TÉCNICO LISBOA

UNIVERSIDADE DE LISBOA

INSTITUTO SUPERIOR TÉCNICO

A Probabilistic Approach for Stereo Visual Egomotion

Hugo Miguel Gomes da Silva

Supervisor: Doctor Alexandre José Malheiro Bernardino
Co-Supervisor: Doctor Eduardo Alexandre Pereira da Silva

Thesis approved in public session to obtain the PhD Degree in

Electrical and Computer Engineering

Jury final Classification: Pass with Merit

Jury

Chairperson:	Chairman of the IST Scientific Board
Members of the Committee:	Doctor José Alberto Rosado dos Santos Victor
	Doctor Luis Mejias Alvarez
	Doctor Pedro Manuel Urbano de Almeida Lima
	Doctor Jorge Nuno de Almeida e Sousa Almada Lobo
	Doctor Alexandre José Malheiro Bernardino
	Doctor Eduardo Alexandre Pereira da Silva

2014

TÉCNICO LISBOA

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

A Probabilistic Approach for Stereo Visual Egomotion

Hugo Miguel Gomes da Silva

Supervisor: Doctor Alexandre José Malheiro Bernardino
Co-Supervisor: Doctor Eduardo Alexandre Pereira da Silva

Thesis approved in public session to obtain the PhD Degree in
Electrical and Computer Engineering

Jury final classification: Pass with Merit

Jury

Chairperson: Charmain of the IST Scientific Board

Members of the Committee:

Doctor José Alberto Rosado dos Santos Victor, Professor Catedrático do Instituto Superior Técnico, da Universidade de Lisboa
Doctor Luis Mejias Alvarez, Senior Lecturer, Science and Engineering Faculty, Queensland University of Technology, Brisbane, Australia
Doctor Pedro Manuel Urbano de Almeida Lima, Professor Associado (com Agregação) do Instituto Superior Técnico, da Universidade de Lisboa
Doctor Jorge Nuno de Almeida e Sousa Almada Lobo, Professor Auxiliar da Faculdade de Ciências e Tecnologia, da Universidade de Coimbra
Doctor Alexandre José Malheiro Bernardino, Professor Auxiliar do Instituto Superior Técnico, da Universidade de Lisboa
Doctor Eduardo Alexandre Pereira da Silva, Professor Adjunto do Instituto Superior de Engenharia do Porto - Instituto Politécnico do Porto

Funding Institutions - Fundação para a Ciência e Tecnologia
Grant - SFRH / BD / 47468 / 2008

2014

This Thesis is dedicated to my late grandmother Benilde
She will always be in my heart and mind.
One is done the other is for life

Abstract

The development of vision based navigation systems for mobile robotics applications in outdoor scenarios is a very challenging problem due to known outdoor visibility concerns such as changes in contrast and illumination, image blur, pixel noise, lack of image texture, not sufficient image overlap, and other sources of errors that lead to ambiguity of the observed image data. From our point of view probabilistic methods are more robust to these effects, and allow a easier integration with other very well known navigation sensors (IMU). Until now probabilistic methods haven't yet been fully explored due to their high computational cost, but with today's efforts and resources put into the development of parallel hardware (e.g GPGPU), probabilistic techniques due to their parallel nature can be used in real-time applications.

The work conducted in this thesis, focus on the research and development of reliable visual measurements techniques to support visual navigation systems in mobile robotics applications, with special emphasis in estimating robot motion in challenging scenarios where current visual state-of-the-art methods are more prone to failures. When equipped with cameras, robots must determine motion by measuring their displacement relative to static key points in the environment, process which is usually denoted as Visual Odometry(VO). The use of VO methods has been subject of research by the robotics community over the past years. One way of performing VO estimation is by determining instantaneous camera displacement on consecutive frames, a process denoted as visual egomotion estimation, and integrating over time the obtained rotational and translational velocities. In monocular egomotion estimation there is translation scale ambiguity, i.e. in the absence of other sources of information, only the translational velocity direction is possible to measure reliably. Therefore, whenever possible two cameras are used to have a full velocity estimation, usually denoted as stereo egomotion estimation.

In this thesis, we develop a novel fully probabilistic method for estimating stereo egomotion, denoted as Probabilistic Stereo Egomotion Transform (PSET) capable of computing 6-DOF motion parameters solely based on probabilistic correspondence approaches, and without the need to track or commit key point matches between two consecutive frames. The use of probabilistic correspondence methods allows to maintain several match hypothesis for each point, which is an advantage when ambiguous matches occur (which is the rule in image feature correspondences problems), because no commitment is made before analyzing all image information. Another ad-

vantage is that a full probabilistic distribution of motion provides a better sensor fusion with other sensors, e.g. inertial. Experimental results in simulated and real outdoor scenarios are presented. Comparison with other current-state-of-the-art visual motion estimation methods is also provided.

Overall, we show that probabilistic approaches provide better average precision than their deterministic counterparts. The price to pay is a bigger computational cost that can, nevertheless, be mitigated with multi-core implementations due to the inherent parallelism of probabilistic computations.

Keywords

Egomotion, Visual Odometry, Stereo Vision, Fundamental Matrix, Epipolar Geometry, Correlation, Extended Kalman Filter, Computer Vision, Robotics, Graphic Processing Unit

Resumo

O desenvolvimento de sistemas de navegação visuais para aplicações de robótica móvel constitui um desafio complexo, essencialmente devido à problemática relacionada com as condições de visibilidade quando os robôs operam em ambientes exteriores. Os métodos probabilísticos são, do nosso ponto de vista os mais robustos aos efeitos da dinâmica das condições de visibilidade, e podem mais facilmente integrar informação proveniente de outros tipos de sensores de navegação como por exemplo: sensores inerciais. A utilização de métodos probabilísticos tem sido limitada, devido ao fato de necessitarem de elevados recursos computacionais. No entanto, com o aparecimento e desenvolvimento dos sistemas de computação paralela, os métodos probabilísticos dadas as suas caraterísticas intrínsecas de natureza paralela são passíveis de serem utilizados em aplicações de tempo real.

O trabalho efetuado nesta dissertação, aborda o desenvolvimento e utilização de técnicas de perceção visual que sirvam de suporte a sistemas de navegação visuais para aplicações robóticas movéis. A dissertação dedica especial atenção à estimação do movimento do robô em cenários de aplicação dinâmicos, onde os métodos clássicos de estimação de movimento não são imunes a falhas. Os robôs equipados com meios visuais, podem determinar o seu movimento através da mediçao do seu deslocamento relativo a pontos fixos no ambiente. O processo é usualmente denominado de estimação da odometria visual. A comunidade robótica têm vindo a dedicar tempo ao estudo e desenvolvimento de métodos de estimação da odometria visual. Uma das formas de estimar a odometria visual é determinar o deslocamento instantâneo das camaras entre instantes de tempo consecutivos, processo normalmente denominado como estimação de movimento por meios visuais, e seguidamente integrar no tempo as velocidades de rotação e translação. Na estimação do movimento utilizando visão monocular, existe o problema da ambiguidade na determinação do fator de escala. No caso de não existirem outras fontes de informação, apenas a direção da velocidade de translação pode ser determinada com fiabilidade. Portanto, sempre que possível utilizam-se duas camaras para determinar as velocidades do robô, usualmente denominada de estimação de movimento por visão binocular.

Nesta dissertação, desenvolveu-se um novo método de estimação de movimento por visão binocular recorrendo apenas a métodos probabilísticos. O método denominado de PSET é capaz de calcular todos os seis graus de liberdade do movimento, baseado apenas em probabilidades

de correspondência e sem necessidade de seguir ou se comprometer com a identificação de pontos entre imagens consecutivas. A utilização de correspondências probabilísticas permite manter várias hipóteses de correspondência para cada ponto em simultâneo, o que é uma vantagem quando existe ambiguidade (o que é uma regra em problemas de correspondência de pontos entre imagens), porque não se estabelece uma relação entre pontos enquanto toda a informação da imagem não for analisada. Outra vantagem é que a utilização de uma distribuição probabilística do movimento proporciona uma melhor fusão sensorial com outros sensores. Nesta dissertação são apresentados resultados experimentais em ambiente de simulação e em cenários reais em ambiente exterior. Também foram efetuados testes comparativos com outros métodos de estimação de movimento.

No geral, é demonstrado que a nossa abordagem probabilística em linha com as demais, têm melhor exatidão que as abordagens determinísticas na estimação do movimento, com o inconveniente de consumir mais recursos computacionais, o que pode ser minimizado utilizando uma implementação multi-core aproveitando o paralelismo inerente às computações probabilísticas.

Palavras Chave

Estimação de movimento, Odometria visual, Visão binocular, Matriz fundamental, Geometria Epipolar, Correlação, Filtro de Kalman Extendido, Visão por Computador, Robótica, Unidade de Processamento Gráfico.

Acknowledgments

This doctoral thesis would not been possible without the support of all my family, friends and colleagues, for all my immense gratitude.

First, I would like to express my gratitude to my supervisor Prof. Alexandre Bernardino for his knowledge, guidance and insights throughout the completion of this doctoral thesis. Without our fruitful discussions the work would never be completed. To my co-supervisor Prof. Eduardo Silva, I would like to express my immense gratitude for the work over the last 10 years, it has been a long road. I would never reach this state of my academic life without his guidance and support. To him and his friendship, I will forever remain indebted.

I would like to extend my gratitude to all my friends and colleagues from the Autonomous System Laboratory, specially those belonging to the Advanced Field Robotics: André Dias , Carlos Almeida, Hugo Ferreira, Luís Lima, Nuno Dias, Prof. José Almeida and Prof. Alfredo Martins. A special word to Prof. João Paulo Baptista and Betina Neves for their help in the creation of a proper work environment, as well as, their nice gestures throughout this years.

To my parents, there are no words to describe their endless efforts throughout my youth until now. For their loving tender and inspiration my sincere gratitude. This doctoral thesis is also theirs.

To Xana, who has been my strength throughout all this time, I am grateful for having you by my side.

Contents

Abstract i

Resumo iii

Acknowledgments v

List of Figures xi

List of Tables xv

1 Introduction 1

1.1 Motivation . . . 1

1.2 Objectives . . . 3

1.3 Contributions . . . 4

1.4 Thesis Overview and Organization . . . 5

2 Related Work 9

2.1 Introduction . . . 9

2.2 Image Information . . . 10

2.2.1 Dense Methods . . . 11

2.2.2 Point Correspondence . . . 12

2.2.3 Summary on Image Information . . . 14

2.3 Egomotion Estimation . . . 14

2.3.1 Visual Odometry . . . 15

2.3.2 Visual SLAM . . . 19

2.3.3 Inertial and Visual Sensing . . . 20

2.4 Parallel Programming . . . 22

2.5 Summary on Egomotion Estimation . . . 23

3 Fundamentals 25

3.1 Introduction . . . 25

3.2 Image Formation . . . 25

3.2.1 Pinhole Camera Model . . . 25

3.3 Geometry of Multiple Views . . . 27
3.3.1 Epipolar Geometry . . . 27
3.3.2 Planar Homography . . . 30
3.3.3 Stereo Vision . . . 31
3.4 Egomotion Recovery . . . 32
3.5 Visual Odometry Estimation . . . 33

4 Sparse and Dense Stereo Visual Egomotion 37
4.1 Introduction . . . 37
4.2 A mixed approach to stereo visual egomotion: combining sparse and dense methods 38
4.2.1 Probabilistic Correspondence . . . 40
4.2.2 Probabilistic Egomotion Estimation . . . 42
4.2.3 Scale Estimation . . . 43
4.2.3.A Procrustes Analysis and Scale Factor Recovery . . . 44
4.2.3.B Bucketing . . . 44
4.2.4 Linear and Angular Velocity Estimation . . . 45
4.2.5 Kalman Filter . . . 46
4.3 Results . . . 46
4.3.1 Computational Implementation . . . 47
4.3.2 6DP-raw-Harris vs 5-point . . . 47
4.3.3 6DP-raw-Harris vs 6DP-raw-SIFT . . . 48
4.3.4 6DP-KF vs LIBVISO . . . 50
4.4 Summary . . . 51
4.5 Related Publications . . . 52

5 Probabilistic Stereo Egomotion Transform 53
5.1 Introduction . . . 53
5.2 Probabilistic Stereo Egomotion Estimation . . . 54
5.2.1 The Geometry of Stereo Egomotion . . . 55
5.2.1.A Degenerate Cases . . . 58
5.2.2 Translational Scale Estimation . . . 58
5.2.3 PSET Accumulator . . . 59
5.2.4 Dealing with calibration errors . . . 61
5.2.5 Synthetic Image Sequences . . . 63
5.2.5.A Computational Implementation . . . 63
5.2.5.B Motion Setup . . . 64
5.2.5.C Qualitative Analysis . . . 64
5.2.5.D Motion Quantitative Analysis . . . 66

5.2.6 Real Image Sequences 67
5.3 Summary 70
5.4 Related Publications 70

6 Conclusions and Future Work **71**
6.1 Conclusions 71
6.2 Future Work 72

A Appendix 1 **87**
A.1 Zero Normalized Cross Correlation 87

List of Figures

1.1 INESC-TEC Mobile Robotics platforms on land, see and air application scenarios. All robotic platforms are equipped with one or more visual sensors to perform visual navigation, or other complementary tasks. 2

1.2 Flow Diagram representing the thesis chapter organization, where each box represents a different chapter or appendix where, related topics display the same color. 6

2.1 Thesis related work, egomotion estimation can be performed using a monocular or stereo camera configuration setup. Motion information from image measurements can be obtained using dense methods (optical flow) or point correspondence methods. Egomotion applications in a computer vision and mobile robotics context include but are not limited to: Structure from Motion, detect moving independent objects in the image, Visual Odometry, Inertial and Visual Sensing and also Simultaneous Localization and Mapping. 10

2.2 Brightness constancy principle of an 2D image pixel representation over a short period of time. The image pattern at position (x, y, t) is the same of position $(x + u\delta t, y + u\delta t, t + \delta t)$ 11

2.3 Image motion measurements methods chronology, in blue (optical flow methods), in red (key point methods). 14

2.4 Opportunity Mars Exploration Rover 15

3.1 Pinhole Camera Model 26

3.2 Epipolar Geometry showing two camera reference frames $\{L\}$ and $\{R\}$, that are related via pose transformation C_R^L. The world point **P** and the two cameras centers form the epipolar plane, and the intersection of the epipolar plane with the image plane forms the epipolar lines 28

3.3 The four possible solutions for obtaining left and right camera pose from E. Only in solution (1) the point is in front of both cameras (l,r). 29

3.4 The homography geometry consists on having two cameras with coordinate frames $\{L\}, \{R\}$. The 3D world point **P** belongs to a plane with surface normal Π. The homography H allows to map point $\mathbf{p_L}$ to $\mathbf{p_R}$ 30

3.5 Depth estimation uncertainty over the epipolar line. In the left figure the epipolar geometry shows the point depth variation of points **P**,**P'** along the epipolar line in the second image. In the right figure is shown the triangulation procedure to estimate point **P** 3D camera reference frame coordinates 31

3.6 Model that formalizes the displacement in time and visual space of image sequences (k, k+1), according to Longuet-Higgins and Pradzny model 32

4.1 Example of Acquisition Setup for a vehicle-like robot, with the use of stereo cameras for providing estimates of vehicle angular and linear velocities. 38

4.2 6DP architecture 39

4.3 Image feature point correspondence for ZNCC matching, with window size N_W between points $\mathbf{x}$ and $\mathbf{x}'$ represented in red and green respectively 41

4.4 Likelihood of a point $\mathbf{x}$ in image I_k^L with all matching candidates $\mathbf{x}'$ in I_{k+1}^L, for the case of Fig. 4.3. Points with high likelihood are represented in lighter colour 41

4.5 Image feature point marked in colour green in image I_k^L lies in the epipolar line (blue) estimated between I_k to I_{k+1}. The point with higher correlation score, marked in red in image I_{k+1}^L is chosen as the matching feature point. 43

4.6 Feature detection bucketing technique used to avoid biased samples in the RANSAC method stage. The image is divided in buckets where feature points are assigned to and pulled according to the bucket probability. 45

4.7 Comparison of angular velocity estimation results between IMS/GPU (red), raw 6DP measurements (blue) and a native 5-point implementation (black). The obtained 6DP raw measurements are similar to the data estimated by the IMU/GPS, contrary to the 5-point implementation that has some periods of large errors (e.g. the regions indicated with arrows in the plots). 48

4.8 Comparison of linear velocity estimation results, where the 5-point implementation (black) exhibits a closer match to the IMU/GPS information (red). The 6DP method (blue) displays some highlighted outliers due to the use of the Harris feature detection matching in the sparse method stage. 48

4.9 Translation scale factor comparison between 5-point and 6DP-raw-Harris, where the 5-point method exhibits a more constant behavior for the translation scale factor estimation. 49

4.10 Number of Features at different steps of 6DP-raw-Harris and 5-point. SIFT features display a more robust matching behavior between images. Contrary to Harris Corners, most of the SIFTS are not eliminated in the RANSAC stage. 49

4.11 Results for angular velocities estimation between IMU/GPS information (red), raw 6DP measurements 6DP-raw-SIFTS (blue), filtered 6DP measurements 6DP-KF (black), and 6D Visual Odometry Library LIBVISO (green). Even though all exhibit similar behaviors the filtered implementation 6DP-KF is the one which is closer to the "ground truth" IMU/GPS measurements (see also Table 1). 50
4.12 Results for linear velocities estimation, where the LIBVISO implementation and 6DP-KF display similar performance when compared to IMU/GPS performance. . 51

5.1 ZNCC matching used to compute the PSET transform 54
5.2 Example of probabilistic correspondence ($\rho_s(r)$, $\rho_s(q)$, $\rho_s(q)$) obtained by ZNCC matching for a given point **s** for an image triplet (I_k^R,I_{k+1}^L,I_{k+1}^R) 55
5.3 Stereo Egomotion Geometry 56
5.4 Point correspondence hypotheses along the epipolar lines 59
5.5 Probabilistic correspondence $\rho_s(r)$ for a point **s** along the epipolar line E_{sr}. In the left hand side figure, it is shown all known hypotheses (red), the local maximum probabilistic correspondences (peaks) of $\rho_s(r)$ (blue), and the global maximum of $\rho_s(r)$ (green). On the right hand side figure, we see sample point **s** in I_k^L and the local maximum (peaks) probabilistic correspondences represented in I_k^R 60
5.6 PSET H_j 2D table accumulator 61
5.7 Epipolar lines on I_k^R computed by the different fitting methods i.e no-interpolation, parabolic fitting and gaussian fitting 62
5.8 Image used in the synthetic image sequence to perform egomotion estimation . . 63
5.9 Generated Motion Trajectory computed by the VISLAB simulator to evaluate PSET egomotion accuracy while undergoing a pure translational movement in all 3 axes. 64
5.10 Sequence 1 translational motion in the x axis corresponding to a stereo camera pair movement to the right 65
5.11 Sequence 2 translational motion in the x axis in the opposite direction at double velocity 65
5.12 Sequence 3 translational movement in the x axis and y axis, that corresponds to a left-right downwards diagonal movement 65
5.13 Sequence 4 translational movement in the y axis and z axis, that corresponds to a frontal upward movement 65
5.14 Generated Motion Trajectory used in the sinthetic image translational motion experiment using PSET (blue), LIBVISO (red) and ground-truth (black) information . 66
5.15 Zoom Top view of the global translational motion trajectory using PSET, LIBVISO and ground-truth information 66
5.16 Error Statistics for ||V|| linear velocities obtained by PSET and LIBVISO egomotion estimation 67

5.17 Error Statistics for the linear velocities estimation obtained by PSET and LIBVISO in all 3 axes (V_x, V_y, V_z) 67
5.18 Results for the angular velocities estimation of 300 frames: ground truth(GPS-IMU information), filtered PSET measurements (PST-EKF) and 6D Visual Odometry Library (LIBVISO). Even though all exhibit similar behaviors the filtered implementation PSET-EKF is the one which is closer to GT(GPS-IMU)(see also table 1). . . . 68
5.19 Estimated linear velocities of 300 frames estimation. The PSET transform exhibits a better performance in V_y compared to LIBVISO, and the opposite occurs in V_z estimation (see Table 1). However in overall linear velocities estimation the PSET is about 50 % better, see Table 1 69
5.20 Zoom view of the first 20 frames results for linear velocities estimation, using PSET, LIBVISO and Inertial Measurement Unit information 69

A.1 ZNCC reference template matching 88
A.2 Integral Window calculation 88

List of Tables

2.1 Visual Odometry methods comparison 24

4.1 Standard Mean Squared Error between IMU and Visual Odometry (LIBVISO and 6DP-KF). The displayed results show a significant improvement of the 6DP-KF method performance specially in the angular velocities estimation case. 51

5.1 Synthetic image sequences ground truth information 64
5.2 Comparison of the standard mean squared error between ground truth information and both stereo egomotion estimation methods (PSET and LIBVISO). 67
5.3 Comparison of the standard mean squared error between IMU and stereo egomotion estimation methods(LIBVISO, 6DP, and PSET). The linear velocities results (V) are presented in (m/s), and the angular velocities results (W) are presented in $(degrees/s)$ 70

List of Acronyms

AO *Absolute Orientation*

ASIC *Application Specific Integrated Circuit*

BA *Bundle Adjustment*

BRISK *Binary Robust Invariant Scalable Keypoints*

BRIEF *Binary Robust Independent Elementary Features*

CPU *Central Processing Unit*

DOF *Degrees of Freedom*

DLT *Direct Linear Transform*

DSP *Digital Signal Processor*

EKF *Extended Kalman Filter*

FPGA *Field Programmable Gate Array*

GPGPU *General Purpose Graphic Processing Unit*

GPU *Graphic Processing Unit*

GPP *General Purpose Processor*

GPS *Global Positioning System*

IMU *Inertial Measurement Unit*

ICP *Iterative Closest Point*

JPL *Jet Propulsion Laboratory*

KF *Kalman Filter*

KNN *K-Nearest Neighbors*

KLT *Kanade Lucas Tomasi Tracker*

LIBVISO *Library Visual Odometry*

MER *Mars Exploration Rover*

MLE *Maximum Likelihood Estimation*

MSER *Maximum Stable Extreme Region*

PnP *Perspective number Points*

PSET *Probabilistic Stereo Ego-motion Transform*

RANSAC *Random Sample Consensus*

ROS *Robotic Operating System*

SSD *Sum Square Differences*

SAD *Sum Absolute Differences*

SFM *Structure From Motion*

SIFT *Scale Invariant Feature Transform*

SURF *Speeded Up Robust Feature*

UAV *Unmanned Aerial Vehicle*

VO *Visual Odometry*

VSLAM *Visual Simultaneous Localization and Mapping*

ZNCC *Zero Normalized Cross Correlation*

6DP *6 Degrees-of-freedom Dense Probabilistic*

1

Introduction

"The perception of motion in the visual field, when recognized as a psychological problem instead of something self-evident, is often taken to present the same kind of problem as the perception of color or of form. Movement is thought to be simply one of the characteristics of an object, and the only question is "how do we see it?" Actually, the problem cuts across many of the unanswered questions of psychology, including those concerned with behavior. It involves at least three separable, but closely related problems: How do we see the motion of an object? How do we see the stability of the environment? How do we perceive ourselves as moving in a stable environment?"

– J.J. Gibson (1954) , *Journal of Psychological review*

1.1 Motivation

The perception of motion is a key step in mobile robot navigation tasks. Since the beginning of the robotics era, mobile robot navigation has been considered utmost important, and extensive research has been continuously devoted to solve the robot navigation problem. As Gibson pointed out for the human case, roboticists have also been trying to answer those same questions applied to robots. *How can robots see the motion of an object? How can robots see the stability of the environment? How can robots perceive their own movement in a stable environment?.* The answer to those questions starts by inferring robot self-motion relative to his surrounding environment. In this thesis, we focus on the inference of robot self-motion, from now on denoted as egomotion, based on visual observations of the world environment. Although egomotion can

Figure 1.1: INESC-TEC Mobile Robotics platforms on land, see and air application scenarios. All robotic platforms are equipped with one or more visual sensors to perform visual navigation, or other complementary tasks.

also be estimated without visual information, using other sensor types such as: Inertial Measurement Units(**IMU**), or Global Positioning Systems(**GPS**), the use of visual information plays an important role specially in mobile robots IMU/GPS denied environments, such as: urban crowded areas or underwater mobile robotics applications scenarios. Furthermore, in today's modern mobile robots, visual sensor information usefulness far exceeds the motion estimation problem, and visual sensors are becoming ubiquitous in modern mobile robots, as displayed in Fig.1.1.

The main questions that motivate us, *Why do robots need to compute their egomotion*, and *What advantage does egomotion knowledge brings to robots?*

The main answer is that the egomotion estimation is of key importance to any robot that wishes to navigate and interact with its environment. Many robotic navigation tasks require egomotion knowledge e.g, to find the 3D structure of a scene, denoted as structure from motion **SFM**. In the monocular case, the 3D recovery of a scene is achieved by computing the relative motion between two camera positions on consecutive images, as for the stereo case, the 3D position of the different points may be inferred directly in the same time instant. Other mobile robot navigation task that requires egomotion estimation is Visual Odometry **VO** computation. The egomotion estimates are integrated over time to compute robot pose, from VO estimates the robot can obtain knowledge of the distance and direction traveled. The egomotion knowledge is also of valuable importance to detect independently moving objects. This is an important requirement, that still has not been accomplished even though significant amount of work has been and continues to be done both by academic and industry partners. We strongly believe that for mobile robots to correctly addresses this problem, a robot must have robust visual estimation methods that help to perceive not only static targets, but moving targets as well, making it act as a "moving observer",

capable of distinguish between his self-motion from a target motion, based on image perception. Later on in chapter 2, related work regarding these autonomous navigation applications that require egomotion estimation will be discussed in detail.

Up until now, visual egomotion estimation was approached from the robot standpoint, let us now detail how to compute visual egomotion, and what is visual egomotion precisely.

Visual Egomotion is the process of determining instantaneous camera displacement on consecutive frames. The camera and all the points present in the scene undergo a purely rigid-body motion, whose kinematics can be described by a translation (**t**) and rotation(R) of the camera, with respect to the world. The translation and rotation of the camera are computed based on the displacement that image points undergo between consecutive frames. Usually, this movement is denoted as image motion, and methods to compute image motion are categorized into two standard approaches: sparse key point correspondences or dense approaches. Both approaches have advantages and disadvantages that we will discuss in detail throughout this thesis work.

The process of determining visual egomotion can be accomplished using a monocular or stereo camera setup. However in monocular egomotion estimation there is translation scale ambiguity, i.e. in the absence of other sources of information, only the translation velocity direction is possible to measure reliably. Therefore, whenever possible two cameras are used to have a full velocity estimation, usually denoted as stereo visual egomotion estimation.

1.2 Objectives

Our main objective is to robustly estimate robot visual egomotion using a stereo camera setup. For achieving such purpose, a few factors that influence the desired outcome must be taken into consideration.

Despite all benefits of having vision sensors as source of information for robot egomotion estimation, vision is inherently noisy, specially in mobile robotics outdoor scenarios, due to changes in contrast and illumination, image blur, pixel noise, lack of image texture, not sufficient image overlap, and other sources of errors that lead to ambiguity of the observed image data.

To this end, this thesis proposes the use of novel probabilistic approaches to solve the robot visual egomotion estimation problem. Most approaches to the stereo egomotion estimation problem, rely on non-probabilistic correspondences methods. Common approaches try to detect, match, and track key points between images on adjacent time frames and afterwards use the largest subset of point correspondences that yield a consistent motion. In probabilistic correspondence methods, matches are not fully committed during the initial phases of the algorithm and multiple matching hypotheses are accounted for. Therefore motion is only computed at a latter stage of the method, when most of the visual information was already analyzed, resulting in more accurate estimates. Of course, there is a price to pay, which is higher computational cost, that

can nevertheless be diminished by taking advantage of the inner parallel nature of the probabilistic methods, and their implementation into multi-core processing hardware.

1.3 Contributions

In our work, we contribute to the development of novel visual navigation methods. Specifically, we propose a probabilistic approach for stereo visual egomotion estimation.

- The thesis first contribution, and also our first approach to stereo visual egomotion estimation problem, denoted as **6DP**, combines sparse feature detection and tracking for stereo-based depth estimation, using highly distinctive key points, and a variant of the dense probabilistic egomotion method developed by Domke et al [DA06] to estimate camera motion up to a translational scale factor. Upon obtaining two registered point sets in consecutive time frames, an Absolute Orientation(**AO**) method, defined as an orthogonal Procrustes problem is used to recover yet undetermined motion scale. The velocities obtained by the proposed method are then filtered with a Kalman Filter **KF** approach to reduce sensor noise, and provide frame-to-frame filtered linear and angular velocity estimates. The developed method is compared to other state-of-the-art methods and also with Inertial Measurement Unit information. Results show that our method presents significant improvements in the estimation of angular velocities and a slight improvement in performance for linear velocities. The benefits of using dense probabilistic approaches are validated in a real world scenario with practical significance.

