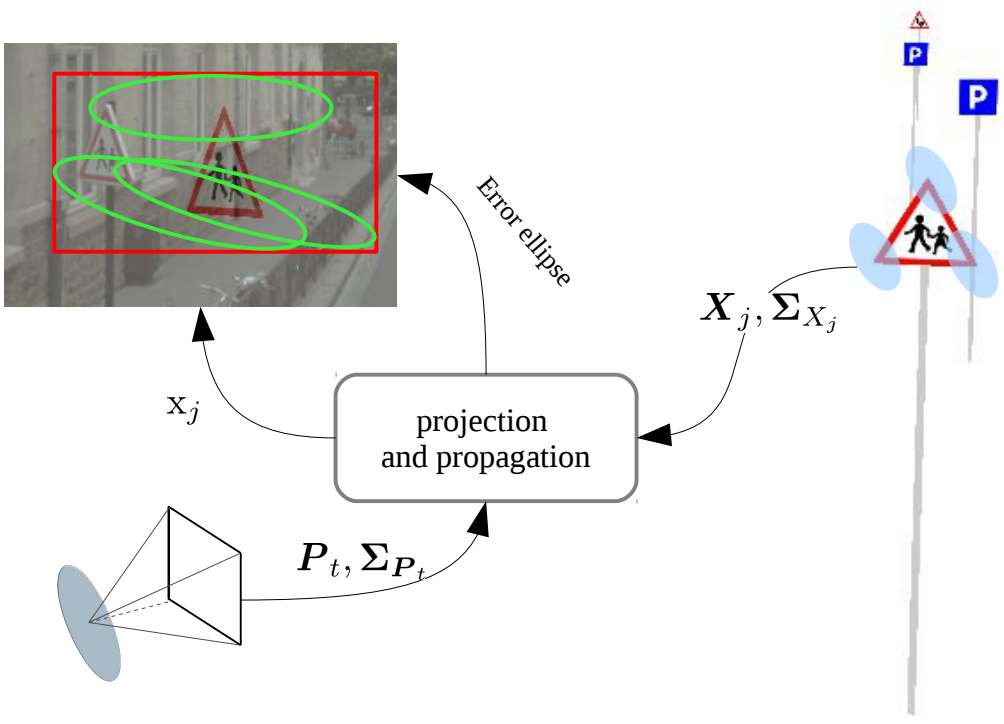


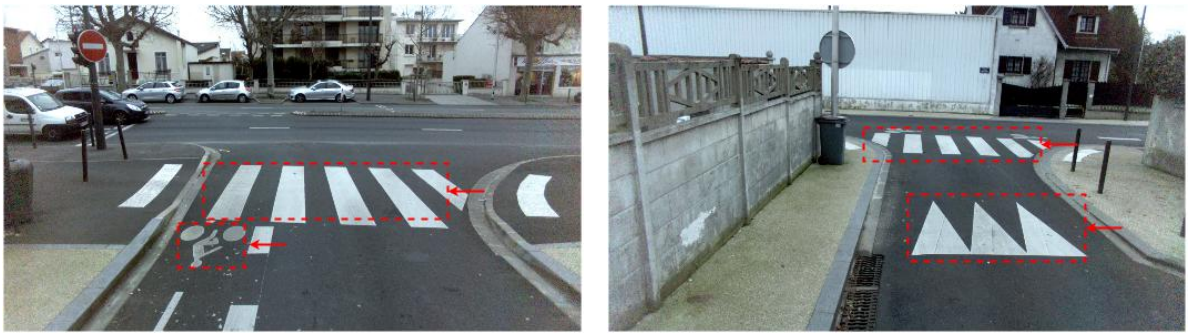
Figure 5.8: The image pose and geo-referenced road sign are provided with their uncertainties. The road sign can be projected into image plane and the uncertainty for every vertex of the projected shape is estimated with error propagation. All the ellipses determine the region for detection.

The traffic signs are extracted automatically within the searching region using the method pro-

posed by Soheilian et al. [2013a]. It recognizes both location and attribute of 2D traffic sign. To validate the extracted results in image, we compare their attribute between extraction and the traffic sign from database, if they have the same category, the extraction is correct. We take the center of each 3D traffic as the GCP and the image control point are fitted from the shape detected in image.

5.3.3.2 Registering by road marking matching

Not all the landmarks have rich strong visual and geometric properties as traffic signs. Figure 5.9 shows some road markings in images captured by an on-board camera. Most road markings can be expressed with polygons, but there are some complex markings such as bicycle box in figure 5.9(a), which are difficult to describe by simple geometric shapes. Another challenge is the poor texture information, it is hard to distinguish different types of road markings by color. For instance, the dashed lanes and the strips of zebra cross have similar pattern in color and geometry. These problems make it difficult to extract road markings in image according



(a) (b)

Figure 5.9: An example of complex situations for road markings.

to their geometric and visual properties. This section introduces a matching based landmark registration. We aim at matching the pattern of 3D landmark with them in image directly. We take road markings for example in this thesis, but the proposed methods can be extended to other landmarks such as traffic signs or build facades.

A: Searching space definition Each road marking in landmark database yields a 4-sided polygon. We estimate the uncertainty of every projection of road marking vertex in image. These four 2D points together with their 2D Gaussian uncertainties allow us to define a sufficiently tight search space. We consider each 2D point in a search area for the refined position. The area is generated by the 2D bounding box of the 99%-confidence region of the Gaussian uncertainty, centered around its estimated 2D position.

Figure 5.10 presents the strategy of matching for a road marking. With the approximate image pose, an initial location of the road marking in image can be predicted, shown with blue

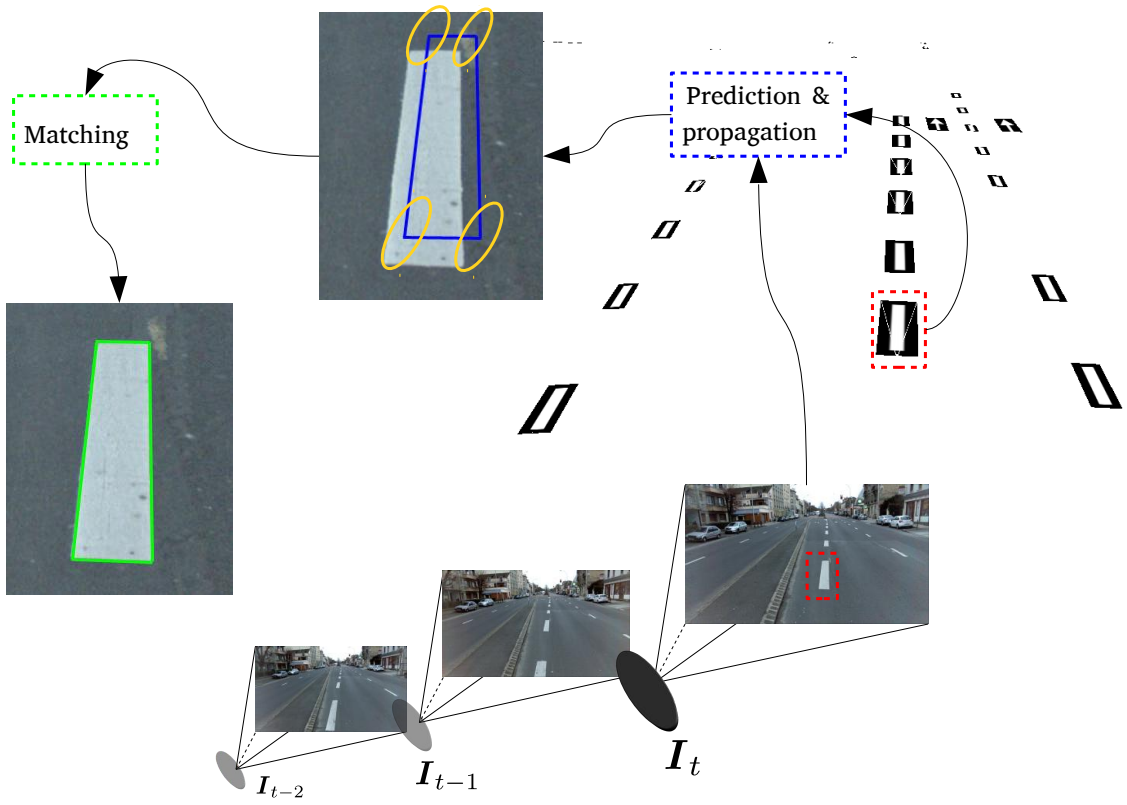


Figure 5.10: Error propagation and searching space generation for road markings.

rectangle in close up window (*cf.* Fig 5.10). Considering the uncertainties of image pose and vertexes of the 4-sided polygon, the uncertain region can be generated in image. Then, the road marking are registered with the image. This conservative search space definition is able to cope with small error underestimation. Starting from the initial position, we define an objective function and then optimize this function to approach the optimal location of road marking in image [Soheilian et al., 2016].

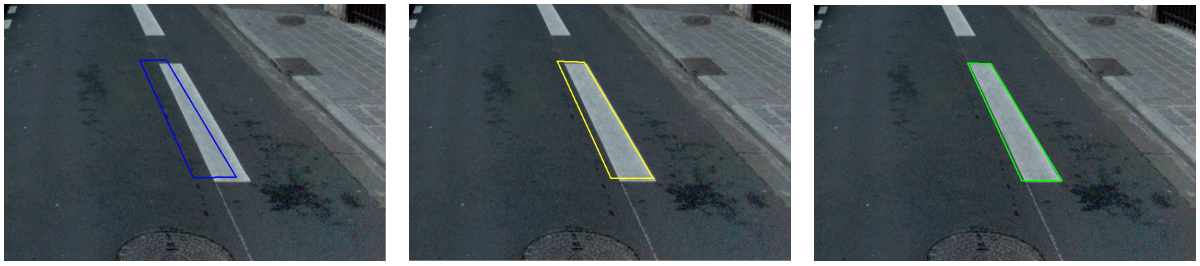
B: Objective function Given four image points defining the homographic projection of a road marking template into the current view, we can assess the quality of this projection by computing the ZMNC of this homographic projection with the image content. We can then formulate our problem as finding the four road marking corner projections within their uncertainty-based bounding boxes such that they maximize this ZMNC score. In order to rule out degenerate set ups, we further impose that the four points define a convex polygon.

C: MCMC Optimization During generation of road marking database, the Reversible-Jump Markov Chain Monte Carlo (RJMCMC) is used for optimization to cope with varying dimensionality problem [Hervieu et al., 2015]. However, the optimization here is defined in a fixed dimension setup, that only has four points with eight coordinates. Given the nature of the objective function, a more specific optimizer is not trivially available, thus we propose to perform regular Markov Chain Monte Carlo (MCMC) optimization.

Given the strong correlation between the errors of the 4 projected points, we propose the following transformation kernels for the MCMC modification proposal step:

- An overall rigid translation of the 4 points
- A translation of one point leaving the three other points fixed with a lower amplitude

In order to rule out degenerate set ups, we also impose that the four points define a convex polygon. The MCMC sampler is coupled with a simulated annealing to optimize the ZMNC objective function, rejecting all modifications that produce a concave polygon [Soheilian et al., 2016]. The initial position is provided by the road marking projection using the approximate pose, the 4-sided polygon is approaching to its precise locations with the progress of iteration. Figure 5.11 shows an example of the iterative steps. After 5000 iterations, the matching of road marking is very close to its real location in image. The number of needed iterations



(a) Initial position (b) After 1000 iterations (c) After 5000 iterations

Figure 5.11: Example of MCMC optimization.

for convergence in the MCMC algorithm depends on the initial estimation and the size of the uncertainty region. Due to some non Gaussian errors, our estimated uncertainty is sometimes underestimated. In practice, we set a large number for iteration threshold (10000) to guarantee the convergence.

After optimization, we can obtain the optimal parameters for homographic transformation. For each geo-referenced road marking object, we choose its centroid as GCP $\boldsymbol{X}_g^i$, its corresponding image point $\mathbf{x}_g^i$ is determined by mapping the position of $\boldsymbol{X}_g^i$ in 3D road marking plane to image using the optimal homography parameters.

5.4 Integration of GCPs with LBA

In this section, we derive formulations for integration of GCPs with LBA that consider the uncertainty of GCPs. First, we propose a method that integrate GCPs for monocular case. Then, we extend the method to multi-camera system.

5.4.1 Formulations for monocular sequences

The geometric relations of every GCP $\boldsymbol{X}_g^i$ to one image control point $\mathbf{x}_g^j$ can be expressed with perspective projection for pinhole camera. Regarding the equation 6 proposed in chapter 3, the formulation can be written as:

$$^i\mathbf{x}_g^j = F(\boldsymbol{P}_j, \boldsymbol{X}_g^i), \tag{13}$$

where, F is projection function and $\boldsymbol{P}_j$ is the pose of image j.

The first type of error equations about GCPs is the back projections errors. In LBA, we divide the image poses into inherited poses $\boldsymbol{P}_p$ and new poses $\boldsymbol{P}_n$. To separate the error equations for GCPs and the regular error equations presented in equation 9 in section 3.3.4.2, we define the back projection equations for GCPs as:

$$^i\boldsymbol{v}_c^j = F(\boldsymbol{P}_p, \boldsymbol{P}_n, \boldsymbol{X}_g^i) - ^i\mathbf{x}_g^j, \tag{14}$$

where, $^iv_c^j$ is the vector of back projection errors for GCPs with respect to image control point $^i\mathbf{x}_g^j$. Although the form of equation 9 in section 3.3.4.2 and equation 14 are same, their meanings are different. $\boldsymbol{X}$ is a set of unknowns in equation 9, $\boldsymbol{X}_g$ is known beforehand, as well as its uncertainty. We denote $\boldsymbol{X}_g^0$ as prior values of GCPs measured from geo-referenced landmarks and the precisions of the coordinates are $\boldsymbol{X}_g^0$ are $\sigma_X, \sigma_Y, \sigma_Z$. In LBA, we regard $\boldsymbol{X}_g$ as a kind of special parameters and $\boldsymbol{X}_g^0$ is an observation of $\boldsymbol{X}_g$. For a GCP $^i\boldsymbol{X}_g$, we can obtain an extra error equation below:

$$^i\boldsymbol{v}_g = ^i\boldsymbol{X}_g - ^i\boldsymbol{X}_g^0 \tag{15}$$

where, $^i\boldsymbol{v}_g$ is the vector of residuals of i^{th} GCP.

Therefore, two types of error equations, expressed in equation 14 and equation 15, are combined within LBA equation system. The weights of these error equations relay on the precision of GCPs in 3D and image point measured in images. We define the precision of image control point as σ_c in image. The value of σ_c is determined by landmark registration algorithms. The covariance matrix for an image control point in image j is:

$$\boldsymbol{\Sigma}_c^j = \sigma_c^2 \boldsymbol{I}$$

Thus, the covariance matrix of every GCP $\bar{\boldsymbol{X}}_g^i$, noted as $\boldsymbol{\Sigma}_g^i$, is:

$$\boldsymbol{\Sigma}_g^i = diag(\sigma_X^2, \sigma_Y^2, \sigma_Z^2)$$

The integration of GCPs in LBA is conducted by combining equation 14 and 15 with normal error equations presented in section 3.3.4.2. Then a new equation array is obtained:

$$\begin{cases} \boldsymbol{v}_p = \boldsymbol{P}_p - \boldsymbol{P}_p^0 \\ \boldsymbol{v}_t = F(\boldsymbol{P}_p, \boldsymbol{P}_n, \boldsymbol{X}_t) - \mathbf{x}_t \\ \boldsymbol{v}_c = F(\boldsymbol{P}_p, \boldsymbol{P}_n, \boldsymbol{X}_g) - \mathbf{x}_g \\ \boldsymbol{v}_g = \boldsymbol{X}_g - \boldsymbol{X}_g^0 \end{cases} \tag{16}$$

where, v_c is a vector of all the back projection errors of GCPs, v_g is a vector of all GCPs residuals, $\boldsymbol{X}_g$ is a vector of all the GCPs appeared in current LBA window and $\mathbf{x}_g$ represents all the corresponding image control points.

The nonlinear least squares technique is employed to solve the problem and it has the same procedure as normal LBA in section 3.3.4. We also estimate the covariance matrix for each pose and then propagate the uncertainty over time. We assume that $\boldsymbol{v}_p, \boldsymbol{v}_t, \boldsymbol{v}_g, \boldsymbol{v}_g$ are independent, thus, the weight matrix for all observations is:

$$\boldsymbol{W} = diag(\boldsymbol{\Sigma}_p, \boldsymbol{\Sigma}_t, \boldsymbol{\Sigma}_c, \boldsymbol{\Sigma}_g)^{-1}, \tag{17}$$

where, $\boldsymbol{\Sigma}_c$ and $\boldsymbol{\Sigma}_g$ are diagonal matrices, which are conducted by a set of $\boldsymbol{\Sigma}_c^j$ and $\boldsymbol{\Sigma}_g^i$. Then we resolve the problem by minimizing the sum of weighted squares of equation 16:

$$[\hat{\boldsymbol{P}}_p, \hat{\boldsymbol{P}}_n, \hat{\boldsymbol{X}}_t, \hat{\boldsymbol{X}}_g] = argmin\{\frac{1}{2}(\boldsymbol{v}_t^T\boldsymbol{\Sigma}_t^{-1}\boldsymbol{v}_t + \boldsymbol{v}_p^T\boldsymbol{\Sigma}_p^{-1}\boldsymbol{v}_p + \boldsymbol{v}_c^T\boldsymbol{\Sigma}_c^{-1}\boldsymbol{v}_c + \boldsymbol{v}_g^T\boldsymbol{\Sigma}_g^{-1}\boldsymbol{v}_g)\}, \tag{18}$$

where, $\hat{\boldsymbol{P}}_p, \hat{\boldsymbol{P}}_n, \hat{\boldsymbol{X}}_t, \hat{\boldsymbol{X}}_g$ are the optimal estimates of the parameters.

The number of GCPs and corresponding image points is very small in comparison to regular tie points, so the extra computation is negligible. Despite the low number of equations, the influence of GCPs is considerable due to their covariance matrix $\boldsymbol{\Sigma}_c$ and $\boldsymbol{\Sigma}_g$.

5.4.2 Integration of GCPs for multi-camera system

In multi-camera case, the GCPs are still generated from geo-referenced landmarks, but the image points are measured in more images, compared with the single camera case. The strategy that integrates GCPs with LBA for multi-camera case, is similar with that used for monocular camera. The back projection error equations are used to cope with the GCPs and images points and the uncertainties of the GCPs are considered in LBA. The only difference is that the image poses should be adopted from the viewpoint pose via rigid transformation. Thus, the error equation about back projections for one GCP $\boldsymbol{X}_g^i$ in j^{th} image is given by:

$$^j\boldsymbol{v}_c^i = F(\boldsymbol{P}_p, \boldsymbol{P}_p, \boldsymbol{\Gamma}_i, \boldsymbol{X}_g^i) - ^j\mathbf{x}_g^i.$$

The $F(\boldsymbol{P}_p, \boldsymbol{P}_p, \boldsymbol{\Gamma}_i, \boldsymbol{X}_g^i)$ is the back projection model for multi-camera system, defined in section 3.4.1.3(*cf.* chapter 3).

The error equations related to constraints for GCPs are same with equation 15 in previous section. Combining the error equations for multi-camera based LBA, presented in section 3.4 (chapter 3), the new error equation array for LBA with GCPs for multi-camera case is:

$$\begin{cases} \boldsymbol{v}_p = \boldsymbol{P}_p - \boldsymbol{P}_p^0 \\ \boldsymbol{v}_\Gamma = \boldsymbol{\Gamma} - \boldsymbol{\Gamma}^0 \\ \boldsymbol{v}_t = F(\boldsymbol{P}_p, \boldsymbol{P}_p, \boldsymbol{\Gamma}, \boldsymbol{X}) - \mathbf{x}_t^i \\ \boldsymbol{v}_c = F(\boldsymbol{P}_p, \boldsymbol{P}_p, \boldsymbol{\Gamma}, \boldsymbol{X}_g) - \mathbf{x}_g^i \\ \boldsymbol{v}_g = \boldsymbol{X}_g - \boldsymbol{X}_g^0 \end{cases} \tag{19}$$

where, $\boldsymbol{v}_\Gamma$ stands for residuals of rigid transformation parameters from camera to viewpoint. The weight matrix for the observations in equation 19 is:

$$\boldsymbol{W} = diag(\boldsymbol{\Sigma}_p, \boldsymbol{\Sigma}_\Gamma, \boldsymbol{\Sigma}_t, \boldsymbol{\Sigma}_c, \boldsymbol{\Sigma}_g)^{-1}, \tag{20}$$

It is also a diagonal block matrix. We minimize the sum of weighted squares of the errors presented in equation 19 to solve the problem:

$$\begin{gathered}[\hat{\boldsymbol{P}}_p, \hat{\boldsymbol{P}}_n, \hat{\boldsymbol{X}}_t, \hat{\boldsymbol{\Gamma}}, \hat{\boldsymbol{X}}_g] = \\ argmin\{\frac{1}{2}(\boldsymbol{v}_p^T\boldsymbol{\Sigma}_p^{-1}\boldsymbol{v}_p + \boldsymbol{v}_\Gamma^T\boldsymbol{\Sigma}_\Gamma^{-1}\boldsymbol{v}_\Gamma + \boldsymbol{v}_t^T\boldsymbol{\Sigma}_t^{-1}\boldsymbol{v}_t + \boldsymbol{v}_c^T\boldsymbol{\Sigma}_c^{-1}\boldsymbol{v}_c + \boldsymbol{v}_g^T\boldsymbol{\Sigma}_g^{-1}\boldsymbol{v}_g)\}\end{gathered} \tag{21}$$

where, $\hat{\boldsymbol{P}}_p, \hat{\boldsymbol{P}}_n, \hat{\boldsymbol{X}}_t, \hat{\boldsymbol{\Gamma}}, \hat{\boldsymbol{X}}_g$ are the optimal estimates of the parameters. We apply nonlinear least squares to resolve the problem.