- The second contribution of this thesis, is a novel fully probabilistic stereo visual egomotion estimation method, denoted as Probabilistic Stereo Egomotion Transform (**PSET**). The method is to the best of our knowledge the first completely probabilistic visual stereo egomotion estimation method. It is capable of computing 6-DOF motion parameters solely based on probabilistic correspondence approaches, and without the need to track or commit key point matches between consecutive frames. The PSET method allows to maintain several match hypothesis for each point, which is an advantage when there are ambiguous matches (which is the rule in image feature correspondences problems), because no hard decision is made before analyzing all image information. The rotation estimation is achieved the same way as in 6DP (with a 5D search over the motion space based on the notion of epipolar constraint), yet the translation scale factor is obtained by exploiting an accumulator array voting scheme based also on epipolar stereo geometry combined with probabilistic distribution hypotheses between the two adjacent stereo image pairs. The obtained results demonstrate a clear performance improvement in the estimation of the linear and angular velocities over current state-of-the-art stereo egomotion estimation methods, and when compared to Inertial Measurement Unit ground-truth information.

1.4 Thesis Overview and Organization

In this section the organization of the dissertation is presented. An illustration containing the thesis chapter organization is displayed in Fig.1.2.

In chapter 1, we mainly introduce the thesis research context, and how our approach relates to visual navigation for mobile robotics research. It describes the role of the visual egomotion in several robotics navigation applications e.g. Visual Odometry, Structure from Motion. The chapter also contains a clear description of the thesis objectives and its main contributions, detailing the main algorithms developed within the thesis scope.

In chapter 2 the related work is described, mainly focusing on other egomotion/VO methods using monocular or stereo vision approaches, either sparse or dense based applications. It also covers subjects related to the fusion of other sensors such as Inertial Measurements Units. Finally it covers related parallel programming implementations of egomotion/VO estimation methods that are currently being applied in mobile robotics applications.

In chapter 3 we introduce key fundamentals concepts of computer vision required for the thesis topic. We underline the geometric principles that are associated with computer vision multiple view geometry for a stereo camera configuration. Epipolar geometry, and egomotion/VO estimation are also discussed with references being provided for each topic.

chapter 4 describes our first approach to the stereo visual egomotion estimation problem, denoted as 6DP. Visual Stereo egomotion estimation is performed using a mixture of probabilistic methods and sparse feature based methods. While the probabilistic methods are responsible for computing the rotational velocities, a feature based approach is used to provide the translation scale factor. Results using a standard car dataset are presented with evaluation against other known state-of-the-art methods for stereo egomotion estimation, and Inertial Measurement Units information.

In chapter 5 we present the novel fully probabilistic stereo visual egomotion method, denoted as Probabilistic Stereo Egomotion Transform **PSET**. It is based on probabilistic correspondences for 5D motion estimation using the epipolar constraint, followed by a voting scheme to compute the missing translation scale factor. Simulation results using synthetic images with ground-truth information are presented, as well as, results using online datasets with Inertial Measurement Units ground-truth information. Evaluation results with other stereo egomotion estimation methods are also provided.

Finally chapter 6 contains conclusions and outcomes of thesis. It also details relevant future work.

The Appendix A contains a detailed explanation of the Zero Normalized Cross Correlation computations, necessary to determine the probabilistic correspondences between images on both 6DP and PSET visual stereo egomotion implementations.

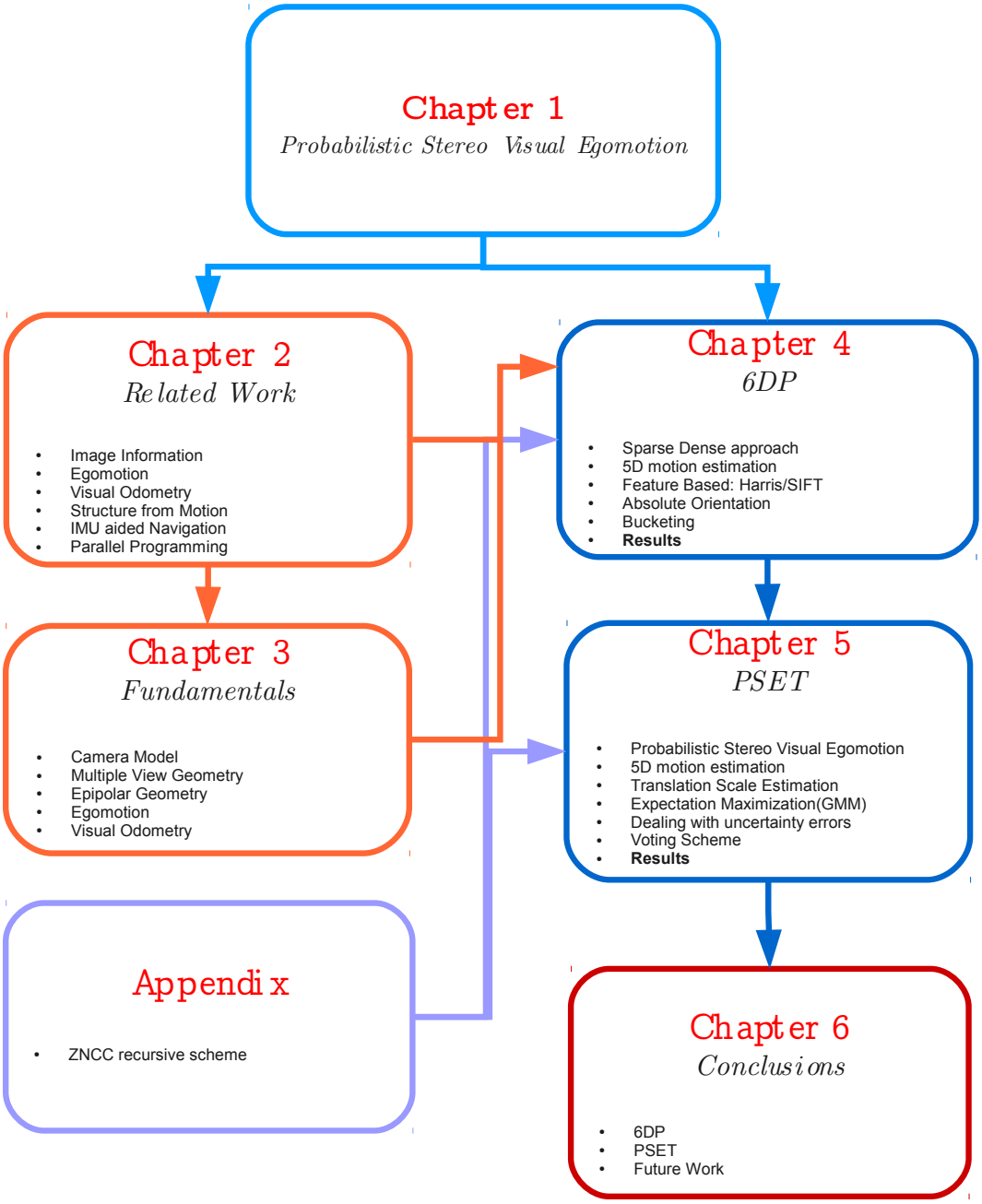

Figure 1.2: Flow Diagram representing the thesis chapter organization, where each box represents a different chapter or appendix where, related topics display the same color.

The implementations of the algorithms developed during the thesis (**6DP**, **PSET**) are described in chapter 4 and chapter 5 respectively. The obtained results for both methods are presented in each individual chapter. In chapter 5 a global comparison of the different implemented methods is provided.

2

Related Work

2.1 Introduction

In this chapter we focus on the thesis related work. First, we turn our attention to the problem of how to obtain motion information from image measurements. Second, we make reference to methods that perform egomotion estimation in the context of robotics and computer vision applications. Third, we make reference to methods that use Graphic Processing Unit (**GPU**) implementations.

Extensive research has been devoted to the estimation of self-motion from image measurements over the past 30 years. In the early stages of motion estimation research, most of the methods utilized were biological inspired methods. Perhaps the most notable one is the optical flow method, which is the spatial shift of brightness patterns in the 2D image reference frame over time due to the movement of the visual observer through an environment. Afterwards, the methods for estimating motion from image measurements evolve to the use of point correspondence methods, usually used in fairly large motion representations, contrary to optical flow methods that are usually employed on small motions, where the brightness constancy assumption holds. Later on, both methods started to be applied in robot navigation tasks, and still today constitute an important vector of visual navigation research for mobile robotics applications, responsible for turning cameras into robots commodity hardware.

In the following sections, we describe some of the existing methodologies for computing egomotion, but also extend the related work to mobile robotics applications that use visual egomotion methods e.g. Visual Odometry (**VO**), Structure from Motion (**SFM**), Visual Simultaneous Localiza-

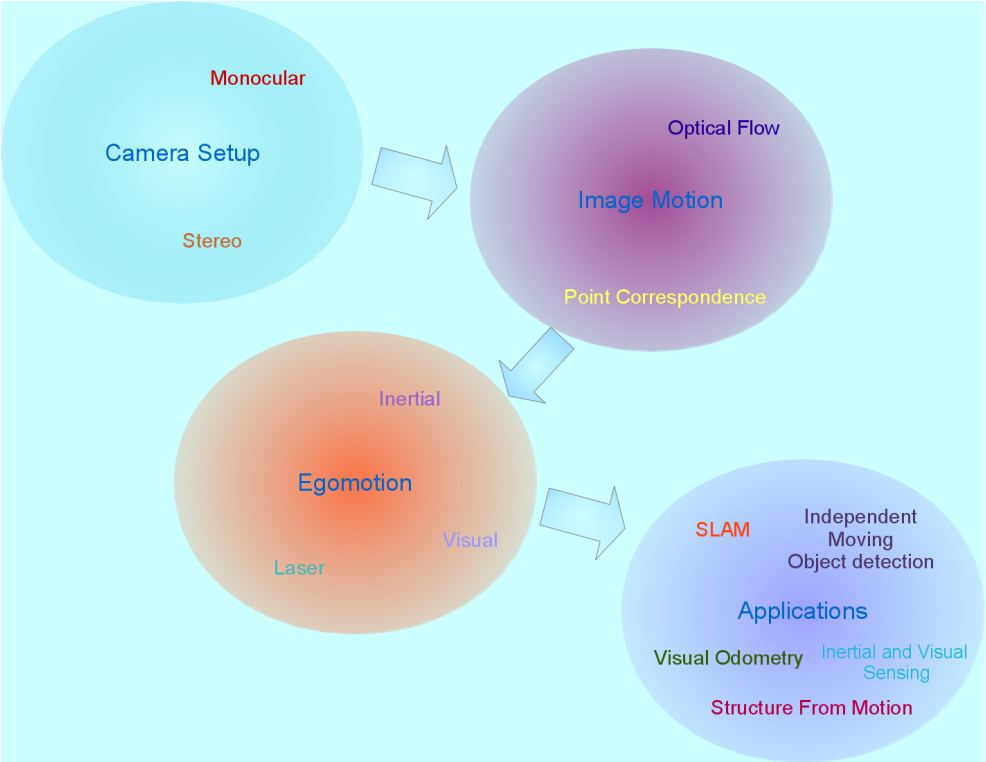

Figure 2.1: Thesis related work, egomotion estimation can be performed using a monocular or stereo camera configuration setup. Motion information from image measurements can be obtained using dense methods (optical flow) or point correspondence methods. Egomotion applications in a computer vision and mobile robotics context include but are not limited to: Structure from Motion, detect moving independent objects in the image, Visual Odometry, Inertial and Visual Sensing and also Simultaneous Localization and Mapping.

tion and Mapping (**VSLAM**), Inertial and Visual Sensing (see Fig.2.1). Finally, we include related work to parallel programming implementation of egomotion estimation methods and related topics.

2.2 Image Information

To recover motion information from a sequence of images taken by a moving camera, we need to identify world points in the environment and measure their relative displacement between images. This is referred in computer vision literature as the correspondence problem [Cor11]. Extensive research has been devoted to solve the correspondence problem by the robotics and computer vision research community. Despite the fact that the taxonomy used by researchers is somewhat confusing, we can roughly divide image motion estimation algorithms into two types: dense methods and key point correspondence methods. Dense methods use all image pixel information to estimate image motion. Some optical flow methods, e.g [HS81], are a particular case of dense methods, where image motion information is computed based on the spatio-temporal patterns of image intensity [BFB94]. Concurrently, point correspondence methods do not use

all image pixel information, and instead find salient points of an image based on pixel intensity values, as well as on the intensity of their neighbors. Then, on each salient point, a signature (*descriptor*) allowing to identify(match or track) the same pixel in other images is extracted.

2.2.1 Dense Methods

Most optical flow methods are based on the principle of brightness (irradiance) constancy over time. Brightness constancy is given by:

$$I(x, y, t) - I(x + u\delta t, y + v\delta t, t + \delta t) = 0 \tag{2.1}$$

where $I(x, y, t)$ is brightness at time t of pixel (x, y), and the optical flow is (u, v). The principle is illustrated in Fig.2.2.

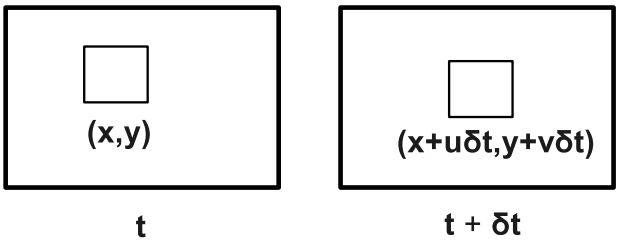

Figure 2.2: Brightness constancy principle of an 2D image pixel representation over a short period of time. The image pattern at position (x, y, t) is the same of position $(x + u\delta t, y + u\delta t, t + \delta t)$

Expanding equation (2.1) into a 1^{st} order Taylor series expansion, and then computing the partial derivatives, we obtain the optical flow constraint:

with

$$I_x = \frac{\delta I}{\delta x}, I_y = \frac{\delta I}{\delta y}, I_t = \frac{\delta I}{\delta t} \tag{2.2}$$

becoming

$$I_x u + I_y v + I_t = 0 \tag{2.3}$$

This imposes a constraint on the vertical and horizontal components of the flow, that depends on the first-order spatio-temporal image derivatives. It is important to notice that the observed motion of image points does not necessarily equate the true motion of the points. In fact because image information is often limited, e.g due to occlusions, brightness changes, and due to the known aperture problem [NS88], it leads to ambiguities in the estimates of world points motion.

Following the taxonomy of the seminal work of Barron *et al.* [BFB94], and extending it to novel applications, optical flow methods can be divided into **differential methods** that compute the velocities from the spatio-temporal derivates or filtered versions of the image e.g.[HS81],[WS03], [LK81],[XJM12] and [SSV97], **correlation methods** that use window image regions to maximize some similarity measure between the regions under the assumption that the region remains undistorted for a short period of time e.g.[Ana89],[SA92],[Sun99],[MCF10], and **frequency methods**

that estimate optical flow, using spatiotemporal filters in the Fourier Domain, like e.g.[Hee87],[Hee88] that uses Gabor filters [Dau85] tuned to different spatiotemporal frequencies, in order to find the strongest velocity orientation vector of an image point.

Optical flow research can be traced back to the early $80s$. However, optical flow methods have been subject of ongoing development until the present time. For example Ogale *et al*.[OFA05] used occlusions for finding independent moving objects instantaneously in a video obtained by a moving camera with a restricted field of view. The problem joins image motion caused by the combined effect of camera motion (egomotion), with the recovery of a structure (depth), as well as the independent motion of scene entities. In Baker *et al*.[BSL+10] a dataset benchmark [1] for evaluating optical flow methods is proposed. The authors provide a sequence of real and synthetic images that constitute the standard for evaluating novel optical flow methods. When trying to apply optical flow methods to the egomotion estimation problem, they tend to perform better on small motions, and breakdown when the camera motions are large, or when the image undergoes significant perspective or affine distortion. In section 2.3, we make reference to related work of egomotion estimation using optical flow methods for estimating image motion.

2.2.2 Point Correspondence

Another method of retrieving image motion information is by using point correspondence methods. The objective of point correspondence methods is to find unique characteristics, sometimes denoted as *descriptors*, in an image point, that makes it easy to match or track in another image. In egomotion estimation, usually the image on the next time instant has undergone some type of motion, and therefore the appearance of the point of interest will have changed due to image noise, differences in rotation, scale and illumination. Point correspondence methods employ techniques that are, in principle, more robust to these effects, since they are based on salient key points.

One of the first key point detectors to be used for obtaining point correspondences was developed by Moravec [Mor80], denoted as Moravec corner detector. In Moravec corner detection algorithm a patch around a pixel is compared with the neighboring patches, through a metric of similarity denoted as Sum-Square-Differences (**SSD**). If a pixel is on a smooth region or an edge, there should exist a neighboring patch that is very similar. For a corner point, all neighboring patches should be different. In that way, Moravec defined the corner strength at a pixel as the smallest sum of squared differences between the center patch and its surrounding patches. The problem with the Moravec corner detector is that only a finite number of neighboring patches in the horizontal, vertical and both diagonal directions are considered. For example if an edge is present, but is not in the direction of its neighbors it will be classified as corner erratically, hence the method is not isotropic. Based on these premises other point correspondence method,

[1] http://vision.middlebury.edu/flow/

the Harris corner detector [HS88] was developed. The Harris corner detector is based on the computation of an auto-correlation matrix of the image gradients whose eigenvalues indicate the magnitudes of two orthogonal gradient vectors. For a corner point, both magnitudes should be large. There are other corner detectors based on the Harris Corner method implementation like e.g. Shi and Tomasi [ST94] that modified the original method corner classification criteria. Later on, SUSAN [SB97] corner detector was developed to improve the sensibility of corner detection algorithms to image noise, and FAST [RD06] corner detection algorithm was develop for use in high frame-rate applications to increase the speed of the corner detection algorithm.

Almost all corner detection algorithms for obtaining point correspondence between images, have difficulties under different conditions of lighting, translation and rotation. To overcome such limitations a novel class of point correspondence methods has been develop, the scale invariant methods e.g Scale Invariant Feature Transform (**SIFT**) [Low04] method, or the Speeded Up Robust Feature (**SURF**) [BETVG08] method. The main issue behind the scale estimation problem, is that an object viewed up close does not occupies the same image area when is viewed far away. This effect produces blur in the image object making more difficult to obtain a point correspondence. Scale-invariant methods search for points robust to these type effects. For example the **SIFT** locates the local maximum and local minimal(scale space extrema detection) by searching a pyramid of Difference of Gaussian images taken at different scales with Gaussian kernels with different standard deviations. Other example of a scale-invariant method is the **MSER** [MCUP04] method. It detects image regions with the following properties: closed under continuous and perspective transformation of image coordinates, and also closed under monotonic transformation of image intensities. Image pixels are grouped in a binary threshold form by their intensity values, followed by a sorting procedure in the image by ascending or descending order where connected components are grouped by an union-find algorithm. More recently the **BRIEF** descriptor developed by Caloender *et al*.[CLSF10], improves the SIFT descriptor by reducing the complexity of the descriptor to binary strings using hash functions. The similarity between descriptors is then measured by the Hamming distance. To improve not only the robustness but also the speed of the scale-invariant detectors the **BRISK** detector developed by [LCS11] was created. The main difference when compared to the popular SIFT method, is the search for local maxima not only in the image plane but also in the scale space using FAST [RD06]detector score for measuring saliency. The keypoints scale is estimated in a continuous scale-space. Other method of identifying point correspondence is the DAYSY descriptor developed by Tola *et al*.[TLF10]. It tackles the problem using a multi-scale approach to be able to match image points even when their scale changes.

In Fig.2.3 a timeline with the image motion methods previously mentioned in the text is presented.

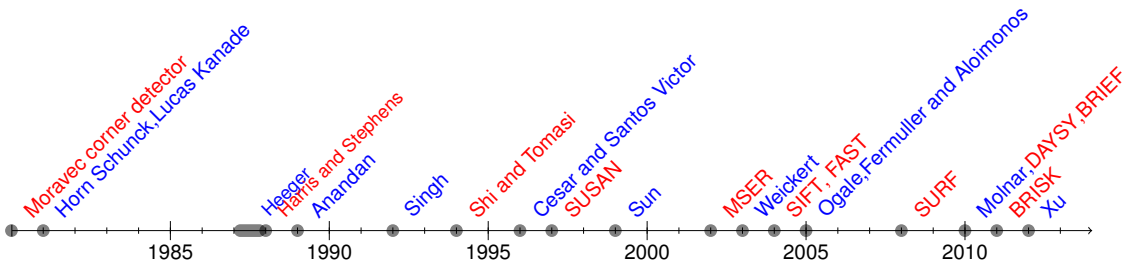

Figure 2.3: Image motion measurements methods chronology, in blue (optical flow methods), in red (key point methods).

2.2.3 Summary on Image Information

The ability to discriminate image points that undergone motion during two consecutive frames is a crucial step in any visual egomotion estimation method. We adopted a taxonomy that divides methods of obtaining motion information from image measurements into two types: dense methods and key point correspondence methods. Both methodologies have been extensible employed in egomotion estimation. In [Lim10], it is stated that dense methods (optical flow) are more suitable for small motion displacements, while point correspondence methods perform better when large motion occurs. It is clear from Fig.2.3, that point correspondences methods are more recent than optical flow methods. Most mobile robotics applications that use image motion algorithms to compute visual egomotion estimation employ point correspondence methods. In the following section we discuss egomotion estimation in detail.

2.3 Egomotion Estimation

Egomotion estimation, as defined in Raudies and Neumann [RN12] is the estimation, from a sequence of images recorded from a moving camera, of the 3D camera movement, as well as the relative depth of the pictured environment. In the paper, Raudies and Neumann focus on three estimation problems: First, how to compute the optical flow. Second, how to estimate the egomotion using the computed optical flow combined with a model of the visual image motion. Finally, the estimation of the relative depth, with respect to the translational speed of the observer.

In the previous section, we addressed the problem of how to compute the optical flow, and other methods to obtain image motion information. We now turn our attention to the egomotion estimation problem per se. To answer this problem, we classified each of the reference egomotion estimation methods according to their end application e.g. Visual Odometry (**VO**) monocular and stereo, Structure From Motion (**SFM**), Visual SLAM (**VSLAM**), and Inertial and Visual Sensing.

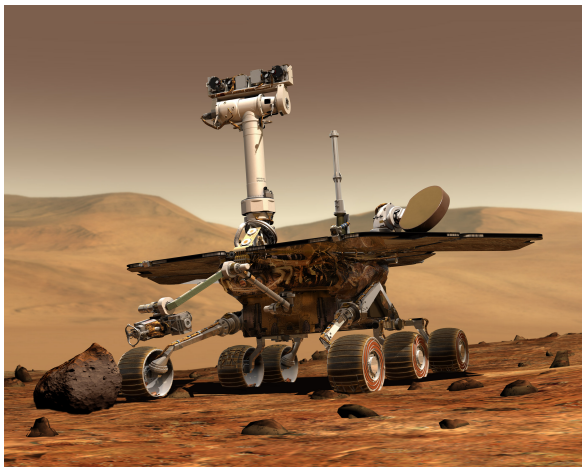

Figure 2.4: Opportunity Mars Exploration Rover

2.3.1 Visual Odometry

In robotics applications, egomotion estimation is directly linked to Visual Odometry(VO) [SF11]. The use of VO methods for obtaining robot motion has been continuously subject of research by the robotics and automotive industry over the past years. One way of performing VO estimation is by determining instantaneous camera displacement on consecutive frames, and integrating over time the obtained rotational and translational velocities. The need to develop such applications urged from the fact that there is an increasing use of mobile robots on modern world tasks, as well as their application scenarios. One of the most complex tasks is navigation where typically IMU/GPS sensor information is used. Typical robotic application scenarios (e.g. urban areas, underwater **GPS** denied environments) are prone to IMU/GPS failures, making it necessary to use other alternative or complementary sensors such as vision cameras. When using visual sensors (cameras), robots must determine motion measuring their displacement relative to static key points in the environment.

In monocular egomotion estimation there is translation scale ambiguity, i.e. in the absence of other sources of information, only the translational velocity direction is possible to measure reliably, therefore whenever possible a stereo camera setup is used. This method is denoted as Stereo VO or stereo egomotion estimation.

If one considers that stereo VO algorithms account for 3D key points position estimation by using triangulation between left and right image stereo pair, then relative body motion can be obtained based upon aligning 3D key points position between consecutive image frames. Most of the work on stereo visual odometry methods was driven by Matthies *et al.*[MMC05],[MMC07] outstanding work on the famous Mars Rover Project, denoted as **MER** method (see Fig.2.4). The system was able to determine all 6 Degrees-of-Freedom (**DOF**) of the rover (x, y, z, roll, pitch, yaw) by tracking "interest" 2D image pixel motion between stereo image pairs, and inherently by triangulation obtain their 3D world point coordinates. Concerning the way image motion informa-

tion is obtained, the method employs a key point detector scheme using Fornster [FG87] or Harris [HS88] corner detector combined with a grid scheme to sample key points over the image. After 3D point position has been triangulated using stereo correspondence, a fixed number of point is used inside a **RANSAC**[FB81] framework to obtain an initial motion estimation using least-squares-estimation. Subsequently a maximum likelihood estimation (batch estimation) procedure uses the Rotation matrix (R_{lsq}) and translation vector ($\mathbf{t_{lsq}}$) obtained by least-square-estimation, as well as the "inlier" points to produce a more accurate motion estimation.

The stereo VO method implemented in the Mars Rover Project was inspired by Olson *et al*.[OMSM03] it was when visual odometry estimation methods started to surface as replacement for wheel odometry dead reckoning methods, urged by the need to develop methods able to correctly estimate robot motion over long distances. In order to avoid large drift in robot position over time, Olson *et al*. method combines a primitive form of the stereo egomotion estimation procedure used in [MMC05] with absolute orientation sensor information.

The taxonomy adopted by the robotics and computer vision community to denominate stereo VO estimation methods, divides stereo VO methods into two categories based either on feature detection scheme, or by pose estimation procedure. The most utilized methods for pose estimation are: 3D Absolute Orientation(**AO**) methods, and Perspective-n-Point(**PnP**) methods. In Alismail *et al*.[ABD10b] a benchmark study is performed to evaluate both AO and PnP techniques for robot pose estimation using stereo VO methods. The authors concluded that PnP methods, perform better than AO methods due to stereo triangulation uncertainty, specially in the presence of small stereo rig baselines. For more insights on AO and PnP techniques, the reader is invited to see chapter 3.

The AO methods consists on 3D points triangulation for every stereo pair. Then motion estimation is solved, by using point alignment algorithms e.g. Procrustes method [Goo91], the absolute orientation using unit quaternions method of [Hor87], or the Iterative-Closest-Point(**ICP**) method [RL01] such as the one utilized by Milella and Siegwart [MS06] for estimating motion of an all-terrain rover.

The influential work of Nister *et al*.[NNB06], was one of the first Perspective-n-Point method implementations. It utilized the Perspective-three-point method (P3P- which deals with 3D world points to 2D image point correspondences developed by [HLON94]), it is computed in real-time combined with an outlier rejection scheme **RANSAC**. Nonetheless, despite the fact of having instantaneous 3D information from a stereo camera setup, the authors use a P3P method instead of a more easily implementable AO method. The authors concluded that P3P pose estimation method deals better with depth estimation ambiguity, which corroborates the conclusions drawn by [ABD10b].

In a similar line of work Ni *et al*.[KD06],[NDK09], tries to avoid having a great dependency of feature matching and tracking algorithms, and tested both three-point and one-point stereo visual

odometry implementations using a quadrifocal setting within a RANSAC framework [KD06]. Later on, the same authors in [NDK09] decouple the rotation and translation estimation into two different estimation problems. The method starts with the computation of a stereo putative matching, followed by a classification of features based on their disparity. Afterwards, distant points are used to compute the rotation using a two-point RANSAC method (the underlying idea is to reduce the problem of the rotation estimation to the monocular case), and the closer points with a disparity above a given threshold (Θ_t) are used together with the estimated rotation to compute the 1-point RANSAC translation.

Most of stereo VO methods differ on the way stereo information is acquired and computed, sparse feature or dense stereo approaches. One of the most relevant dense stereo VO applications was developed by Howard [How08] for ground vehicle applications. The method does not assume prior knowledge over camera motion and so can handle very large image translations. However, due to the fact of not having feature detectors invariant to rotation and scaling has some restrictions: only works on low-speed applications and with high frame-rate, since large motions around the optical axis will result in poor performance.

In [MBG07] a visual odometry estimation method using stereo cameras is presented. A closed form solution of an absolute orientation method [Hor87] is derived for the incremental movement of the cameras and combines distinctive features (**SIFT**)[Low04] with sparse optical flow (**KLT**)[LK81].

Recent work on stereo VO has been enforced not by planetary rover applications but more on the development of novel intelligent vehicles and by the automotive industry. Obdrzalek *et al.*[OM10] developed a voting scheme strategy for egomotion estimation, where 6-**DOF** problem was divided into a four dimensions problems and then decomposed into two sub-problems for rotation and translation estimation. Another related work to the automotive industry that uses stereo VO methods, is the one developed by Kitt *et al.*[KGL10]. The proposed method, is available as an open-source visual odometry library named **LIBVISO**. The stereo egomotion estimation approach is based on image triples and online estimation of the trifocal tensor [HZ04]. It uses rectified stereo image sequences and produces an output 6D vector with linear and angular velocities estimation within an Iterative Extended Kalman filter approach. Comport *et al.*[CMR10] also developed a stereo VO method based on the quadrifocal tensor [HZ04]. It computes the image motion using a dense optical flow method developed by Ogale and Aloimonos [OA07].

Recent developments on Visual Odometry, have been achieved by the extensive research conducted at the Autonomous System Laboratory of ETH Zurich University [SFS09], [KCS11], [VNH$^+$11],[RGNS12], [KKN$^+$12]. First, with the work developed by Scaramuzza *et al.*[FS12], [SFS09], that takes advantages of applied motion model constraints to help reduce motion model complexity, and allow a much faster motion estimation. This simplification assumes planar motion, which allows a less complex motion model to be used. Also, since the camera is installed

on a non-holonomic wheeled vehicle, motion complexity can be further reduced to a single-point correspondence. More recently having the work of Kneip *et al*.[KSS11] as reference, a novel parametrization for the P3P perspective-n-point was introduced. The method differs from standard algebraic solutions for the P3P estimation problem [HLON94], by computing the aligning transformation directly in a single stage, without the intermediate derivation of the points in the camera frame. This pose estimation method combined with key point detectors [Low04], [BETVG08], [CLSF10], and with Inertial Measurement Unit information was used to estimate monocular VO [KCS11], and also using a stereo camera setup in [VNH$^+$11]. On a different stereo approach Kazik *et al*[KKN$^+$12] developed a framework that allows to perform 6-DOF absolute scale motion and structure estimation using a stereo setup with non-overlapping fields of view in indoor environments. It estimates monocular VO using each camera individually, and afterwards scale is recovered by imposing the known stereo rig transformation between both cameras.

Throughout the years, several applications were developed to compute visual egomotion estimation from a single camera, using different camera models such as: perspective pinhole model (majority of applications) but also omnidirectional cameras, see the work of Corke *et al*.[CSS05] and Tardif *et al*.[TPD08]. Usually, the problem with monocular VO approaches is the lack of image scale knowledge, since monocular VO applications in an instant frame only calculate motion up to a scale factor.

Nister *et al*.[Nis04] developed a Visual Odometry system, based on a 5-point algorithm, that became the standard algorithm for comparison of Visual Odometry techniques. Vatani *et al*.[NvVB11] developed an approach based on correlation template matching. Their method estimates motion by analyzing a image template motion from frame to frame. Guizilini and Ramos [GR11], presented a monocular VO approach for Unmanned-Aerial-Vehicle (**UAV**) applications, the authors argue that vehicle motion estimation should not rely heavily on the geometry of a calibrated camera model, but instead use a learning approach of how image structure and vehicle dynamics can affect camera motion. This is accomplished using sparse optical-flow with a coupled Gaussian Process based on supervised learning for rotation and translation estimation. A similar work based on a machine learning solution for the monocular VO problem was proposed by Roberts *et al*. [RNKB08]. Other Visual Odometry application for **UAV**, was developed by Dusha and Mejias [DM12], who presented a method capable of recovering position and absolute attitude by fusing VO and GPS measurements in an identical manner to a GPS/INS system. The method was tested using data collected from a real-flight. Warren and Upcroft [WU13] developed a specific method for computing VO at high altitude, by relaxing the (typically fixed) stereo transform during bundle adjustment(**BA**), and thus reduce the method dependency on the fixed geometry for triangulation. This procedure allows to obtain VO estimates even in situations where high altitude and structural deformation from vibration would cause other VO methods to fail.

2.3.2 Visual SLAM

Vision Simultaneous Localization and Mapping (**VSLAM**) is one of the most important applications that can benefit from VO approaches. The short term velocity estimates provided Visual Odometry has been shown to improve the localization results of Simultaneous Localization and Mapping (SLAM) methods. There are many different approaches to the VSLAM problem, one the of most notable was developed by Cummins and Newman [CN08] denominated as *FAB-MAP*. The method uses a probabilistic approach to the problem of recognizing places based on their appearance. The system can determine not only the robot localization but also based on new observations determine if the scene corresponds to a previously unseen place. The system uses a learning generative model of place appearance, that can be triggered online by a single observation of new places in the map. The number of places in the map grows in a linear form with the number of observed places, which can be useful for loop closure in mobile robotics scenarios. The VSLAM method was further developed by Cummins and Newman [CN10] by bulding on top of the probabilistic framework introduced in [CN08], and modifying the model structure (using sparse visual data) to be able to support efficient inference over maps that have higher orders of magnitude when compared to previous developed approaches.

In VSLAM robotic applications, loop-closure detection and global consistent localization are two issues that require the capacity to recognize a previously visited place from current camera measurements. In Angeli *et al* [AFDM08] an online method that makes it possible to detect when an image comes from an already perceived scene using local shape and color information is presented. The authors extended the bag-of-words method used in image classification to incremental conditions and rely on Bayesian filtering to estimate loop-closure probability.

Other VSLAM methods such as the one described in [ZT13] use multiple cameras to solve the vision-based simultaneous localization and mapping in dynamic environments problem. The cameras move independently and can be mounted on different platforms, and all contribute to build a global map. The system is based on a inter-camera pose estimation and an inter-camera mapping scheme that allows to recover the 3D position of static background points, as well as the trajectory of moving foreground points. In order to enhance robustness, the system maintains a position uncertainty measurement of each map point and the cameras are grouped according to their field-of-view overlap.

Now we turn our attention to VSLAM methods that employ Visual Odometry techniques. In [ABD10a], visual odometry measurements are used as priors for the prediction step of a robust EKF-SLAM algorithm. VO provides improved quality predictions mainly in cases of sudden robot accelerations that differ from the constant velocity models usually employed in the prediction steps of SLAM systems. Although, VO approaches are only interested in maintaining an accurate estimate of the local trajectory, VSLAM methods need to maintain global map consistency.

Recently, research combining both methods was developed by Williams and Reid [WR10].

Their method combines robust consistency given by VLSAM approach, based on the famous MonoSLAM by Davidson *et al.*[DRMS07] method, allowing to maintain a sparse map of features, with a VO frame to frame motion estimation that provides additional constraints leading to a more accurate pose estimation.

Civera *et al.*[CGDM10], developed a 1-point RANSAC algorithm with applications to visual odometry. The method fused monocular image sequences with wheel odometry information to estimate VO. Others, like Alcantarilla *et al.*[AYAB12], use camera egomotion for building a dense scene flow representation of the environment. Afterwards, using motion likelihood estimates, they detect visual odometry outliers that are located on moving objects and discard these outliers from the set of matching correspondences. VO estimates are then recomputed and used to create priors for VSLAM estimation. In [TPD08] the camera trajectory is recovered without relying on any motion model. No iterative optimization procedure of the camera position and 3D scene structure is used but instead it employs a simple pose estimation method using information obtained from the 3D map, combined with the epipolar constraint. The camera trajectory is computed in urban scenes with large amount of clutter for over to 2.5 kilometers.

Finally, using a different approach to the stereo VSLAM problem for outdoor applications, Marks *et al.*[MHB+07], developed VO estimation methods for a grid-based VSLAM algorithm. It uses a Rao-Blackwellized particle filter for pose distribution estimation, and visual odometry is used to provide particles, with proposal distributions.

2.3.3 Inertial and Visual Sensing

The combined use of inertial and visual sensing has been subject of continuous research, by the robotics community in the past years. The wide spread use of micro-chip gyroscopes, and accelerometers, together with visual information, is turning ubiquitous in today's robots. Even though both sensing modalities can be used separately, their complementary behavior makes suitable its combined used in mobile robotics navigation.

The tutorial work of Lobo *et al.*[CLD07], introduces the topic of the combined use of Inertial and Visual sensors. It is stated, and can be easily proved by the wide range of applications, that cameras and inertial sensors are complementary in terms of the type, rate of information and types of motion estimates that they can provide. Inertial sensors are more capable of dealing, with high profile velocities and accelerations at higher rates, where they exhibit a lower relative uncertainty with minimal computation cost. On the contrary, visual information provides more accurate low velocities estimates at the expense of high processing requirements. With the increase in velocity, visual sensors suffer from the effects of the motion blur, and not sufficient scene overlap to be able to recover relative motion. Therefore, the added value of inertial sensing in perception of egomotion, and structure from motion (**SFM**), is essential in most mobile robotics navigation applications. An example of a valuable use of Inertial sensing is the use of gravity as vertical reference to sim-

plify point correspondence [LD03]. Recently Naroditszky *et al*. [NZG+11], [NZG+12] developed a new minimal method for computing relative pose for monocular visual odometry within a RANSAC framework that uses three image correspondences combined with what authors call directional correspondence(denoted as three-plus-one method). The main concept behind the approach is to reduce the five point estimation problem using a four point minimal solver, as long as the fourth point is at the infinity (providing directional correspondence). The authors improved this approach, by using the Inertial Measurement Unit (**IMU**) measurements to provide the directional correspondence (using the gravity vector). In Saurer *et al*.[SFP12] the gravity vector obtained from IMU measurements is used to help compute the homography of a visual odometry application using a smartphone in urban crowded areas. Egomotion estimation can also be obtained from the combination of vision and inertial measurements [LD04]. In [PZJ12], an egomotion approach based on the fusion of monocular visual with inertial measurements is presented. The system observes features on the ground plane and tracks these features between consecutive images. Based upon the virtual camera concept perpendicular to the ground plane introduced by [MDT07], and also on the concept of epipolar constraints [HZ04], a measurement model that imposes visual constraints on the inertial navigation system is used to perform 6-DOF motion estimation. The work of Jones *et al*.[JVS07],[Jon10],[JS11] also has been devoted to the combination of monocular vision and inertial measurements, with a special focus not only on the egomotion estimation task, but also on the tasks of localization and mapping with real-time concerns.

In Kelly *et al*.[KSS08] an helicopter platform using stereo visual odometry combined with Inertial Measurement Unit estimation is presented. The authors follow the visual odometry estimation method that uses Maximum-Likelihood Estimation (**MLE**) of [MMC05]. Afterwards, the visual measurements are combined with IMU data using an Extended Kalman Filter (**EKF**). Li and Mourikis [LXX12] also improve the accuracy of visual odometry with inertial measurements using an EKF approach. In Trawny *et al*.[TMR+07] an EKF algorithm for estimating the pose and velocity of a spacecraft during entry, descent, and landing is presented. The estimator uses the observations of known pre-defined visual landmarks on the surface of the planet and combines it with rotational velocity and accelerations obtained from the IMU in a tightly coupled manner. The authors argue that the method diminishes the uncertainty ellipses by three orders of magnitude than the method solely based on Inertial Measurement Unit information. Bryson *et al*.[BS11] used Inertial Measurement Unit combined with vision and GPS information to build 3D large-scale terrain reconstructions using UAVs. The camera pose was computed from visual feature matching between frames, based on 1D epipolar constraints combined with IMU/GPS information.

2.4 Parallel Programming

Nowadays, scientific community but also industrial companies have been conducting a substantial amount of work on real-time implementations of vision algorithms. Several commercial equipment such as: Field Programmable Gate Array (**FPGA**), Digital Signal Processor (**DSP**), Graphic Processing Unit (**GPU**), General Purpose Processor (**GPP**) and Application Specific Integrated Circuit (**ASIC**) have been used for developing these implementations.

There are two different types of requirements driving research on these fields of vision and computational expertise. People that need to develop low power, low weight, low size, capable of producing re-configurable software solutions, and others that need quite the opposite, having the need of huge computational resources and massive parallel code optimization.

Due to wide range of parallel programming vision applications is difficult to define a consensual related work on this topic. So, within the scope of our work, we are going to focus on parallel programming vision algorithms for mobile robotics applications, specifically describing novel parallel hardware implementation of egomotion related topics.

One of the most interesting research being performed on parallel programming for mobile robotics application is being conducted by NASA **JPL** laboratory. Howard *et al* [HMM+12] are transforming MER [MMC05] approach, through an FPGA capable of performing stereo and visual odometry. This improved version of the algorithm includes a FPGA implementation of the well known Harris corner detector [HS88], and also the Sum of Absolute Differences (**SAD**) operator. The RANSAC inlier detection and minimization of the re-projection error is already performed on a conventional Central Processing Unit (**CPU**). On a much smaller scale, Goldberg and Matthies [GM11], developed a stereo and IMU visual odometry method using an OMAP3530 System-on-chip for small mobile robotics applications. Their main objective is to develop stereo and visual odometry mobile robotics applications using standard cell-phone hardware.

Parallel hardware implementations of visual egomotion estimation methods, up until now are based on "classical methods" of egomotion estimation such as Shi and Tomasi features with RANSAC [IGN11] and spatio-temporal filters [SRD13]. Research effort is being put into the development of high frame-rate methods able to run faster than real-time, rather than in the robustness and accuracy of egomotion estimation itself.

The seminal work of Newcombe *et al.*[ND11] on the field of Dense Tracking and Mapping in Real Time (DTAM), is based on the construction of dense map (all image points) using a single RGB camera and a **GPGPU** hardware implementation with real-time constraints. The implementation is straightforward, all image points are aligned with a dense model of the scene, and the camera pose is obtained from the images. The model is then used to build dense textured depth maps. Newcombe further developed this work with KinectFusion[NLD11], by taking advantage of low-cost and widespread use of RGB-D cameras as the Microsoft Kinect whose easy integra-

tion with the Robotic Operating System [2] (**ROS**) allows to obtain instantaneous depth information. The sensor pose information is obtained, by simultaneously tracking the live depth (using all image data available) of the sensor frame relative to a global model using a coarse-fine iterative closest point method. This work has become an important research vector and still today is subject of continuous improvement as shown by Whelan *et al.*[WJK13]. They improved the work [NLD11] by adding the integration of multiple 6-DOF camera odometry estimation methods for robust tracking, as well as, a novel GPU based implementation of an existing dense RGB-D visual odometry algorithm develop by [SSC11]. They performed some advances in fusion of real-time surface coloring. Kerl *et al.*[KSC13] develop a real-time odometry method based on dense RGB-D images, by aligning two consecutive RGB-D images and minimizing the photo-metrical error. In Huang *et al.*[HBH+11] a mixture of existing visual odometry estimation methods is used to develop a stereo visual odometry system using a RGB-D camera for an autonomous flight vehicle, the method denoted as FOVIS is freely available as an open source stereo visual odometry library[3] for RGB-D and stereo calibrated cameras. It is important to mention that RGB-D sensors still have unsolved problems when performing on outdoor applications.

2.5 Summary on Egomotion Estimation

To perform visual egomotion estimation, one must extract relevant image information from a camera setup (monocular, or stereo), and then be able to relate the same image information between two consecutive time frames. The egomotion estimation computes the linear and angular velocities that the camera(s) undergone during the differential of time instants. It is a key asset in mobile robotic applications specially in navigation tasks to help realize the localization of a robot in its world environment, and its application crosses many robotic vision related topics. In this chapter, we presented several methods and applications of how to compute egomotion, based on how the image information can be retrieved, or by the way the pose estimation is achieved. In table 2.1 a summary of related work methods (several applications) that use egomotion information are presented. The methods are grouped by their camera configuration setup, image information, other sensors use, pose estimation and end-application.

[2] http://wiki.ros.org/kinect
[3] https://code.google.com/p/fovis/

Table 2.1: Visual Odometry methods comparison

Method	Camera Setup	Image	O.Sensors	Pose Estimation	Scale	Application
MER (Matthies *et al* [MMC05])	Stereo	Keypoint Forstner, Harris	No	**MLE**	Yes	VO
5-point (Nister [Nis04])	Monocular	Keypoint Harris	No	PnP	No	VO,SFM
3-point (Nister *et al*)[NNB06]	Stereo	Keypoint Harris	No	PnP P3P [HLON94]	Yes	VO
Milella and Siegwart [MS06]	Stereo	Mixed SRI, Shi and Tomasi	No	AO ICP [RL01]	Yes	VO
Ni et al [NDK09]	Stereo	Mixed Harris, dense stereo	No	PnP 2P(R),1P($\mathbf{t}$) RANSAC	Yes	VO
Howard [How08]	Stereo	Dense SAD, Census	No	PnP P3P	Yes	VO
Moreno et al [MBG07]	Stereo	Mixed SIFT, KLT	No	AO [Hor87]	Yes	VO
Agrawal and Konolidge [AK07]	Stereo	Keypoint Harris	IMU	AO 3-Point [AKI05] with BA+IMU	Yes	VO
LIBVISO (Kitt *et al* [KGL10])	Stereo	Keypoint Harris, SURF	No	Trifocal Tensor	Yes	VO
Comport *et al* [CMR10]	Stereo	Dense Optical Flow [OA07]	No	Quadrifocal Tensor	Yes	VO
Kazik *et al* [KKN$^+$12]	Stereo [a]	Keypoint BRIEF	IMU	PnP P3P-modified [KSS11]+WBA	Yes	VO
Corke *et al* [CSS05]	Monocular(Omni)	Sparse Optical Flow KLT	No	Shape-from-Motion [SMSH01]	No	VO, SFM
Scaramuzza *et al* [SFS09]	Monocular(Omni)	Keypoint Harris	No	PnP 1P RANSAC + NH constraints	No [b]	VO
Guizilini and Ramos [GR11]	Monocular	KeyPoint SIFT, SURF	No	Learning coupled Gaussian Process	Yes	VO,SFM
Vatani *et al* [NvVB11]	Monocular	Dense Optical Flow [HS81]	IMU	Ackerman Model+IMU	Yes	VO
Roberts *et al* [RNKB08]	Monocular	Sparse Optical Flow Harris with KLT	No	Learning Memory based **KNN**	No	VO
Dusha and Mejias [DM12]	Monocular	Keypoint SURF	GPS	Homography	Yes [c]	VO
Rehder et al [RGNS12]	Stereo	Mixed Stereoscan3D [GZS11]	IMU/GPS	PnP P3P combined with IMU/GPS	Yes	VO
Kneip et al [KCS11]	Monocular	Keypoint SURF, AGAST [MHB$^+$10], BRIEF	IMU	PnP P3P-modified [KSS11]	Yes	VO,SFM
Voigt et al [VNH$^+$11]	Stereo	Keypoint FAST, BRIEF	IMU	PnP P3P-modified [KSS11]	Yes	VO
Alcantarilla et al [ABD10a]	Stereo	Keypoint Harris	No	PnP 2P(R),1P($\mathbf{t}$) RANSAC	Yes	VO,VSLAM
Williams and Reid [WR10]	Monocular	Keypoint FAST	No	PnP 8P VO + Robocentric SLAM [WKR07]	Yes	VO,VSLAM
PTAM (Klein and Murray [KM07])	Stereo	Mixed [d]	No	AO	Yes	SFM, AR
DTAM (Newcombe *et al*[ND11])	Monocular	Dense Depth Maps	No	AO	Yes	SFM, AR
FOVIS (Huang *et al*[HBH$^+$11])	Stereo	Mixed FAST, Gaussian pyramidal	No	AO [Hor87]	Yes	VO,VSLAM

[a] Monocular cameras with non-overlapping FOV
[b] Scale is obtained from the speed of the car read by a CAN sensor
[c] Scale is obtained from filtering GPS and VO
[d] Mapping is based on keyframes using batch optimization techniques, Tracking of features is Dense using the 5-point algorithm and FAST corner detector

3

Fundamentals

3.1 Introduction

In this chapter, we describe some fundamental Computer Vision concepts required for the thesis research topic. It is not our intention to be complete on the broad range of geometrical aspects regarding egomotion estimation, but just to cover the basics to understand the different techniques involved on egomotion/Visual Odometry (VO) estimation. For a more detailed analysis the reader is provided with references to each of the related topics. However, with respect to probabilistic methods, we follow a different approach, and the basic probabilistic methods principles will be presented together with our probabilistic stereo egomotion implementations, since we believe it provides the reader a more fluent analysis of the thesis document and of its main contributions.

3.2 Image Formation

3.2.1 Pinhole Camera Model

In this thesis, we only work with cameras based on the perspective projection camera model, specifically with the pinhole projection model. The camera model relates 3D world point space onto 2D image space by:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} f & 0 & c_x & 0 \\ 0 & f & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3.1}$$

If one considers **X** to be the representation of the point in space by the homogeneous vec-

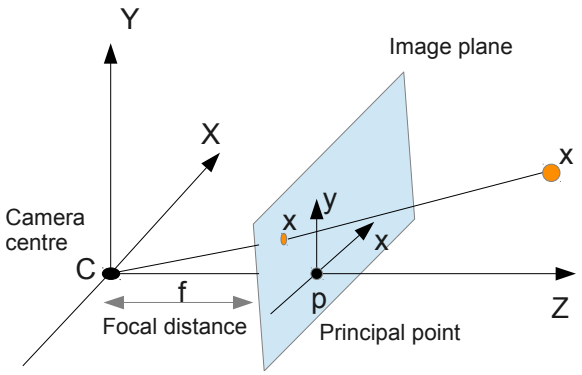

Figure 3.1: Pinhole Camera Model

tor $(X, Y, Z, 1)^T$, **x** for the image point represented by a homogeneous vector (x, y, z) which is computed based on projective geometry mapping $(X, Y, Z)^T \rightarrow (fX/Z, fY/Z, f)^T$, and P as the camera projection matrix, one can represent the camera model by:

$$\mathbf{x} \simeq P\mathbf{X} \tag{3.2}$$

Usually in computer vision notation, parameter f corresponds to the focal distance and parameters (c_x, c_y) represent the principal point coordinates, which is the place where the principal axis intersects the image plane in pinhole camera model, see figure 3.1.

The camera parameters like the focal distance f, principal point (c_x, c_y), and skew s which is the angle between the x and y sensor axes, need to be known for the model to be applied. They form part of matrix K, which is the camera calibration matrix (3.3) and contains the camera intrinsic parameters defined as follows,

$$K = \begin{bmatrix} f & s & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3.3}$$

In Visual Odometry applications, sometimes it is necessary to apply an Euclidean transformation between the camera coordinate frame and the world coordinate frame, that is given by a 3×3 rotation matrix R representing the orientation of the camera coordinate frame and a translation vector **t**, as displayed by:

$$\mathbf{X}_c = R\mathbf{X}_w + \mathbf{t} \tag{3.4}$$

The camera projection matrix becomes:

$$P = K[R|\mathbf{t}] \tag{3.5}$$

3.3 Geometry of Multiple Views

In the previous section we discussed how to map a 3-dimensional coordinate of a corresponding world point to a 2-dimensional image coordinate point using a perspective projection camera model. By doing so, we lost spatial dimension information about the point, and all we know now is that the world point lies along some ray in space corresponding to the pixel coordinate. Therefore, we need to recover the dimension lost, by adding additional information. One of the most utilized means of obtaining this information is to consider information of multiple views from the same scene. The underlying idea is to use information from a different but known camera pose in order to determine the exact position of the world point. Then, the world point will be at the intersection of the two rays originating from each of the cameras, and can be computed in a process denoted in computer vision literature as *triangulation* or *3D reconstruction* [HZ04],[Sze10],[Cor11]. If enough points are available, then we can estimate the 3D motion of the camera (**VO**), as well as to recover the 3D structure of the observed points(**SFM**). To accomplish this objective there is the need to solve the fundamental and non-trivial question of how to obtain *correspondences* between images.

3.3.1 Epipolar Geometry

The epipolar geometry allows to geometrically relate images of a single point (**P**) observed from two different viewpoints $\{L\}$ and $\{R\}$ as shown in Fig.3.2. It can be used to represent the case of two cameras simultaneously viewing the same scene (stereo case scenario), or a single camera taking a snapshot from two different viewpoints (monocular case scenario). Elaborating, the pose of each camera has its origins at $\{O_L\}$ and $\{O_R\}$, and together with the world point **P**, they define the epipolar plane in space. The world point **P** is projected onto the image planes of the two cameras at ($\mathbf{p_L}$) and ($\mathbf{p_R}$) respectively. Using image I^L as image reference, the image point ($\mathbf{e_L}$) known as epipole point, is the projection of $\{O_R\}$ in camera $\{L\}$. The camera center of projection $\{O_L\}$, the epipole point ($\mathbf{e_L}$) and image point ($\mathbf{p_L}$) define the epipolar plane and hence the epipolar line $\mathbf{l_R}$ in image I^R. By definition, point ($\mathbf{p_R}$) must belong to the epipolar line $\mathbf{l_R}$. The opposite is also true, and point $\mathbf{p_L}$ must also lie along epipolar line $\mathbf{l_L}$, that is defined by point $\mathbf{p_R}$ in image I^R.

This is a key geometric concept that limits the correspondence of a given point in image I^L to lie along a line in the other image I^R. This concept is denoted as the epipolar constraint [HZ04].

The epipolar relationship can be described concisely by:

$$\tilde{\mathbf{p}}_\mathbf{R}^\mathbf{T} F \tilde{\mathbf{p}}_\mathbf{L} = 0 \tag{3.6}$$

where points $\tilde{\mathbf{p}}_\mathbf{L}$ and $\tilde{\mathbf{p}}_\mathbf{R}$ are the image points ($\mathbf{p_L}$,$\mathbf{p_R}$) expressed in homogeneous coordinates and F is the fundamental matrix.