5.5 Experiments

The data used for experiments is acquired by STEREOPOLIS [Paparoditis et al., 2012]. Images are captured by high resolution cameras which are calibrated beforehand. The geo-referenced traffic signs and road markings are generated using the methods introduced in section 5.2. A combined navigation system (GPS/INS/odometer) is mounted in STEREOPOLIS , so the ground truth of image poses can be easily obtained.

5.5.1 Experiment of using traffic signs

In this experiment, we test images captured by both single camera and stereo rig. The experimental data is shown in figure 5.12. The binocular images were taken by a forward looking stereo rig while the monocular sequence are conducted by the images captured by the left camera of the stereo rig in STEREOPOLIS. The images are shown in figure 5.12(a) and 5.12(b) which are captured by calibrated cameras. There are 20 geo-referenced traffic signs in test area and the trajectory length is about 1 km (cf. Fig 5.12(c)).

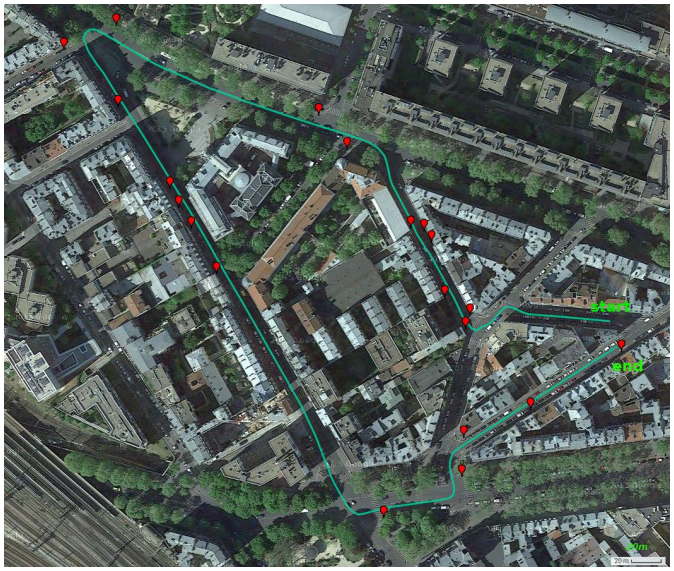
5.5.1.1 Integration of traffic signs for single camera based localization

We start from a known point and give the distance between first two frames to determine the absolute scale. We test our localization method without any traffic signs integration using the method proposed in chapter 3. Then, the same dataset is tested with the method integrated with geo-referenced traffic signs. At last, we compare their localization results.

Figure 5.13 shows the trajectories of ground truth. The vision based localization using LBA for poses refinement has been proposed in chapter 3. For simplified expression, we use "LBA"

(a) Left image

(b) Right image



(c) Trajectory and locations of traffic signs.

Figure 5.12: Experiment data

to stand for vision based localization and use "LBA+TS" to represent the method for localization integrated with geo-referenced traffic signs. As demonstrated in figure 5.13, the drift of localization is growing if no geo-referenced traffic sign is integrated (*cf.* blue line in Fig 5.13). However, the drift is reduced a lot when we integrate the traffic signs into our localization procedure (the red line is very close to the ground truth).

We also compute lateral error and depth error separately for each position, as shown in figure 5.14. The x axis is the index of image and y axis represents the absolute errors. We can see that both lateral and depth accuracy can be reduced with the integration of traffic signs within LBA. In particular, the depth error is more sensitive to the external constraints(*cf.* Fig 5.14(b)). The errors are reduced immediately when the traffic signs are successfully detected in image and combined into LBA, while the lateral errors are decreased slower.

From figure 5.14(a) and figure 5.14(b), we see that the lateral and depth errors are reduced, but they are not decreased too much. To explain this, let's review the method to compute lateral and depth errors introduced in section 3.3.5 (cf. chapter 3), which needs to transform camera center $\boldsymbol{C}_t$ in geo-referenced coordinate system to $\boldsymbol{\nu}_t$ in camera coordinate system. Thus, when we calculate lateral and depth errors, the errors of orientation will also influence the results. To know the real accuracy of position of image center, we investigate absolute errors, comparing the estimated position with ground truth. The diagram of absolute errors will be shown in figure 5.17. The maximum error is reduced from $4.5m$ to $1.5m$ with 18 traffic signs along the whole trajectory.

As we discussed in chapter 3, the uncertainties of localization grow over time. One objective of integrating external data with localization is to limit the uncertainty growing. Figure 5.15 demonstrates the uncertainties of localization with and without integration of traffic signs. The

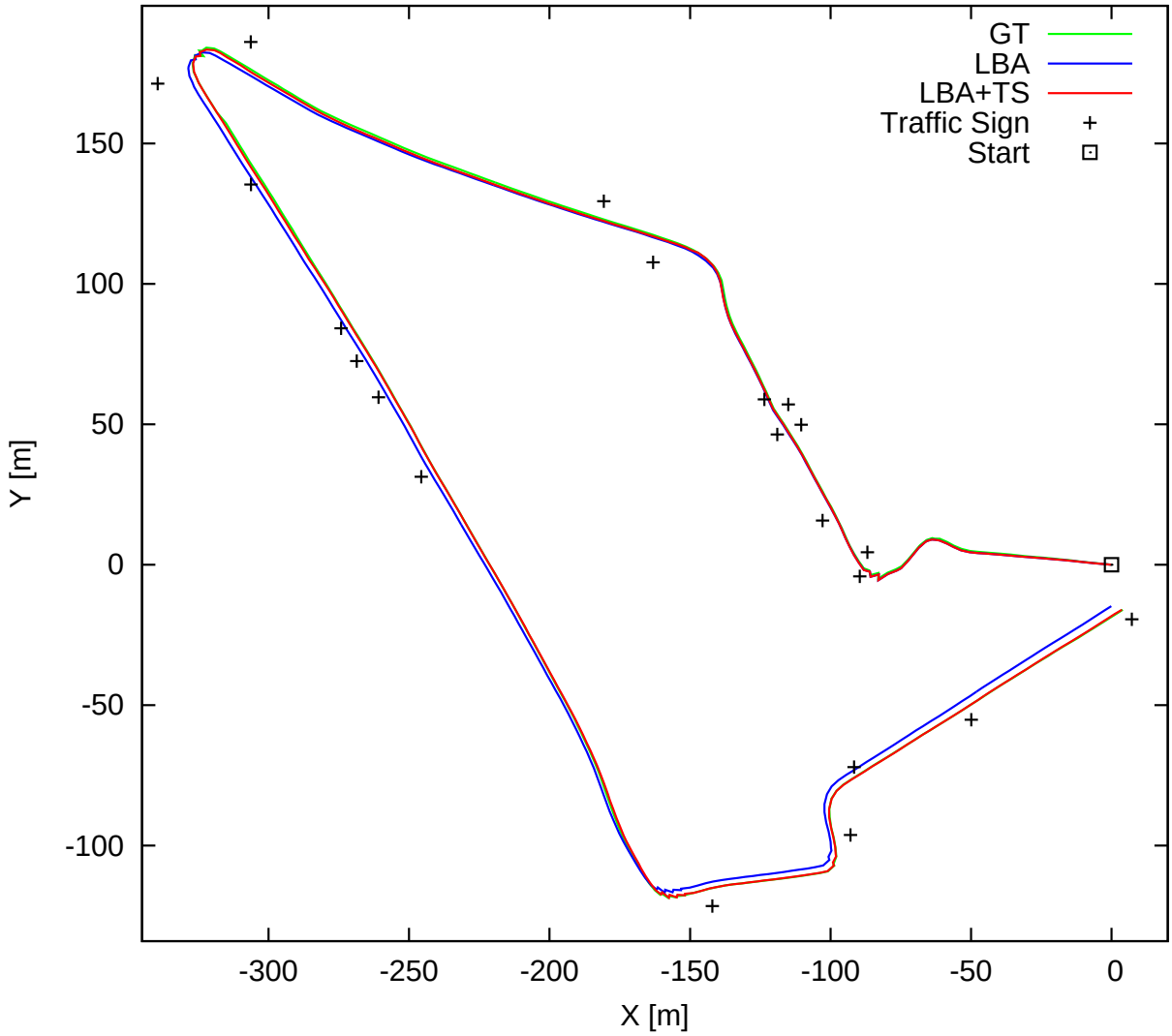
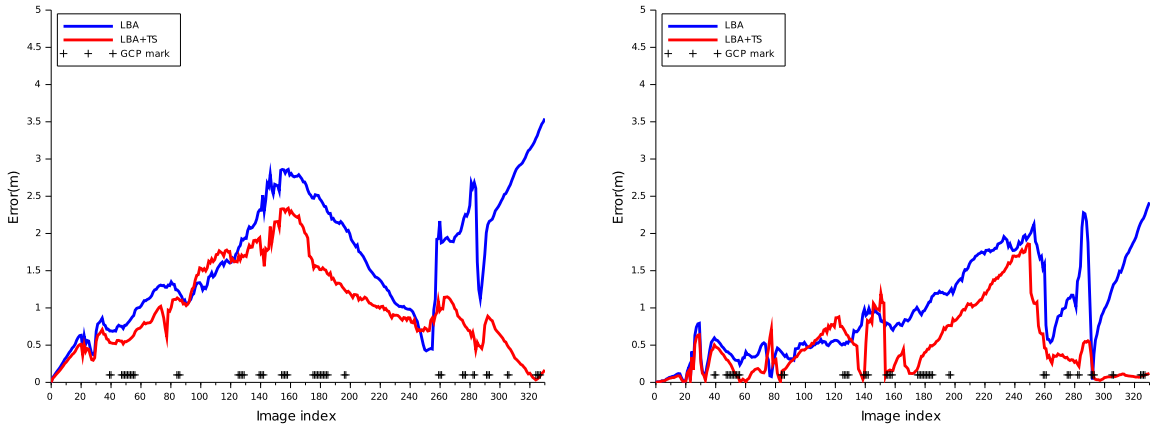


Figure 5.13: Trajectories of Ground Truth (GT), vision based localization (LBA) and integration of Traffic Signs (TS) for localization (LBA+TS). Black crosses stand for the locations of traffic signs.



(a) Lateral errors.

(b) Depth errors.

Figure 5.14: **Blue**: the errors of localization using LBA. **Red**: the errors for the case of traffic signs integration. Black cross: containing GCP in the image.

confident level is set as 99 % for error ellipsoid. The cyan error ellipsoids are results of LBA

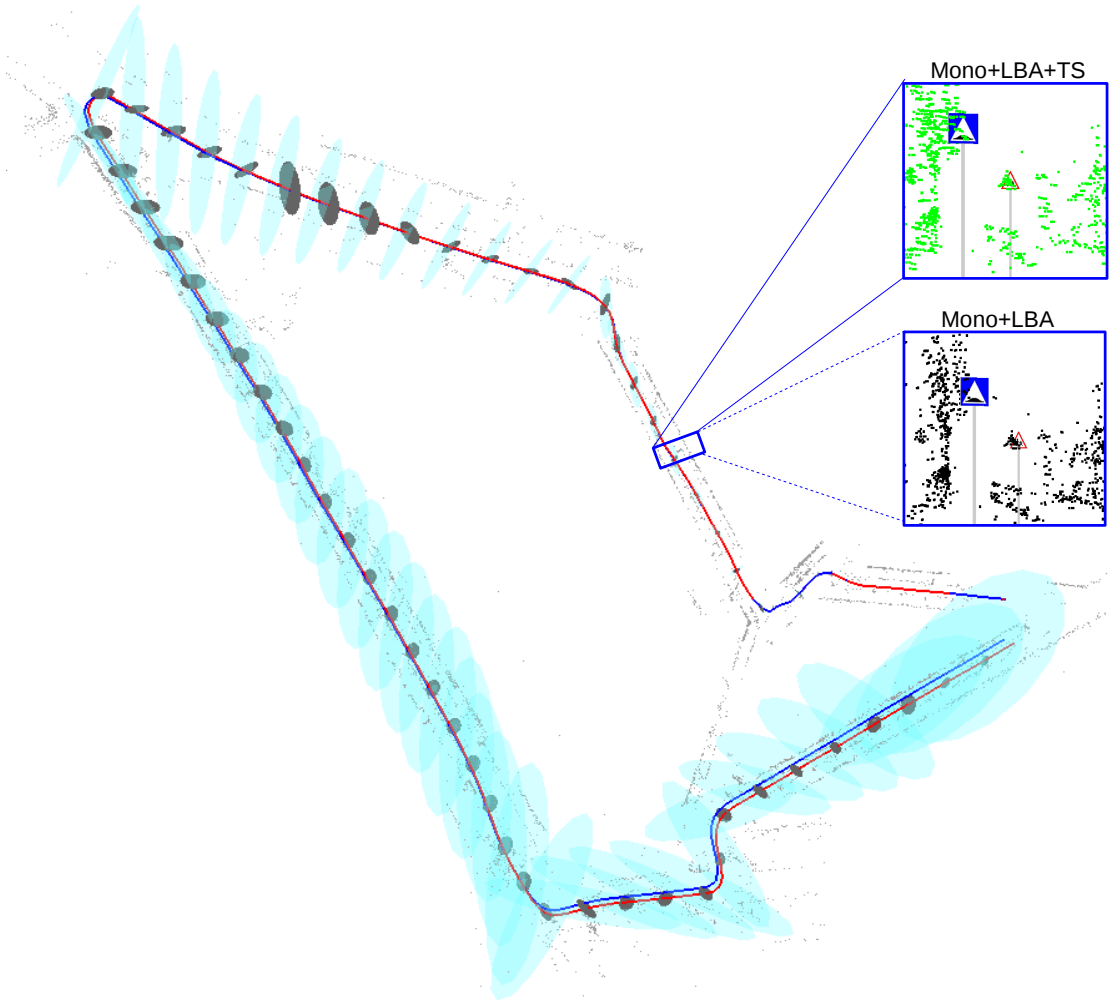


Figure 5.15: Red: LBA+TS based localization. **Blue**: LBA based localization. **Cyan**: error ellipsoids of LBA based localization. **Grey**: error ellipsoids using LBA+TS. **Close-up windows**: accuracy improvement of tie points. All the error ellipsoids are ten times larger than their original size for visualization.

without traffic sign integration and the gray ellipsoids represent the uncertainty of localization with the integration of traffic signs (*cf.* Fig 5.15). We can see that the size of error ellipsoids are reduced with the integration of traffic signs. We also observe that the uncertainties of localization are growing continuously over time. When the traffic signs are integrated, the uncertainties only grow between two successive observations of traffic signs. This can avoid the error propagation over sequence.

Because traffic signs are distinctive features in image, thus some tie points on traffic signs can be reconstructed. The two sub-windows in figure 5.15 show the tie points optimized with LBA and constrained LBA by traffic signs. Regarding the tie points on warning sign (red triangle), the displacement between tie points and the sign are reduced with the integration of geo-referenced traffic signs. Tie points on the traffic sign almost coincide with the road sign, seen from the green points in figure 5.15.

5.5.1.2 Integration of traffic signs for stereo based localization

We test the binocular image for localization in the same district. For stereo case, the absolute scale is determined by the length of baseline of stereo rig. Figure 5.12 presents one image pair used in this experiment. The length of baseline is $1.5m$. We consider the uncertainties of stereo rig parameters in our approach. In stereo sequences, each geo-referenced traffic sign is searched in both left and right images with the proposed method in section 5.3.

We compare monocular and stereo based localization with and without the integration of traffic signs, thus four experiments need to be conducted. We note:

- **Mono+LBA**. Pure vision based localization using monocular images.
- **Mono+LBA+TS**. Integrating geo-referenced traffic signs with localization using monocular images.
- **Stereo+LBA**. Pure vision based localization based on stereo image sequences.
- **Stereo+LBA+TS**. Integrating traffic signs with stereo based localization.

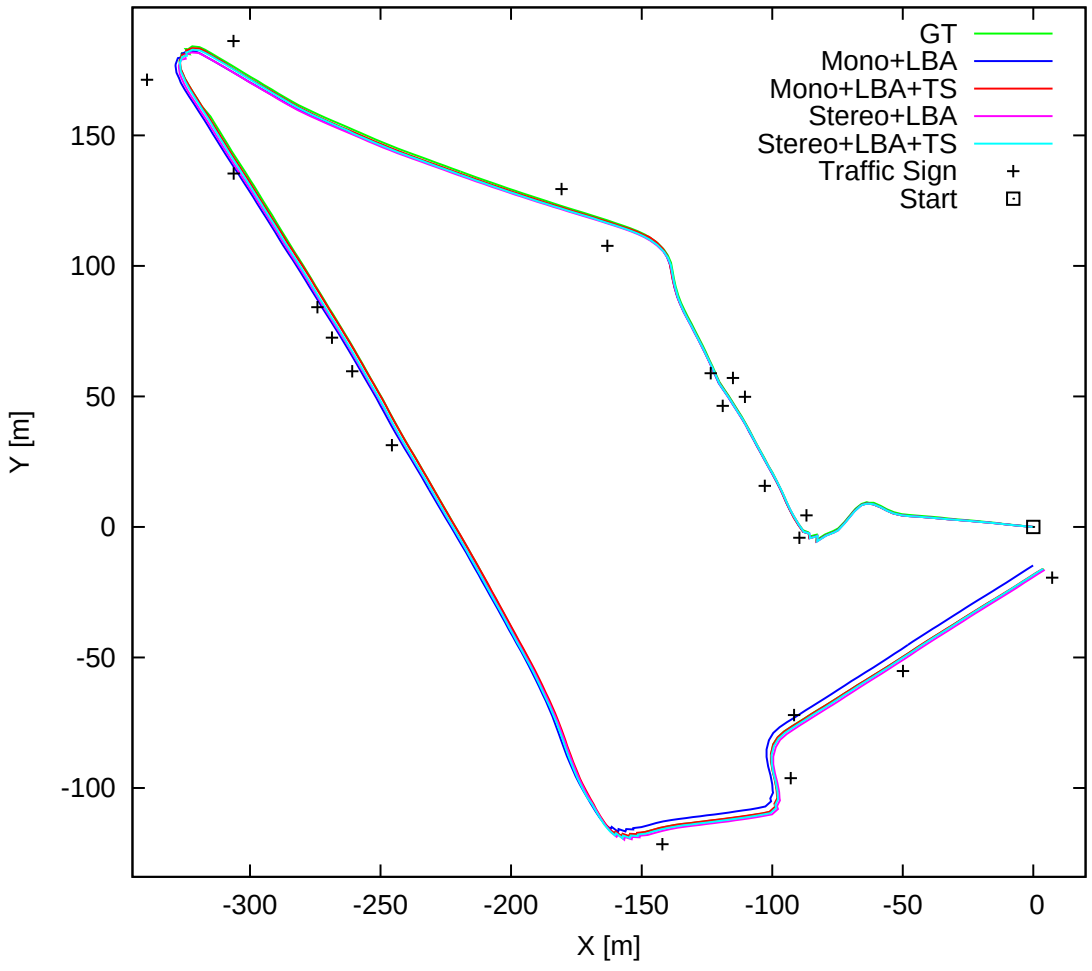


Figure 5.16: Comparing estimated trajectories with ground truth.

Figure 5.16 shows four estimated trajectories and ground truth. Compared with other three approaches, the drift of Mono+LBA is the largest. However, the trajectories obtained by monocular and stereo are very close when we integrate traffic signs (*cf.* cyan and red in Fig 5.16).

To know exact accuracy of the four approaches for localization, we calculate absolute errors for the localization in 3D that is the Euclidean distance from estimates to ground truth. The diagram of absolute errors is shown in figure 5.17. For pure vision based localization, the stereo

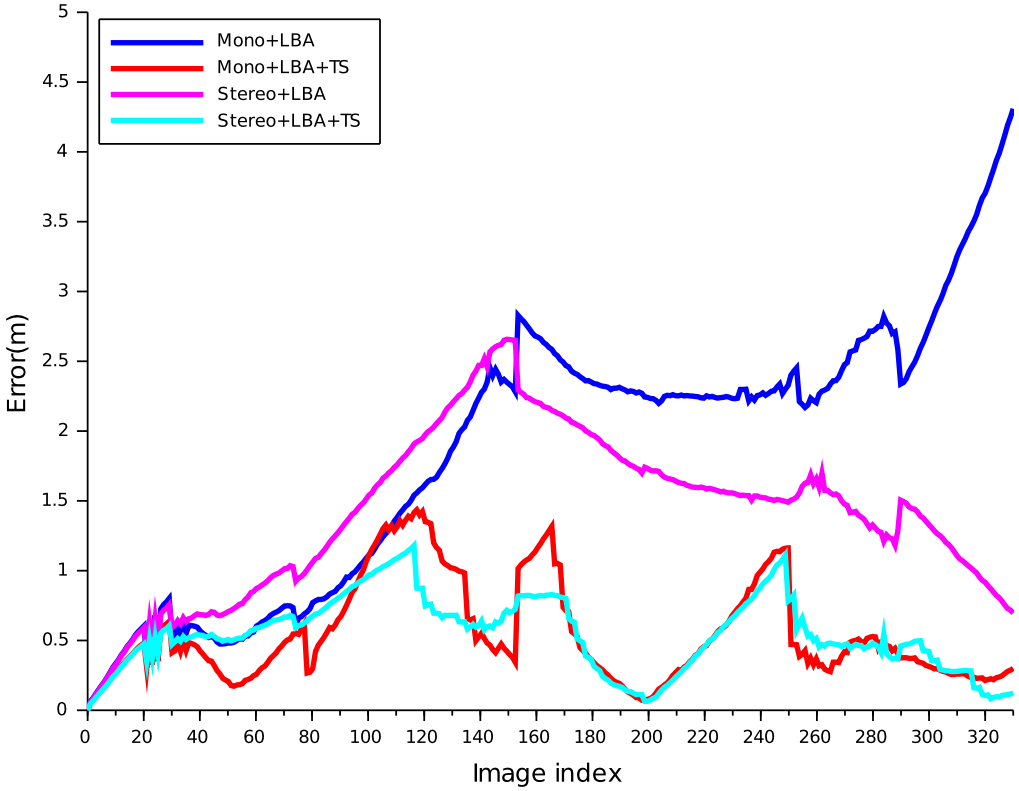


Figure 5.17: Absolute errors of localization.

rig can achieve better results (*cf.* Fig 5.17) than monocular (see blue and magenta lines in Fig. 5.17). This has been studied in chapter 3. Our aim is to analyze the performance of traffic sign integration. The red and cyan lines represent the absolute errors for monocular and stereo images integrated with geo-referenced traffic signs. Both of them are below the absolute error lines of monocular and stereo based localization, which are shown with blue and magenta in figure 5.17. The maximum localization error is reduced from $4.5m$ to $1.5m$ for monocular case and $3.0m$ to $1.4m$ for stereo, when the traffic signs are integrated. From the diagram, we also observe that the stereo case is slightly better than monocular case under the integration of traffic signs.

We estimate the error ellipsoids for each key frame, as shown in figure 5.18. We can see the improvement where the size of error ellipsoids (gray ellipsoids) are reduced in comparison to the cyan error ellipsoids for pose estimated with pure vision based localization. The two close windows also show some tie points. Let's focus on the points on warning sign. We can see

the improvement of localization accuracy using traffic signs. The tie points match the warning traffic sign better when we use the GCPs in LBA.

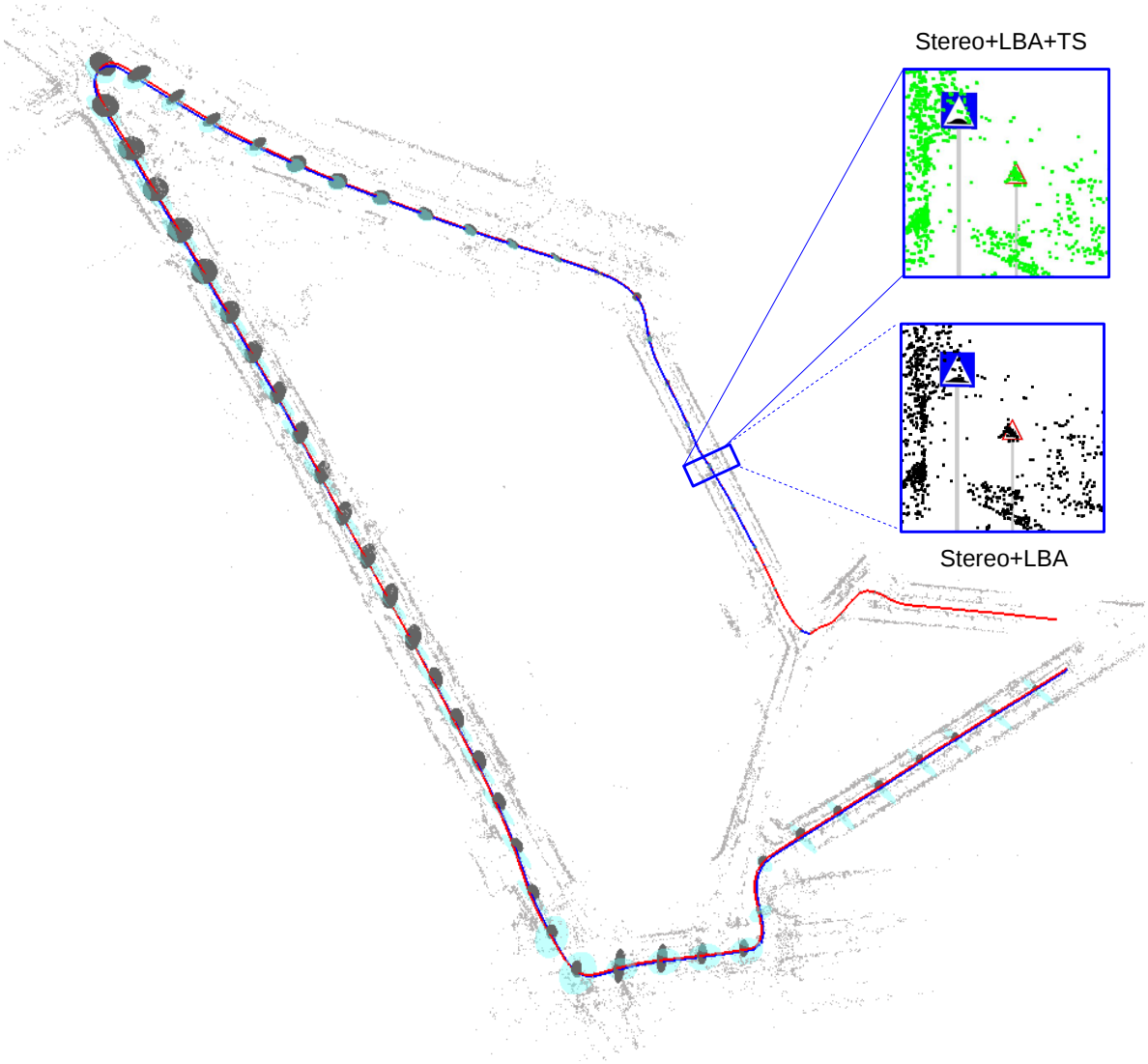


Figure 5.18: Red: estimated trajectory using stereo rig and traffic signs. **Blue**: estimated trajectory using only stereo rig. **Cyan**: error ellipsoids of pose estimation using only stereo rig (LBA). **Grey**: error ellipsoids of pose estimation using stereo rig integrated with traffic signs (LBA+RS). **Close-up windows**: tie points in the selected areas. The size of all the error ellipsoids are exaggerated ten times for visualization.

5.5.1.3 Efficiency of traffic signs detection

The most time-consuming part in GCPs generation from traffic signs is the 2D traffic sign detection. Figure 5.19 shows two traffic signs detected in searching areas marked with yellow rectangles, which is generated based on uncertainty propagation. The efficiency of traffic sign detection algorithm proposed by Soheilian et al. [2013a], is related to the size of region. As shown in figure 5.19, it needs $0.14s$ to detect the sign in the larger searching area while only

$0.039s$ is spent to detect the traffic sign in the smaller searching area. It is much more efficient than detecting the traffic signs in entire image, which spends $1.6s$ with the same method, this shows the interest of proposed method of guiding traffic signs detection.

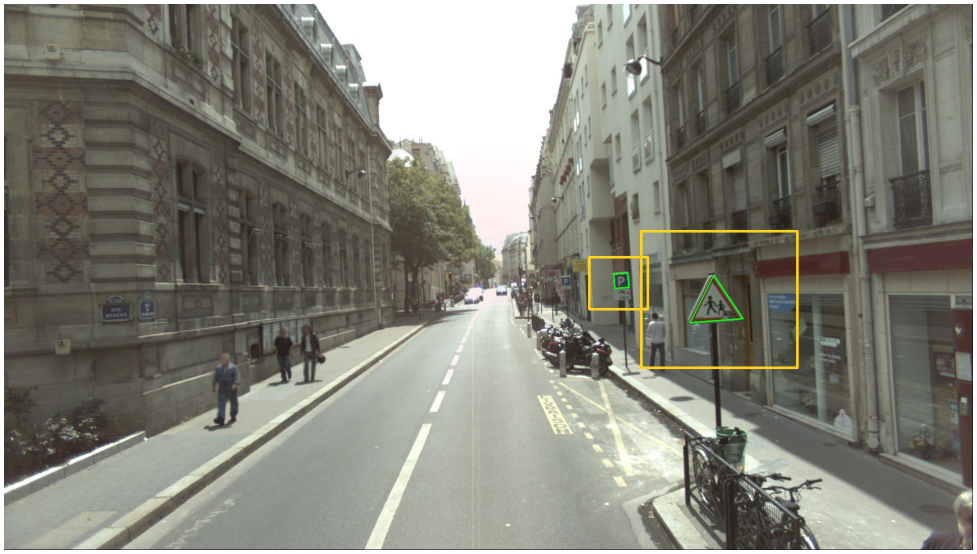


Figure 5.19: Traffic signs detection. Yellow rectangle: searching area for each traffic sign. Green polygon: detected traffic signs.

In previous experiments, there are 18 geo-referenced traffic signs along $1km$ trajectory with 330 key frames from mono and 330 image pairs for stereo. For mono case, 61 images out of 330 can observe the traffic signs. In order to give an idea about the number of tie points, GCPs and image control points in LBA, we explore one processing window which contains the most image control points and GCPs over entire sequence. The statistic of the number of parameters and observations are listed in table below:

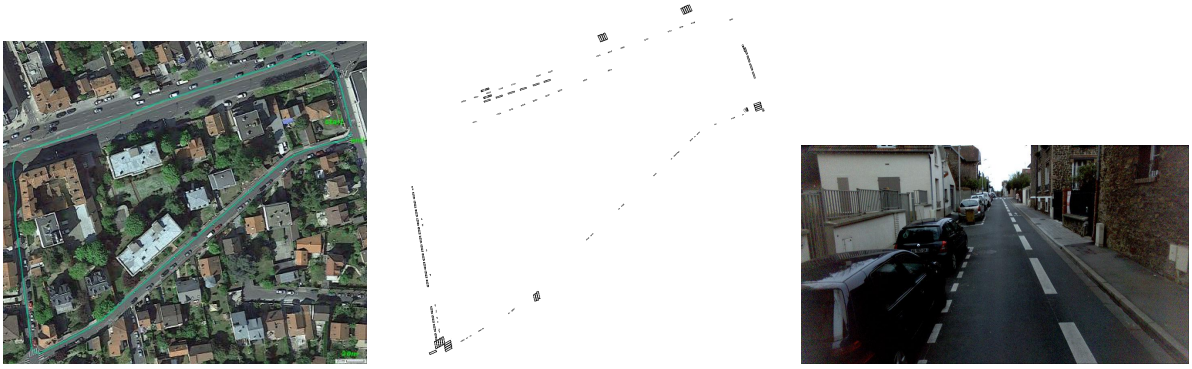
Table 5.1: Extra computation caused by GCPs.

	Image	GCPs	Image control point	Tie points	2D image points
Number	10	3	12	1068	3059

There are 3 GCPs generated from traffic signs. The size of processing window is ten ($N = 10$) and seven images in window can detect the 2D traffic signs. The total number of image control points is 12, thus 24 error equations are generated for back-projection errors. Besides 9 constrained equations are obtained for the 3 GCPs. In this case, we have 33 extra error equations. However, there are 1068 tie points and 3059 image points are extracted, that is, 6118 error equations generated for solution. So we can even ignore the additional computation caused by the extra 33 equations in LBA. This is the case with the highest number of GCPs. It means that most of LBA windows have less additional computation caused by GCPs.

5.5.2 Experiment of using road markings

In order to validate the integration of geo-referenced road markings with localization. A $520m$ trajectory in urban environment as shown in figure 5.20(a), is selected for experiment. We apply monocular sequences for this experiment, which are captured by a forward looking camera embedded on STEREOPOLIS . The geo-referenced road markings shown in figure 5.20(b) are generated off-line with the algorithm presented in section 5.2. About 200 objects along the trajectory were reconstructed, and 150 out of them were kept after manually editing to remove some false objects.



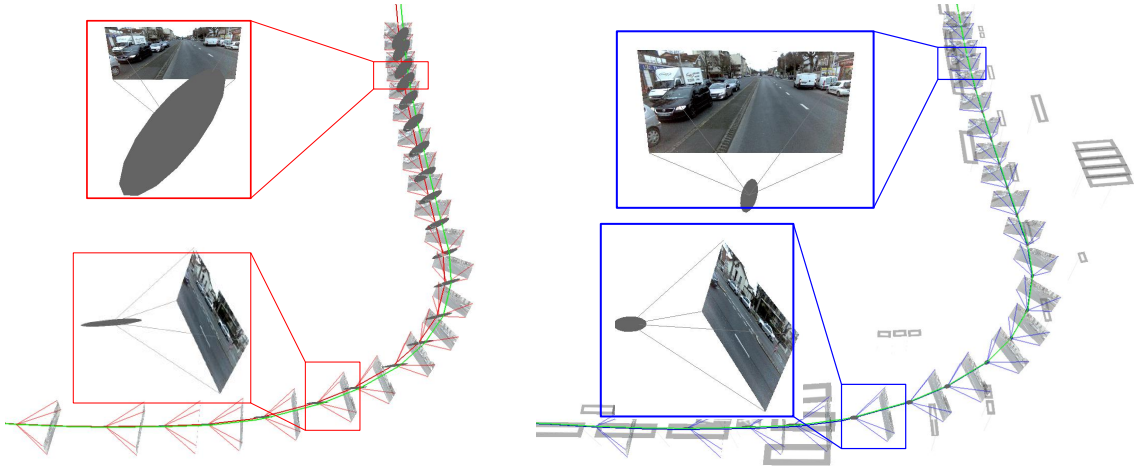
(a) (b) (c)

Figure 5.20: (a) Testing area. (b) Geo-referenced road marking objects after manually editing. (c) An example of image acquired by the monocular camera.

5.5.2.1 Interest of road marking for localization

For monocular case, we still need to initialize localization method with known absolute point and given the absolute scale at the beginning. The first processing window for LBA run with the conventional LBA, then we propagate the uncertainty of image poses over sequence. In this experiment, the size of the processing window is set as seven ($N = 7$) and progressed with one image ($n = 1$) for LBA.

The geo-referenced road markings are successfully matched from the beginning of the trajectory. Thus, the GCPs and image control points are generated for LBA from first step. Figure 5.21 shows the interest of road marking at the beginning. The green line shows the ground truth trajectory. Figure 5.21(a) shows the localization results optimized with LBA. The red trajectory shows the estimated path. The gowning drift and the size of error ellipsoids can be noticed for pure vision based localization. However, the results are becoming better when the geo-referenced road markings are integrated as GCPs (*cf.* Fig 5.21(b)), where the blue line is much closer to the ground truth and the size or error ellipsoids are always small as shown in the two close-up windows in figure 5.21(b).



(a) Without using road markings. (b) Using road markings.

Figure 5.21: The uncertainties of localization at the first $20m$. The error ellipsoids are exaggerated ten times and the ground truth trajectory is drawn in green.

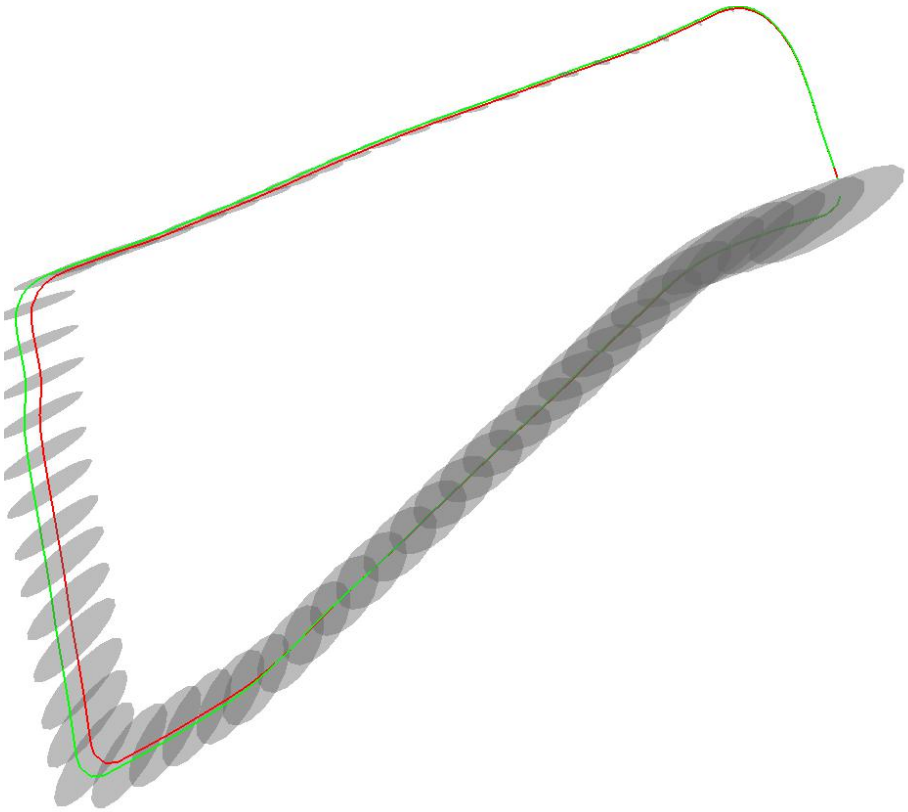
5.5.2.2 Accuracy of localization integrated with road marking

The results of localization with and without integration of road marks are shown in figure 5.22. The trajectories are shown in XY planes. If no road markings are integrated, we can see the growing of drift and uncertainty (*cf.* Fig 5.22(a)). When we integrate the road markings in localization, the estimated trajectory is very close to ground truth and the uncertainties are limited to small level. We notice that, most of the images have very small ellipsoid except for the images close to the end of the trajectory. The error ellipsoids are growing for those image. Let's observe the distribution of the geo-referenced markings in figure 5.22(b), the density of the road markings is lower, thus less GCPs are generated at this area. That is why the uncertainties of image poses are larger.