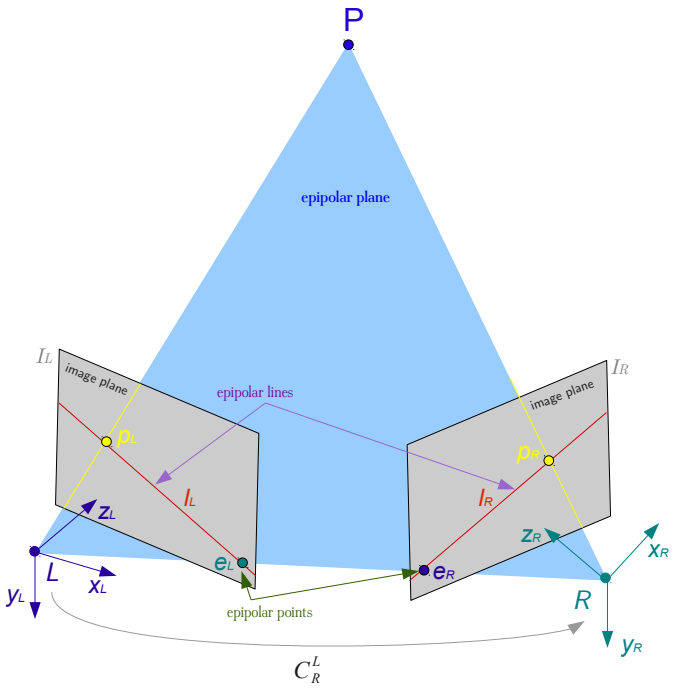

Figure 3.2: Epipolar Geometry showing two camera reference frames $\{L\}$ and $\{R\}$, that are related via pose transformation C_R^L. The world point **P** and the two cameras centers form the epipolar plane, and the intersection of the epipolar plane with the image plane forms the epipolar lines

The epipolar line $\tilde{\mathbf{l}}_{\mathbf{R}}$ defined in homogeneous coordinates can be obtained by:

$$\tilde{\mathbf{l}}_{\mathbf{R}} \simeq F\tilde{\mathbf{p}}_{\mathbf{L}} \tag{3.7}$$

where ($\simeq$) means equal up to a possible unknown scale value. This is the equation of the epipolar line, along which the conjugate point in image I^R must lie

$$\tilde{\mathbf{p}}_{\mathbf{R}}^{\mathbf{T}}\tilde{\mathbf{l}}_{\mathbf{R}} = 0 \tag{3.8}$$

The fundamental matrix (F) is a function of the camera intrinsic parameters (K), and also encodes the relative camera poses between the different viewpoints and can be given by:

$$F \simeq K^{-1}R[\tilde{\mathbf{t}}]_{\mathbf{x}}K \tag{3.9}$$

Another form of describing the epipolar geometric constraint is by using the essential matrix [HZ04]. The essential matrix is the specialization of the fundamental matrix to the case of calibrated cameras (known intrinsic parameters) and it was introduced by Longuet-Higgins [LH87]. It has fewer degrees of freedom, and some additional properties compared to the fundamental matrix [Cor11].

$$\tilde{\mathbf{x}}_{\mathbf{R}}^{\mathbf{T}}E\tilde{\mathbf{x}}_{\mathbf{L}} = 0 \tag{3.10}$$

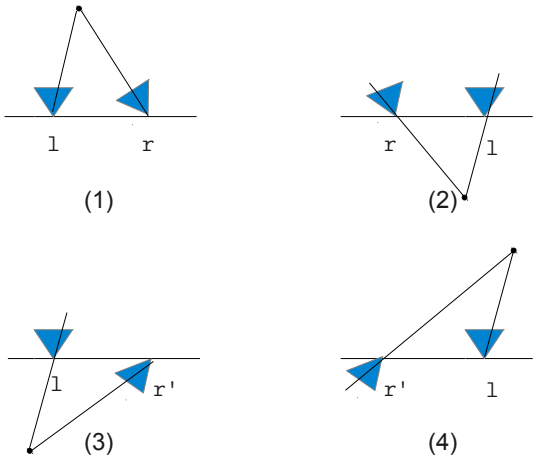

Figure 3.3: The four possible solutions for obtaining left and right camera pose from E. Only in solution (1) the point is in front of both cameras (l,r).

where E is the essential matrix [HZ04], $\tilde{\mathbf{x}}_\mathbf{L}$ and $\tilde{\mathbf{x}}_\mathbf{R}$ are conjugate points in homogeneous image coordinates. The essential matrix E can be computed directly by:

$$E \simeq R[\tilde{\mathbf{t}}]_\mathbf{x} \tag{3.11}$$

The essential matrix has 5-**DOF**, and is defined by three rotational and two translation parameters. It can also be computed from the fundamental matrix by assuming $\tilde{\mathbf{p}} \simeq K\tilde{\mathbf{x}}$, and by substituting in (3.10) we obtain

$$\tilde{\mathbf{p}}_\mathbf{R}^\mathbf{T} K_R^{-T} E K_L^{-1} \tilde{\mathbf{p}}_\mathbf{L} = 0 \tag{3.12}$$

and thus establish a relationship between the essential and fundamental matrices

$$E = K_R^T F K_L \tag{3.13}$$

Once the essential matrix is known, the camera matrices may be retrieved up to an unknown scale factor. There are four possible solutions (see figure 3.3) that are computed from the factorization of E into the product of $R[\tilde{\mathbf{t}}]_x$. Although having four possible solutions, only one solution will have a point in front of both cameras (solution 1). Therefore a single point is usually sufficient to decide between the four solutions for obtaining the cameras pose.

The key point is that the fundamental and essential matrix encode the geometry that allows to relate two cameras or the same camera from two different viewpoints. The fundamental matrix (F) and a point in one image define an epipolar line in the other image on which its conjugate point must lie. The essential matrix (E) encodes the relative pose between the two camera frames, and their pose can be extracted with translation scaled by an unknown factor. The geometric relations provide valuable information to the visual egomotion estimation problem, and a number of techniques for monocular and stereo visual egomotion estimation are based on these assumptions e.g. [Nis04],[NNB06],[DA06],[SBS13b],[SBS13a].

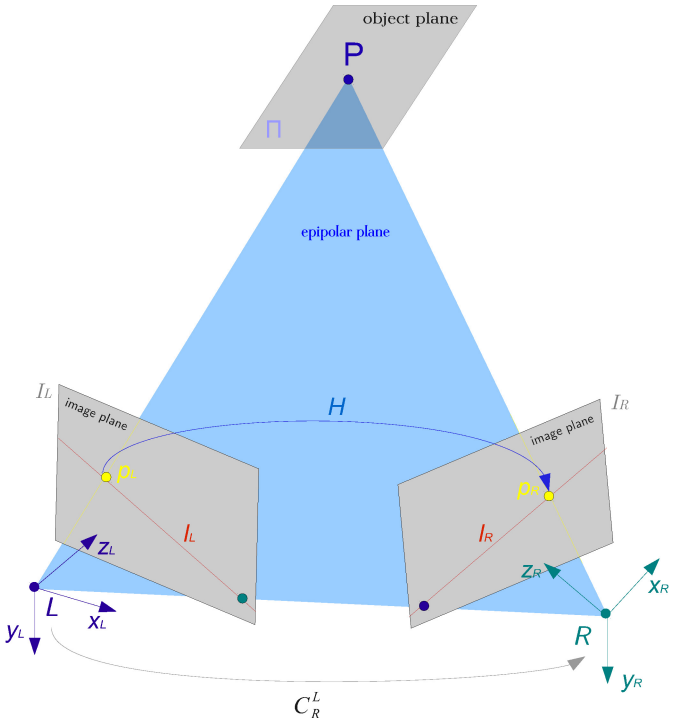

Figure 3.4: The homography geometry consists on having two cameras with coordinate frames $\{L\}, \{R\}$. The 3D world point **P** belongs to a plane with surface normal Π. The homography H allows to map point $\mathbf{p_L}$ to $\mathbf{p_R}$

3.3.2 Planar Homography

The planar homography represents the geometry of a camera viewing a group of world points $\mathbf{P_i}$ that lie on a plane. The points are viewed by two different cameras $\{L\}, \{R\}$, and the projection of the points onto the cameras is given by:

$$\tilde{\mathbf{p}}_{\mathbf{R_i}} \simeq H\tilde{\mathbf{p}}_{\mathbf{L_i}} \tag{3.14}$$

where H is 3$\times$ 3 non singular up to a scalar matrix, known in computer vision literature [HZ04], [Cor11] as homography. The homography H as the fundamental matrix F, can be computed from two sets of corresponding points. From the homography H, it is possible to tell exactly where the conjugate point will be in the other image, as long as the point lies on a plane. One way of estimating the homography is to use robust estimation methods such as **RANSAC** [FB81], to establish the best relationship between the sets of points $\tilde{\mathbf{p}}_{\mathbf{L_i}}$,$\tilde{\mathbf{p}}_{\mathbf{R_i}}$. The homography estimation geometry is displayed in figure 3.4.

Similarly to the essential matrix E, the homography can also be express in normalized image coordinates by:

$$\tilde{\mathbf{x}}_{\mathbf{R}} \simeq H_E\tilde{\mathbf{x}}_{\mathbf{L}} \tag{3.15}$$

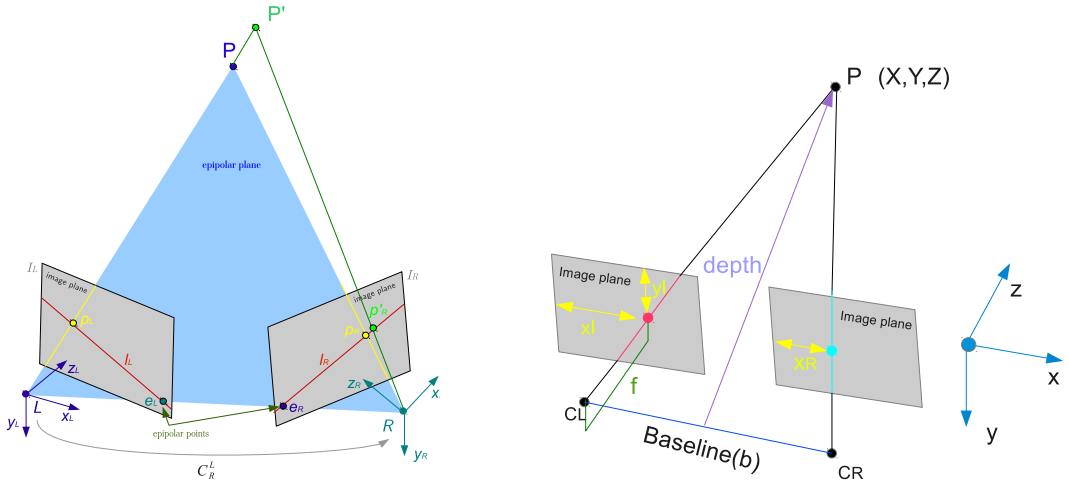

Figure 3.5: Depth estimation uncertainty over the epipolar line. In the left figure the epipolar geometry shows the point depth variation of points **P**,**P'** along the epipolar line in the second image. In the right figure is shown the triangulation procedure to estimate point **P** 3D camera reference frame coordinates

where the Euclidean homography H_E is given by:

$$H_E \simeq R + \frac{\mathbf{t}}{d}\Pi^T \tag{3.16}$$

relating the motion (R,$\mathbf{t}$) $\sim C_R^L$ and the plane $\Pi^T\mathbf{P} + d = 0$ with respect to $\{L\}$. Both the Euclidean and projective homographies are related by:

$$H_E \simeq K^{-1}HK \tag{3.17}$$

being K the camera parameter matrix.

Analogously to the essential matrix, the homography can also be decomposed to recover the relative pose between the two cameras. There are multiple solutions to the camera pose estimation problem using the homography that need to be disambiguated using additional information to determine the correct solution. Complementary, the translational component of the transformation matrix is computed up to an unknown scale factor [HZ04].

3.3.3 Stereo Vision

Stereo vision is the technique of estimating the 3-D structure of the world from two images taken from different viewpoints [Cor11].

Usually in order to simplify the stereo vision estimation procedure both cameras images undergo a transformation process called *stereo rectification*. The underlying idea of the rectification procedure, is to determine a transformation of the image planes such as the conjugate epipolar lines become collinear and parallel to one of the image axes, hence reducing the stereo correspondences problem to a 1D horizontal search over the new rectified images. In figure 3.5, is shown the depth ambiguity that affects the world points position (**P**,**P'**) over the epipolar line in

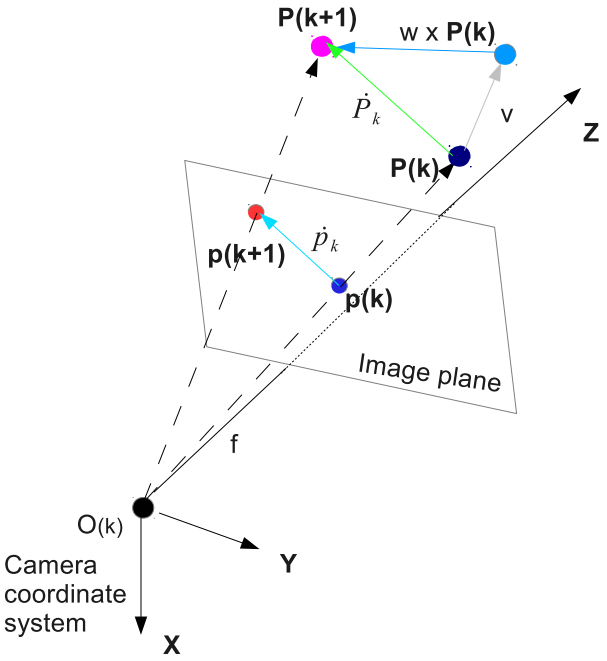

Figure 3.6: Model that formalizes the displacement in time and visual space of image sequences (k, k+1), according to Longuet-Higgins and Pradzny model

the other image. On the right hand side of the figure, it is displayed the stereo camera setup after calibration (with known camera position, baseline and rectification).

The *stereo triangulation* procedure uses each image point correspondences to estimate world points depth by:

$$\mathbf{P} = \begin{cases} \mathbf{X} = \frac{X_l \mathbf{Z}}{f} \\ \mathbf{Y} = \frac{Y_l \mathbf{Z}}{f} \\ \mathbf{Z} = \frac{fb}{X_l - X_r} \end{cases} \tag{3.18}$$

The world point **P** depth is inversely proportional to the disparity d, that is obtained by subtracting the x axes coordinates of the two image points $d = \mathbf{x_l} - \mathbf{x_r}$, being d>0 so that the 3D point is placed in front of the cameras.

3.4 Egomotion Recovery

Egomotion estimation methods are usually divided into two different motion types, those that employ differential camera motion and others that assume a discrete camera motion. Differential motion assumes instantaneous camera translational and angular velocity between time frames, while discrete motion are used when there is a large translation and rotation between views.

The differential motion models, follow a visual motion field generated by egomotion through a rigid environment, that has been proposed by Longuet-Higgins and Prazdny [LHP80].

In figure 3.6 the visual motion field model of Longuet-Higgins and Prazdny describes visual motion as the 3D velocities of a point $\dot{\mathbf{P}} = (\dot{\mathbf{X}}, \dot{\mathbf{Y}}, \dot{\mathbf{Z}})^T$ projected onto an image plane. The 3D

velocities are directly linked with the image point $\mathbf{P} = (\mathbf{X}, \mathbf{Y}, \mathbf{Z})^T$, and the velocity of the camera frame by:

$$\dot{\mathbf{P}} = -\mathbf{v} - \mathbf{w} \times \mathbf{P} \tag{3.19}$$

where $\mathbf{v} = (v_x, v_y, v_z)$ and $\mathbf{w} = (w_x, w_y, w_z)$ the translational and angular velocity respectively, and $\times$ denotes the cross product. It can be represented in scalar form as:

$$\begin{aligned} \dot{\mathbf{X}} &= \mathbf{Y} w_z \mathbf{Z} - w_y - v_x \\ \dot{\mathbf{Y}} &= \mathbf{Z} w_x - \mathbf{X} w_z - v_y \\ \dot{\mathbf{Z}} &= \mathbf{X} w_y - \mathbf{Y} w_x - v_z \end{aligned} \tag{3.20}$$

From the pinhole model, we can obtain 2D point **p** image coordinates as:

$$x = f\frac{\mathbf{X}}{\mathbf{Z}}, y = f\frac{\mathbf{Y}}{\mathbf{Z}} \tag{3.21}$$

Computing the temporal derivative of this expression and linking it with definition of instantaneous motion given by translational and rotational velocity (3.19) we obtain:

$$\dot{\mathbf{p}} = \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \frac{1}{\mathbf{Z}} \underbrace{\begin{pmatrix} -f & 0 & x \\ 0 & -f & y \end{pmatrix}}_{\text{translational part}} \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} + \frac{1}{f} \underbrace{\begin{pmatrix} xy & -(f^2 + x^2) & fy \\ (f^2 + y^2) & -xy & -fx \end{pmatrix}}_{\text{rotational part}} \begin{pmatrix} w_x \\ w_y \\ w_z \end{pmatrix} \tag{3.22}$$

It is important to note that in the translational part the depth **Z** and the translational speed $||\mathbf{v}||$ are invariant to each other. Therefore, only the translation direction and relative depth can be recovered, with the unknown parameter being the translation scale factor.

The discrete motion model follows the Longuet-Higgins stereo model [LH87]. Where the same two viewpoints are linked by a rigid body transformation comprising a rotation and translation (R,**t**), following the geometry of Fig.3.2. The egomotion estimation problem in this case relies on point correspondences from the two viewpoints in order to recover the transformation matrix.

3.5 Visual Odometry Estimation

In the tutorial work of Scaramuzza and Fraundofer [SF11][FS12] Visual Odometry **VO** is defined as the process of estimating robot egomotion using solely the input of a single or multiple cameras.

In this section we make a brief summary of the VO estimation methods, and the way camera motion is obtained from a sequence of images.

If one considers two camera positions at consecutive time instants (k, $k+1$), they are related by a rigid body motion transformation $M_k^{k+1} \in R^{4x4}$:

$$M_k^{k+1} = \begin{bmatrix} R_k^{k+1} & \mathbf{t}_k^{k+1} \\ 0 & 1 \end{bmatrix} \tag{3.23}$$

The camera motion can be obtained if one can ground perception to static features in the environment, and then measure camera displacement based on relative feature displacement on adjacent time instants. When computing VO estimates, the main objective is to continuously compute pose transformations between adjacent time instants, therefore due to VO methods incremental nature pose uncertainty grows unbounded with time. To reduce the error there is the need to refine local trajectory estimation employing iterative refinement techniques or combining other sensors information e.g. Global Positioning Systems (**GPS**) or Inertial Measurement Unit (**IMU**) information.

Like previously mentioned in chapter 2, there are two major alternatives to compute image motion information. The first approach, is the most used and it is known as feature based approach. It is according to some authors like [SF11] easier to implement, faster and more accurate and consists on finding and robustly match (or track)features across adjacent time frames. The second approach, defined as appearance based methods in [SF11], and denoted as dense methods in chapter 2, uses intensity pixel level information of all pixels in two images, being more complex and therefore more difficult to implement. Until now most of them only have been applied to the monocular VO estimation case.

To recover the camera path one must concatenate all relative motions. VO methods for motion estimation are classified in [SF11], according to how point correspondences dimensions are specified.

The first approach is the 2D-2D motion estimation case. The VO methods using 2D to 2D motion estimation step are based on 2D point correspondences. The geometric relation between two images of a calibrated camera setup is established by the essential matrix E. If E is computed from the 2D to 2D correspondences using the epipolar constraint, one can recover motion parameters directly from E, being R the camera rotation and $\tilde{\mathbf{t}}$, the camera translation vector up to a unknown scale factor α, with $\tilde{\mathbf{t}}_\times$ the skew symmetric representation of $\tilde{\mathbf{t}}$ as in (3.24) [HZ04].

$$E = R\left[\tilde{\mathbf{t}}\right]_\times \tag{3.24}$$

The epipolar constraint determines that point **x'**, which is the 2D correspondence of point **x** at I_k, lies on a line on image I_{k+1}. It can be formulated by:

$$\mathbf{x'}^T E\mathbf{x} = 0 \tag{3.25}$$

The Essential Matrix E can be computed if one uses at least 5-point 2D correspondences solving a linear equation system $AE = 0$ using singular value decomposition (SVD). Then R and $\tilde{\mathbf{t}}$ can be found by factorization of E as in [Nis04]. Usually, one obtains four possible solutions

for $(R,\ \tilde{\mathrm{t}})$ estimation, being the correct solution the one that obtains a positive image depth, as already illustrated in figure 3.3. Since VO 2D-2D estimation method can only recover translation direction and not its magnitude, translation scale must be computed by alternative methods e.g. triangulating 3D points $\mathbf{X_k}$ and $\mathbf{X_{k+1}}$ from two subsequent image pairs [SF11].

Other type of motion estimation parametrization is the 3D to 3D. In this case, keypoints in image I_k and image I_{k+1} are defined in 3D dimensions. It is usually applied in VO stereo vision estimation, where 3D point information is easily obtainable by stereo triangulation. The camera motion can be estimated directly by determining the alignment transformation between 3D points in two consecutive image pairs $(\mathbf{X_{k,i}}, \mathbf{X_{k+1,i}})$, see equation (3.26).

These VO methods are also denoted as Absolute Orientation **AO** methods e.g. Procrustes orientation method [Goo91], and Iterative Closest Point method [RL01]. The difference to the previous 2D-2D estimation case, is that, since point correspondences are defined in a camera reference frame (3D) instead of an image plane (2D), translation scale is already included:

$$\arg\min_{M_k^{k+1}} \sum_i ||\mathbf{X}_{k+1,i} - M_k^{k+1}\mathbf{X}_{k,i}|| \tag{3.26}$$

Most monocular VO estimation methods employ a 3D-2D motion estimation procedure, even though it can also be applied to stereo visual odometry estimation. It is more accurate then previous 3D-3D motion according to [SF11] [NNB06], because it tries to minimize the re-projection error instead of the keypoint position error (3.26), as displayed in (3.27).

$$\arg\min_{M_k^{k+1}} \sum_i ||\mathbf{x}_{k+1,i} - \mathbf{x}_{k,i}||^2 \tag{3.27}$$

This method is also denoted as the Perspective-n-Point problem (**PnP**). Solutions for solving are still being developed e.g. [LXX12], [WH], [LMNF09] or using Direct Linear Transform **DLT** [HZ04]. It can be described as the problem of determining the calibrated camera pose from n correspondences between 3D camera reference points and their 2D projections. The PnP problem solutions are classified as iterative or non-iterative methods. Non-iterative methods are computationally more efficient, but are unstable when n < 5. The minimal solution of points to compute the PnP is the 3-point solution denoted as P3P. The standard approach for VO applications that typically include n points is based on a P3P in a RANSAC scheme [FB81] in order to remove point outliers, and then use other PnP estimation method on all remaining inliers.

4

Sparse and Dense Stereo Visual Egomotion

4.1 Introduction

In this chapter, we describe our first approach to the stereo egomotion estimation problem using probabilistic methods.

Having a good own vehicle motion estimation is of key importance if one wants to act as a "independent moving observer" and detect targets based upon motion estimation capabilities.

Despite the amount of research in Egomotion/VO during the past few years, almost all approaches employ feature based methods. These methods have the advantage of being fast, since only a subset of the image points is processed, but depend critically on the features to track between consecutive pairs of frames and are often sensitive to noise. On the opposite, dense methods combined with probabilistic approaches have demonstrated higher robustness to noise and outliers. In Domke et al [DA06], a method for estimating the epipolar geometry describing the motion of a camera is proposed using dense probabilistic methods. Instead of deterministically choosing matches between two images, a probability distribution is computed over all possible correspondences. By exploiting a larger amount of data, a better performance is achieved under noisy measurements. However, that method is more computationally expensive and does not recover the translational scale factor.

One solution to overcome such limitation was to develop a probabilistic egomotion estimation method using a stereo vision approach. The developed method allows us to determine vehicle linear and angular velocities and was developed using a stereo camera system setup, similar to

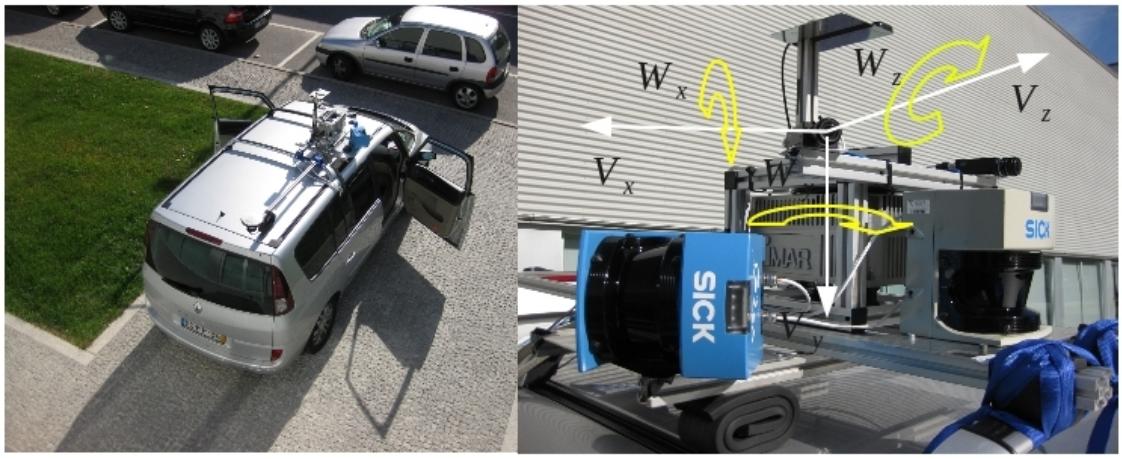

Figure 4.1: Example of Acquisition Setup for a vehicle-like robot, with the use of stereo cameras for providing estimates of vehicle angular and linear velocities.

the one displayed in figure 4.1. The method estimates camera rotation and translation direction using an adaptation of the dense probabilistic method of [DA06] and solves the translational velocity magnitude with a deterministic, feature-based, stereo algorithm. This was our first approach to the stereo visual egomotion estimation problem and, as we show in this chapter, allows better angular velocity estimation than current state-of-the-art approaches, and led to the publications ([SBS13a], [SBS13b]).

4.2 A mixed approach to stereo visual egomotion: combining sparse and dense methods

In this chapter we propose a method to estimate the linear and angular velocities (V, W) of a vehicle equipped with a calibrated stereo vision setup. Let the images acquired by the left and right cameras of the stereo vision system in consecutive time instants be represented as the 4-tuple $\mathbf{I_{k+1}} = (I_k^L, I_k^R, I_{k+1}^L, I_{k+1}^R)$, where the subscripts k and $k+1$ denote time, and the superscripts R and L denote the right and left cameras, respectively. From point correspondences between the observations in $\mathbf{I_{k+1}}$ we can compute the rigid transformation describing the incremental motion of the setup and, thus, estimate its velocity at instant k, (V_k, W_k). Our method, denoted 6DP, combines sparse feature based methods and dense probabilistic methods [SBS13a] to compute the point correspondences between the 4-tuple of images. While feature based methods are less computational expensive and are used in real-time applications, dense correlation methods tend to be computational intensive and used in more complex applications. However, when combined with probabilistic approaches, dense methods are usually more robust and tend to produce more precise results. Therefore we developed a solution that tries to exploit the advantages of both methods.

Our 6DP method, as schematically illustrated in Fig 4.2, can be roughly divided into three main

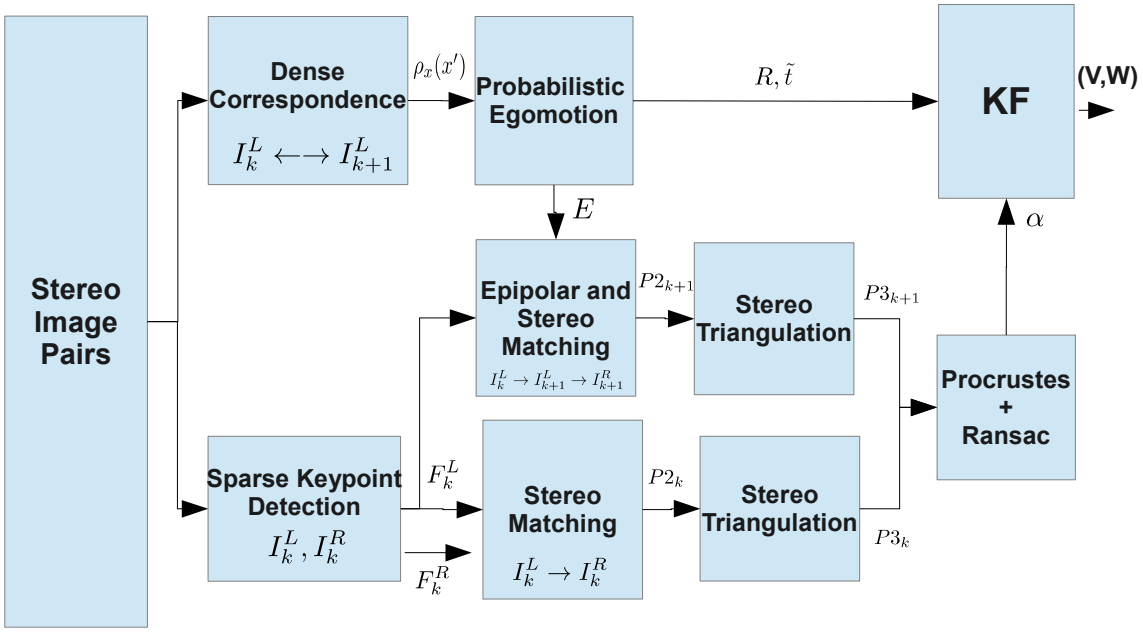

Figure 4.2: 6DP architecture

steps:

- **Dense Correspondence and Egomotion estimation**

  In order to be able to estimate egomotion, first there is the need to compute correspondence information between images I_k and I_{k+1}, where k and $k+1$ are consecutive time instants. For egomotion estimation a variant of the dense probabilistic egomotion estimation method of [DA06] is used. By doing so, we establish a probabilistic correspondence between the left images at consecutive time steps, I_k^L and I_{k+1}^L, and estimate camera rotation (R) and translation ($\tilde{\mathbf{t}}$) up to a scale factor (α), by maximizing likelihood in the space of Essential Matrices [DA06].

- **Sparse Keypoint and Stereo Matching**

  The sparse keypoint detection consists on obtaining salient features in both images at time k (I_k^L, I_k^R). To obtain the keypoints a feature detector such as the Harris corner [HS88] or a SIFT detector [Low04] is used. The result is a set of feature points F_k^L, F_k^R that will be used in a stereo matching procedure to obtain point correspondence $P2_k$ at time k, and together with the Essential Matrix (E), correspondences $P2_{k+1}$ at time $k+1$.