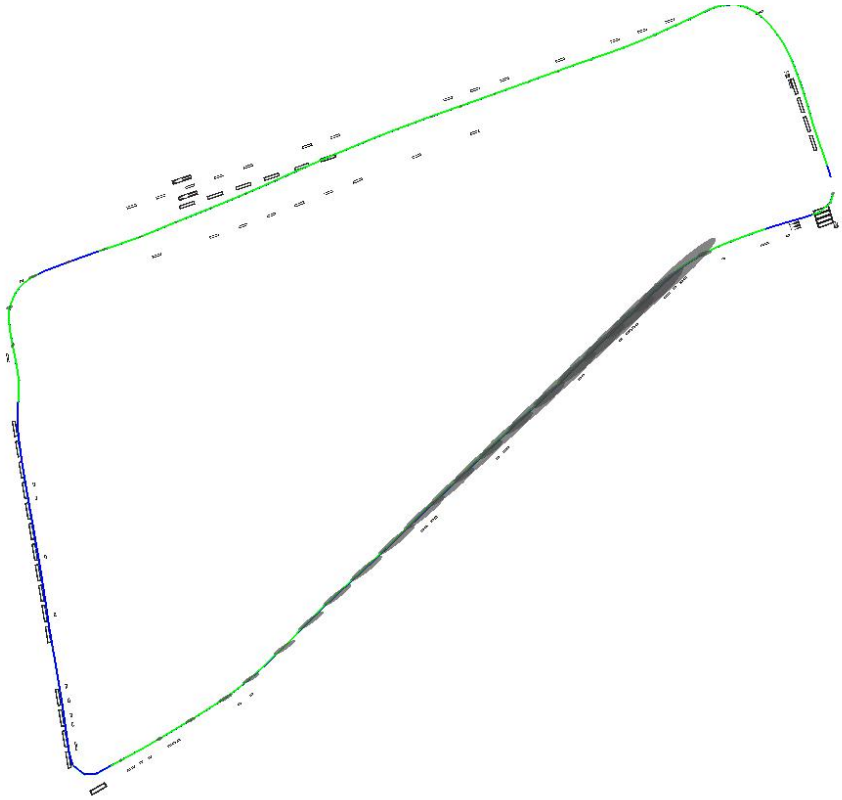
The localization errors are computed at depth and lateral direction. The errors are shown in figure 5.23 which indicates that both types of errors are reduced with the integration of road markings. Particularly, most of the depth errors are less than $0.2m$. We also compare the position with ground truth to compute absolute errors of image position (*cf.* Fig 5.24). The maximum error is up to $4m$, if no road marks are integrated in. However, it can be decreased to $0.4m$ with the integration of road markings. Moreover, the error is around or less $0.1m$ for most part of the sequence.

5.5.2.3 Efficiency of road marking based localization

101 road markings are successfully matched and generated GCPs over the sequence. Compared with the number of tie points used in LBA, which is usually over one hundred per image, the number of GCPs is still small. The computing cost caused by the integration of GCPs for LBA is too small to be considered.

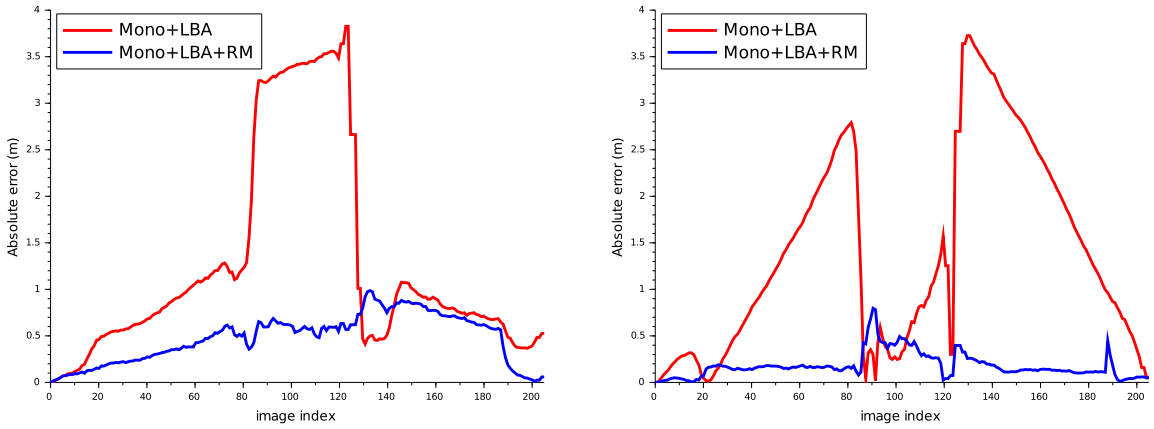


(a) LBA without using road marks.



(b) LBA using road marks.

Figure 5.22: Trajectories of localization without and with road marks, compared with ground truth. Error ellipsoids are exaggerated 10 times. Ground-truth trajectory is drawn in green.



(a) Lateral errors.

(b) Deoth errors

Figure 5.23: The lateral and depth errors of localization. The blue line presents the errors for localization using LBA. The red line shows the errors of localization with the integration of road markings.

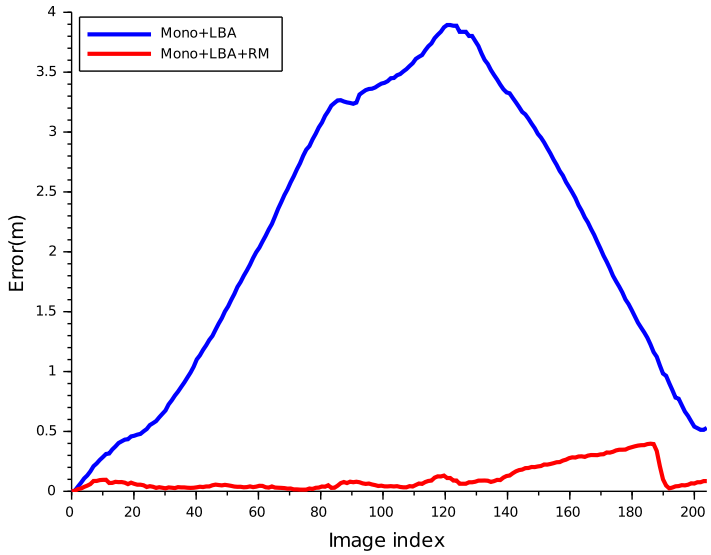


Figure 5.24: Absolute errors for localization without and with road marks.

In fact, most of the computation time goes into road mark matching. The number of iterations for convergence in the MCMC algorithm depends on the initial estimation and the size of the uncertainty region. Due to some non Gaussian errors, our estimated uncertainty is sometimes underestimated. That is why we enlarge the search area to guarantee the convergence in practice, but it slows down the algorithm. In addition, the computation time for each iteration is proportional to the number of the pixels in the pattern. The current algorithm takes $10 - 40s$ for each object and the average of pixels for the matched patterns in image is about 20000 pixels. Our current road markings based localization approach is far from being real-time application. However, it can achieve very precise localization.

5.6 Conclusion and perspectives

Two different landmarks (traffic sign and road marking) are tested for localization in previous experiments. A brief comparison of the performance of integrating traffic signs and road markings, is shown in the table below:

Table 5.2: Experiments summary.

	Trajectory(m)	GCP number	Localization accuracy(m)			
			Mean		Max	
Traffic sign	1000	18	Mono	Stereo	Mono	Stereo
			0.69	0.70	1.5	1.4
Road marking	520	101	0.11		0.38	

Although the images used for experiments were captured at different time and districts by STEREOPOLIS, the trajectories have different length, we can still say that the localization using road markings can achieve very precise localization; its average error of position is only $0.11m$ which is smaller than the average error of image position than localization using traffic signs. It is of that the accuracy of localization is related to the density of landmarks over sequences.

The extra computation caused by the integration of GCPs for LBA can be ignored for both traffic sign and road marking cases. The traffic signs detection is efficient which only need approximately $0.15s$ per object in our experiment. However, the road marking matching is off-line processing at current time. It takes $10 - 40s$ per object, that is related to the size of projected object in image using MCMC optimization. The high computing cost on road marking matching is the bottleneck for real time applications. One interesting idea for accelerating road marking matching would be to use image gradient to match the contours and/or corners of objects instead of using all the pixels inside the road marking for costly correlation score computation. Moreover, a smarter adaptation of MCMC parameters (number of the iterations, starting temperature and temperature decrease rate) for each object can also help to avoid useless iterations and save computation time. Finally, we believe that the real-time Jurie-Dhome [Jurie and Dhome, 2002] tracker can be adapted to the problem of road mark matching.

It should obtain better localization results if we combine the traffic signs and road marking together as external data. In this case, we need to generate the geo-referenced traffic signs and road markings in a same testing area. This can be done in our further work. Meanwhile, a top-down approach regarding the localization system as a whole throughout the development as the method proposed by Aynaud et al. [2014], can be used in localization with the consideration of uncertainty. Not only landmarks can be integrated, but also the data from different sensors and maps with different precision, can be combined together for localization. We can select the most relevant approach for localization according to the requirement of application.

Chapter 6

Summary

The conventional localization solution for Mobile Mapping system (MMS) is to use GNSS, but this kind of methods suffer from multi-path and mask problems in urban canyons which could lead to outage or inaccurate localization. In order to obtain accurate localization, a precise IMU is often combined with GNSS, sometimes the data from odometer is also fused, to provide accurate and high rate localization for MMS. However, the combined system is too expensive for commercial applications in large scale. Thus, a cost-effective localization solution is desired. In order to achieve a low-cost but precise system for localization, we propose a solution that integrates geo-referenced landmarks with vision based localization. The geo-referenced landmarks are generated with a precise MMS beforehand. During the operation of localization, only one or multiple cameras are applied in the system. A low-cost GPS is used to provide the initial information and the drift of localization is compensated by the integration of geo-referenced landmarks.

A full framework for localization was achieved in this thesis (*cf.* Fig 6.1). The input of the system was image sequences, then the interest points were detected, matched and tracked automatically over sequence in real-time, thus the pose of every frame can be estimated instantly. Keyframes were selected from the sequences and Local Bundle Adjustment (LBA) was applied to optimize the poses of keyframes and propagate the uncertainties in an incremental scheme. At the same time, the geo-referenced landmarks (traffic signs and road markings) were matched within keyframes to generate a set of constraints for LBA considering the uncertainty propagation, to reduce drift of localization.

The incremental approach provides online localization. It means that precise poses can be estimated in a shortly fixed time. This enables us to geo-reference the data acquired by MMS during the mission of mapping. In this case, we detect and edit the change of maps during mapping. Our LBA based approach can be easily extended to global bundle adjustment (only change the size of processing window) based approach, which produces more precise localization results and can be used for offline mapping. The main contributions are introduced in following sections.

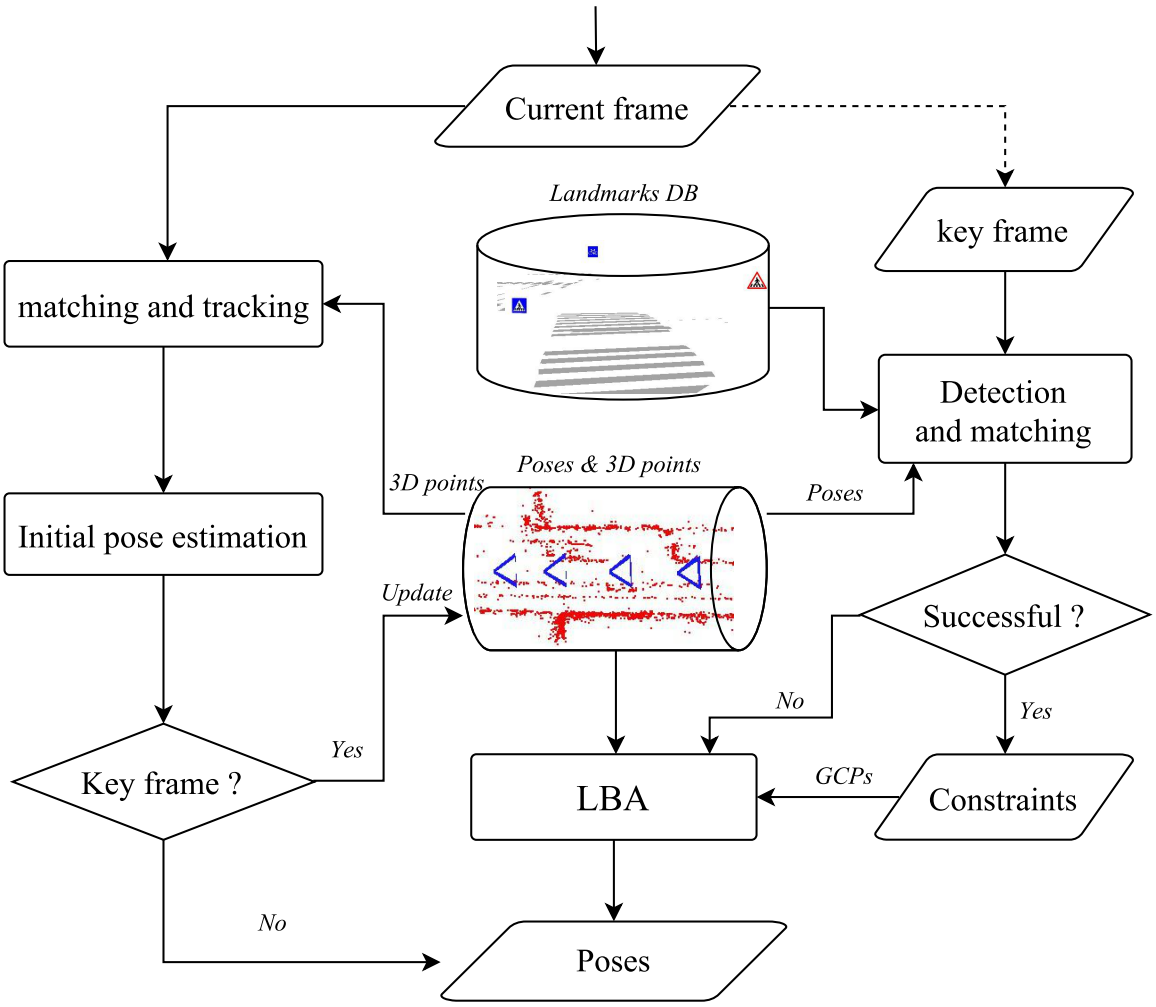


Figure 6.1: The framework of our localization.

6.1 Contributions

6.1.1 Multi-camera based localization

A single camera based approach was implemented considering uncertainty propagation in chapter 3, then we extended the method adopting to multi-camera system. It is well known that vision system with larger FOV enables us to observe larger informative scene and track the tie points in a longer period. Meanwhile, high angular resolution is also desired for localization in large scale environment which can provide more precise image measurements. It is very difficult to obtain images with both large FOV and high angular resolution at the same time for single camera case, but they can be approached using the multi-camera system designed in our approach. In particular, rigorous projection model was proposed considering uncertainty into bundle adjustment to adapt multi-camera case.

The proposed localization method can be easily adopted to different camera configurations according to the requirement of mission without any change of mathematic model for LBA. The

cameras were mounted rigidly on the body of vehicle, we didn't need any special configurations between images captured by different cameras (e.g. overlap, parallel view). In our research, the rigid transformation from camera to the viewpoint which was defined at an arbitrary location on vehicle body, were calibrated by offline precess before localization. In order to overcome the impact of calibration error, we considered the uncertainties of these rigid transformation during optimizing in LBA.

6.1.2 Integration of geo-referenced landmarks

In order to compensate error accumulation, the geo-referenced semantic landmarks were integrated with the vision based localization. In this thesis, both traffic signs and road markings

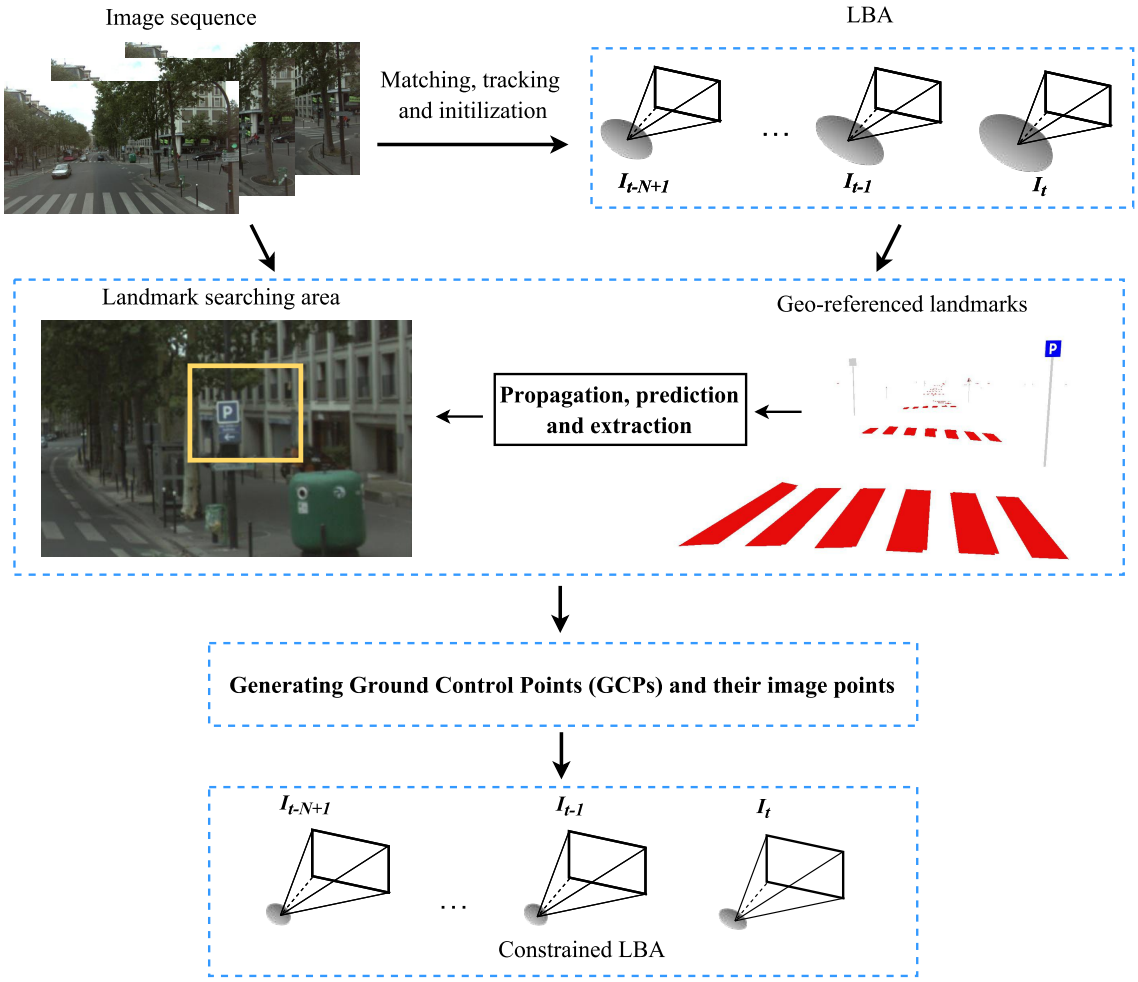


Figure 6.2: Integration of landmarks with localization.

were regarded as landmarks. We aimed at generating a set of GCPs from geo-referenced landmarks. Then, the image points linked to GCPs were measured by matching the 3D landmarks in images. To do this, we generated a searching area for each landmark in image, considering the uncertainty propagation from image pose obtained by LBA and geo-referenced landmarks (*cf.* figure 6.2). With GCPs and their measurements in image, a group of error equations were

generated for LBA. We also considered the uncertainties of GCPs that can reduce the impact of GCPs errors. With constraints of GCPs, the drift was reduced, the drift accumulation only occurred amongst the images between two successive GCPs. Moreover, few extra error equations were added in LBA equation system, thus the integration of GCPs didn't increase the computation time of LBA.

Two different landmark extracting methods were introduced: bottom-up and top-down strategies, considering the characteristic of traffic signs and road markings in images. The bottom-up detection method was used for traffic signs extraction that can get benefit from special color and geometric information of the signs, while the top-down method was used for the road marking detection which had much more complex geometry and poor color information. The top-down approach was a more generic method that can also be used for the registration of traffic signs.

6.1.3 Propagation based matching and tracking

In our approach, tie points were tracked over image sequence to obtain a set of 3D-2D correspondences for pose estimation. This process was built based on interest points matching. Feature based matching is widely used, but lots of low quality matches are usually obtained for poor texture areas (e.g. road surface, building facades). Considering the characteristic of vehicle movement on road which is restricted because of inertial moment, we proposed a propagation based matching and tracking method to enhance the performance of localization (*cf.* figure 6.3).

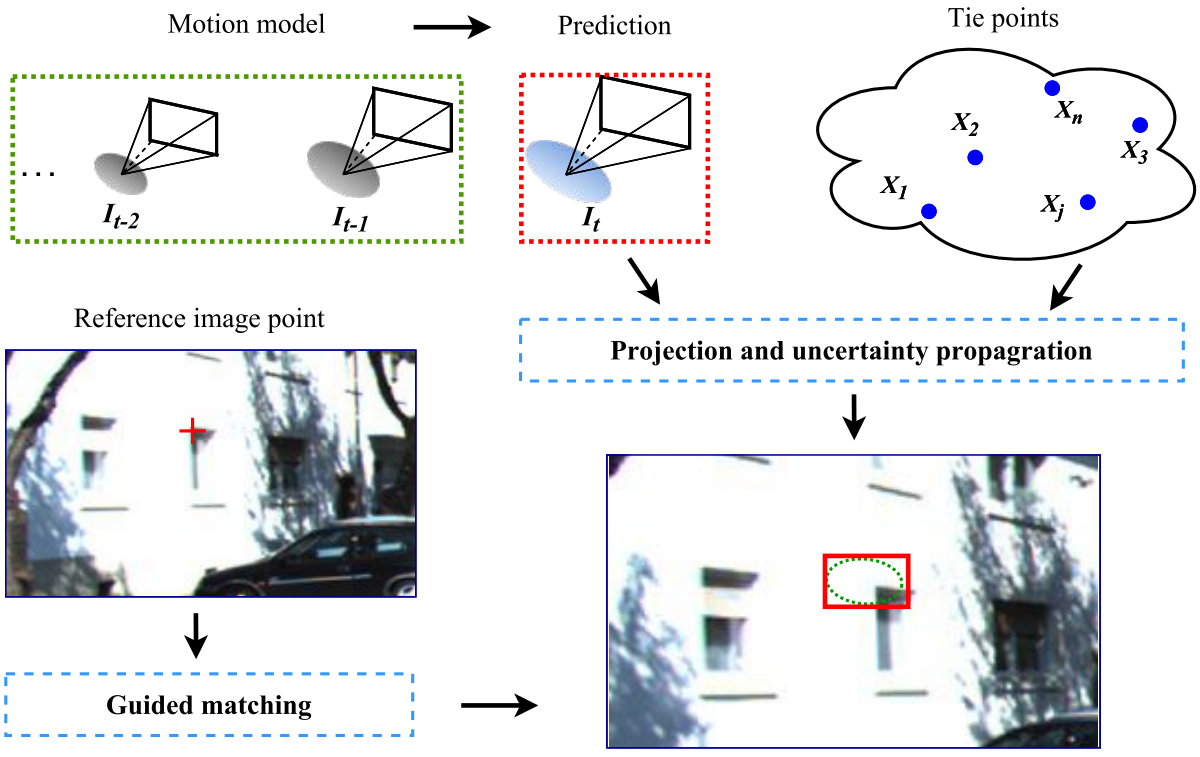


Figure 6.3: Propagation based matching and tracking for tie points.

The aim of propagation based matching and tracking was to restrict the searching of every tie point in new frame, so that robust and accurate matches for pose estimation can be obtained. To approach this, a motion model was defined by learning from previous poses to predict the pose for every new frame. Meanwhile, the covariances of prediction was estimated via uncertainty propagation. According to the predicted pose and its covariance, a uncertain region can be generated for each tie point in image. Then the matching of tie point was limited within uncertain region, in contrast to searching in entire image for the common methods. The searching area was determined by the bounding box of uncertain region (red rectangle in figure 6.3). The precise correspondences of tie points were determined using template matching in every searching area. This method can reduce the false matches by limiting searching scope. At the same time, the efficiency of matching was improved due to reduction of searching size. With the robust tracks of tie points, high quality pose estimation was obtained, so that the accuracy of localization was improved. The motion model was updated dynamically with the precise pose and uncertainty to adapt the new frames.

6.1.4 Uncertainty analysis

The uncertainty is important to characterize the state and performance of a process, it presents the potential errors with given confidences. In this thesis, we estimated uncertainties for poses to quantify the performance of localization. Furthermore, the uncertainty propagation was maintained in our system.

Modeling constraints based on uncertainty. The observations in localization were weighted according to their uncertainties. In our LBA equation system, there were four different kinds of observations: image points, estimated poses, camera rigid parameters and GCPs generated from landmarks. The weight for each observation was determined by its uncertainty. Thus, the observations with low precision can also be integrated for localization without worrying about the decreasing of localization accuracy. In particular, we estimated uncertainty for pose estimation considering uncertainty propagation to evaluate the performance of localization.

Uncertainty propagation in the system. In our approach, the uncertainties was also used to guide data association (tie point matching & GCP registration)matching and tracking. A propagation based matching and tracking method was proposed(*cf.* chapter 4) to explore tie points in new frame for pose estimation. We propagated the uncertainties from image poses to determine the searching area for each tie point. Besides, the uncertainty propagation was also considered for landmark matching. Geo-referenced landmarks usually have different precision and some gross errors sometimes are contained in landmark database. In this thesis, we considered the uncertainties of landmarks in matching and integration.We generated a searching area in image for each 3D landmark considering the uncertainty propagation of image pose and landmark to

limit the searching scope for landmark matching in image.

6.1.5 Evaluation

We tested our localization method using different datasets. In MATIS, a precise MMS (STEREOPOLIS) has been developed for street-level mapping. Thus, geo-referenced road markings and traffic signs can be generated by means of imagery and dense point clouds acquired by STEREOPOLIS. Meanwhile, the image sequences captured by on-board cameras gave us various options for evaluation of the proposed localization method. We evaluated the methods after introduction of theory in the three technical chapters (chapter 3, chapter 4, chapter 5).

Vision based localization Chapter 3 explained the LBA based localization using only cameras. Different camera configurations were adopted and the uncertainty propagation was considered over sequence. The high resolution cameras mounted on STEREOPOLIS made it easy to compose different types of camera configuration. We tested four camera configurations (monocular, stereo, no-overlap stereo and multi-camera) using the data collected by STEREOPOLIS. The ground truth was acquired using a precise GNSS/INS/odometer system. From experiment results, the accuracy was improved with growing of FOV and the multi-camera rig (four cameras) obtained the highest accuracy. We also noticed that the non-overlap stereo obtained better accuracy than forward-looking stereo rig most of the time, because the non-overlap stereo system has larger FOV. However, more computation time was needed for feature detection and matching as well as the LBA with growing of cameras in vision system.

Evaluation on KITTI benchmark We also tested our vision based localization using the datasets for visual odometry on KITTI websites. The matching and tracking method proposed in chapter 3 suffered from ambiguity problem in some challenging scenarios (high speed road) for vision based approach. To improve the robustness of localization, a propagation based matching and tracking method was proposed in chapter 4. The interest points were detected under the condition of buckets, based on FAST detector to obtain uniform distribution of the points. Eleven binocular sequences were used for training sequences. The accuracy of localization was up to 0.95% for translation and 0.0033 deg/m for rotation with the new approach for matching and tracking for training data. Then we summit our estimated poses for testing data to KITTI benchmark. The accuracy was 1.25% for translation and 0.0041 deg/m for rotation which was not as good as the performance of training data, because there were more sequences captured on highway for testing datasets that contained many moving vehicles in images and leaded to inaccurate matches for pose estimation.

Localization integrated with landmarks We presented the method for landmark integration with LBA for localization in chapter 5. Two typical semantic features in street environment:

traffic signs and road markings were used for experiments. The images were also captured by STEREOPOLIS and the ground truth was acquired by GNSS/INS/Odometer system. For traffic signs based localization, we reduced the maximum error from $4.5m$ to $1.5m$ and the mean error was reduced from 1.9 to $0.7m$ only using the GCPs generated from 18 traffic signs over a $1km$ path. The road markings based localization achieved more precise localization, because over two hundred road markings were matched for GCPs over $520m$ path. The maximum localization error decreased from $3.9m$ to $0.38m$ while mean absolute error of localization with road marking was only $0.11m$ in comparison to $1.3m$ without the integration of road markings.

6.1.6 Integration of the methods in THINGS2DO

The THINGS2DO project is focused on the design & development ecosystem for FDSOI (Fully Depleted Silicon On Insulator) technology, which is developed in the field of semiconductors. The FDSOI has the characteristics of energy efficiency, large dynamic range, higher absolute performance and higher radiation tolerance, which make it very useful for many applications. The contribution of IGN is to participate in the study and implementation of a portable system which can help pedestrian navigation in urban environment. Our vision based localization method has been integrated into the system. Six DOF pose is provided and uncertainties for the poses are estimated to quantify the performance of localization. The algorithms such as feature detection, matching, landmark detection are being implemented on electrical chips to achieve real-time localization.

6.2 Perspectives

6.2.1 Landmark based localization using top-down approach

A top-down approach regards the system as a whole throughout the development which drives the subsystems all the way down to base elements. As the method proposed by Aynaud et al. [2014], the top-down process can also be used in localization. A Bayesian network based approach is developed for the integration of map with a localization system containing GNSS, laser scanner and odometer.

In our future work, a top-down approach for localization can be developed with the consideration of uncertainty. In this kind of approach, different sensors (e.g. camera, odometer, GNSS, IMU, etc.) and different maps (e.g. OSM, DEM, 3D model, geo-referenced aerial images) with different precision, density and ambiguity, can be combined together for localization. According to the requirement of different applications (e.g. accuracy, processing time, confidence etc.), the most relevant method is determined for each one to chose which sensor we should use for data perception, which kind of landmarks should be integrated and which algorithm should be applied for landmark matching. By considering the uncertainty, the measurements can be

correctly weighted throughout localization though some inaccurate observations are included.

6.2.2 Integration of low-cost sensors

6.2.2.1 Low-cost GNSS

In our localization system, we use GNSS to provide position of start point for initialization. Therefore, it is natural that integrates the measurements from a GNSS throughout our localization system. The GNSS measurements can be integrated which can improve the accuracy and robustness of localization further. We tested the integration of GPS data with monocular sequence using simulated GPS positions which are are generated by adding Gaussian noises into the ground truth positions ($mean = 0, \sigma = 5m$). The simulated experiment shows the potential of the GPS integration (*cf.* figure 6.4). We can see the drift for LBA based localization due to error accumulation from interest point detection and matching (blue path), but the drift can be reduced if we integrate the noisy simulation of GPS positions (red path is closer to ground truth most of the time). The average error for localization is reduced from $1.6m$ to $1.2m$ using the inaccurate GPS measurements (*cf.* figure 6.4). In practice, the system should be able to reject the gross errors in real GPS measurements since the noises of the real data are not always Gaussian distribution.

The loosely coupled integration of GPS with LBA for vision based localization has been investigated by Lhuillier [2012], where a constrained LBA was developed. The outliers of GPS measurements are rejected by enforcing an upper bound for the back-projection errors. Recently, the tightly coupled integration of GPS is proposed for localization based on EKF [Aumayer et al., 2014] or particle filter [Schreiber et al., 2016]. In these approaches, the error of every pseudo-range measurement can be taken into account. In particular, even though measurement from only one satellite is obtained precisely, the pseudo-range can still be used for the integration. Thus, higher localization accuracy can be reached. In our future work, tightly coupled integration of low-cost GNSS considering the uncertainty propagation, would be an interesting way to enhance the localization.

6.2.2.2 Low-cost IMU

Precise IMU is very expensive, but we can use low cost IMU in the localization. The IMU can compute the relative motion by accumulating the acceleration and rotary rate. Our interest is to use IMU for guided matching and relative constraining. The relevant research was started by Roumeliotis et al. [2002] who used IMU for feature points tracking and then fused the IMU measurements with image based pose estimation into a general Kalman filter methodology. Nevertheless, we can promote this method by integrating the IMU measurements in LBA considering the uncertainty propagation.

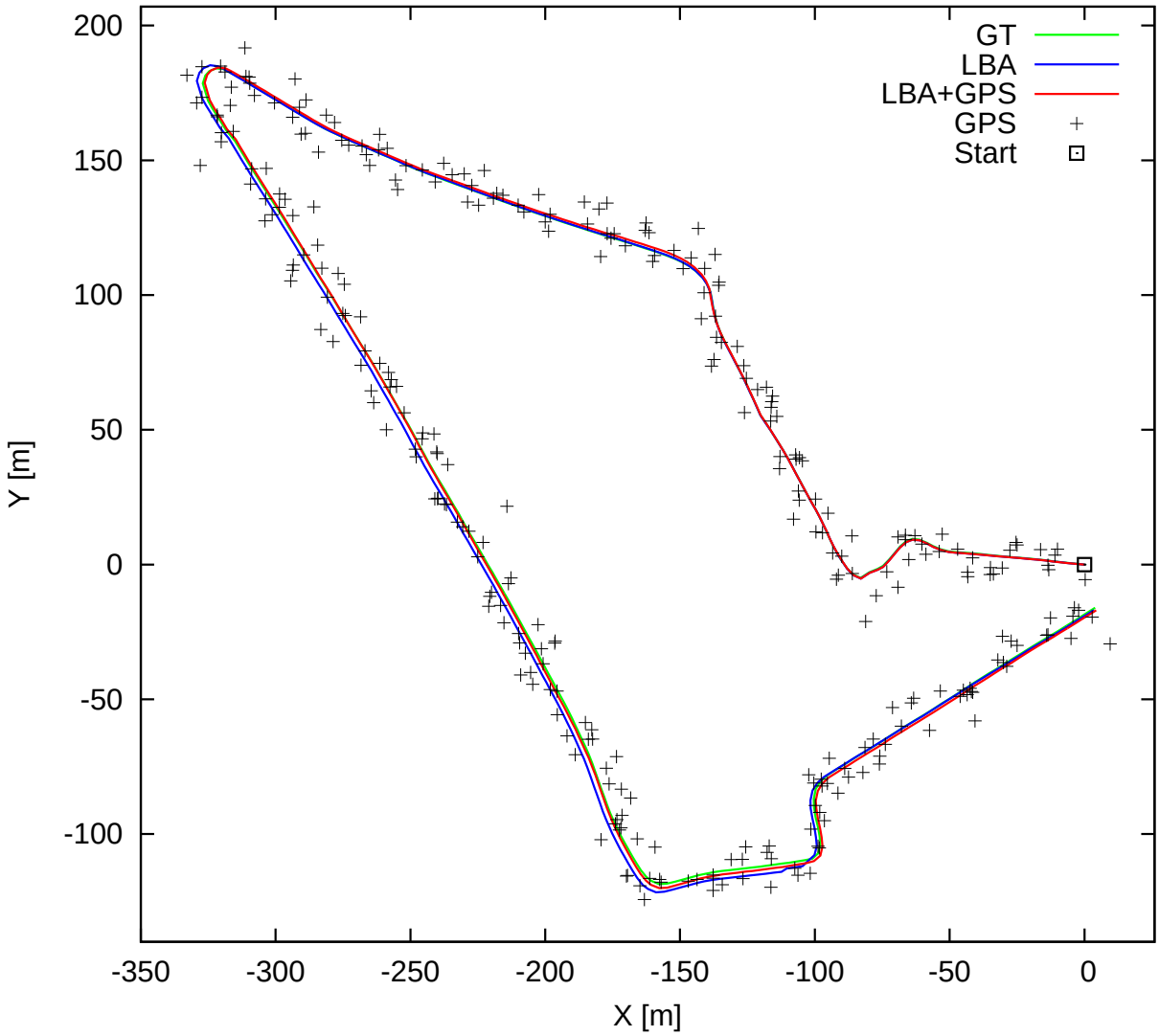


Figure 6.4: Integrating simulated GPS points with monocular based localization. Green: ground truth. Blue: LBA based localization. Red: GPS data integration with LBA based localization.

Guiding the matching and tracking. For low cost IMU, the drift would accumulate very fast over time, but the relative motion would be precise enough to predict the motion for new images. In this case, we can use IMU to replace our current method for motion prediction in propagation based matching tracking. Our current motion model for prediction is built based on the assumption of smooth moving. However, this is not the case when the robot or vehicle move slowly or turns suddenly. With the help of IMU measurement, the prediction can be more robust, so that we can obtain more accurate matching and tracking results.

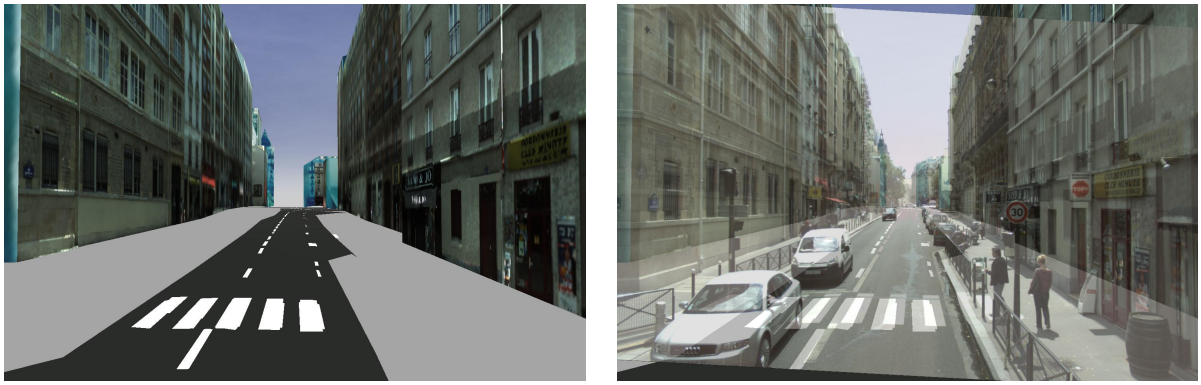
Constraints for pose estimation with uncertainty analysis The relative pose between two successive locations can be measured by IMU, thus the LBA can be conducted under the constraints of these relative poses. It means every two adjacent image poses are constrained by the relative pose from IMU. In particular, the uncertainties of the IMU measurements can be considered into data fusion, thus, the error accumulation can be controlled despite some low

quality measurements are included. Therefore, the robustness of localization can be increased.

6.2.3 Integrating multi-level landmarks

Matching with multi-level landmarks. Our future experiments need to integrate both traffic signs and road markings as landmarks for localization at the same time. The fusion of both landmarks should obviously provide more accurate localization. Furthermore, our current localization system assumes that there are no false objects in the landmark map and that the uncertainty of initial pose is sufficiently low to avoid ambiguities in landmark matching step. To improve the robustness of landmark association, we would match all the visible landmarks in a view at the same time instead of matching each object separately.

Integration of 3D model With aerial and street-based mapping, the integrated 3D city model can be generated (*cf.* Fig 6.5(a)) [Soheilian et al., 2013b]. The building models are reconstructed using aerial photogrammetry. The building facades are textured with the ground based images captured by high resolution camera on MMS. In particular, some street features (road marking and road signs) are extracted and reconstructed by means of the data acquired by MMS. In



(a) Integrated 3D city model (b) Projecting 3D model into image

Figure 6.5: Projecting the 3D model into image with the approximate image pose.

general, 3D building models are not as precise as the semantic features like traffic signs or road markings. On the one hand, we consider uncertainty of 3D build model that can give the correct weight for the integration. On the other hand, the 3D city model can be used to guide the matching and segmentation of images. With approximate image poses and 3D model, we can have knowledge that which parts would be the road and which parts would be the building facades in images (*cf.* Fig 6.5). Thus, a semantic segmentation can be done for the images. Then, these informations can help us to detect salient feature points and obtain robust matches. In addition, based on the semantic segmentation, the perspective deformation can be rectified for the image patches located on the road and building facade with known mounted orientation

of on-board camera on vehicle. This can improve the accuracy of interest point matching based on window correlation.

6.2.4 Integrating image segmentation with localization

In vision based localization, massive interest points might be detected on trees or moving objects, which can generate inaccurate matches and lead to low quality pose estimation. In order to detect robust interest points in image, one solution would be to combine image segmentation with feature detection.

Semantic segmentation Figure 6.6 demonstrates a semantic segmentation of image based on deep learning. It recognizes objects and recovers the 2D outline of the object in image [Zheng

(a) Input image (b) Semantic segmentation

Figure 6.6: Semantic segmentation based on deep learning.

et al., 2015]. As shown in figure 6.6(b), the cars (gray mask) and pedestrian (pink mask) are recognized automatically. We can therefore remove the interest points detected on these moving objects when we use the image for localization. To do this, we need to select a huge amounts of samples for training before segmentation.

Saliency maps Another way is to produce a visual saliency map for each image by analyzing image content. We only use the points located in the salient parts in image. Figure 6.7 is a saliency map. The white areas are considered as the key parts in image (building facades), which can be used to filter the low quality interest points (e.g. interest points on trees). To generate previous salient map, the line segments are detected over image. Then the gradient direction for the pixels on line segments are analyzed via direction histogram in a local area. If the area contains only one major direction (edge) or two major directions (corner), the area is considered as salient [Guissous, 2017]. The areas such as trees or lawn which have multiple directions, are recognized. This method do not need training and is easy to be integrated with localization. The downside of using this kind of saliency map is that they can not identity the points on moving object. Besides, some small objects such as road markings are ignored when

Figure 6.7: Visual salient map (white parts).

generated the map, but the points detected on those objects are very important in poor texture scenarios such as localization on high speed road.