- **Scale Estimation**

  The missing translation scale factor (α), is obtained by stereo triangulation with the point correspondences at time k and $k+1$, ($P2_k$, $P2_{k+1}$), thus obtaining corresponding point clouds $P3_k$ and $P3_{k+1}$ with point match information. Afterwards, we use an AO method like the Procrustes method [Goo91] to obtain the best alignment between the two sets of points

Algorithm 1: 6DP Method

Input: 2 stereo Image pairs (I_k^L, I_k^R) and (I_{k+1}^L, I_{k+1}^R), E_{rig} (stereo calibration)
Output: (Velocities) V, W

Step 1. Compute the probabilistic correspondences between images I_k^L and I_{k+1}^L, $\rho_x(x')$. Eqs. (4.1),(4.2), (4.3)

Step 2. Compute probabilistic egomotion, E. Eqs. (4.5), 4.6), (4.7),

Step 3. Compute sparse keypoints in images I_k^L and I_k^R, F_k^L and F_k^R respectively. We conducted experiments using both Harris corners and Scale Invariant Features (SIFT)

Step 4. Perform stereo matching in between features F_k^L and F_k^R to obtain matches $P2_k$.

Step 5. Perform epipolar and stereo matching between images I_k^L, I_{k+1}^L and I_{k+1}^L , I_{k+1}^R, respectively, to obtain point matches $P2_{k+1}$.

Step 6. Stereo triangulate matches $P2_k$ and $P2_{k+1}$ to obtain corresponding point clouds $P3_k$ and $P3_{k+1}$, respectively.

Step 7. Perform Translation scale estimation using an Absolute Orientation method (Procrustes) to align point clouds $P3_k$ and $P3_{k+1}$. Use RANSAC to reject outliers. Eqs. (4.9), (4.10)

Step 8. Estimate Linear and Angular Velocities , V and W Eqs. (4.11), (4.12), (4.14)

Step 9. Standard Kalman Filtering Eqs. (4.15) and (4.16)

and determine the value of the translation scale factor (α). A RANSAC algorithm [FB81] is used to discard outliers in the 3D point cloud matches.

- **Kalman Filtering**
  To achieve a more robust egomotion estimation, we use a Kalman Filter approach for the linear and angular velocity estimates.

4.2.1 Probabilistic Correspondence

The key to the proposed method relies on a robust probabilistic computation of the epipolar geometry relating the camera's relative pose on consecutive time steps. This will speed-up and simplify the search for 3D matches on the subsequent phases of the algorithm. Given two images taken at different times, I_k and I_{k+1}, the probabilistic correspondence between point $\mathbf{x} \in R^2$ in image I_k and point $\mathbf{x}' \in R^2$ in image I_{k+1}, is defined as a belief:

$$\rho_{\mathbf{x}}(\mathbf{x}') = \mathrm{match}(\mathbf{x}, \mathbf{x}' | I_k, I_{k+1}) \tag{4.1}$$

where the function $\mathrm{match}(\cdot)$ outputs a value between 0 and 1 expressing similarity in the appearance of the two points in local neighborhoods.

Thus, all points $\mathbf{x}'$ in image I_{k+1}^L are candidates for matching with point $\mathbf{x}$ in image I_k^L with a likelihood proportional to $\rho_{\mathbf{x}}(\mathbf{x}')$. One can consider $\rho_{\mathbf{x}}$ as images (one per each pixel in image I_k^L) whose value in $\mathbf{x}'$ is proportional to the likelihood of $\mathbf{x}'$ matching with $\mathbf{x}$. In Fig.4.4, we can

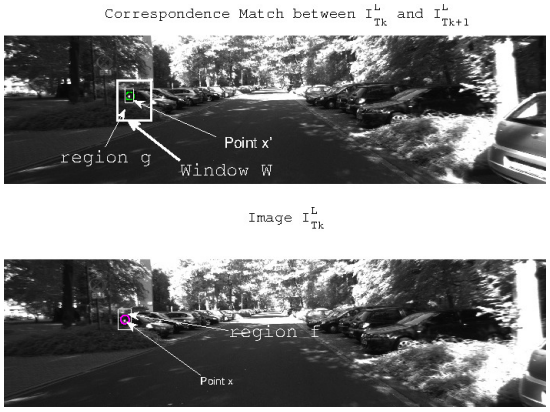

Figure 4.3: Image feature point correspondence for ZNCC matching, with window size N_W between points $\mathbf{x}$ and $\mathbf{x}'$ represented in red and green respectively

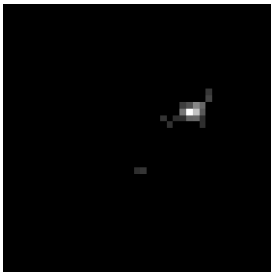

Figure 4.4: Likelihood of a point $\mathbf{x}$ in image I^L_k with all matching candidates $\mathbf{x}'$ in I^L_{k+1}, for the case of Fig. 4.3. Points with high likelihood are represented in lighter colour

observe the correspondence likelihood of a point $\mathbf{x}$ in image I^L_k with all matching candidates $\mathbf{x}'$ in I^L_{k+1}. For the sake of computational cost, likelihoods are not computed for the whole range in image I^L_{k+1} but just on windows around $\mathbf{x}$, or suitable predictions based on prior information (see Fig. 4.3).

In [DA06] the probabilistic correspondence images computed via the differences between the angle of a bank of Gabor filter responses in $\mathbf{x}$ and $\mathbf{x}'$. The motivation for using a Gabor filter bank is its robustness to changes in the brightness and contrast of the image. However, it demands a significant computational effort, thus we propose to perform the computations with the well known Zero Mean Normalized Cross Correlation function **ZNCC**:

$$C_{x,y}(u,v) = \frac{\sum_{x,y \in N_W} (f(x,y) - \bar{f})(g(x+u, y+v) - \bar{g})}{\sqrt{\sum_{x,y \in N_W} (f(x,y) - \bar{f})^2} \sqrt{\sum_{x,y \in N_W} (g(x+u, y+v) - \bar{g})^2}} \tag{4.2}$$

The ZNCC method allows to compute the correlation factor $C_{x,y}(u,v)$ between regions of two images f and g by using a correlation window around pixel $\mathbf{x} = (x,y)$ in image f and pixel $\mathbf{x}'$ = x+(u,v) in image g, being the correlation window size $N_W = 20$. The value $N_W = 20$ is a compromise between match quality and computational cost that we found adequate for this

problem through our empirical studies, $\bar{f}$ and $\bar{g}$ are the mean values of the images in the regions delimited by the window size. This correlation factor is then transformed into a likelihood match between $\mathbf{x}$ and $\mathbf{x}'$.

$$\rho_{\mathbf{x}}(\mathbf{x}') = \frac{C_{x,y}(u,v)}{2} + 0.5 \tag{4.3}$$

The ZNCC function is known to be robust to brightness and contrast changes and recent efficient recursive schemes developed by Huang *et al.* [HZP+11] render it suitable to real-time implementations. The method is faster to compute and yields similar results to the implemented by Domke [DA06].

4.2.2 Probabilistic Egomotion Estimation

From two images of the same camera, one can recover its motion up to the translation scale factor. Given the camera motion, image motion can be represented by the epipolar constraint which, in homogeneous normalized coordinates, can be written as:

$$(\tilde{\mathbf{x}'})^T E \tilde{\mathbf{x}} = 0 \tag{4.4}$$

where E is the so called Essential Matrix [HZ04], as already explained in chapter 3.

To obtain the Essential matrix from the probabilistic correspondences, [DA06] proposes the computation of a probability distribution over the 5-dimensional space of essential matrices. Each dimension of the space is discretized in 10 bins, thus leading to 100000 hypotheses E_i. For each point $\mathbf{x}$ the likelihood of these hypotheses is evaluated by:

$$\rho(E_i|\mathbf{x}) \propto \max_{(\tilde{\mathbf{x}'})^T E_i \tilde{\mathbf{x}}=0} \rho_{\mathbf{x}}(\mathbf{x}') \tag{4.5}$$

Intuitively, for a single point $\mathbf{x}$ in image I_k^L, the likelihood of a motion hypothesis is proportional to the likelihood of the best match obtained along the epipolar line generated by the essential matrix. After the dense correspondence probability distribution has been computed for all points, the method [DA06] computes a probability distribution over motion hypotheses represented by the epipolar constraint. Assuming statistical independence between the measurements obtained at each point, the overall likelihood of a motion hypothesis is proportional to the product of the likelihoods for all points:

$$\rho(E_i) \propto \prod_{\mathbf{x}} \rho(E_i|\mathbf{x}) \tag{4.6}$$

Finally, having computed all the motion hypotheses, a Nelder-Mead simplex method [NM65] is used to refine the motion estimate around the highest scoring samples E_i. The Nelder-Mead simplex method is a local search method for problems whose derivatives are not known. The method was already applied in [DA06] to search for the local maxima of likelihood around the top ranked motion hypotheses:

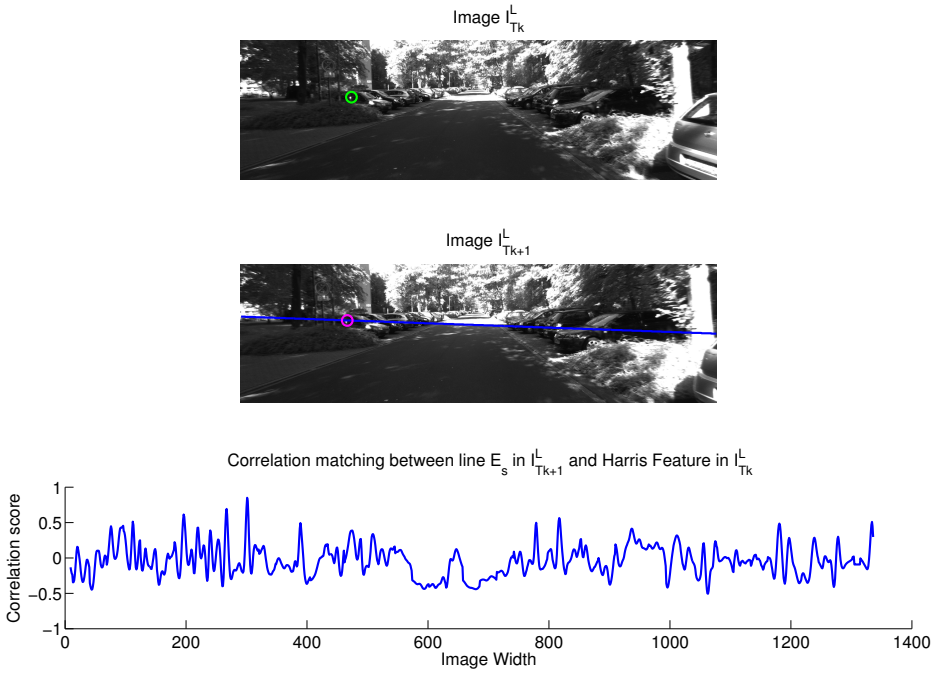

Figure 4.5: Image feature point marked in colour green in image I_k^L lies in the epipolar line (blue) estimated between I_k to I_{k+1}. The point with higher correlation score, marked in red in image I_{k+1}^L is chosen as the matching feature point.

$$E_i^* = arg \max_{E_i+\delta E} \rho(E_i + \delta E) \tag{4.7}$$

where δE are perturbations to the initial solution E_i computed by the Nelder-Mead optimization procedure.

Then, the output of the algorithm is the solution with the highest likelihood

$$E^* = \max_i E_i^* \tag{4.8}$$

4.2.3 Scale Estimation

By using the previous method, we compute the 5D transformation $(R, \tilde{\mathbf{t}})$ between the camera frames at times k and $k+1$. However, $\tilde{\mathbf{t}}$ does not contain translation scale information. This type of information, will be calculated by an Absolute Orientation(AO) method like the Procrustes method.

Once the essential matrix between images I_k^L and I_{k+1}^L has been computed by the method described in the previous section, we search along the epipolar lines for matches F_{k+1}^L in I_{k+1}^L to the features F_k^L computed in I_k^L, as displayed in Fig. 4.5.

Then, these matches are propagated to I_{k+1}^R by searching along horizontal stereo epipolar lines for matches F_{k+1}^R. From stereo triangulation we compute 3D point clouds at instant k and

$k+1$, respectively $P3_k$ and $P3_{k+1}$, with known point correspondence. Points whose matches are unreliable or were not found are discarded from the point clouds.

4.2.3.A Procrustes Analysis and Scale Factor Recovery

The Procrustes method allows to recover rigid body motion between frames through the use of 3D point matches, obtained in the previous steps

$$\mathbf{P3^i_{k+1}} = R'\mathbf{P3^i_k} + \mathbf{t}' \tag{4.9}$$

where i is a point cloud element.

In order to estimate the motion $[R', \mathbf{t}']$, a cost function that measures the sum of squared distances between corresponding points is used.

$$c^2 = \sum_i^n \left\| \mathbf{P3^i_{k+1}} - (R'\mathbf{P3^i_k} + \mathbf{t}') \right\|^2 \tag{4.10}$$

Performing minimization of equation (4.10) is possible to estimate $[R', \mathbf{t}']$. However these estimates are only used to obtain the missing translation scale factor α, since rotation (R) and translation direction $(\tilde{\mathbf{t}})$ were already obtained by the probabilistic method. Although conceptually simple, some aspects regarding the practical implementation of the Procrustes method must be taken into consideration. Namely, since this method is sensitive to data noise, obtained results tend to vary in the presence of outliers. To overcome this difficulty, RANSAC [Fis81] is used to discard possible outliers within the set of matching points.

4.2.3.B Bucketing

For a correct motion scale estimation, it is necessary to have a proper spatial feature distribution through out the image. For instance, if the Procrustes method uses all obtained image feature points without having their image spatial distribution into consideration, the obtained motion estimation $[R', \mathbf{t}']$ between two consecutive images could turn out biased. To avoid having biased samples in the RANSAC phase of the algorithm a bucketing technique [ZDFL95] is implemented to assure a balanced image feature distribution sample. In Fig. 4.6 a possible division of the image is displayed. The image region is divided into $L_x \times L_y$ buckets, based on minimum and maximum coordinates of the feature points. Afterwards, image feature points are classified as belonging to one of the buckets. In case a bucket does not contain any feature, it will be disregarded. The bucket size must be previously defined: in our case we divided the image into a 8 $\times$ 8 buckets. Assuming we have l buckets, the interval between $[0...1]$ is divided into l intervals such that the width (i^{th}) of each interval is defined as $n_i / \sum_i n_i$, where n_i is the number of matches assigned to the i^{th} bucket. The bucket selection procedure, consists on retrieving a number using a uniform random generator in the interval $[0...1]$. The number that falls in the i^{th} interval, gives origin to the i^{th} bucket being selected. Finally, we select a random point of the selected i^{th} bucket.

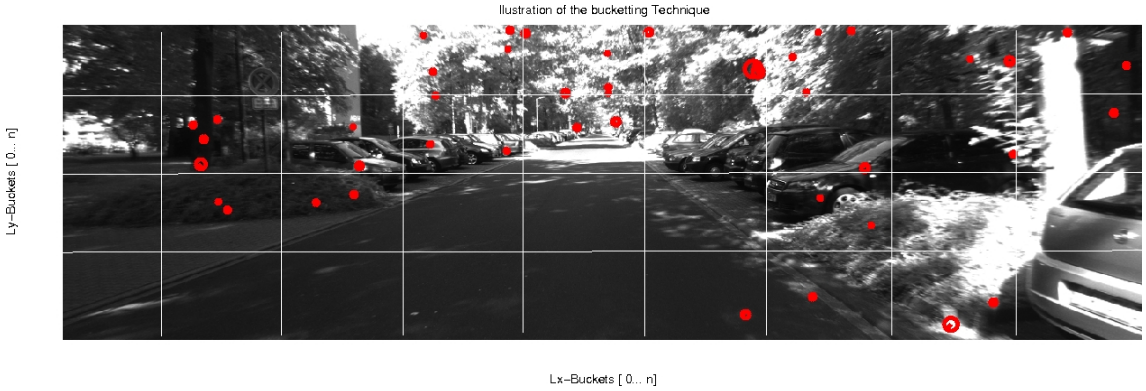

Figure 4.6: Feature detection bucketing technique used to avoid biased samples in the RANSAC method stage. The image is divided in buckets where feature points are assigned to and pulled according to the bucket probability.

4.2.4 Linear and Angular Velocity Estimation

To sum up the foregoing, we determine camera motion up to a scale factor using a probabilistic method, and by adding stereo vision combined with Procrustes estimation method, we are able to determine the missing motion scale α:

$$\alpha = \frac{\|\mathbf{t}'\|}{\|\tilde{\mathbf{t}}\|} \tag{4.11}$$

Then, the instantaneous linear velocity is given by:

$$V = \frac{\alpha\tilde{\mathbf{t}}}{\Delta T} \tag{4.12}$$

where ΔT is the sampling interval:

$$\Delta T = T_{k+1} - T_k \tag{4.13}$$

Likewise, the angular velocity is computed by:

$$W = \frac{r}{\Delta T} \tag{4.14}$$

where $r = \theta u$, the angle-axis representation of the incremental rotation R [Cra89].

Thus, using motion scale information given by the Procrustes method, we can estimate vehicle linear velocity between instants k and $k+1$. The AO orientation method is only used for linear velocity estimation (motion scale). For the angular velocity estimation we use the rotation matrix R calculated by Domke's probabilistic method, that is more accurate than the rotation obtained by the AO method.

4.2.5 Kalman Filter

In order to achieve a more robust estimation, we also use a Kalman filter to the linear and angular velocity estimates having state equation $X = [V, W]^T$, where V is the vehicle linear velocity, W is the vehicle angular velocity. The constant velocity Kalman filter [GKN+74] considers a state transition model with zero-mean stochastic acceleration:

$$X_k = FX_{k-1} + \xi_k \tag{4.15}$$

where the state transition matrix is the identity matrix, $F = I_{6x6}$, and the stochastic acceleration vector ξ_k is distributed according to a multivariate zero-mean Gaussian distribution with covariance matrix Q, $\xi_k \sim \mathcal{N}(0,Q)$. The observation model considers state observations with additive noise:

$$Y_k = HX_k + \eta_k \tag{4.16}$$

where the observation matrix H is identity, $H = I_{6x6}$, and the η_k measurement noise is zero-mean Gaussian with covariance R.

We set the covariance matrices Q and R empirically, according to our experiences, to:

$$Q = \mathrm{diag}(q_1, \cdots, q_6) \tag{4.17}$$

$$R = \mathrm{diag}(r_1, \cdots, r_6) \tag{4.18}$$

where $q_i = 10^{-3}, i = 1, \cdots, 6$, $r_3 = 10^{-3}$ and $r_i = 10^{-4}, i \neq 3$.

The r_3 differs from the other (r_i) measurement noises values, due to the fact that it corresponds to the translation on the z axis which is inherently noisier due to the uncertainty of the t_z estimates in the stereo triangulation step.

4.3 Results

In this section, we present results of 3 implementations of the 6DP method. The first experiment compares 6DP non-filtered (raw) estimates using the Harris corner detector [HS88] as the sparse feature detector, here on denoted as 6DP-raw-Harris and compares it against a native 5-point implementation. Afterwards, we present results of the other 2 implementations: (i) 6DP-raw-SIFT where we replaced the Harris corner for a more robust and invariant to scale detector (SIFT)[Low04]; (ii) 6DP-KF that also uses SIFT features but this time integrated in a Kalman Filter framework. The results of both implementations are compared with the state-of-the art visual odometry estimation method LIBVISO [KGL10] using their dataset reference (2009-09-08-drive-0021).

4.3.1 Computational Implementation

The code used to compute 6DP was written in MATLAB as a proof of concept, without using any kind of code optimization. The experiments were performed using an Intel I5 Dual Core 3.2 GHz. For the evaluation we used a section of the dataset [KGL10] reference (2009-09-08-drive-0021), which has been used for benchmarking visual odometry methods in other works against which we compare our method. During our experiments several parts of the dataset were tried and results were consistent across the dataset. The dataset images have resolution of 1344 $\times$ 391, which consumes a considerable amount of computational and memory resources ($\sim$ 0.5MB per point) making unfeasible the computation of all image points using the Matlab implementation on standard CPU hardware. Thus, the results shown in this chapter were obtained using 1000 randomly selected points in image I_k^L. The method takes about 12 sec per image pair. Most of time is consumed in the first stage of the implementation, with the dense probabilistic correspondences and the motion up to a scale factor estimates. The recursive ZNCC approach allowed to reduce Domke Gabor Filter [DA06] processing time by 20 %.

Even so, the approach is feasible and can be implemented in real-time for use on mobile robotics applications. The main option is to develop a **GPU** version of the method. Since the method deals with multiple hypothesis of correspondence, and motion, it is suitable to be implemented into parallel hardware.

4.3.2 6DP-raw-Harris vs 5-point

In this section, one can observe results comparing our approach versus the 5-point RANSAC algorithm [Nis04]. Linear and angular velocities estimation results are presented in the camera reference frame.

In Fig. 4.7, one can observe the angular velocity estimation of the 6DP method, IMU/GPS information and the 5-point RANSAC. We also show the Inertial Navigation System data (IMU/GPS OXTS RT 3003), which is considered as "ground-truth" information. The displayed results demonstrate a high degree of similarity between performance obtained using 6DP and IMU/GPS information. Results obtained by 6DP were performed without using any type of filtering technique, thus the display of one or two clear outliers. Most importantly, when it comes to angular velocities estimation, the 6DP method performance is better than the performance exhibited by the 5-point RANSAC algorithm.

However, for linear velocities as displayed in Fig. 4.8, the 5-point RANSAC algorithm implementation performance is smoother than our 6DP approach, especially in Z axis T_z. As shown in Fig. 4.10, the 5-point algorithm contains more image features when performing Procrustes Absolute Orientation method (after RANSAC) which may also explain the higher robustness in motion scale estimation in Fig. 4.9, where the 5-point algorithm displays a constant translation scale value.

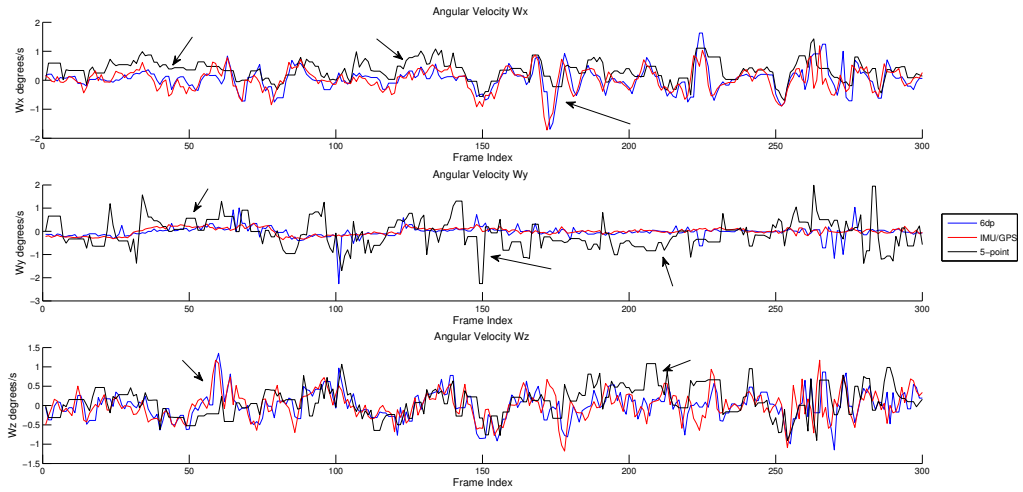

Figure 4.7: Comparison of angular velocity estimation results between IMS/GPU (red), raw 6DP measurements (blue) and a native 5-point implementation (black). The obtained 6DP raw measurements are similar to the data estimated by the IMU/GPS, contrary to the 5-point implementation that has some periods of large errors (e.g. the regions indicated with arrows in the plots).

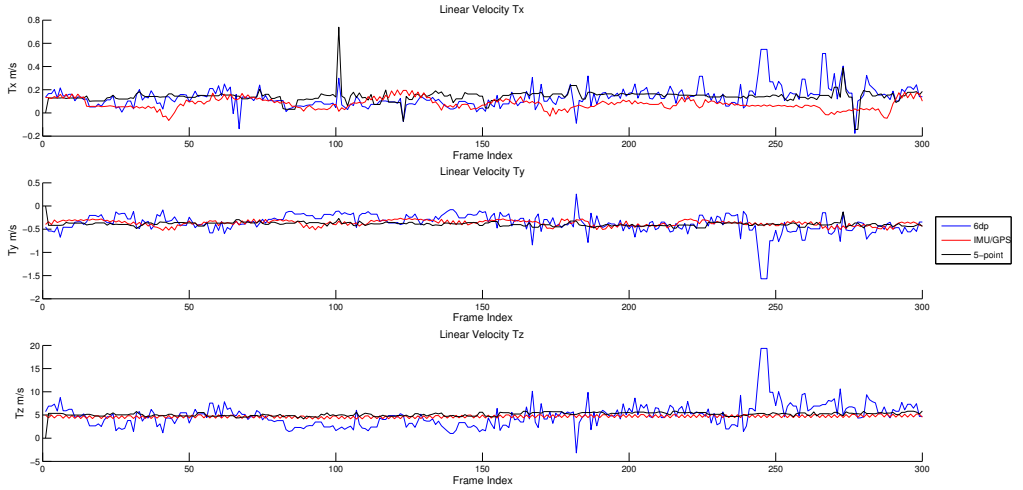

Figure 4.8: Comparison of linear velocity estimation results, where the 5-point implementation (black) exhibits a closer match to the IMU/GPS information (red). The 6DP method (blue) displays some highlighted outliers due to the use of the Harris feature detection matching in the sparse method stage.

The results demonstrate complementary performances, one more suitable for linear motion estimation and the other more suitable for angular motion estimation.

4.3.3 6DP-raw-Harris vs 6DP-raw-SIFT

The obtained results using 6DP-raw-Harris in the translation scale (α) estimation were not sufficiently accurate, mostly due to the use of the Harris corner detector. We modified the 6DP

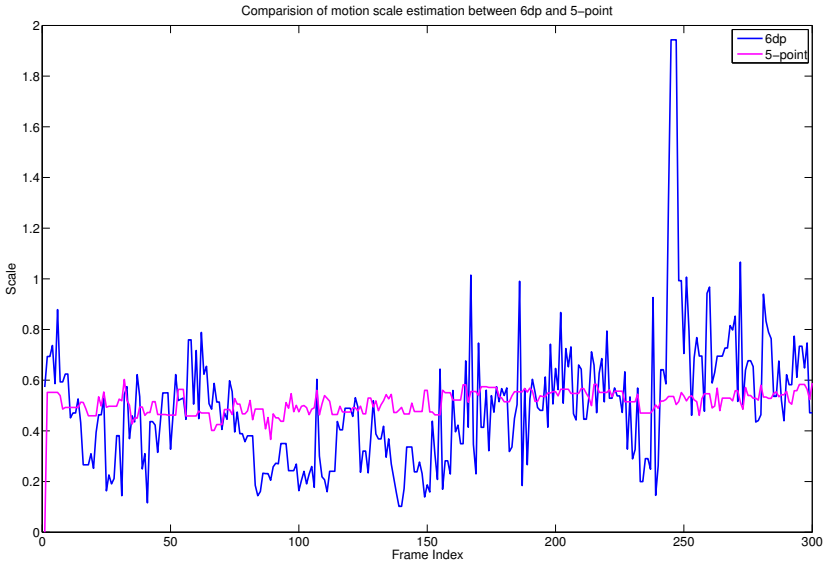

Figure 4.9: Translation scale factor comparison between 5-point and 6DP-raw-Harris, where the 5-point method exhibits a more constant behavior for the translation scale factor estimation.

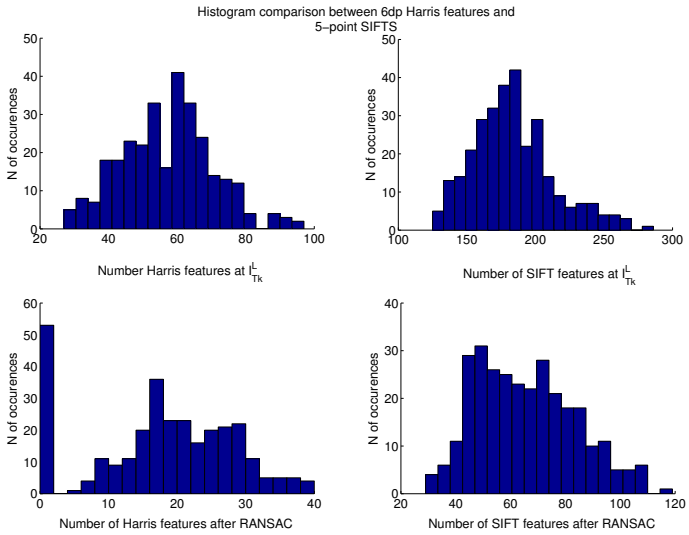

Figure 4.10: Number of Features at different steps of 6DP-raw-Harris and 5-point. SIFT features display a more robust matching behavior between images. Contrary to Harris Corners, most of the SIFTS are not eliminated in the RANSAC stage.

method, by replacing the Harris corner feature detector [HS88] for the more robust and invariant to rotation and scale SIFT detector [Low04]. We can observe in figure 4.10 that SIFT features are more stable after the RANSAC step when compared to the Harris corner approach, and thus can provide more accurate point correspondence between I_k^L and I_{k+1}^L.