6.2.5 Network design

All the above-mentioned perspectives aim at improving the performance of localization. In IGN, there are a lot research about photogrammetry. An open-source photogrammetry software which is MicMac, is developed for 3D reconstruction [Deseilligny and Clery, 2011]. It has been used in various fields such as environment [Rosu et al., 2015], forestry [Lisein et al., 2013], monitoring [Galland et al., 2016]. For future work, it could be interesting to integrate uncertainty analysis with MicMac.

Apart from this, the network design technique can be used to generate the best configuration for photogrammetric task. In geodesy, network design refers to establishing the best geometric configuration of a new geodetic network to satisfy the given quality. There are Zero-Order Design (ZOD): the datum problem (solution being free from impact of reference system), First-Order Design (FOD): the configuration problem (position and observations to be made), Second-Order Design (SOD): the weight of observations, Third-Order Design (TOD): the densification problem [Schmitt, 1982]. Our purpose is to describe where the camera should be placed in order to satisfy the requirements of a photogrammetric task. This is a problem of FOD to obtain optimal imaging geometry and 3D modeling. In our future work, we can develop a tool based on network design for photogrammetric task that enables us know where to put the cameras for image acquiring. A application of such as system is in deformation measurement of engineering construction such as bridges and dams.

6.3 Conclusion

In this thesis, we present a low cost but precise localization system based on vision system and geo-referenced landmarks. In order to quantify the potential errors, the uncertainties of

landmarks and image points are taken into account for the integration and we estimate the uncertainties for poses. Furthermore, a propagation based interest points matching and tracking method and propagation based landmark matching strategies are proposed to improve the accuracy and robustness of localization. According to the experiments, the accuracy of localization is improved greatly with the integration of traffic signs and road markings. But we also notice that the accuracy is related to the density of landmarks along street, higher landmark density provides more accurate localization. In our future work, multi-level geo-referenced data can be integrated into localization (e.g. DEM, 3D model, road network, etc.). Although, the external data such as DEM and 3D building model are not as precise as landmarks we used in this work, we can integrate them considering their uncertainties so that the proper weight can be determined and integrated in LBA. In addition, our integrating methods for landmark based localization can be easily extended for data acquired by low cost GNSS and IMU. Furthermore, an efficient landmark matching method could be developed based on template tracking while semantic segmentation can be applied for stable interest points detection.

Bibliography

I Abuhadrous, F Nashashibi, and C Laurgeau. 3d land vehicle localization: A real-time multi-sensor data fusion approach using rtmaps. In *International Conference on Advanced Robotics*, pages 71–76, 2003. 21

Sameer Agarwal, Keir Mierle, and Others. Ceres solver. http://ceres-solver.org. 54, 74

Motilal Agrawal and Kurt Konolige. Real-time localization in outdoor environments using stereo vision and inexpensive gps. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 3, pages 1063–1068. IEEE, 2006. 31, 105

Mirza Tahir Ahmed, Matthew N Dailey, Jose Luis Landabaso, and Nicolas Herrero. Robust key frame extraction for 3d reconstruction from video streams. In *VISAPP (1)*, pages 231–236, 2010. 52

Ignacio Parra Alonso, David Fernández Llorca, Miguel Gavilán, Sergio Ávarez Pardo, Miguel Ángel García-Garrido, Ljubo Vlacic, and Miguel Ángel Sotelo. Accurate global localization using visual odometry and digital maps on urban environments. *Intelligent Transportation Systems, IEEE Transactions on*, 13(4):1535–1545, 2012. 33, 105

Fawaz Alsaade. Fast and accurate template matching algorithm based on image pyramid and sum of absolute difference similarity measure. *Research Journal of information Technology*, 4 (4):204–211, 2012. 85

Antonio Angrisano. Gnss/ins integration methods. *Dottorato di ricerca (PhD) in Scienze Geodetiche e Topografiche Thesis, Universita'degli Studi di Napoli PARTHENOPE, Naple*, 2010. 21

Xavier Armangué and Joaquim Salvi. Overall view regarding fundamental matrix estimation. *Image and vision computing*, 21(2):205–220, 2003. 93

Leopoldo Armesto, Josep Tornero, and Markus Vincze. Fast ego-motion estimation with multi-rate fusion of inertial and vision. *The International Journal of Robotics Research*, 26(6):577–589, 2007. 20

Clemens Arth, Christian Pirchheim, Vincent Lepetit, and Jonathan Ventura. Global 6dof pose estimation from untextured 2d city models. *arXiv preprint arXiv:1503.02675*, 2015a. 34

Clemens Arth, Christian Pirchheim, Jonathan Ventura, Dieter Schmalstieg, and Vincent Lepetit. Instant outdoor localization and slam initialization from 2.5d maps. In *Proceedings of the International Symposium on Mixed and Augmented Reality*, 2015b. 105

Bernhard M Aumayer, Mark G Petovello, and Gérard Lachapelle. Development of a tightly coupled vision/gnss system. In *Proc. ION GNSS*, 2014. 31, 37, 144

Claude Aynaud, Coralie Bernay-Angeletti, Roland Chapuis, Romuald Aufrère, Christophe Debain, and Nadir Karam. Real-time vehicle localization by using a top-down process. In *Information Fusion (FUSION), 2014 17th International Conference on*, pages 1–6. IEEE, 2014. 30, 135, 143

Paramvir Bahl and Venkata N Padmanabhan. Radar: An in-building rf-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 775–784. Ieee, 2000. 14

Yashar Balazadegan Sarvrood, Siavash Hosseinyalamdary, and Yang Gao. Visual-lidar odometry aided by reduced imu. *ISPRS International Journal of Geo-Information*, 5(1):3, 2016. 20

Mónica Ballesta, Arturo Gil, Oscar Reinoso, and O Martinez Mozos. Evaluation of interest point detectors for visual slam. *International Journal of Factory Automation, Robotics and Soft Computing*, 4:86–95, 2007. 45

M. Barnada, C. Conrad, H. Bradler, M. Ochs, and R. Mester. Estimation of automotive pitch, yaw, and roll using enhanced phase correlation on multiple far-field windows. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 481–486, June 2015. doi: 10.1109/IVS.2015.7225731. 18, 77, 79

Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Computer vision–ECCV 2006*, pages 404–417. Springer, 2006. 18, 45, 87

Norman Beck. *Guide to GPS positioning*. Larry d Hothem, 1986. 13

S Belongie. Notes on corner detection. 2000. 88

Amani Ben-Afia, Lina Deambrogio, Daniel Salós, Anne-Christine Escher, Christophe Macabiau, Laurent Soulier, and Vincent Gay-Bellile. Review and classification of vision-based localisation techniques in unknown environments. *IET Radar, Sonar & Navigation*, 8(9):1059–1072, 2014. 26

Ouided Bentrah, Nicolas Paparoditis, Marc Pierrot-Deseilligny, and Radu Horaud. Estimating sensor pose from images of a stereo rig. *ISPRS, Istanbul*, 2004. 2

Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992. 23

Masahiro Bessho, N Koshizuka, Shinsuke Kobayashi, and Ken Sakamura. Location systems for ubiquitous computing. *J. IEICE*, 92(4):249–255, 2009. 15

David Betaille, Sébastien Peyraud, Florian Mougel, Stéphane Renault, Miguel Ortiz, Dominique Meizel, and François Peyret. Using road constraints to progress in real-time nlos detection. In *Workshop Navigation, Perception, Accurate Positioning and Mapping for Intelligent Vehicles*, pages 6–p, 2012. 22

Joydeep Biswas and Manuela Veloso. Depth camera based indoor mobile robot localization and navigation. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1697–1702. IEEE, 2012. 20

Christoph Bodensteiner, Wolfgang Hübner, Kai Jüngling, Peter Solbrig, and Michael Arens. Monocular camera trajectory optimization using lidar data. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 2018–2025. IEEE, 2011. 32, 37

Henry Bradler, Birthe Anne Wiegand, and Rudolf Mester. The statistics of driving sequences–and what we can learn from them. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 17–25, 2015. 77, 78, 79

Sotiris Brakatsoulas, Dieter Pfoser, Randall Salas, and Carola Wenk. On map-matching vehicle tracking data. In *Proceedings of the 31st international conference on Very large data bases*, pages 853–864. VLDB Endowment, 2005. 22

Claus Brenner. Vehicle localization using landmarks obtained by a lidar mobile mapping system. *Int. Arch. Photogramm. Remote Sens*, 38:139–144, 2010. 35

Marcus Brubaker, Andreas Geiger, and Raquel Urtasun. Lost! leveraging the crowd for probabilistic visual self-localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3057–3064, 2013. 33

Martin Buczko and Volker Willert. Flow-decoupled normalized reprojection error for visual odometry. In *19th IEEE Intelligent Transportation Systems Conference (ITSC)*, 2016a. 100

Martin Buczko and Volker Willert. How to distinguish inliers from outliers in visual odometry for high-speed automotive applications. In *IEEE Intelligent Vehicles Symposium (IV)*, 2016b. 100, 101

Jean-Pascal Burochin, Olivier Tournaire, and Nicolas Paparoditis. An unsupervised hierarchical segmentation of a facade building image in elementary 2d-models. In *Proceedings of the ISPRS Workshop on Object Extraction for 3D City Models, Road Databases and Traffic Monitoring, Paris, France*, pages 3–4. Citeseer, 2009. xi, 6

Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792. Springer, 2010. 18

B. Cannelle, N. Paparoditis, M. Pierrot-Deseilligny, and J.-P. Papelard. Off-line vs. on-line calibration of a panoramic-based mobile mapping system. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, I-3:31–36, 2012. doi: 10.5194/isprsannals-I-3-31-2012. URL http://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/I-3/31/2012/. 3

L Caraffa, M Brédif, and B Vallet. 3d octree based watertight mesh generation from ubiquitous data. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(3):613, 2015. 7

Rodrigo Carceroni, Ankita Kumar, and Kostas Daniilidis. Structure from motion with known camera positions. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 477–484. IEEE, 2006. 31

Guillaume Caron, Amaury Dame, and Eric Marchand. Direct model based visual tracking and pose estimation using mutual information. *Image and Vision Computing*, 32(1):54 – 63, 2014. ISSN 0262-8856. xii, 30

Gerardo Carrera, Adrien Angeli, and Andrew J Davison. Lightweight slam and navigation with a multi-camera rig. In *ECMR*, pages 77–82, 2011. 30

David Caruso, Jakob Engel, and Daniel Cremers. Large-scale direct slam for omnidirectional cameras. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 141–148. IEEE, 2015. 29

B. Charmette, E. Royer, and F. Chausse. Efficient planar features matching for robot localization using gpu. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 16–23, June 2010. 32

Chu-Song Chen and Wen-Yan Chang. On pose recovery for generalized visual sensors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7):848–861, July 2004. ISSN 0162-8828. doi: 10.1109/TPAMI.2004.34. 64

Dixiao Cui, Jianru Xue, and Nanning Zheng. Real-time global localization of robotic cars in lane level via lane marking detection and shape registration. *IEEE Transactions on Intelligent Transportation Systems*, 17(4):1039–1050, 2016. 33

Igor Cvišić and Ivan Petrović. Stereo odometry based on careful feature selection and tracking. In *Mobile Robots (ECMR), 2015 European Conference on*, pages 1–6. IEEE, 2015. 92, 100

A.J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1403–1410 vol.2, Oct 2003. doi: 10.1109/ICCV.2003.1238654. 17, 18, 27, 36, 79, 80, 87

Misganu Debella-Gilo and Andreas Kääb. Sub-pixel precision image matching for measuring surface displacements on mass movements using normalized cross-correlation. *Remote Sensing of Environment*, 115(1):130–142, 2011. 86, 87

Frank Dellaert, Dieter Fox, Wolfram Burgard, and Sebastian Thrun. Monte carlo localization for mobile robots. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 2, pages 1322–1328. IEEE, 1999. 25

Jérôme Demantké, Bruno Vallet, and Nicolas Paparoditis. Facade reconstruction with generalized 2.5 d grids. *Int. Ann. Photogramm. Remote Sens. Spat. Inf. Sci*, pages 67–72, 2013. xi, 6, 7

A. Desai and D. J. Lee. Visual odometry drift reduction using syba descriptor and feature transformation. *IEEE Transactions on Intelligent Transportation Systems*, 17(7):1839–1851, July 2016. ISSN 1524-9050. doi: 10.1109/TITS.2015.2511453. 37

M Pierrot Deseilligny and I Clery. Apero, an open source bundle adjusment software for automatic calibration and orientation of set of images. In *Proceedings of the ISPRS Symposium, 3DARCH11*, volume 269277, 2011. 148

Martin Dodge, Simon Doyle, A Hudson-Smith, and Stephen Fleetwood. Towards the virtual city: Vr & internet gis for urban planning. 1998. 1

Arnaud Doucet and Adam M Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12(656-704):3, 2009. 25

Norman Richard Draper and Harry Smith. Applied regression analysis. In *Applied regression analysis*. John Wiley & Sons, 1981. 84

Vincent Drevelle and Philippe Bonnifait. Global positioning in urban areas with 3-d maps. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 764–769. IEEE, 2011. 22

Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *Robotics & Automation Magazine, IEEE*, 13(2):99–110, 2006. 18

Venkatesan N Ekambaram and Kannan Ramchandran. Distributed high accuracy peer-to-peer localization in mobile multipath environments. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–5. IEEE, 2010. 22

Naser El-Sheimy. *The development of VISAT-A mobile survey system for gis applications*. PhD thesis, University of Calgary, 1996. 1

Cameron Ellum. Integration of raw gps measurements into a bundle adjustment. *IAPRS series vol. XXXV*, 3025, 2006. 31

Felix Endres, Jürgen Hess, Nikolas Engelhard, Jürgen Sturm, Daniel Cremers, and Wolfram Burgard. An evaluation of the rgb-d slam system. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1691–1696. IEEE, 2012. 19

Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *Computer Vision–ECCV 2014*, pages 834–849. Springer, 2014. 28

Jakob Engel, Jorg Stuckler, and Daniel Cremers. Large-scale direct slam with stereo cameras. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 1935–1942. IEEE, 2015. 28, 61

Chris Engels, Henrik Stewénius, and David Nistér. Bundle adjustment rules. *Photogrammetric computer vision*, 2:124–131, 2006. 26

Ovid Wallace Eshbach, Byron D Tapley, and Thurman R Poston. *Eshbach's handbook of engineering fundamentals*. John Wiley & Sons, 1990. 17

Andreas Ess, Alexander Neubeck, and Luc J Van Gool. Generalised linear pose estimation. In *BMVC*, pages 1–10, 2007. 64

Alexandre Eudes and Maxime Lhuillier. Error propagations for local bundle adjustment. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2411–2418. IEEE, 2009. 27, 36, 45, 52, 57, 65

Alexandre Eudes, Sylvie Naudet-Collette, Maxime Lhuillier, and Michel Dhome. Weighted local bundle adjustment and application to odometry and visual slam fusion. In *Proceedings of the British Machine Vision Conference*, pages 25.1–25.10. BMVA Press, 2010. ISBN 1-901725-40-5. doi:10.5244/C.24.25. 18, 27

Huaan Fan. *Theory of errors and least squares adjustment*. Tekniska högskolan, 1997. 84

Michela Farenzena, Andrea Fusiello, and Riccardo Gherardi. Structure-and-motion pipeline on a hierarchical cluster tree. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 1489–1496. IEEE, 2009. 27

Jay Farrell. *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill, Inc., New York, NY, USA, 1 edition, 2008. ISBN 0071493298, 9780071493291. 21

Olivier Faugeras, Bernard Hotz, Hervé Mathieu, Thierry Viéville, Zhengyou Zhang, Pascal Fua, Eric Théron, Laurent Moll, Gérard Berry, Jean Vuillemin, et al. Real time correlation-based stereo: algorithm, implementations and applications. Technical report, Inria, 1993. 85

D Flamanc, G Maillet, and H Jibrini. 3d city models: an operational approach using aerial images and cadastral maps. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(3/W8):53–58, 2003. 1

Georgios Floros, Benito van der Zander, and Bastian Leibe. Openstreetslam: Global vehicle localization using openstreetmaps. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 1054–1059. IEEE, 2013. xii, 30

Jeffrey Forshaw and Gavin Smith. *Dynamics and relativity*. John Wiley & Sons, 2014. 79

Wolfgang Förstner and Eberhard Gülch. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *Proc. ISPRS intercommission conference on fast processing of photogrammetric data*, pages 281–305, 1987. 88

Jorge Fuentes-Pacheco, José Ruiz-Ascencio, and Juan Manuel Rendón-Mancha. Visual simultaneous localization and mapping: a survey. *Artificial Intelligence Review*, 43(1):55–81, 2015. ISSN 1573-7462. doi: 10.1007/s10462-012-9365-8. URL http://dx.doi.org/10.1007/s10462-012-9365-8. 23, 27, 30

Sae Fujii, Atsushi Fujita, Takaaki Umedu, Shigeru Kaneda, Hirozumi Yamaguchi, Teruo Higashino, and Mineo Takai. Cooperative vehicle positioning via v2v communications and onboard sensors. In *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, pages 1–5. IEEE, 2011. 22

Paul Furgale and Tim Barfoot. Stereo mapping and localization for long-range path following on rough terrain. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 4410–4416. IEEE, 2010. 32

Olivier Galland, Håvard S Bertelsen, Frank Guldstrand, Luc Girod, Rikke F Johannessen, Fanny Bjugger, Steffi Burchardt, and Karen Mair. Application of open-source photogrammetric software micmac for monitoring surface deformation in laboratory models. *Journal of Geophysical Research: Solid Earth*, 121(4):2852–2872, 2016. 148

Bernard A Galler and Michael J Fisher. An improved equivalence algorithm. *Communications of the ACM*, 7(5):301–303, 1964. 48

Sundaram Ganapathy. Decomposition of transformation matrices for robot vision. *Pattern Recognition Letters*, 2(6):401–412, 1984. 47, 50

Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE transactions on pattern analysis and machine intelligence*, 25(8):930–943, 2003. 51

Emilio Garcia-Fidalgo and Alberto Ortiz. Vision-based topological mapping and localization methods: A survey. *Robotics and Autonomous Systems*, 64:1–20, 2015. 15

Steffen Gauglitz, Tobias Höllerer, and Matthew Turk. Evaluation of interest point detectors and feature descriptors for visual tracking. *International journal of computer vision*, 94(3):335–360, 2011. 45

Andreas Geiger, Julius Ziegler, and Christoph Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 963–968. IEEE, 2011. 28, 37, 61, 92, 93

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and PatternRecognition (CVPR)*, 2012. 76, 96, 97

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, page 0278364913491297, 2013. 96

Haokun Geng, Hsiang-Jen Chien, Radu Nicolescu, and Reinhard Klette. Egomotion estimation and reconstruction with kalman filters and gps integration. In *International Conference on Computer Analysis of Images and Patterns*, pages 399–410. Springer, 2015. 31

Simon Gibson, Jon Cook, Toby Howard, Roger Hubbold, and Dan Oram. Accurate camera calibration for off-line, video-based augmented reality. In *Proceedings of the 1st International Symposium on Mixed and Augmented Reality*, page 37. IEEE Computer Society, 2002. 52

R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. 1992. 85

Adam Goodliss, Christian Manasseh, Venkatesan N Ekambaram, Raja Sengupta, and Kannan Ramchandran. Cooperative high-accuracy location (c-halo) service for intelligent transportation systems: A cost benefit study. In *Proc. 24th Int. Tech. Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2011), Portland, OR*, pages 2220–2232, 2011. 22

Neil J Gordon, David J Salmond, and Adrian FM Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *Radar and Signal Processing, IEE Proceedings F*, volume 140, pages 107–113. IET, 1993. 25

Joshua S Greenfeld. Matching gps observations to locations on a digital map. In *Transportation Research Board 81st Annual Meeting*, 2002. 22

Michael D Grossberg and Shree K Nayar. A general imaging model and a method for finding its parameters. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 108–115. IEEE, 2001. 64

B Grush. The case against map-matching. *Eur. J. Navig*, 6(3):22–25, 2008. 22

Haiyan Guan, Jonathan Li, Yongtao Yu, Cheng Wang, Michael Chapman, and Bisheng Yang. Using mobile laser scanning data for automated extraction of road markings. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87:93–107, 2014. 109

Kamel Guissous. *Saillance visuelle en imagerie urbaine*. PhD thesis, Université Paris Est, 2017. 147

Erico Guizzo. How google's self-driving car works. *IEEE Spectrum Online, October*, 18, 2011. 1

A. Gupta, H. Chang, and A. Yilmaz. Gps-denied geo-localisation using visual odometry. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, III-3:263–270, 2016. doi: 10.5194/isprs-annals-III-3-263-2016. URL http://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/III-3/263/2016/. 33, 49, 105

Peter Hansen, Peter Corke, and Wageeh Boles. Wide-angle visual feature matching for outdoor localization. *The International Journal of Robotics Research*, 2009. 29

Chris Harris and Mike Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Citeseer, 1988. 18, 87

Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster. The anatomy of a context-aware application. *Wireless Networks*, 8(2/3):187–197, 2002. 14

Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. xii, 23, 42, 51, 90, 93

Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox. Rgb-d mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments. *The International Journal of Robotics Research*, 31(5):647–663, 2012. 20

Miguel Heredia, Felix Endres, Wolfram Burgard, and Rafael Sanz. Fast and robust feature matching for rgb-d based localization. *arXiv preprint arXiv:1502.00500*, 2015. 20

A. Hervieu, B. Soheilian, and M. Brédif. Road marking extraction using a model&data-driven rj-mcmc. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W4:47–54, 2015. doi: 10.5194/isprsannals-II-3-W4-47-2015. URL http://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/II-3-W4/47/2015/. xi, xv, 6, 35, 109, 110, 111, 113, 119

Alexandre Hervieu and Bahman Soheilian. Road side detection and reconstruction using lidar sensor. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 1247–1252. IEEE, 2013. xi, 6

Joel A Hesch, Faraz M Mirzaei, Gian Luca Mariottini, and Stergios I Roumeliotis. A laser-aided inertial navigation system (l-ins) for human localization in unknown indoor environments. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 5376–5382. IEEE, 2010. 20

Jeffrey Hightower and Gaetano Borriello. Location sensing techniques. *IEEE Computer*, 34(8): 57–66, 2001. xii, 14

Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and James Collins. Global positioning system (gps). theory and practice. *Wien: Springer, 1992*, 1, 1992. 13

Dirk Holz, Stefan Holzer, Radu Bogdan Rusu, and Sven Behnke. Real-time plane segmentation using rgb-d cameras. In *Robot Soccer World Cup*, pages 306–317. Springer, 2011. 20

Radu Horaud, Bernard Conio, Olivier Leboulleux, and Bernard Lacolle. An analytic solution for the perspective 4-point problem. In *Computer Vision and Pattern Recognition, 1989. Proceedings CVPR'89., IEEE Computer Society Conference on*, pages 500–507. IEEE, 1989. 23

Arnold Irschara, Christopher Zach, Jan-Michael Frahm, and Horst Bischof. From structure-from-motion point clouds to fast location recognition. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2599–2606. IEEE, 2009. xii, 16, 48

Marion Jaud, Raphaël Rouveure, Patrice Faure, and Marie-Odile Monod. Methods for fmcw radar map georeferencing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 84:33–42, 2013. 105

Andrew H Jazwinski. *Stochastic Processes and Filtering Theory*, volume 64. Academic Press, 1970. 24

S Ji and X Yuan. A generic probabilistic model and a hierarchical solution for sensor localization in noisy and restricted conditions. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 41, 2016. 26, 52

Shunping Ji, Yun Shi, Jie Shan, Xiaowei Shao, Zhongchao Shi, Xiuxiao Yuan, Peng Yang, Wenbin Wu, Huajun Tang, and Ryosuke Shibasaki. Particle filtering methods for georeferencing panoramic image sequence in complex urban scenes. *ISPRS Journal of Photogrammetry and Remote Sensing*, 105:1–12, 2015. 25, 32, 105

Kichun Jo, Keonyup Chu, and Myoungho Sunwoo. Gps-bias correction for precise localization of autonomous vehicles. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 636–641. IEEE, 2013. 33

Simon J Julier and Jeffrey K Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, 2004. 25

F. Jurie and M. Dhome. Hyperplane approximation for template matching. *TPAMI*, 24(7): 996–1000, Jul 2002. 135

Michael Kaess and Frank Dellaert. Probabilistic structure matching for visual slam with a multi-camera rig. *Comput. Vis. Image Underst.*, 114(2):286–296, February 2010. ISSN 1077-3142. doi: 10.1016/j.cviu.2009.07.006. URL http://dx.doi.org/10.1016/j.cviu.2009.07.006. 36

Nadir Karam, Frédéric Chausse, Romuald Aufrère, and Roland Chapuis. Localization of a group of communicating vehicles by state exchange. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 519–524. IEEE, 2006. 22

N Karbo and R Schroth. Oblique aerial photography: a status review. In *Photogrammetric week*, volume 9, pages 119–125, 2009. 1

Myron Kayton and Walter R Fried. *Avionics navigation systems*. John Wiley & Sons, 1997. 14

Tim Kazik, Laurent Kneip, Janosch Nikolic, Marc Pollefeys, and Roland Siegwart. Real-time 6d stereo visual odometry with non-overlapping fields of view. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1529–1536. IEEE, 2012. 37

Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2938–2946, 2015. 16

Solmaz S Kia, Stephen Rounds, and Sonia Martınez. Cooperative localization for mobile agents. *arXiv preprint arXiv:1505.05908*, 2015. 22

Jin Gon Kim, Dong Yeob Han, Ki Yun Yu, Yong Il Kim, and Sung Mo Rhee. Efficient extraction of road information for car navigation applications using road pavement markings obtained from aerial images. *Canadian Journal of Civil Engineering*, 33(10):1320–1331, 2006. 109

Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pages 225–234. IEEE, 2007. 26, 27, 36, 38

L. Kneip, P. Furgale, and R. Siegwart. Using multi-camera systems in robotics: Efficient solutions to the npnp problem. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 3770–3776, May 2013. doi: 10.1109/ICRA.2013.6631107. 18, 30, 37, 64, 65

Laurent Kneip, Margarita Chli, Roland Siegwart, et al. Robust real-time visual odometry with a single camera and an imu. In *BMVC*, pages 1–11, 2011a. 20

Laurent Kneip, Davide Scaramuzza, and Roland Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2969–2976. IEEE, 2011b. 51

Kurt Konolige and James Bowman. Towards lifelong visual maps. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1156–1163. IEEE, 2009. 32

Hakan Koyuncu and Shuang Hua Yang. A survey of indoor positioning and object locating systems. *IJCSNS International Journal of Computer Science and Network Security*, 10(5):121–128, 2010. 15

Hideyuki Kume, Takafumi Taketomi, Tomokazu Sato, and Naokazu Yokoya. Extrinsic camera parameter estimation using video images and gps considering gps positioning accuracy. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 3923–3926. IEEE, 2010. 31

Hideyuki Kume, Tomokazu Sato, and Naokazu Yokoya. Bundle adjustment using aerial images with two-stage geometric verification. *Computer Vision and Image Understanding*, 138:74–84, 2015. 32

Rainer Kümmerle, Bastian Steder, Christian Dornhege, Michael Ruhnke, Giorgio Grisetti, Cyrill Stachniss, and Alexander Kleiner. On measuring the accuracy of slam algorithms. *Autonomous Robots*, 27(4):387–407, 2009. 96

Dorra Larnaout, Steve Bourgeois, Vincent Gay-Bellile, and Michel Dhome. Towards bundle adjustment with gis constraints for online geo-localization of a vehicle in urban center. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*, pages 348–355. IEEE, 2012. 34, 37, 105

Dorra Larnaout, Vincent Gay-Bellile, Steve Bourgeois, and Michel Dhome. Vehicle 6-dof localization based on slam constrained by gps and digital elevation model information. In *2013 IEEE International Conference on Image Processing*, pages 2504–2508. IEEE, 2013. 34, 37

Khaoula Lassoued, Philippe Bonnifait, and Isabelle Fantoni. Cooperative localization of vehicles sharing gnss pseudoranges corrections with no base station using set inversion. In *Intelligent Vehicles Symposium (IV), 2016 IEEE*, pages 496–501. IEEE, 2016. 22

Henning Lategahn, Markus Schreiber, Julius Ziegler, and Christoph Stiller. Urban localization with camera and inertial measurement unit. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 719–724. IEEE, 2013. 20

J Ph Lauffenburger, Benazouz Bradai, Michel Basset, and Fawzi Nashashibi. Navigation and speed signs recognition fusion for enhanced vehicle location. *IFAC Proceedings Volumes*, 41 (2):2069–2074, 2008. 34

Sang-Seol Lee, Sung-Joon Jang, Jungho Kim, Youngbae Hwang, and Byeongho Choi. Memory-efficient surf architecture for asic implementation. *Electronics Letters*, 50(15):1058–1059, 2014. 45

Keith Yu Kit Leung, Christopher M Clark, and Jan P Huissoon. Localization in urban environments by matching ground level video images with an aerial image. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 551–556. IEEE, 2008. 32

Jesse Levinson, Michael Montemerlo, and Sebastian Thrun. Map-based precision vehicle localization in urban environments. In *Robotics: Science and Systems*, volume 4, page 1. Citeseer, 2007. xii, 23, 30, 32

JP Lewis. Fast normalized cross-correlation. In *Vision interface*, volume 10, pages 120–123, 1995. 85

Maxime Lhuillier. Automatic structure and motion using a catadioptric camera. In *Proceedings of the 6th Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras*, 2005. 29

Maxime Lhuillier. Toward automatic 3d modeling of scenes using a generic camera model. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 29

Maxime Lhuillier. Fusion of gps and structure-from-motion using constrained bundle adjustments. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3025–3032. IEEE, 2011. xii, 30, 31, 36

Maxime Lhuillier. Incremental fusion of structure-from-motion and gps using constrained bundle adjustments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(12): 2489–2495, 2012. 31, 105, 144

Hao Li and Fawzi Nashashibi. Cooperative multi-vehicle localization using split covariance intersection filter. *IEEE Intelligent transportation systems magazine*, 5(2):33–44, 2013. 22

Hao Li, Fawzi Nashashibi, and Gwénaëlle Toulminet. Localization for intelligent vehicle by fusing mono-camera, low-cost gps and map data. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 1657–1662. IEEE, 2010. 22

Reoxiang Li, Bing Zeng, and Ming L Liou. A new three-step search algorithm for block motion estimation. *IEEE transactions on circuits and systems for video technology*, 4(4):438–442, 1994. 85

Rongbing Li, Jianye Liu, Ling Zhang, and Yijun Hang. Lidar/mems imu integrated navigation (slam) method for a small uav in indoor environments. In *2014 DGON Inertial Sensors and Systems (ISS)*, pages 1–15. IEEE, 2014. 20

F Liebold and HG Maas. Integrated georeferencing of lidar and camera data acquired from a moving platform. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(3):191, 2014. 20

Fredrik Lindsten, Jonas Callmer, Henrik Ohlsson, David Törnqvist, Thomas B Schön, and Fredrik Gustafsson. Geo-referencing for uav navigation using environmental classification. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 1420–1425. IEEE, 2010. 15

Jonathan Lisein, Marc Pierrot-Deseilligny, Stéphanie Bonnet, and Philippe Lejeune. A photogrammetric workflow for the creation of a forest canopy height model from small unmanned aerial system imagery. *Forests*, 4(4):922–944, 2013. 148

Vadim Litvinov, Maxime Lhuillier, and France Aubière. Incremental solid modeling from sparse and omnidirectional structure-from-motion data. In *BMVC*, volume 2, page 4, 2013. 29

JG Liu and J McM Moore. Hue image rgb colour composition. a simple technique to suppress shadow and enhance spectral signature. *International Journal of Remote Sensing*, 11(8):1521–1530, 1990. 85

Pierre Lothe, Steve Bourgeois, Fabien Dekeyser, Eric Royer, and Michel Dhome. Towards geographical referencing of monocular slam reconstruction using 3d city models: Application to real-time accurate vision-based localization. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2882–2889. IEEE, 2009. 34, 36

David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 18, 45, 47, 57, 87

Feng Lu and Evangelos Milios. Globally consistent range scan alignment for environment mapping. *Autonomous robots*, 4(4):333–349, 1997a. 19

Feng Lu and Evangelos Milios. Robot pose estimation in unknown environments by matching 2d range scans. *Journal of Intelligent and Robotic Systems*, 18(3):249–275, 1997b. 19

Yan Lu, Dezhen Song, and Jingang Yi. High level landmark-based visual navigation using unsupervised geometric constraints in local bundle adjustment. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 1540–1545. IEEE, 2014. 35, 37

Thomas Luhmann, Clive Fraser, and Hans-Gerd Maas. Sensor modelling and camera calibration for close-range photogrammetry. *ISPRS Journal of Photogrammetry and Remote Sensing*, 115: 37–46, 2016. 29

Quan-Tuan Luong and Olivier D Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *International journal of computer vision*, 17(1):43–75, 1996. 93

Marc Luxen. Variance component estimation in performance characteristics applied to feature extraction procedures. In *Pattern Recognition*, pages 498–506. Springer, 2003. 57

Will Maddern, Alexander D Stewart, and Paul Newman. Laps-ii: 6-dof day and night visual localisation with prior 3d structure for autonomous road vehicles. In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pages 330–337. IEEE, 2014. 33

H Malmström. Measuring ground control points for satellite image rectification. *Schriftenreihe des Instituts für Photogrametrie der Universität Stuttgart. Proceedings*, 11:127–135, 1986. 105, 106

J. Chris McGlone, Edward M. Mikhail, and James S. Bethel. *Manual of Photogrammetry*. American Society of Photogrammetry and Remote Sensing, 5th eddition edition, 2004. 26, 50

Julien Michot, Adrien Bartoli, and François Gaspard. Bi-objective bundle adjustment with application to multi-sensor slam. *3DPVT'10*, 3025, 2010. 31

Annalisa Milella and Roland Siegwart. Stereo-based ego-motion estimation using pixel tracking and iterative closest point. In *Computer Vision Systems, 2006 ICVS'06. IEEE International Conference on*, pages 21–21. IEEE, 2006. 23, 28, 61

A Miraliakbari, M Hahn, and HG Maas. Development of a multi-sensor system for road condition mapping. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(1):265, 2014. 1

Takeo Miyasaka, Yoshihiro Ohama, and Yoshiki Ninomiya. Ego-motion estimation and moving object tracking using multi-layer lidar. In *Intelligent Vehicles Symposium, 2009 IEEE*, pages 151–156. IEEE, 2009. 19

Fabrice Monnier, Bruno Vallet, and Bahman Soheilian. Trees detection from laser point clouds acquired in dense urban areas by a mobile mapping system. *Proceedings of the ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Annals), Melbourne, Australia*, 25:245–250, 2012. xi, 7

Michael Montemerlo, Sebastian Thrun, Daphne Koller, and Ben Wegbreit. Fastslam: A factored solution to the simultaneous localization and mapping problem. In *Eighteenth National Conference on Artificial Intelligence*, pages 593–598, Menlo Park, CA, USA, 2002. American Association for Artificial Intelligence. ISBN 0-262-51129-0. URL http://dl.acm.org/citation.cfm?id=777092.777184. 18, 25

Frank Moosmann and Thierry Fraichard. Motion estimation from range images in dynamic outdoor scenes. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 142–147. IEEE, 2010. 19

Frank Moosmann and Christoph Stiller. Velodyne slam. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 393–398. IEEE, 2011. 19

Mohamed MR Mostafa and Joseph Hutton. Direct positioning and orientation systems: How do they work? what is the attainable accuracy. In *Proceedings, The American Society of Photogrammetry and Remote Sensing Annual Meeting, St. Louis, MO, USA, April*, pages 23–27, 2001. 21

Tarek Mouats, Nabil Aouf, Angel Domingo Sappa, Cristhian Aguilera, and Ricardo Toledo. Multispectral stereo odometry. *IEEE Transactions on Intelligent Transportation Systems*, 16 (3):1210–1224, 2015. 18

Pierre Moulon and Pascal Monasse. Unordered feature tracking made fast and easy. In *CVMP 2012*, page 1, 2012. 48, 76

Pierre Moulon, Pascal Monasse, and Renaud Marlet. Adaptive structure from motion with a contrario model estimation. In *Asian Conference on Computer Vision*, pages 257–270. Springer, 2012. 26, 47

E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Real time localization and 3d reconstruction. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 363–370, 2006. doi: 10.1109/CVPR.2006.236. 18, 27, 28, 36, 52, 65, 87

Etienne Mouragnon, Maxime Lhuillier, Michel Dhome, Fabien Dekeyser, and Patrick Sayd. Generic and real-time structure from motion using local bundle adjustment. *Image and Vision Computing*, 27(8):1178–1193, 2009. 36

Marius Muja and David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *International Conference on Computer Vision Theory and Application VISSAPP'09)*, pages 331–340. INSTICC Press, 2009. 46

A. C. Murillo, J. J. Guerrero, and C. Sagues. Surf features for efficient robot localization with omnidirectional images. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 3901–3907, April 2007. doi: 10.1109/ROBOT.2007.364077. 45

DK Naidu and Robert B Fisher. A comparative analysis of algorithms for determining the peak position of a stripe to sub-pixel accuracy. In *BMVC91*, pages 217–225. Springer, 1991. 86, 87

Sergiu Nedevschi, Voichita Popescu, Radu Danescu, Tiberiu Marita, and Florin Oniga. Accurate ego-vehicle global localization at intersections through alignment of visual data with digital map. *Intelligent Transportation Systems, IEEE Transactions on*, 14(2):673–687, 2013. 33

D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–652–I–659 Vol.1, June 2004. doi: 10.1109/CVPR.2004.1315094. 18, 28, 36, 61, 87

David Nistér. An efficient solution to the five-point relative pose problem. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(6):756–770, 2004. 47, 50

David Nistér and Henrik Stewénius. A minimal solution to the generalised 3-point pose problem. *Journal of Mathematical Imaging and Vision*, 27(1):67–79, 2007. xiii, 64

Henry Bradler Nolang Fanani, Matthias Ochs and Rudolf Mester. Keypoint trajectory estimation using propagation based tracking. In *IEEE Intelligent Vehicles Symposium (IV)*, 2016. 77

Andreas Nüchter, Kai Lingemann, Joachim Hertzberg, and Hartmut Surmann. 6d slam—3d mapping outdoor environments. *Journal of Field Robotics*, 24(8-9):699–722, 2007. 19, 23

Benjamin Ochoa and Serge Belongie. Covariance propagation for guided matching. In *Proceedings of the Workshop on Statistical Methods in Multi-Image and Video Processing (SMVP)*, 2006. 83

C. F. Olson, L. H. Matthies, M. Schoppers, and M. W. Maimone. Stereo ego-motion improvements for robust rover navigation. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 2, pages 1099–1104 vol.2, 2001. doi: 10.1109/ROBOT.2001.932758. 61

Clark F Olson, Larry H Matthies, H Schoppers, and Mark W Maimone. Robust stereo ego-motion for long distance navigation. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 2, pages 453–458. IEEE, 2000. 18, 28, 36

Nicolas Paparoditis, Jean-Pierre Papelard, Bertrand Cannelle, Alexandre Devaux, Bahman Soheilian, Nicolas David, and Erwann Houzay. Stereopolis ii: A multi-purpose and multi-sensor 3d mobile mapping system for street visualisation and 3d metrology. *Revue française de photogrammétrie et de télédétection*, 200(1):69–79, 2012. 2, 58, 107, 123

Lionel Pénard, Nicolas Paparoditis, and Marc Pierrot-Deseilligny. 3d building facade reconstruction under mesh form from multiple wide angle views. *Proceedings of the ISPRS Working Group*, 4, 2005. xi, 6, 7

M. Persson, T. Piccini, M. Felsberg, and R. Mester. Robust stereo visual odometry from monocular techniques. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 686–691, June 2015. doi: 10.1109/IVS.2015.7225764. 27, 37, 79

Tom Pilutti and A Galip Ulsoy. Identification of driver state for lane-keeping tasks. *IEEE transactions on systems, man, and cybernetics-Part A: Systems and humans*, 29(5):486–502, 1999. 34

Oliver Pink. Visual map matching and localization using a global feature map. In *Computer vision and pattern recognition workshops*, pages 1–7, 2008. 35

Marc Pollefeys, Luc Van Gool, Maarten Vergauwen, F Verbiest, K Cornelis, and Jan Tops. Video-to-3d. *International archives of photogrammetry remote sensing and spatial information sciences*, 34(3/A):252–257, 2002. 52

Marc Pollefeys, Luc Van Gool, Maarten Vergauwen, Frank Verbiest, Kurt Cornelis, Jan Tops, and Reinhard Koch. Visual modeling with a hand-held camera. *International Journal of Computer Vision*, 59(3):207–232, 2004. 76

Nissanka B Priyantha, Anit Chakraborty, and Hari Balakrishnan. The cricket location-support system. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 32–43. ACM, 2000. 14

X. Qu, B. Soheilian, E. Habets, and N. Paparoditis. Evaluation of Sift and Surf for Vision Based Localization. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 685–692, June 2016. doi: 10.5194/isprs-archives-XLI-B3-685-2016. 46