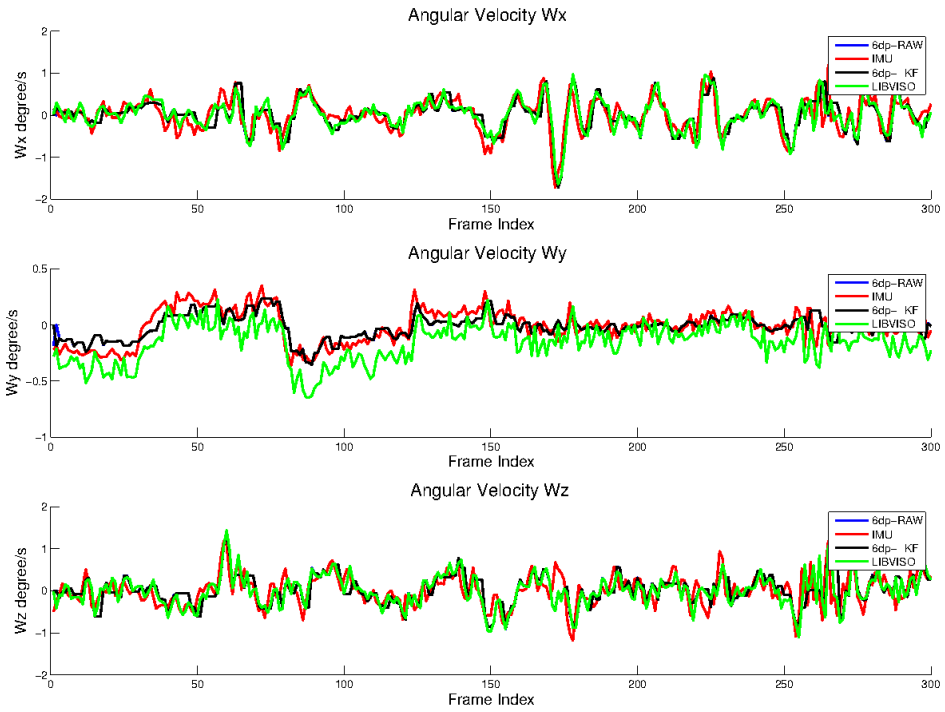

Figure 4.11: Results for angular velocities estimation between IMU/GPS information (red), raw 6DP measurements 6DP-raw-SIFTS (blue), filtered 6DP measurements 6DP-KF (black), and 6D Visual Odometry Library LIBVISO (green). Even though all exhibit similar behaviors the filtered implementation 6DP-KF is the one which is closer to the "ground truth" IMU/GPS measurements (see also Table 1).

4.3.4 6DP-KF vs LIBVISO

To illustrate the performance of the 6DP-KF method, we compared our system performance against LIBVISO [KGL10], which is a standard library for computing 6-DOF visual Odometry. We also compared our performance against IMU/GPS acting as ground truth information using the same Kitt *et al.*[KGL10] Karlsruhe dataset sequences.

In Fig. 5.18 one can observe angular velocity estimation from both IMU/GPS and LIBVISO, together with 6DP-raw-SIFT and 6DP-KF filtered measurements. All approaches obtained results consistent with the IMU/GPS, but the 6DP-KF displays a better performance with respect to angular velocities. These results are stated in Table 5.3, where root mean square error between IMU/GPS, LIBVISO and 6DP-KF estimation are displayed. The 6DP-KF method shows 50% lower error than LIBVISO for the angular velocities estimation.

Although not as good as for the angular velocities, the 6DP-KF method also displays a better performance in obtaining linear velocity estimates as displayed in Fig. 5.19 and in Table 1. Overall, our 6DP-KF shows an important precision improvement over LIBVISO.

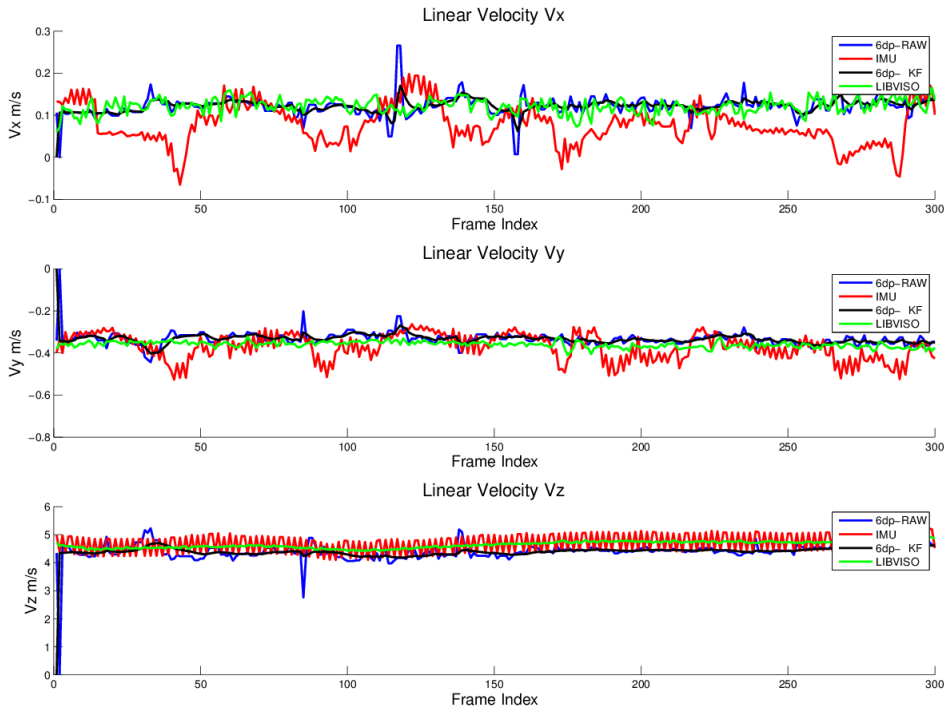

Figure 4.12: Results for linear velocities estimation, where the LIBVISO implementation and 6DP-KF display similar performance when compared to IMU/GPS performance.

Table 4.1: Standard Mean Squared Error between IMU and Visual Odometry (LIBVISO and 6DP-KF). The displayed results show a significant improvement of the 6DP-KF method performance specially in the angular velocities estimation case.

	V_x	V_y	V_z	W_x	W_y	W_z	$\|\|V\|\|$	$\|\|W\|\|$
LIBVISO	0.0674	0.7353	0.3186	0.0127	0.0059	0.0117	1.1213	0.0303
6DP-KF	0.0884	0.0748	0.7789	0.0049	0.0021	0.0056	0.9421	0.0126

4.4 Summary

In this chapter, we developed a novel method of stereo visual odometry using sparse and dense egomotion estimation methods. We utilized dense egomotion estimation methods for estimating the rotation and translation up to scale and then complement the method with the use of a sparse feature approach for recovering the scale factor.

First, we compared the raw estimates of our 6DP-raw-Harris algorithm against a native 5-point implementation without any type of filtering. The results obtained proved that 6DP-raw-Harris performed better in the angular velocities estimation but compared unfavorably in the linear velocities estimation due to lack of robustness in the translation scale factor(α) estimation. On a second implementation, we replaced the Harris feature detector with the more robust SIFT detector, implemented a Kalman filter on top of the raw estimates and tested the proposed algorithm against a

state-of-the-art 6D visual Odometry Library such as LIBVISO. The presented results demonstrate that 6DP-KF performs more accurately when compared to other techniques for stereo VO estimation, yielding robust motion estimation results, most notably in the angular velocities. However, we were unable to achieve a significant improvement in the linear velocities estimation mainly because it uses deterministic approaches, and therefore there we need to develop a fully stereo probabilistic egomotion method.

4.5 Related Publications

The work presented in this chapter, related to sparse and dense approach to stereo egomotion estimation was initially published in [SBS13b] Computer Vision and Applications Conference in Barcelona, February 2013 and in [SBS13a] Autonomous Mobile Robotics Conference and Competitions in April 2013). It was later invited for a special issue in Springer Journal of Intelligent Robotics Systems where it has been accepted for publishing.

5

Probabilistic Stereo Egomotion Transform

5.1 Introduction

Most approaches to the stereo egomotion estimation problem, rely on non-probabilistic correspondences methods. Common approaches try to detect, match, and track key points between images on adjacent time frames and afterwards use the largest subset of point correspondences that yield a consistent motion. In probabilistic correspondence methods matches are not fully committed during the initial phases of the algorithm and multiple matching hypotheses are accounted for. Our previous work in egomotion estimation (6DP)[SBS13a][SBS13b], described in the previous chapter has shown that probabilistic correspondence methods are a viable way to estimate egomotion with advantages in precision over classical feature based methods. Nevertheless 6DP method was unable to estimate the translation scale factor based only on probabilistic approaches, and required a mixed approach to be able to recover all motion parameters.

In this chapter, we develop a novel probabilistic stereo egomotion method (PSET) capable of computing 6-DOF motion parameters solely based on probabilistic correspondence approaches, and without the need to track or commit key point matches between consecutive frames. The use of probabilistic correspondence methods allows to maintain several match hypothesis for each point, which is an advantage when there are ambiguous matches (which is the rule in image feature correspondences problems), because no commitment is made before analyzing all image information. Another advantage is that a full probabilistic distribution of motion provides a better sensor fusion with other sensors, e.g. inertial.

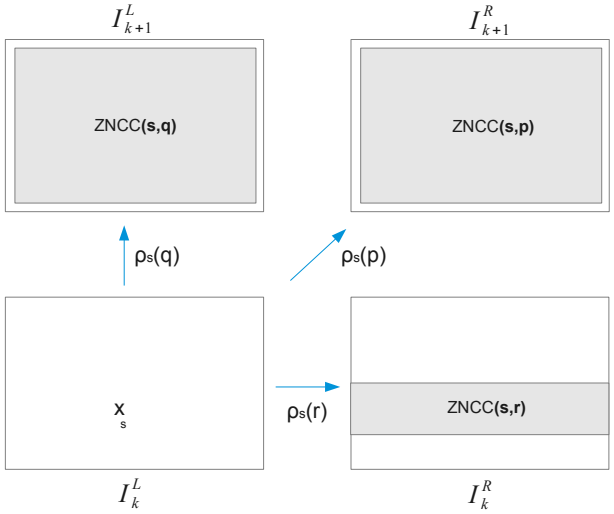

Figure 5.1: ZNCC matching used to compute the PSET transform

The work presented in this chapter improves the work conducted in [SBS13a], [SBS13b] and proposes a fully probabilistic algorithm to perform stereo egomotion estimation, which we denote as Probabilistic Stereo Egomotion Transform (PSET). While in 6DP [SBS13a], a mixed probabilistic and deterministic approach was used to estimate rotation and translation parameters, PSET only employs probabilistic correspondences. The rotation estimation is achieved the same way as in 6DP (with a 5D search over the motion space based on the notion of epipolar constraint), yet the translation scale factor is obtained by exploiting an accumulator array voting scheme based also on epipolar stereo geometry combined with probabilistic distribution hypotheses between the two adjacent stereo image pairs. The obtained results demonstrate a clear performance improvement in the estimation of the linear and angular velocities over current state-of-the-art stereo egomotion estimation methods, when compared to Inertial Measurement Unit ground-truth information. Furthermore, since real-time is a concern in today modern mobile robotics applications the algorithm can be easily implemented using a multi-core architecture.

5.2 Probabilistic Stereo Egomotion Estimation

In this chapter we extend the notion of probabilistic correspondence and probabilistic egomotion estimation already presented in the previous chapter (4.2.1, 4.2.2) to the stereo case, which allow us to compute the whole 6D motion information in a probabilistic way. In a stereo setup we consider images I^L_k, I^L_{k+1}, I^R_k and I^R_{k+1}, where superscripts L and R denote respectively the left and right images of the stereo pair. Probabilistic matches of a point $\mathbf{s}$ in I^L_k are now computed not only for points $\mathbf{q}$ in I^L_{k+1} but also for points $\mathbf{r}$ in I^R_k and $\mathbf{p}$ in I^R_{k+1} (see figure 5.1 and also figure 5.2):

$$\rho_s(r) = \frac{ZNCC(s,r)+1}{2} \tag{5.1}$$

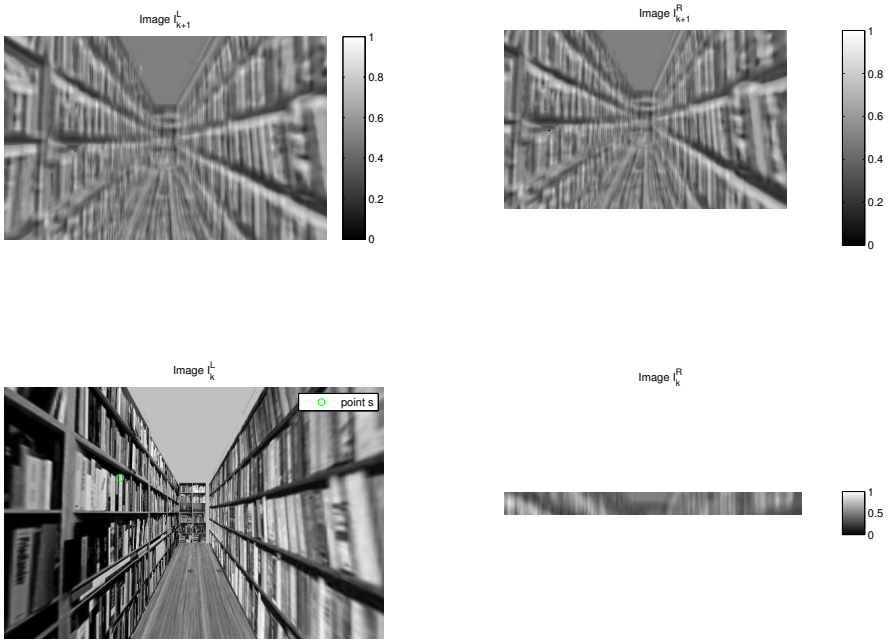

Figure 5.2: Example of probabilistic correspondence ($\rho_s(r)$, $\rho_s(q)$, $\rho_s(q)$) obtained by ZNCC matching for a given point **s** for an image triplet (I_k^R,I_{k+1}^L,I_{k+1}^R)

$$\rho_s(p) = \frac{ZNCC(s,p)+1}{2} \tag{5.2}$$

For the sake of computational efficiency, analysis can be limited to sub-regions of the images given prior knowledge about the geometry of the stereo system or the motion given by other sensors like IMU's. In particular, for each point s, coordinates $\mathbf{r}$ can be limited to a band around the epipolar lines according to the stereo setup epipolar geometry.

5.2.1 The Geometry of Stereo Egomotion

In this section we describe the geometry of the stereo egomotion problem, i.e. analyze how world points project in the four images acquired from the stereo setup in two consecutive instants of time according to its motion. This analysis is required to derive the expressions to compute the translational scale factor.

Let us consider the 4×4 rototranslations T_L^R and M_k^{k+1} that describe, respectively, the rigid transformation between the left and right cameras of the stereo setup, and the transformation describing the motion of the left camera from time k to $k+1$:

$$T_L^R = \begin{bmatrix} R_L^R & \mathbf{t_L^R} \\ 0 & 1 \end{bmatrix} \quad M_k^{k+1} = \begin{bmatrix} R_k^{k+1} & \mathbf{t_k^{k+1}} \\ 0 & 1 \end{bmatrix} \tag{5.3}$$

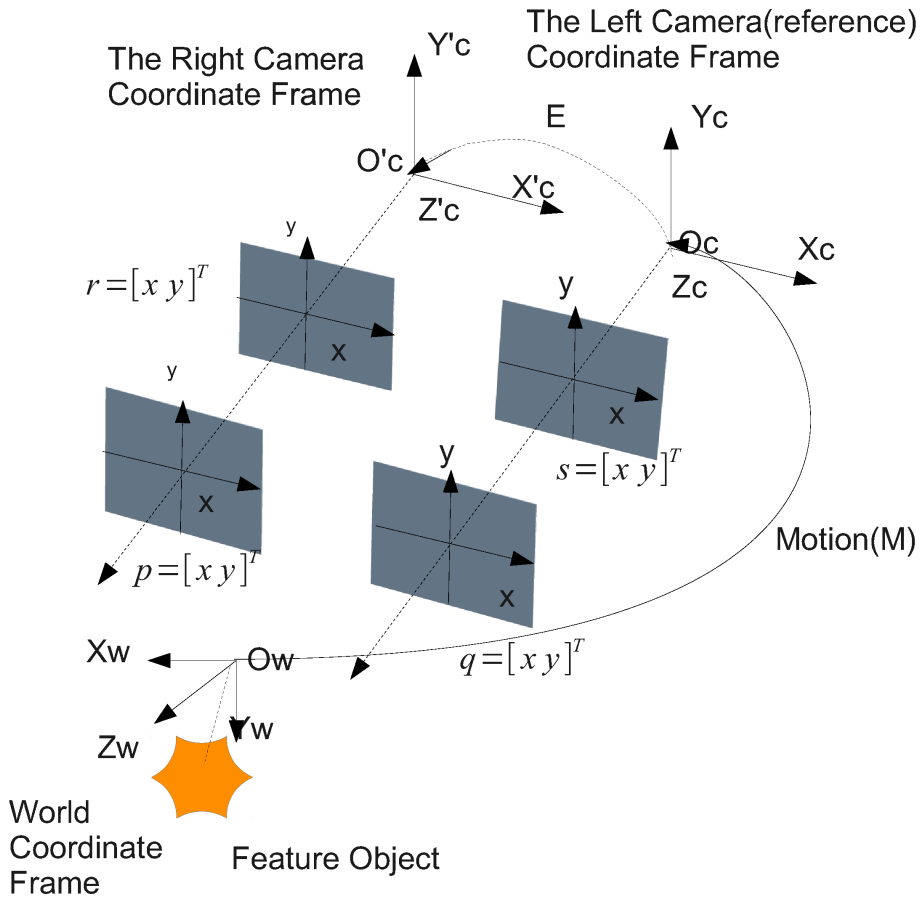

Figure 5.3: Stereo Egomotion Geometry

where $R_:$ and $\mathbf{t}_:$ denote the rotational and translational components. We factorize the translational motion $\mathbf{t}$ in its direction $\hat{\mathbf{t}}$ and amplitude α:

$$\mathbf{t} = \alpha\hat{\mathbf{t}} \tag{5.4}$$

Given that rotational motion and translation direction are computed by the method described in the previous section, the computation of α is the objective to pursue.

Let us consider an arbitrary 3D point $\mathbf{X} = (X_x, X_y, X_z)^T$ expressed in the reference frame of the left camera at time k. Considering normalized intrinsic parameters (unit focal distance $f = 1$, zero central point $c_x = c_y = 0$, no skew), the homogeneous coordinates of the projection of $\mathbf{X}$ in the 4 images is given by:

$$\begin{cases} \tilde{\mathbf{s}} = \mathbf{X} \\ \tilde{\mathbf{r}} = R_L^R\mathbf{X} + \mathbf{t_L^R} \\ \tilde{\mathbf{q}} = R_k^{k+1}\mathbf{X} + \alpha\hat{\mathbf{t}} \\ \tilde{\mathbf{p}} = R_L^R R_k^{k+1}\mathbf{X} + \alpha R_L^R\hat{t} + \mathbf{t_L^R} \end{cases} \tag{5.5}$$

To illustrate the solution, let us consider the particular case of parallel stereo. This will allow us to obtain the form of the solution with simple equations but does not compromise generality because the procedure to obtain the solution in the non parallel case is analogous. In parallel

stereo the cameras are displaced laterally with no rotation. The rotation component is the 3×3 identity ($R_L^R = I_{3\times3}$) and the translation vector is an offset (baseline b) along the x coordinate, $\mathbf{t_L^R} = (b, 0, 0)^T$. In this case, expanding the equations for $\mathbf{s} = (s_x, s_y)^T$ and $\mathbf{r} = (r_x, r_y)^T$ we obtain:

$$\begin{cases} s_x = \frac{X_x}{X_z} \\ s_y = r_y = \frac{X_y}{X_z} \\ r_x = \frac{(X_x+b)}{X_z} \end{cases} \tag{5.6}$$

Introducing the disparity d as $d = r_x - s_x$ we have $d = \frac{b}{X_z}$ and we can reconstruct the 3D coordinates of point $\mathbf{X}$ as a function of the image coordinates $\mathbf{r}$ and $\mathbf{s}$ and the known baseline value b:

$$\mathbf{X} = \begin{pmatrix} \frac{s_x b}{d} & \frac{s_y b}{d} & \frac{b}{d} \end{pmatrix}^T \tag{5.7}$$

Replacing this value now in (5.5) we obtain:

$$\mathbf{r} = \begin{bmatrix} \frac{(\frac{s_x b}{d}+b)d}{b} \\ s_y \end{bmatrix} \tag{5.8}$$

$$\mathbf{q} = \begin{bmatrix} \frac{r11 s_x b + r12 s_y b + r13 b + \alpha t_x d}{r31 s_x b + r32 s_y b + r33 b + \alpha t_z d} \\ \\ \frac{r21 s_x b + r22 s_y b + r23 b + \alpha t_y d}{r31 s_x b + r32 s_y b + r33 b + \alpha t_z d} \end{bmatrix} \tag{5.9}$$

$$\mathbf{p} = \begin{bmatrix} \frac{r11 s_x b + r12 s_y b + r13 b + \alpha t_x d + bd}{r31 s_x b + r32 s_y b + r33 b + \alpha t_z d} \\ \\ \frac{r21 s_x b + r22 s_y b + r23 b + \alpha t_y d}{r31 s_x b + r32 s_y b + r33 b + \alpha t_z d} \end{bmatrix} \tag{5.10}$$

We determine the translation scale factor α, using (5.9) by:

$$q_x = \frac{\overbrace{r11 s_x b + r12 s_y b + r13 b}^{\text{A}} + \alpha t_x d}{\underbrace{r31 s_x b + r32 s_y b + r33 b}_{\text{C}} + \alpha t_z d} \tag{5.11}$$

being α given by:

$$\alpha = \frac{A - q_x C}{q_x t_z d - t_x d} \tag{5.12}$$

The same procedure is applied to q_y:

$$q_y = \frac{\overbrace{r21 s_x b + r22 s_y b + r23 b}^{\text{B}} + \alpha t_y d}{\underbrace{r31 s_x b + r32 s_y b + r33 b}_{\text{C}} + \alpha t_z d} \tag{5.13}$$

being α given by:

$$\alpha = \frac{B - q_y C}{q_y t_z d - t_y d} \tag{5.14}$$

The translation scale factor alpha can also be determine using point p coordinates by:

$$p_x = \frac{\overbrace{r11s_xb + r12s_yb + r13b}^{\text{A}} + \alpha t_x d + bd}{\underbrace{r31s_xb + r32s_yb + r33b}_{\text{C}} + \alpha t_z d} \tag{5.15}$$

being α given by:

$$\alpha = \frac{A + bd - p_x C}{p_x t_z d - t_x d} \tag{5.16}$$

The same procedure is applied to p_y:

$$p_y = \frac{\overbrace{r21s_xb + r22s_yb + r23b}^{\text{B}} + \alpha t_y d}{\underbrace{r31s_xb + r32s_yb + r33b}_{\text{C}} + \alpha t_z d} \tag{5.17}$$

being α given by:

$$\alpha = \frac{B - p_y C}{p_y t_z d - t_y d} \tag{5.18}$$

Therefore, being α an over-determined parameter since there are four equations to one unknown, we choose the α with the highest denominator to minimize the effect of numerical errors. In case both denominators are low due to very low disparity or degenerate motions, this particular point can not be used for the estimation.

5.2.1.A Degenerate Cases

In some cases for a given point $\mathbf{s}$ it is not possible to determine the translation scale factor α. This occurs when there is not enough disparity between point $\mathbf{s}$ in I_k^L and probabilistic correspondence hypotheses $\mathbf{r}$ in I_k^R, and $d \approx 0$. The denominator of equations 5.12, 5.14, 5.16, 5.18 tends to zero, and α becomes undetermined. In our implementation we empirically set a predetermined minimal value for d, $d \geq d_{min}$, below which point $\mathbf{s}$ is not used in the following implementation steps. Other undetermined translation scale factor α case, is when a degenerate motion occurs. If motion E is imprecisely determined it will be difficult to correctly determine the probabilistic correspondence points $\mathbf{q}$ and $\mathbf{p}$ in time $k+1$, and therefore α becomes undetermined. In both cases, motion can only be determine up to an unknown scale factor.

5.2.2 Translational Scale Estimation

In the previous section we have seen that it is possible to estimate the translational scale α from the observation of a single static point $\mathbf{s}$, if point correspondences $\mathbf{r}, \mathbf{q}$ and $\mathbf{p}$ are known and there are no degeneracies. In practice, two major problems arise: (i) it is hard to determine what are the static points in the environment given that the cameras are also moving; and (ii) it is very hard to obtain reliable matches due to the noise and ambiguities present in natural images.

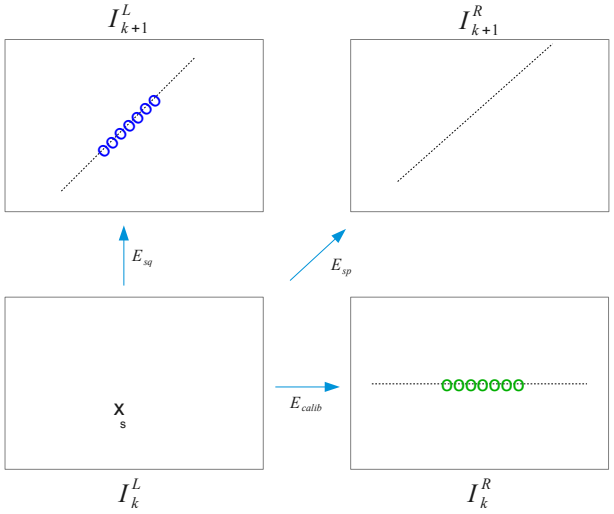

Figure 5.4: Point correspondence hypotheses along the epipolar lines

Therefore using a single point to perform this estimation is doomed to failure. We must therefore use multiple points and apply robust methodologies to discard outliers.

In [SBS13a], this was achieved by computing the rigid transformation between point clouds obtained from stereo reconstruction at times k and $k+1$ with a robust method RANSAC. Point correspondences were deterministically assigned by searching for the best matches along epipolar lines in space (from camera L to camera R) and time (from time k to time $k+1$) see figure 5.4.

In PSET, we extend the probabilistic notion of correspondence to the stereo case. Instead of deterministically committing to matches in space and time, we create a probabilistic observation model for possible matches:

$$P_{match}(s,r,p,q) = \rho_s(r)\rho_s(q)\rho_s(p) \tag{5.19}$$

where we assume statistical independence in the measurements obtained in the pairwise probabilistic correspondence functions $\rho_s(\cdot)$, as shown in figure 5.5 for the $\rho_s(r)$ case.

From the pairwise probabilistic correspondence, we obtain all possible combination of corresponding matches. Then, because each possible match (s,r,p,q) will correspond to a value of α, we will create an accumulator of α hypotheses, weighted by $P_{match}(s,r,p,q)$. Searching for peaks in the accumulator will provide us the best (most agreed) hypothesis for α given all the information in the images.

5.2.3 PSET Accumulator

Here we detail how the method is implemented computationally. We assume E has been computed by the methods described previously and the system calibration is known.

First a large set of points $s_j, j = 1 \cdots J$ is selected. Selection can be random, uniform or based

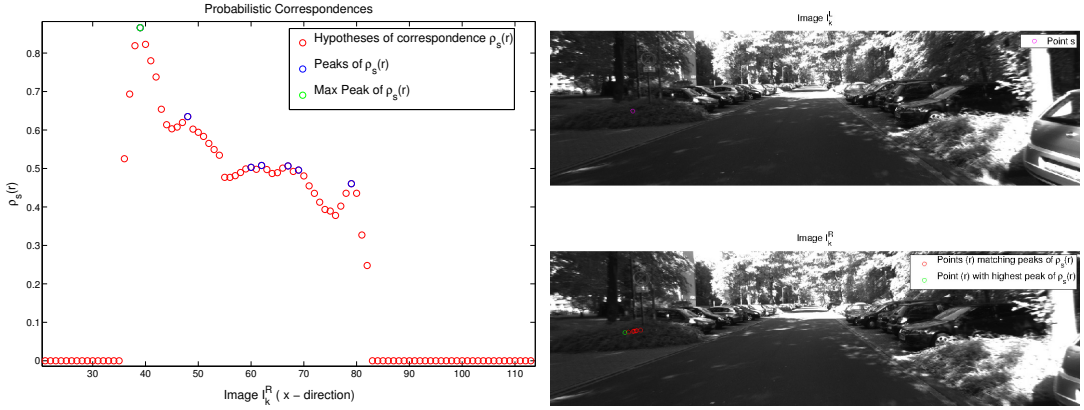

Figure 5.5: Probabilistic correspondence $\rho_s(r)$ for a point **s** along the epipolar line E_{sr}. In the left hand side figure, it is shown all known hypotheses (red), the local maximum probabilistic correspondences (peaks) of $\rho_s(r)$ (blue), and the global maximum of $\rho_s(r)$ (green). On the right hand side figure, we see sample point **s** in I_k^L and the local maximum (peaks) probabilistic correspondences represented in I_k^R

on key points, e.g. Harris corners [HS88] or Scale-Invariant features [Low04].

For each point s_j, the epipolar lines $E_{calib} = \tilde{\mathbf{s}}_j^T S$ and $E_{sq} = \tilde{\mathbf{s}}_j^T E$ are sampled at points r_l^j and q_m^j, in images I_k^R and I_{k+1}^L, respectively. Again sample point selection can be uniform along the epipolar lines or based on match quality. In our implementation we compute local maxima of match quality over the epipolar lines.

At this point we create a PSET table (see figure 5.6). Having $j = 1...J$ we obtain a 2D table $H_j(l,m)$, $l = 1...L_j, m = 1...M_j$, in order to associate to each triplet (s_j, r_l^j, q_m^j) a disparity value d_{jl} and a scale value α_{jlm}, determined by either (5.12) or (5.14). Given this information the value of p becomes uniquely determined by (5.10) and is stored as p^{jlm}, the unique match in the right camera time k+1 corresponding to s_j, r_l^j, q_m^j. The likelihood of this triplet is then computed by:

$$\lambda_{jlm} = \rho_{s_j}(r_l^j)\rho_{s_j}(q_m^j)\rho_{s_j}(p^{jlm}) \tag{5.20}$$

After each PSET table entries $H^j(l,m)$ has been filled with points s_j information,$d_{jl}, \alpha_{jlm}, \lambda_{jlm}$, the global maximum of the associated weights $\max \lambda_{jlm}$ for each s_j are selected by:

$$(l_{\max}^j, m_{\max}^j) = argmax\lambda_{jlm} \tag{5.21}$$

Thus, each point s_j votes for a certain motion scale factor, according the most agreeing matches in all other images.

Finally, the α_{jlm} values associated to the $\max \lambda_{jlm}$ of each s_j, $\alpha_{jl_{max}^j, m_{max}^j}$, are chosen, and a method for estimating the highest density of α votes [Par62] is used to determine the missing translation scale factor.

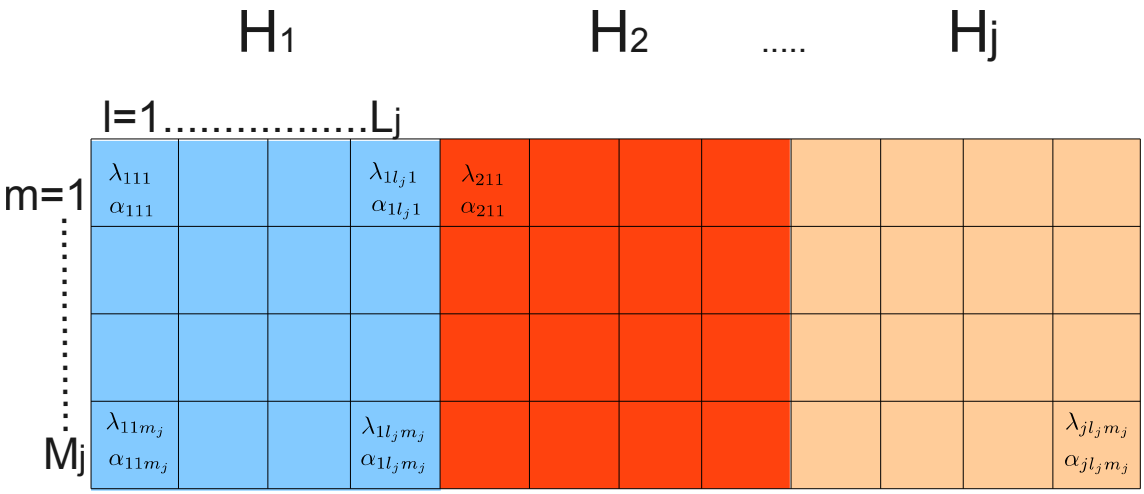

Figure 5.6: PSET H_j 2D table accumulator

5.2.4 Dealing with calibration errors

A common source of errors in a camera stereo setup is the uncertainty in the calibration parameters. Both intrinsic and extrinsic parameter errors will deviate the epipolar lines from their nominal values and influence the computed correspondence probability values. To minimize these effects we modify the correspondence probability function when evaluating sample points such that a neighborhood of the point is analyzed and not only the exact coordinate of the sample point:

$$\rho'_s(q) = \max_{q' \in \mathcal{N}(q)} \left[\rho_s(q') \exp \frac{(q-q')^2}{2\sigma^2} \right] \tag{5.22}$$

where $N(q)$ denotes a neighborhood of the sample point q which, in our experiments, is a 7×7 window.

Other method used to diminish the uncertainty of the correspondence probability function when performing Zero Normalized Cross Correlation (**ZNCC**) is to use sub-pixel refinement. In [DGK11], four methods are presented to perform sub-pixel refinement of the normalized cross correlation computation. Despite bi-cubic intensity and bi-cubic convolution present better accuracy on the datasets experiments performed in [DGK11], they also consume more computational resources and their processing time is up to four times slower than the other two methods that were implemented (parabola fitting and 2D gaussian fitting). Based on these assumptions, we opted to use only parabola and 2D gaussian fitting.

When using parabola fitting, the shape of the correlation surface fits two orthogonal parabolic curves. Furthermore, the location of the real (maximum or peak) is computed by independently fitting 1D quadratic function that helps compute the location of the peak. Let's us assume we already computed the peak $\rho'_s(q)$ and it has integer position $[x_0, y_0]$. The pixel position is surrounded by two neighbors in each of its cartesian orthogonal direction: x-direction

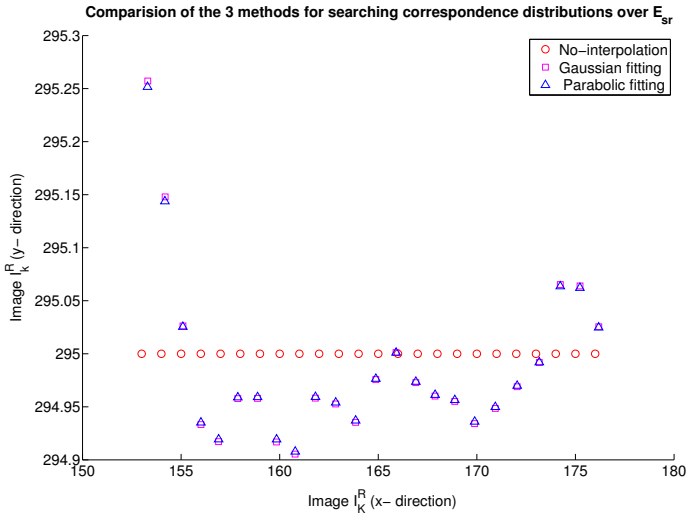

Figure 5.7: Epipolar lines on I_k^R computed by the different fitting methods i.e no-interpolation, parabolic fitting and gaussian fitting

$(x_0 - 1, x_0 + 1)$ and y-direction $(y_0 - 1, y_0 + 1)$. The sub-pixel peak position in each direction is given by $(x_0 + \Delta X, y_0 + \Delta Y)$. Afterwards, a parabolic curve connecting the three poins of that direction is defined by us and the positions where the curve attains the peak is computed. By equations (5.23) and (5.24) is possible to obtain the non-integer location of the peak which will be added to the already known integer location of the peak, thus obtaining the "real" peak location by:

$$\Delta X = \frac{\rho(x_0 - 1, y_0) - \rho(x_0 + 1, y_0)}{2\rho(x_0 - 1, y_0) - 4\rho(x_0, y_0) + 2\rho(x_0 + 1, y_0)} \tag{5.23}$$

For the y direction, we have:

$$\Delta Y = \frac{\rho(x_0, y_0 - 1) - \rho(x_0, y_0 + 1)}{2\rho(x_0, y_0 - 1) - 4\rho(x_0, y_0) + 2\rho(x_0, y_0 + 1)} \tag{5.24}$$

Analogously for the gaussian fitting case, the sub-pixel precision is computed by fitting a second-order polynomial to the logarithm of the peak and of its direct neighbors as expressed by:

$$\Delta X = \frac{\ln(\rho(x_0 - 1, y_0)) - \ln(\rho(x_0 + 1, y_0))}{2\ln(\rho(x_0 - 1, y_0)) - 4\ln(\rho(x_0, y_0)) + 2\ln(\rho(x_0 + 1, y_0))} \tag{5.25}$$

For the y direction, we have:

$$\Delta Y = \frac{\ln(\rho(x_0, y_0 - 1)) - \ln(\rho(x_0, y_0 + 1))}{2\ln(\rho(x_0, y_0 - 1)) - 4\ln(\rho(x_0, y_0)) + 2\ln(\rho(x_0, y_0 + 1))} \tag{5.26}$$

In figure 5.7, an example for a given sample point **s** is shown, it is possible to see the different correspondences over the epipolar lines using all of three methods (no interpolation, gaussian fitting and parabolic fitting).

Figure 5.8: Image used in the synthetic image sequence to perform egomotion estimation

Using the No-interpolation method and assuming a calibrated camera stereo setup, the epipolar line E_{sr} in I_k^R is strictly horizontal, while by performing parabolic and gaussian fitting the best match coordinates display an oscillate behavior, hence diminishing the uncertainty error associated with the computation of each sample point to be less then a 1 pixel.

Having determined the translation scale factor(α), and the results of the rotation (R) and translation $(\hat{t})$, we can estimate the linear and angular velocities as described in the previous chapter (4.2.4, 4.2.5).

5.2.5 Synthetic Image Sequences

As a first test for evaluating the egomotion estimation accuracy of the PSET method, we utilized a sequence of synthetic stereo images. These sequences were created using the **ISR Vislab Simulator**, and they implement a quite difficult scene in which to test egomotion estimation, the images contain a great deal of repetitive structure that cause ambiguity in image point correspondence, as can be observed in figure 5.8.

The sequence is a translation sequence in all 3 axes (x, y, z) at different velocities, where we have different translation scale factor α values in order to evaluate PSET and LIBVISO [KGL10] results in comparison with ground-truth information.

5.2.5.A Computational Implementation

For the synthetic image sequence, we assume a stereo camera pair calibrated setup constituted by a $10cm$ baseline, 576 $\times$ 380 image resolution, with **ZNCC** window $N_w = 7$. For computational reasons, we used 1000 uniform selected points s_j for the dense probabilistic egomotion estimation, and only a subgroup of 100 points of s_j in r_l^j and q_m^j are used for computing the PSET accumulator. If probabilistic correspondence were to be computed for the entire $(I_k^R, I_{k+1}^L, I_{k+1}^R)$ images with would require $\sim 0.5MB$ per point s_j. The PSET method takes about 20 seconds to compute stereo egomotion per image pair.

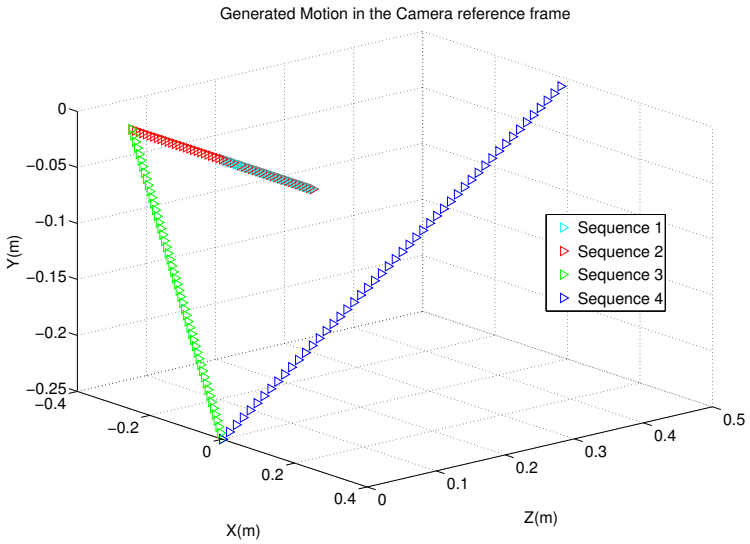

Figure 5.9: Generated Motion Trajectory computed by the VISLAB simulator to evaluate PSET egomotion accuracy while undergoing a pure translational movement in all 3 axes.

Table 5.1: Synthetic image sequences ground truth information

	Axis	Direction	Scale (mm/frame)	Distance Travelled (cm)
Sequence 1	X	right	0.005	0.25
Sequence 2	X	left	0.010	0.50
Sequence 3	YX	down-right	0.010	0.25/0.25
Sequence 4	YZ	up-forward	0.005	0.25/0.50

5.2.5.B Motion Setup

The first experiment for evaluating the accuracy of PSET method, is based on the following sequencial motions. The stereo pair starts by undergoing a translation in the x axis (sequence 1), followed by another translation in the same x axis, but this time performed with 2$\times$ velocity and conducted in the opposite direction (sequence 2). The third sequence is a diagonal translation in both x and y axes with the stereo camera pair moving down and to the right (sequence 3), and finally we have a combined upward and forward translation sequence on both the y and z axes (sequence 4). The 4 sequences are shown in figure 5.9, and in table 5.1.

5.2.5.C Qualitative Analysis

In this section, we present a qualitative analysis of the egomotion estimation for the sequences presented in figure 5.9. We compared the PSET egomotion estimation accuracy with the one obtained using LIBVISO [KGL10] versus known ground-truth information obtained from the simulator. In figures 5.10, 5.11, 5.12, 5.13 we can observe the results of the egomotion estimation between consecutive time frames. It displays both PSET and LIBVISO results in comparison with ground truth information on all 4 translational motion sequences in all 3 coordinate axes. It is possible to state from the referred figures that PSET displays a more stable performance than the

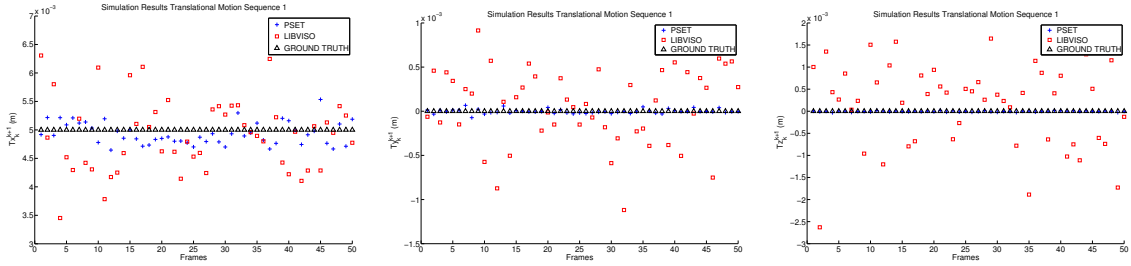

Figure 5.10: Sequence 1 translational motion in the x axis corresponding to a stereo camera pair movement to the right

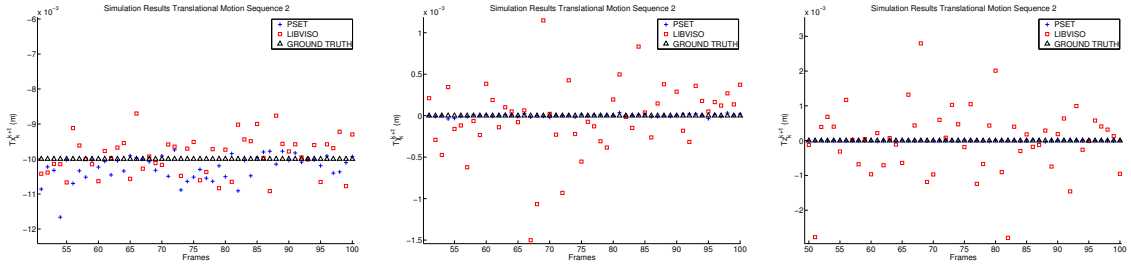

Figure 5.11: Sequence 2 translational motion in the x axis in the opposite direction at double velocity

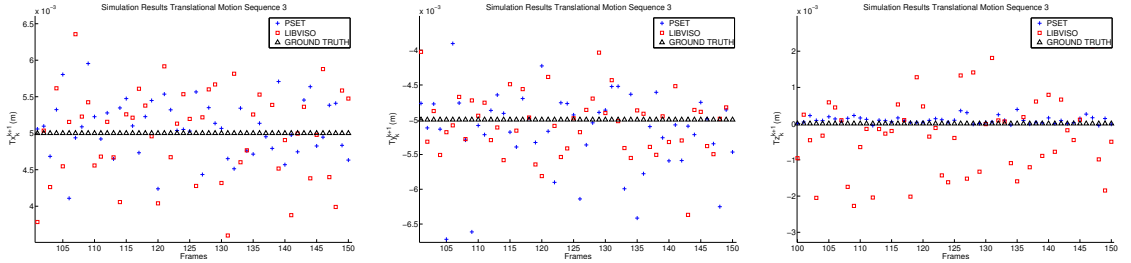

Figure 5.12: Sequence 3 translational movement in the x axis and y axis, that corresponds to a left-right downwards diagonal movement

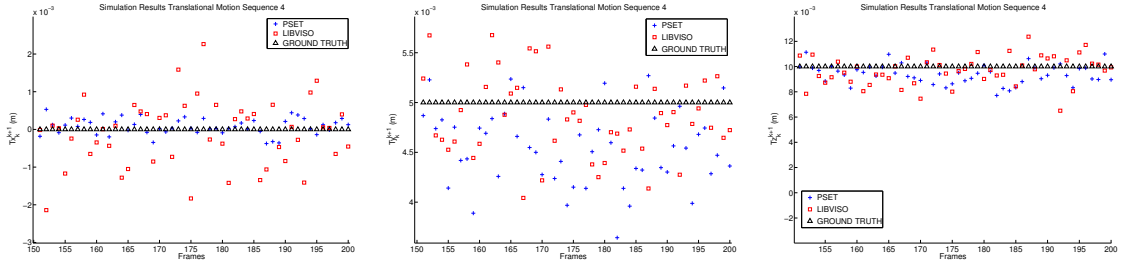

Figure 5.13: Sequence 4 translational movement in the y axis and z axis, that corresponds to a frontal upward movement

one exhibited by LIBVISO when compared to the ground-truth information.

The better performance obtained by PSET compared to LIBVISO is clear, if one computes the overall trajectories from integrating both methods egomotion estimations, as shown in figure 5.14.

The same global trajectory using a different viewpoint (zoom topview) is also displayed in figure

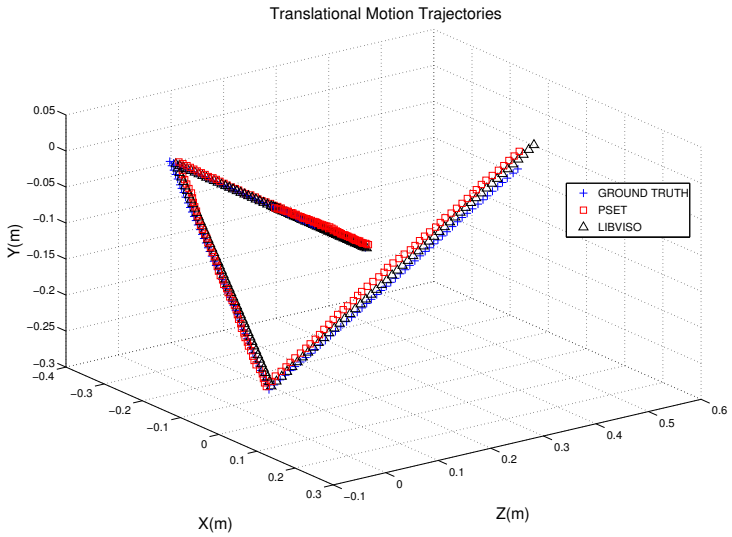

Figure 5.14: Generated Motion Trajectory used in the sinthetic image translational motion experiment using PSET (blue), LIBVISO (red) and ground-truth (black) information

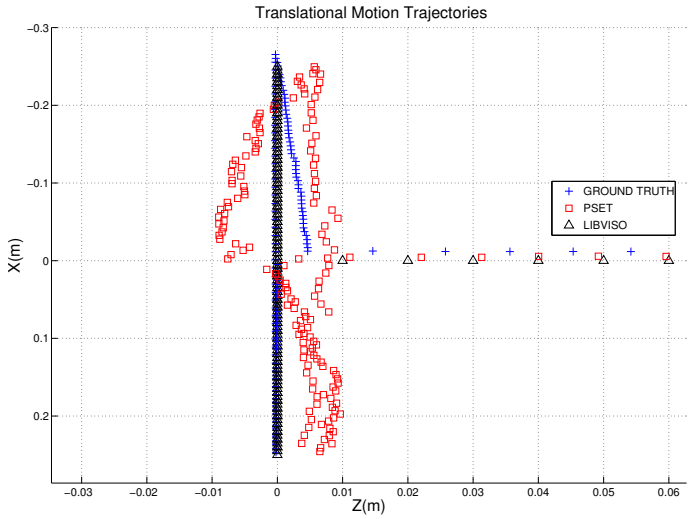

Figure 5.15: Zoom Top view of the global translational motion trajectory using PSET, LIBVISO and ground-truth information

5.15. Despite both methods similar performance with ground-truth information, it is clear from the figure that PSET displays a better overall accuracy in egomotion estimation.

5.2.5.D Motion Quantitative Analysis

In this section we present a quantitative analysis of the accuracy on egomotion estimation while performing the global trajectory presented in figure 5.14 by both PSET and LIBVISO methods.

From table 5.2 it is possible to infer that PSET displays a more accurate egomotion estimation, having less root mean square error than the one exhibit by LIBVISO in linear velocities estimation V_x, V_y, and V_z in all translational estimated trajectory. This turns out to be more evident in the

Table 5.2: Comparison of the standard mean squared error between ground truth information and both stereo egomotion estimation methods (PSET and LIBVISO).

	$V_x(m/f)$	$V_y(m/f)$	$V_z(m/f)$	$\|\|V\|\|(m/f)$
LIBVISO	0.000690	0.000456	0.0011	**0.0022**
PSET	0.000336	0.000420	0.000487	**0.0012**

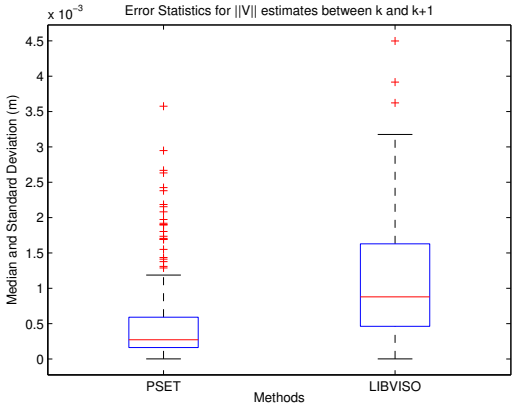

Figure 5.16: Error Statistics for ||V|| linear velocities obtained by PSET and LIBVISO egomotion estimation

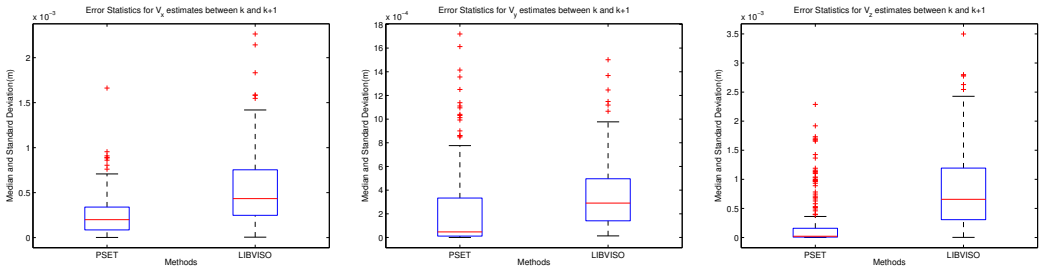

Figure 5.17: Error Statistics for the linear velocities estimation obtained by PSET and LIBVISO in all 3 axes (V_x, V_y, V_z)

computation of the velocity norm over the global motion trajectory, where PSET results are almost 50% more accurate than the one showed by LIBVISO.

In figure 5.16 we can observe the error statistics for each instantaneous motion between time k and k+1 during all 4 sequences, and it is clear the best performance accuracy of the PSET egomotion estimates that have a lower median and a much smaller standard deviation than the LIBVISO estimates. The PSET linear velocities estimates (V_x, V_y, V_z) are more accurate when compared with LIBVISO estimates for all 3 axes, as shown in figure 5.17.

5.2.6 Real Image Sequences

In order to evaluate the PSET results using real image sequences, we utilized the same sequence of the Karlsruhe dataset (2009-09-08-drive-0021). We compared PSET performance against LIBVISO [KGL10], and with our previous implementation of the 6DP [SBS13a], using the Inertial Measurement Unit (RTK-GPS information) as ground-truth information.

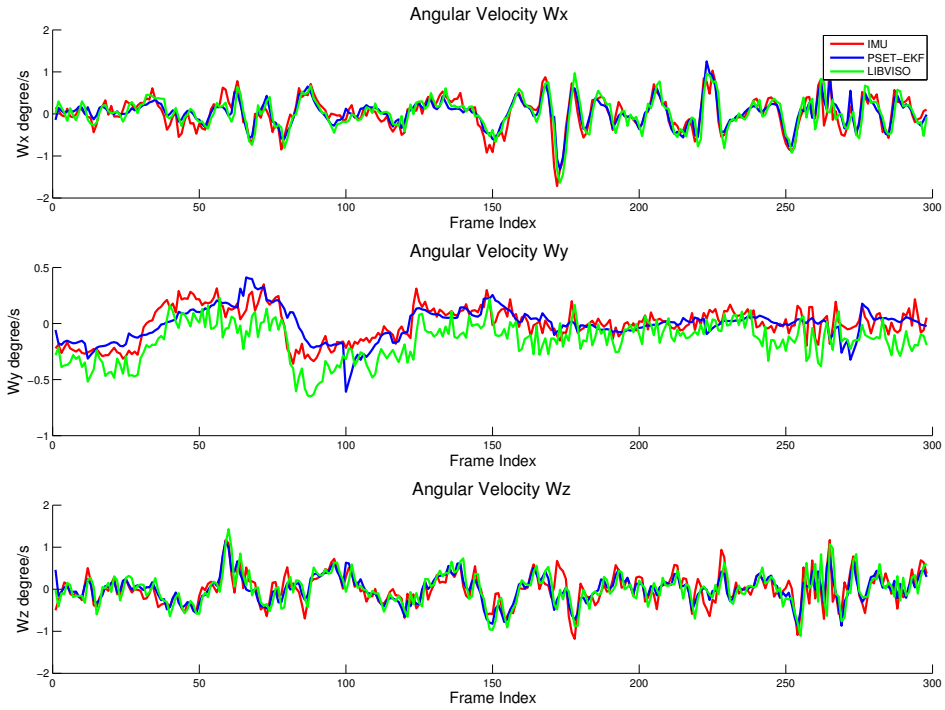

Figure 5.18: Results for the angular velocities estimation of 300 frames: ground truth(GPS-IMU information), filtered PSET measurements (PST-EKF) and 6D Visual Odometry Library (LIBVISO). Even though all exhibit similar behaviors the filtered implementation PSET-EKF is the one which is closer to GT(GPS-IMU)(see also table 1).