Long Quan and Zhongdan Lan. Linear n-point camera pose determination. *IEEE Transactions on pattern analysis and machine intelligence*, 21(8):774–780, 1999. 23, 51

Mohammed A Quddus, Washington Y Ochieng, and Robert 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. 22

David Ribas, Pere Ridao, Juan Domingo Tardós, and José Neira. Underwater slam in man-made structured environments. *Journal of Field Robotics*, 25(11-12):898–921, 2008. 19

Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *Computer Vision–ECCV 2006*, pages 430–443. Springer, 2006. 18, 87

Edward Rosten, Reid Porter, and Tom Drummond. Faster and better: A machine learning approach to corner detection. *IEEE transactions on pattern analysis and machine intelligence*, 32(1):105–119, 2010. 87

Ana-Maria Rosu, Michel Assenbaum, Ywenn De la Torre, and Marc Pierrot-Deseilligny. Coastal digital surface model on low contrast images. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(3):307, 2015. 148

Stergios I Roumeliotis and George A Bekey. Collective localization: A distributed kalman filter approach to localization of groups of mobile robots. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 3, pages 2958–2965. IEEE, 2000. 22

Stergios I Roumeliotis, Andrew E Johnson, and James F Montgomery. Augmenting inertial navigation with image-based motion estimation. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 4, pages 4326–4333. IEEE, 2002. 20, 144

Eric Royer, Maxime Lhuillier, Michel Dhome, and Jean-Marc Lavest. Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision*, 74(3):237–260, 2007. 32

Renaud Ruskoné. *Extraction automatique du réseau routier par interprétation locale du contexte: application à la production de données cartographiques*. PhD thesis, Université de Marne-la-Vallée, 1996. 1

Milton CP Santos, Lucas V Santana, Alexandre S Brandão, and Mário Sarcinelli-Filho. Uav obstacle avoidance using rgb-d system. In *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*, pages 312–319. IEEE, 2015. 20

Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Fast image-based localization using direct 2d-to-3d matching. In *2011 International Conference on Computer Vision*, pages 667–674. IEEE, 2011. 16

D. Scaramuzza and R. Siegwart. Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles. *IEEE Transactions on Robotics*, 24(5):1015–1026, Oct 2008. ISSN 1552-3098. doi: 10.1109/TRO.2008.2004490. 29, 36, 61

Davide Scaramuzza. *Omnidirectional Camera*, pages 552–560. Springer US, Boston, MA, 2014. ISBN 978-0-387-31439-6. doi: 10.1007/978-0-387-31439-6_488. URL http://dx.doi.org/10.1007/978-0-387-31439-6_488. xii, 28, 29

Davide Scaramuzza and Friedrich Fraundorfer. Visual odometry [tutorial]. *IEEE Robotics & Automation Magazine*, 18(4):80–92, 2011. 18, 23, 105

Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A flexible technique for accurate omnidirectional camera calibration and structure from motion. In *Fourth IEEE International Conference on Computer Vision Systems (ICVS'06)*, pages 45–45. IEEE, 2006. 29

Toni Schenk. Introduction to photogrammetry. *The Ohio State University, Columbus*, 2005. 106

David Schleicher, Luis M Bergasa, Manuel Ocaña, Rafael Barea, and Elena López. Real-time hierarchical gps aided visual slam on urban environments. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 4381–4386. IEEE, 2009. 31

A. Schlichting and C. Brenner. Localization using automotive laser scanners and local pattern matching. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 414–419, June 2014. 35

A Schlichting and C Brenner. Vehicle localization by lidar point correlation improved by change detection. *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 703–710, 2016. 19, 33

Günter Schmitt. Optimization of geodetic networks. *Reviews of Geophysics*, 20(4):877–884, 1982. 148

Johannes Schneider and Wolfgang Förstner. Real-time accurate geo-localization of a mav with omnidirectional visual odometry and gps. In *Workshop at the European Conference on Computer Vision*, pages 271–282. Springer, 2014. 37

Markus Schreiber, Carsten Knoppel, and Uwe Franke. Laneloc: Lane marking based localization using highly accurate maps. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 449–454. IEEE, 2013. 35

Markus Schreiber, Hendrik Königshof, André-Marcel Hellmund, and Christoph Stiller. Vehicle localization with tightly coupled gnss and visual odometry. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 858–863. IEEE, 2016. 31, 37, 144

F Schuster, M Wörner, CG Keller, M Haueis, and C Curio. Robust localization based on radar signal clustering. In *Intelligent Vehicles Symposium (IV), 2016 IEEE*, pages 839–844. IEEE, 2016. 19

Gerald Schweighofer and Axel Pinz. Globally optimal o (n) solution to the pnp problem for general camera models. In *BMVC*, pages 1–10, 2008. 64

S. Se, D. Lowe, and J. Little. Vision-based mobile robot localization and mapping using scale-invariant features. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 2, pages 2051–2058 vol.2, 2001. doi: 10.1109/ROBOT.2001.932909. 45

Stephen Se, David Lowe, and Jim Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *The international Journal of robotics Research*, 21(8):735–758, 2002. 18

Yung-Ho Seo, Sang-Hoon Kim, Kyoung-Soo Doo, and Jong-Soo Choi. Optimal keyframe selection algorithm for three-dimensional reconstruction in uncalibrated multiple images. *Optical Engineering*, 47(5):053201–053201, 2008. 52

Michael Shaw, Kanwaljit Sandhoo, and David Turner. Modernization of the global positioning system. Technical report, DTIC Document, 2000. 13

Jianbo Shi and Carlo Tomasi. Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*, pages 593–600. IEEE, 1994. 18, 87

Yun Shi, Shunping Ji, Zhongchao Shi, Yulin Duan, and Ryosuke Shibasaki. Gps-supported visual slam with a rigorous sensor model for a panoramic camera in outdoor environments. *Sensors*, 13(1):119–136, 2012. 30, 31, 105

Chanop Silpa-Anan, Richard Hartley, et al. Visual localization and loop-back detection with a high resolution omnidirectional camera. In *Workshop on Omnidirectional Vision*. Citeseer, 2005. 29, 61

Robert Sim, Pantelis Elinas, Matt Griffin, James J Little, et al. Vision-based slam using the rao-blackwellised particle filter. In *IJCAI Workshop on Reasoning with Uncertainty in Robotics*, volume 14, pages 9–16, 2005. 18

Sayanan Sivaraman and Mohan Manubhai Trivedi. Integrated lane and vehicle detection, localization, and tracking: A synergistic approach. *IEEE Transactions on Intelligent Transportation Systems*, 14(2):906–917, 2013. 34

Gregory G Slabaugh. Computing euler angles from a rotation matrix. *Retrieved on August*, 6 (2000):39–63, 1999. 44

Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM transactions on graphics (TOG)*, volume 25, pages 835–846. ACM, 2006. 26, 48, 76

B Soheilian and M Brédif. Multi-view 3d circular target reconstruction with uncertainty analysis. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2 (3):143, 2014. 109, 116

B. Soheilian, X. Qu, and M. Brédif. Landmark based localization: Lba refinement using mcmc-optimized projections of rjmcmc-extracted road marks. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 940–947, June 2016. doi: 10.1109/IVS.2016.7535501. 119, 120

Bahman Soheilian, Nicolas Paparoditis, and Didier Boldo. 3d road marking reconstruction from street-level calibrated stereo pairs. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65 (4):347–359, 2010. 5, 35, 109

Bahman Soheilian, Nicolas Paparoditis, and Bruno Vallet. Detection and 3d reconstruction of traffic signs from multiple view color images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 77:1–20, 2013a. xi, 5, 6, 35, 108, 109, 118, 129

Bahman Soheilian, Olivier Tournaire, Nicolas Paparoditis, Bruno Vallet, and Jean-Pierre Papelard. Generation of an integrated 3d city model with visual landmarks for autonomous navigation in dense urban areas. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 304–309. IEEE, 2013b. xi, 7, 8, 146

Frank Steinbrücker, Jürgen Sturm, and Daniel Cremers. Real-time visual odometry from dense rgb-d images. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 719–722. IEEE, 2011. 19

Frank Steinbrucker, Christian Kerl, and Daniel Cremers. Large-scale multi-resolution surface reconstruction from rgb-d sequences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3264–3271, 2013. 20

Hauke Strasdat, JMM Montiel, and Andrew J Davison. Real-time monocular slam: Why filter? In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 2657–2664. IEEE, 2010a. xii, 24, 27, 38, 52

Hauke Strasdat, JMM Montiel, and Andrew J Davison. Scale drift-aware large scale monocular slam. In *Robotics: Science and Systems*, volume 2, page 5, 2010b. 28

Jae Kyu Suhr and Ho Gi Jung. Fast symbolic road marking and stop-line detection for vehicle localization. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 186–191. IEEE, 2015. 34

Ivan E Sutherland. Three-dimensional data input by tablet. *Proceedings of the IEEE*, 62(4): 453–461, 1974. 50

Daniel Svensson and Joakim Sörstedt. Ego lane estimation using vehicle observations and map information. In *Intelligent Vehicles Symposium (IV), 2016 IEEE*, pages 909–914. IEEE, 2016. 22

Juan D Tardós, José Neira, Paul M Newman, and John J Leonard. Robust mapping and localization in indoor environments using sonar data. *The International Journal of Robotics Research*, 21(4):311–330, 2002. 19

Sarah Tariq and Frank Dellaert. A multi-camera 6-dof pose tracker. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 296–297. IEEE Computer Society, 2004. 64

Timothy B Terriberry, Lindley M French, and John Helmsen. Gpu accelerating speeded-up robust features. In *Proc. Int. Symp. on 3D Data Processing, Visualization and Transmission (3DPVT)*, pages 355–362. Citeseer, 2008. 45

Thorsten Thormählen, Hellward Broszio, and Axel Weissenfeld. Keyframe selection for camera motion and structure estimation from multiple views. In *European Conference on Computer Vision*, pages 523–535. Springer, 2004. 26, 52

Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005. 11, 12, 16, 24, 25

David Törnqvist, Thomas B Schön, Rickard Karlsson, and Fredrik Gustafsson. Particle filter slam with high dimensional vehicle model. *Journal of Intelligent and Robotic Systems*, 55(4-5): 249–266, 2009. 25

Philip HS Torr, Andrew W Fitzgibbon, and Andrew Zisserman. The problem of degeneracy in structure and motion recovery from uncalibrated image sequences. *International Journal of Computer Vision*, 32(1):27–44, 1999. 52

O Tournaire and N Paparoditis. A geometric stochastic approach based on marked point processes for road mark detection from high resolution aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(6):621–631, 2009. 109

O. Tournaire, B. Soheilian, and N. Paparoditis. Towards a sub-decimetric georeferencing of ground-based mobile mapping systems in urban areas: matching ground-based and aerial-based imagery using roadmarks. In *Proc. of the ISPRS Commission I Symposium*, volume Part A, Marne-la-Vall ee, France, jul 2006a. Interne. 34

O Tournaire, B Soheilian, and N Paparoditis. Towards a sub-decimetric georeferencing of groundbased mobile mapping systems in urban areas: Matching ground-based and aerial-based imagery using roadmarks. *International Archives of Photogrammetry and Remote Sensing. Proceedings..., Paris*, 2006b. 2

Olivier Tournaire, Nicolas Paparoditis, Franck Jung, and Bernard Cervelle. 3d roadmarks reconstruction from multiple calibrated aerial images. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(3):73–78, 2006c. 109

Guillaume Trehard, Evangeline Pollard, Benazouz Bradai, and Fawzi Nashashibi. On line mapping and global positioning for autonomous driving in urban environment based on evidential slam. In *Intelligent Vehicles Symposium-IV 2015*, 2015. 31

Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *Vision algorithms: theory and practice*, pages 298–372. Springer, 2000. 26, 54, 56, 57

Stephen Tully, George Kantor, and Howie Choset. Leap-frog path design for multi-robot cooperative localization. In *Field and service robotics*, pages 307–317. Springer, 2010. 22

Iwan Ulrich and Illah Nourbakhsh. Appearance-based place recognition for topological localization. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 2, pages 1023–1029. Ieee, 2000. 15

Christoffer Valgren and Achim J Lilienthal. Sift, surf and seasons: Long-term outdoor localization using local features. In *EMCR*, 2007. 45

Bruno Vallet and Jean-Pierre Papelard. Road orthophoto/dtm generation from mobile laser scanning. *International Annals of Photogrammetry Remote Sensing and Spatial Information Sciences*, 3:W5, 2015. 6

Bruno Vallet, Wen Xiao, and Mathieu Brédif. Extracting mobile objects in images using a velodyne lidar point cloud. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3):247, 2015. xi, 7

A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008. 74

George Vosselman. Design of an indoor mapping system using three 2d laser scanners and 6 dof slam. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3):173, 2014. 19

Xue Wan, Jianguo Liu, Hongshi Yan, and Gareth LK Morgan. Illumination-invariant image matching for autonomous uav localisation based on optical sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 119:198–213, 2016. 15

Jingchuan Wang and Weidong Chen. A novel localization system based on infrared vision for outdoor mobile robot. In *Life System Modeling and Intelligent Computing*, pages 33–41. Springer, 2010. 18

Ling Wang, JianWei Wan, YunHui Liu, and JinXin Shao. Cooperative localization method for multi-robot based on pf-ekf. *Science in China Series F: Information Sciences*, 51(8):1125–1137, 2008. 22

Lijun Wei, Cindy Cappelle, Yassine Ruichek, and Frédérick Zann. Gps and stereovision-based visual odometry: application to urban scene mapping and intelligent vehicle localization. *International Journal of Vehicular Technology*, 2011, 2011. 31, 36

Lijun Wei, Bahman Soheilian, and Valérie Gouet-Brunet. Augmenting vehicle localization accuracy with cameras and 3d road infrastructure database. In *Computer Vision-ECCV 2014 Workshops*, pages 194–208. Springer, 2014. 35, 79

MARTIN Weinmann, CLÉMENT MALLET, and MATHIEU BRÉDIF. Segmentation and localization of individual trees from mms point cloud data acquired in urban areas. *Tagungsband der Dreiländertagung der DGPF, der OVG und der SGPF, Bern, Switzerland*, 25:351–360, 2016. 7

David Wilkie, Jason Sewall, and Ming C Lin. Transforming gis data into functional road models for large-scale traffic simulation. *IEEE Transactions on Visualization and Computer Graphics*, 18(6):890–901, 2012. 1

William J. Wolfe, Donald Mathis, Cheryl Weber Sklair, and Michael Magee. The perspective view of three points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(1): 66–73, 1991. 51

David Wong, Daisuke Deguchi, Ichiro Ide, and Hiroshi Murase. Vision-based vehicle localization using a visual street map with embedded surf scale. In *Workshop at the European Conference on Computer Vision*, pages 167–179. Springer, 2014. xii, 15, 16

RVC Wong, KP Schwarz, and ME Cannon. High-accuracy kinematic positioning by gps-ins. *Navigation*, 35(2):275–287, 1988. 21

Changchang Wu. Siftgpu: A gpu implementation of scale invariant feature transform (sift). 2007. 45

Changchang Wu. Towards linear-time incremental structure from motion. In *3D Vision-3DV 2013, 2013 International Conference on*, pages 127–134. IEEE, 2013. 26

Jianxin Wu, Henrik I Christensen, and James M Rehg. Visual place categorization: Problem, dataset, and algorithm. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4763–4770. IEEE, 2009. 15

Tao Wu and Ananth Ranganathan. Vehicle localization using road markings. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 1185–1190. IEEE, 2013. 35

Wen Xiao, Bruno Vallet, Mathieu Brédif, and Nicolas Paparoditis. Street environment change detection from mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 107:38–49, 2015. 7

Wen Xiao, Bruno Vallet, Konrad Schindler, and Nicolas Paparoditis. Street-side vehicle detection, classification and change detection using mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 114:166–178, 2016. xi, 7

Jianping Xie, Fawzi Nashashibi, Michel Parent, and Olivier Garcia-Favrot. A real-time robust slam for large-scale outdoor environments. In *17th ITS world congress (ITSwc'2010)*, page S_EU00913, 2010. 19

Lu Yang. A simplified algorithm for solution classification of the perspective-three-point problem. *MM Research Preprints*, 17:135–145, 1998. 51

Y. Yang, Y. Song, F. Zhai, Z. Fan, Y. Meng, and J. Wang. A high-precision localization algorithm by improved sift key-points. In *Image and Signal Processing, 2009. CISP '09. 2nd International Congress on*, pages 1–6, Oct 2009. doi: 10.1109/CISP.2009.5303161. 45

Lifan Yao, Hao Feng, Yiqun Zhu, Zhiguo Jiang, Danpei Zhao, and Wenquan Feng. An architecture of optimised sift feature detection for an fpga implementation of an image matcher. In *Field-Programmable Technology, 2009. FPT 2009. International Conference on*, pages 30–37. IEEE, 2009. 45

Anthony G-O Yeh. Urban planning and gis. *Geographical Information Systems*, 2:877–888, 1999. 1

Keisuke Yoneda, Hossein Tehrani, Tomomi Ogawa, Naohisa Hukuyama, and Seiichi Mita. Lidar scan feature for localization with highly precise 3-d map. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 1345–1350. IEEE, 2014. 33

Yongtao Yu, Jonathan Li, Haiyan Guan, Cheng Wang, and Jun Yu. Automated detection of road manhole and sewer well covers from mobile lidar point clouds. *IEEE Geoscience and Remote Sensing Letters*, 11(9):1549–1553, 2014a. 109

Yufeng Yu, Huijing Zhao, Franck Davoine, Jinshi Cui, and Hongbin Zha. Monocular visual localization using road structural features. In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pages 693–699. IEEE, 2014b. 35

JS-C Yuan. A general photogrammetric method for determining object position and orientation. *ieee Transactions on Robotics and Automation*, 5(2):129–142, 1989. 51

Ji Zhang and Sanjiv Singh. Loam: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems Conference (RSS)*, pages 109–111, 2014. 18, 19

Ji Zhang and Sanjiv Singh. Visual-lidar odometry and mapping: Low-drift, robust, and fast. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2174–2181. IEEE, 2015. 20

Zhengyou Zhang and Ying Shan. Incremental motion estimation through local bundle adjustment. *Technical rep. MSR-TR-01-54, Microsoft Research, Redmond, WA*, 2001. 27, 52

Zichao Zhang, H. Rebecq, C. Forster, and D. Scaramuzza. Benefit of large field-of-view cameras for visual odometry. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 801–808, May 2016. doi: 10.1109/ICRA.2016.7487210. 29, 35, 61

Sheng Zhao and Jay A Farrell. 2d lidar aided ins for vehicle positioning in urban environments. In *Control Applications (CCA), 2013 IEEE International Conference on*, pages 376–381. IEEE, 2013. 20

Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1529–1537, 2015. 147

Dingfu Zhou, Yuchao Dai, and Hongdong Li. Reliable scale estimation and correction for monocular visual odometry. In *Intelligent Vehicles Symposium (IV), 2016 IEEE*, pages 490–495. IEEE, 2016. 49

Barbara Zitova and Jan Flusser. Image registration methods: a survey. *Image and vision computing*, 21(11):977–1000, 2003. 85

Landmark based localization: detection, matching and update of landmark with uncertainty analysis

This thesis proposed to use one or more cameras on a vehicle as a georeferencing system. The vehicle's trajectory can be estimated using visual odometry techniques. To limit the drift of the trajectory due to the accumulation of errors, we propose a registration on a set of visual landmarks that are precisely georeferenced. These landmarks are reconstructed using the reference data generated by precise and expensive mapping systems. Natural road features such as road markings and traffic signs were chosen as visual landmarks.

A local bundle adjustment algorithm has been adapted to estimate the pose of the vehicle using a sequence of images acquired by one or more embedded cameras. A rigorous approach that takes into account the uncertainties enables to tune automatically the weights of every constraint in the equation system of the adjustment and to estimate the uncertainties of the parameters. They are used in a propagation based matching algorithm that accelerates the process of tracking the interest points between the images and eliminate a large number of false matches. This significantly reduces the drift of the visual odometry by reducing the sources of errors. The remaining part of the drift is removed using georeferenced visual landmarks. The process of matching the image sequence with the landmarks is guided by the uncertainty of the poses. It adds a set of absolute constraints in the equation system of bundle adjustment. The drift is drastically reduced. Each step of the algorithm is evaluated on real image sequences with ground truths.

Keywords : *Localization, Landmark, Local Bundle Adjustment (LBA), Uncertainty analysis, Ground Control Points (GCPs), Error propagation, Multi-camera, Traffic signs, Road markings, Feature extraction, Matching and tracking, Motion model, Tie points, Pose estimation.*