In Fig.5.18, PSET and LIBVISO estimation of the angular velocities are presented together with IMU ground-truth information. The 6DP results are not presented in the linear and angular velocities figures, due to the fact that for angular velocities case PSET and 6DP use the same method of computation and thus obtain almost identical results. One can observe that PSET has a much closer performance to IMU than LIBVISO. This is stated in Table 1, where the RMS error for the LIBVISO method is about twice the error of the PSET/6DP method.

In Fig.5.19, one can observe the behavior of both methods in the linear velocity estimation case. Both LIBVISO and PSET present similar results for the linear velocity estimation case, but PSET has about 50 % less overall RMS error, as can be checked in Table 5.3. The results confirm that estimating the translation scale using probabilistic approaches produces better results than using deterministic correspondences, as displayed in table 1. In Fig.5.20, one can observe a zoom for the first 20 frames, where is possible to see that PSET is closer to IMU ground-truth information.

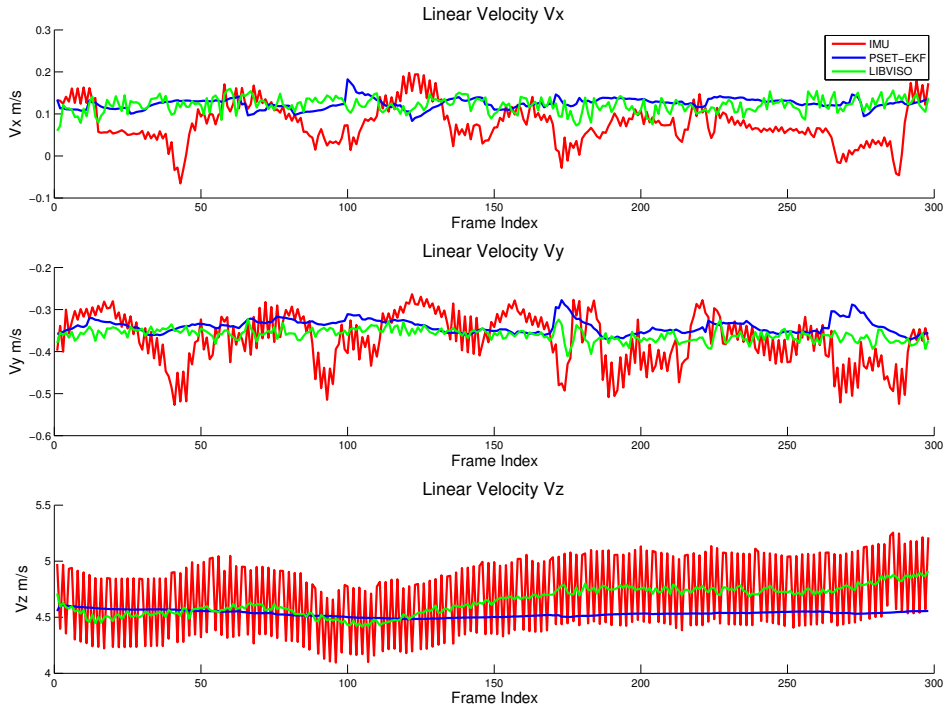

Figure 5.19: Estimated linear velocities of 300 frames estimation. The PSET transform exhibits a better performance in V_y compared to LIBVISO, and the opposite occurs in V_z estimation (see Table 1). However in overall linear velocities estimation the PSET is about 50 % better, see Table 1

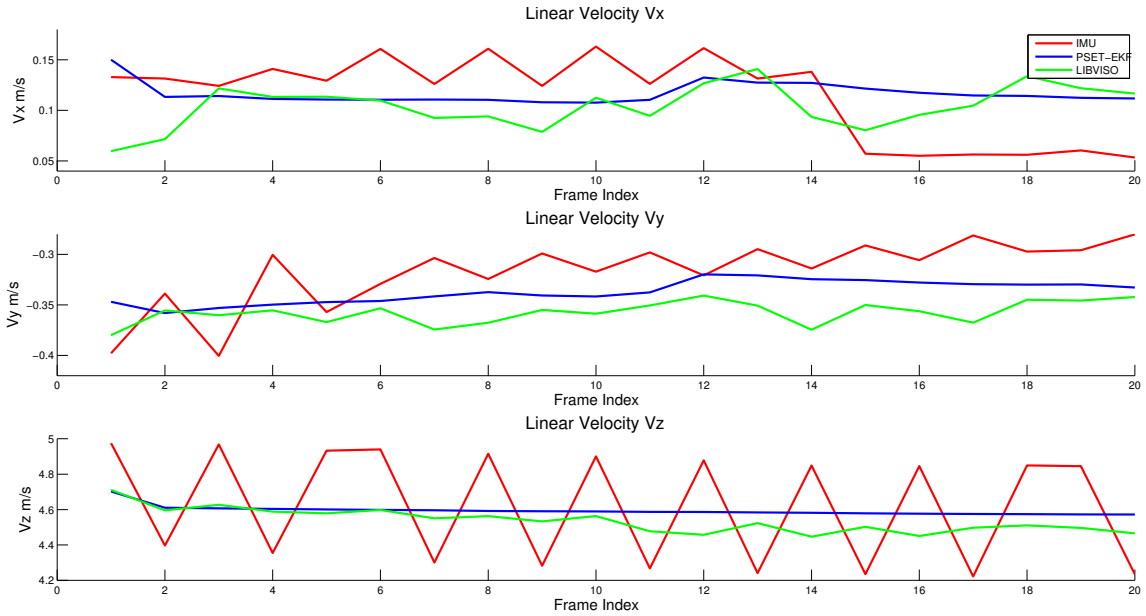

Figure 5.20: Zoom view of the first 20 frames results for linear velocities estimation, using PSET, LIBVISO and Inertial Measurement Unit information

Table 5.3: Comparison of the standard mean squared error between IMU and stereo egomotion estimation methods(LIBVISO, 6DP, and PSET). The linear velocities results (V) are presented in (m/s), and the angular velocities results (W) are presented in $(degrees/s)$

	V_x	V_y	V_z	W_x	W_y	W_z	$\|\|V\|\|$	$\|\|W\|\|$
LIBVISO	0.0674	0.7353	0.3186	0.0127	0.0059	0.0117	1.1213	0.0303
6DP	0.0884	0.0748	0.7789	0.0049	0.0021	0.0056	0.9421	0.0126
PSET	0.0700	0.0703	0.3686	0.0034	0.0019	0.0055	0.5089	0.0108

5.3 Summary

The PSET methodology described in this work has proven to be an accurate method of computing stereo egomotion. The proposed approach is very interesting because no explicit matching or feature tracking is necessary to compute the vehicle motion. To the best of our knowledge this is the first implementation of a full dense probabilistic method to compute stereo egomotion. The results demonstrate that PSET is more accurate then other state-of-the-art stereo egomotion estimation methods, improving the overall accuracy in about 50 % in angular velocity estimation then LIBVISO and 50 % better accuracy performance in linear velocity, over both LIBVISO and 6DP previous methods. At the moment PSET consumes more computational resources than LIBVISO, but this will be mitigated by an implementation of the PSET method using **GPU**.

5.4 Related Publications

The work presented in this chapter, related to the PSET method for stereo egomotion estimation has been accepted for publishing in IEEE International Conference for Robotics and Automation (ICRA 2014) May 31-June 5, 2014 Hong Kong

6

Conclusions and Future Work

6.1 Conclusions

In this thesis, our work focused on the development of novel techniques based on probabilistic approaches for stereo visual egomotion estimation. We developed the following methods:

- **6DP-1st [SBS13a]**

  We presented a novel 6D visual odometry algorithm (named 6DP), based on sparse and dense feature based mixture of probabilistic ego-motion methods with stereo vision. We tested our algorithm performance against other known methods for visual odometry estimation, namely against the 5-point RANSAC algorithm.

  The obtained results demonstrated that probabilistic methods are a viable way to conduct visual odometry estimation, especially by providing additional evidence that this type of approach performs particularly well on estimating camera rotation movement and translation up to a scale factor.

  However, results presented also show that for obtaining translation scale estimation, the performance of using Harris corners propagated through E from sequential time frame images are not as accurate as the one obtained using highly distinctive features such as SIFT.

- **6DP-2st [SBS13b]**

  To overcome the translation scale estimation problem, we modified the 6DP algorithm, and used SIFT features instead of Harris Corners to reduce translation scale estimation ambiguity, we complemented this method with a sparse feature approach for estimating image

depth. We tested the proposed algorithm against an open-source 6D visual odometry library, such as LIBVISO.

The obtained results demonstrate that 6DP performs accurately when compared to other techniques for stereo visual egomotion estimation, yielding robust motion estimation results, mainly in the angular velocities estimation results where 50 % improvement was achieved.

However concerning linear angular velocities estimation, obtained results are only similar to the ones obtained using state-of-the-art feature based ego-motion estimation algorithms, and no significant improvement is achieved.

- **PSET**

  With 6DP implementation, we proved that probabilistic approaches were a viable way of computing stereo egomotion. However probabilistic methods were unable to estimate the translational scale, and deterministic methods were used as complement to estimate the scale.

  One of the thesis objectives was to develop a fully probabilistic approach to the stereo egomotion estimation problem. The **PSET** methodology described in this work has proven to be an accurate method of computing stereo egomotion. The proposed approach is very interesting because no explicit matching or feature tracking is necessary to compute the vehicle motion. To the best of our knowledge this is the first implementation of a full dense probabilistic method to compute stereo egomotion. The results demonstrate that PSET is more accurate then other state-of-the-art 3D egomotion estimation methods, improving the overall accuracy in about 50 % in angular velocity estimation then LIBVISO and 50 % better accuracy performance in linear velocity, over both LIBVISO and 6DP previous methods.

6.2 Future Work

Despite the relevance of the proposed probabilistic stereo visual egomotion approach demonstrated in this thesis, there are many secondary aspects of the method that need to addressed and form the core objectives to pursue on our ongoing and future work. The relevant ones are enumerated as follows.

- Our ongoing work is to implement the PSET OpenCL framework in a GPU. From a standard CPU point of view, the PSET method demonstrated clear improvements over other state-of-the-art methods but the effectiveness and usefulness in mobile robotics scenarios requires to have it run in real-time, which is more feasible in a multi-core implementation.

- Other ongoing work, is to use the results obtained from the PSET accumulator to map and track independent moving targets. The probabilistic nature of the PSET transform method,

allows with high degree of likelihood to determine key points that are moving independently from the image motion. This information can be useful in mobile robotics applications scenarios here not only vehicle egomotion is necessary but also there is a clear need to map the world environment.

- In future work we plan to explore the fusion of the PSET implementation with other navigation sensors e.g, Inertial Measurement Units. The probabilistic approaches theoretically provide better information, so a loosely-coupled solution and tightly-coupled solution between PSET method and IMU sensors should be tested.

- In future work, we plan to implement in a general mobile robotics navigation architecture a Visual Odometry system containing the PSET stereo visual egomotion estimation method. The objective is to pursue further validation of the PSET OpenCL in other heterogeneous mobile robotics scenarios. Specially in the aerial and underwater robotics scenario, where the lack of image texture combined with high matching ambiguity with image repetitiveness, provides an ideal scenario for the PSET advantageous use.

- In future work, we plan to use the probabilistic point correspondences methodologies to other computer vision geometric constraints besides the epipolar geometry namely for the computation of probabilistic distributions of correspondence based on the homography matrix (H).

Bibliography

[ABD10a] P.F. Alcantarilla, L.M. Bergasa, and Frank Dellaert. Visual odometry priors for robust EKF-SLAM. In *IEEE International Conference on Robotics and Automation,ICRA 2010*, pages 3501–3506. IEEE, 2010.

[ABD10b] Hatem Alismail, Brett Browning, and M. Bernardine Dias. Evaluating pose estimation methods for stereo visual odometry on robots. In *In proceedings of the 11th International Conference on Intelligent Autonomous Systems (IAS-11)*, 2010.

[AFDM08] Adrien Angeli, David Filliat, StÃľphane Doncieux, and Jean-Arcady Meyer. Fast and incremental method for loop-closure detection using bags of visual words. *IEEE Transactions on Robotics*, 24(5):1027–1037, 2008.

[AK07] Motilal Agrawal and Kurt Konolige. Rough terrain visual odometry. In *Proceedings of the International Conference on Advanced Robotics (ICAR)*, 2007.

[AKI05] Motilal Agrawal, Kurt Konolige, and Luca Iocchi. Real-time detection of independent motion using stereo. In *IEEE workshop on Motion (WACV/MOTION)*, January 2005.

[Ana89] P Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 310:283–310, 1989.

[AYAB12] PF Alcantarilla, JJ Yebes, J Almazan, and L Bergasa. On Combining Visual SLAM and Dense Scene Flow to Increase the Robustness of Localization and Mapping in Dynamic Environments. *IEEE International Conference on Robotics and Automation*, 2012.

[BETVG08] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, June 2008.

[BFB94] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *Internacional Journal of Computer Vision*, 12:43–77, 1994.

[BS11] Mitch Bryson and Salah Sukkarieh. A comparison of feature and pose-based mapping using vision, inertial and GPS on a UAV. *IEEE International Conference on Intelligent Robots and Systems, IROS*, pages 4256–4262, 2011.

[BSL+10] Simon Baker, Daniel Scharstein, J. P. Lewis, Stefan Roth, Michael J. Black, and Richard Szeliski. A Database and Evaluation Methodology for Optical Flow. *International Journal of Computer Vision*, 92(1):1–31, November 2010.

[CGDM10] Javier Civera, O.G. Grasa, A.J. Davison, and JMM Montiel. 1-Point RANSAC for EKF filtering. Application to real-time structure from motion and visual odometry. *Journal of Field Robotics*, 27(5):609–631, 2010.

[CLD07] Peter Corke, Jorge Lobo, and Jorge Dias. An introduction to inertial and visual sensing. *The International Journal of Robotics Research*, 2007.

[CLSF10] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. *European Conference on Computer Vision*, 2010.

[CMR10] A.I Comport, E. Malis, and P. Rives. Real-time quadrifocal visual odometry. *The International Journal of Robotics Research*, 29(2-3):245–266, January 2010.

[CN08] Mark Cummins and Paul Newman. FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. *The International Journal of Robotics Research*, 27(6):647–665, 2008.

[CN10] Mark Cummins and Paul Newman. Appearance-only SLAM at large scale with FAB-MAP 2.0. *The International Journal of Robotics Research*, 2010.

[Cor11] Peter Corke. *Robotics, Vision and Control - Fundamental Algorithms in MATLAB*. Springer Tracts in Advanced Robotics. Springer, 2011.

[Cra89] John J. Craig. *Introduction to Robotics: Mechanics and Control*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1989.

[CSS05] Peter Corke, D. Strelow, and S. Singh. Omnidirectional visual odometry for a planetary rover. In *IEEE International Conference on Intelligent Robots and Systems, IROS 2004*, volume 4, pages 4007–4012, 2005.

[DA06] Justin Domke and Yiannis Aloimonos. A Probabilistic Notion of Correspondence and the Epipolar Constraint. In *Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, pages 41–48. IEEE, June 2006.

[Dau85] J G Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America. A, Optics and image science*, 2(7):1160–1169, July 1985.

[DGK11] 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.

[DM12] Damien Dusha and Luis Mejias. Error analysis and attitude observability of a monocular gps/visual odometry integrated navigation filter. *The International Journal of Robotics Research*, 31(6):714–737, 2012.

[DRMS07] Andrew J. Davison, Ian D. Reid, Nicholas D. Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):1052–1067, 2007.

[FB81] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[FG87] W Förstner and E Gülch. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *ISPRS Intercommission Workshop*, 1987.

[Fis81] Bolles C. Fischler, M.A. Random sample consensus a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[FS12] F. Fraundorfer and D. Scaramuzza. Visual odometry : Part ii: Matching, robustness, optimization, and applications. *Robotics Automation Magazine, IEEE*, 19(2):78–90, 2012.

[GKN+74] Arthur Gelb, Joseph F. Kasper, Raymond A. Nash, Charles F. Price, and Arthur A. Sutherland, editors. *Applied Optimal Estimation*. MIT Press, Cambridge, MA, 1974.

[GM11] S.B. Goldberg and L. Matthies. Stereo and imu assisted visual odometry on an omap3530 for small robots. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, pages 169 –176, june 2011.

[Goo91] Colin Goodall. Procrustes Methods in the Statistical Analysis of Shape. *Journal of the Royal Statistical Society. Series B (Methodological)*, 53(2):285–339, 1991.

[GR11] V. Guizilini and F. Ramos. Visual odometry learning for unmanned aerial vehicles. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 6213 –6220, may 2011.

[GZS11] Andreas Geiger, Julius Ziegler, and Christoph Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *Intelligent Vehicles Symposium*, pages 963–968. IEEE, 2011.

[HBH+11] Albert S. Huang, Abraham Bachrach, Peter Henry, Michael Krainin, Daniel Maturana, Dieter Fox, and Nicholas Roy. Visual odometry and mapping for autonomous flight using an rgb-d camera. In *Int. Symposium on Robotics Research (ISRR)*, Flagstaff, Arizona, USA, Aug. 2011.

[Hee87] DJ Heeger. Model for the extraction of image flow. *Journal Optics Society American*, 1987.

[Hee88] David J. Heeger. Optical flow using spatiotemporal filters. *International Journal of Computer Vision*, 1(4):279–302, January 1988.

[HLON94] Robert M. Haralick, Chung-Nan Lee, Karsten Ottenberg, and Michael Nölle. Review and analysis of solutions of the three point perspective pose estimation problem. *Int. J. Comput. Vision*, 13(3):331–356, December 1994.

[HMM+12] T.M. Howard, A. Morfopoulos, J. Morrison, Y. Kuwata, C. Villalpando, L. Matthies, and M. McHenry. Enabling continuous planetary rover navigation through fpga stereo and visual odometry. In *Aerospace Conference, 2012 IEEE*, pages 1–9, march 2012.

[Hor87] Berthold K P Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America. A*, 4(4):629–642, April 1987.

[How08] Andrew Howard. Real-time stereo visual odometry for autonomous ground vehicles. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2008*, pages 3946–3952. Ieee, sep 2008.

[HS81] Berthold K.P. Horn and Brian G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203, August 1981.

[HS88] C. Harris and M. Stephens. A combined corner and edge detection. In *Proceedings of The Fourth Alvey Vision Conference*, pages 147–151, 1988.

[HZ04] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[HZP+11] J Huang, T Zhu, X Pan, L Qin, X Peng, C Xiong, and Jing Fang. A high-efficiency digital image correlation method based on a fast recursive scheme. *Measurement Science and Technology*, 21(3), 2011.

[IGN11] Szakats Istvan, Catalin Golban, and Sergiu Nedevschi. Fast vision based ego-motion estimation from stereo sequences âĂŤ A GPU approach. *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 538–543, October 2011.

[Jon10] E.S. Jones. *Large scale visual navigation and community map building*. PhD thesis, 2010.

[JS11] Eagle S. Jones and Stefano Soatto. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *International Journal Robotics Research*, 30(4):407–430, April 2011.

[JVS07] E. Jones, A. Vedaldi, and S. Soatto. Inertial structure from motion with autocalibration. In *Proceedings of the International Conference on Computer Vision - Workshop on Dynamical Vision*, 2007.

[KCS11] L Kneip, M Chli, and R Siegwart. Robust real-time visual odometry with a single camera and an imu. In *Proc. of the British Machine Vision Conference (BMVC)*, 2011.

[KD06] Ni Kai and Frank Dellaert. Stereo tracking and three-point/one-point algorithms - a robust approach. In *Visual Odometry, In Intl. Conf. on Image Processing (ICIP*, pages 2777–2780, 2006.

[KGL10] Bernd Kitt, Andreas Geiger, and Henning Lategahn. Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme. In *IEEE Intelligent Vehicles Symposium (IV), 2010*, pages 486–492. IEEE, 2010.

[KKN+12] T. Kazik, L. Kneip, J. Nikolic, M. Pollefeys, and R. 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, june 2012.

[KM07] Georg Klein and David Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, Nara, Japan, November 2007.

[KSC13] Christian Kerl, Jurgen Sturm, and Daniel Cremers. Robust odometry estimation for RGB-D cameras. *2013 IEEE International Conference on Robotics and Automation*, pages 3748–3754, May 2013.

[KSS08] Jonathan Kelly, Srikanth Saripalli, and GS Sukhatme. Combined visual and inertial navigation for an unmanned aerial vehicle. *Field and Service Robotics*, 2008.

[KSS11] L Kneip, D Scaramuzza, and R Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *Proc. of The IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, Colorado Springs, USA, June 2011.

[LCS11] Stefan Leutenegger, Margarita Chli, and Roland Y. Siegwart. BRISK: Binary Robust invariant scalable keypoints. *2011 International Conference on Computer Vision*, pages 2548–2555, November 2011.

[LD03] Jorge Lobo and Jorge Dias. Vision and inertial sensor cooperation using gravity as a vertical reference. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1597–1608, 2003.

[LD04] Jorge Lobo and Jorge Dias. Inertial sensed ego-motion for 3d vision. *Journal of Robotic Systems*, 21(1):3–12, 2004.

[LH87] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. In M. A. Fischler and O. Firschein, editors, *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*, pages 61–62. Kaufmann, Los Altos, CA., 1987.

[LHP80] H.˜C. Longuet-Higgins and K Prazdny. The Interpretation of a Moving Retinal Image. *Royal Society of London Proceedings Series B*, 208:385–397, 1980.

[Lim10] John Lim. *Egomotion Estimation with Large Field-of-View Vision*. PhD thesis, Australian National University, 2010.

[LK81] BD Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI*, pages 674–679, 1981.

[LMNF09] V. Lepetit, F. Moreno-Noguer, and P. Fua. Epnp: An accurate o(n) solution to the pnp problem. *International Journal Computer Vision*, 81(2), 2009.

[Low04] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[LXX12] Shiqi Li, Chi Xu, and Ming Xie. A Robust O(n) Solution to the Perspective-n-Point Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1444–1450, 2012.

[MBG07] FA Moreno, JL Blanco, and J González. An efficient closed-form solution to probabilistic 6D visual odometry for a stereo camera. In *Proceedings of the 9th international conference on Advanced concepts for intelligent vision systems*, pages 932–942. Springer-Verlag, 2007.

[MCF10] József Molnár, Dmitry Chetverikov, and Sándor Fazekas. Illumination-robust variational optical flow using cross-correlation. *Computer Vision and Image Understanding*, 114(10):1104–1114, October 2010.

[MCUP04] J Matas, O Chum, M Urban, and T Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, September 2004.

[MDT07] Luiz G. B. Mirisola, Jorge Dias, and A. Traça de Almeida. Trajectory recovery and 3D mapping from rotation-compensated imagery for an airship. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1908–1913, October 2007.

[MHB+07] TK Marks, Andrew Howard, Max Bajracharya, G Cottrell, and L Matthies. Gamma-SLAM: Stereo visual SLAM in unstructured environments using variance grid maps. 2007.

[MHB+10] Elmar Mair, Gregory D. Hager, Darius Burschka, Michael Suppa, and Gerhard Hirzinger. Adaptive and generic corner detection based on the accelerated segment test. In *European Conference on Computer Vision (ECCV'10)*, September 2010.

[MMC05] M. Maimone, L. Matthies, and Y. Cheng. Visual Odometry on the Mars Exploration Rovers. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 903–910. IEEE, 2005.

[MMC07] Mark Maimone, Larry Matthies, and Yang Cheng. Two years of visual odometry on the mars exploration rovers: Field reports. *J. Field Robot.*, 24(3):169–186, March 2007.

[Mor80] Hans Moravec. Obstacle avoidance and navigation in the real world by a seeing robot rover. In *tech. report CMU-RI-TR-80-03, Robotics Institute, Carnegie Mellon University doctoral dissertation, Stanford University*, number CMU-RI-TR-80-03. September 1980.

[MS06] Annalisa Milella and Roland Siegwart. Stereo-based ego-motion estimation using pixel tracking and iterative closest point. In *in IEEE International Conference on Computer Vision Systems*, page 21, 2006.

[ND11] RA Newcombe and AJ Davison. KinectFusion: Real-time dense surface mapping and tracking. *In 2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, October 2011.

[NDK09] Kai Ni, Frank Dellaert, and Michael Kaess. Flow separation for fast and robust stereo odometry. In *IEEE International Conference on Robotics and Automation ICRA 2009*, volume 1, pages 3539–3544, 2009.

[Nis04] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26:756–777, June 2004.

[NLD11] Richard a. Newcombe, Steven J. Lovegrove, and Andrew J. Davison. DTAM: Dense tracking and mapping in real-time. *2011 International Conference on Computer Vision*, pages 2320–2327, November 2011.

[NM65] J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965.

[NNB06] David Nister, Oleg Naroditsky, and James Bergen. Visual odometry for ground vehicle applications. *Journal of Field Robotics*, 23:2006, 2006.

[NS88] Ken Nakayama and Gerald H. Silverman. The aperture problem I. Perception of nonrigidity and motion direction in translating sinusoidal lines. *Vision Research*, 28(6):739–746, 1988.

[NvVB11] Navid Nourani-vatani, Paulo Vinicius, and Koerich Borges. Correlation-Based Visual Odometry for Ground Vehicles. *Journal of Field Robotics*, 28(5):742–768, 2011.

[NZG+11] Oleg Naroditsky, XS Zhou, JH Gallier, Stergios I Roumeliotis, and Kostas Daniilidis. Structure from motion with directional correspondence for visual odometry. *GRASP Laboratory Technical Report MS-CIS-11-15*, 2011.

[NZG+12] Oleg Naroditsky, Xun S Zhou, Jean Gallier, Stergios I Roumeliotis, and Kostas Daniilidis. Two efficient solutions for visual odometry using directional correspondence. *IEEE transactions on pattern analysis and machine intelligence*, 34(4):818–24, April 2012.

[OA07] Abhijit S. Ogale and Yiannis Aloimonos. A roadmap to the integration of early visual modules. *International Journal of Computer Vision*, 72(1):9–25, April 2007.

[OFA05] A S Ogale, C Fermuller, and Y Aloimonos. Motion segmentation using occlusions. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 27(6):988–992, 2005.

[OM10] Stepan Obdrzalek and Jiri Matas. A voting strategy for visual ego-motion from stereo. In *2010 IEEE Intelligent Vehicles Symposium*, volume 1, pages 382–387. IEEE, 2010.

[OMSM03] C. Olson, L. Matthies, M. Schoppers, and M. Maimone. Rover navigation using stereo ego-motion. *Robotics and Autonomous Systems*, 43:215–229, February 2003.

[Par62] Emanuel Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 1962.

[PZJ12] Ghazaleh Panahandeh, Dave Zachariah, and Magnus Jansson. Exploiting ground plane constraints for visual-inertial navigation. *Proceedings of the 2012 IEEE/ION Position, Location and Navigation Symposium*, pages 527–534, April 2012.

[RD06] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, May 2006.

[RGNS12] Joern Rehder, Kamal Gupta, Stephen T. Nuske, and Sanjiv Singh. Global pose estimation with limited gps and long range visual odometry. In *IEEE Conference on Robotics and Automation*, May 2012.

[RL01] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the ICP algorithm. In *Third International Conference on 3D Digital Imaging and Modeling (3DIM)*, June 2001.

[RN12] Florian Raudies and Heiko Neumann. A review and evaluation of methods estimating ego-motion. *Computer Vision and Image Understanding*, 116(5):606–633, May 2012.

[RNKB08] R. Roberts, Hai Nguyen, N. Krishnamurthi, and T. Balch. Memory-based learning for visual odometry. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 47 –52, may 2008.

[SA92] Ajit Singh and Peter Allen. Image-flow computation: An estimation-theoretic framework and a unified perspective. *CVGIP: Image understanding*, 56(2):152–177, September 1992.

[SB97] Stephen M. Smith and J. Michael Brady. Susan - a new approach to low level image processing. *Int. J. Comput. Vision*, 23(1):45–78, May 1997.

[SBS13a] Hugo Silva, Alexandre Bernardino, and Eduardo Silva. Combining sparse and dense methods for 6d visual odometry. *13th IEEE International Conference on Autonomous Robot Systems and Competitions, Lisbon Portugal*, Abril 2013.

[SBS13b] Hugo Silva, Alexandre Bernardino, and Eduardo Silva. 6d visual odometry with dense probabilistic egomotion estimation. *8th International Conference on Computer Vision Theory and Applications, Barcelona Spain*, February 2013.

[SF11] D. Scaramuzza and F. Fraundorfer. Visual odometry [tutorial]. *Robotics Automation Magazine, IEEE*, 18(4):80 –92, dec. 2011.

[SFP12] Olivier Saurer, Friedrich Fraundorfer, and Marc Pollefeys. Homography based visual odometry with known vertical direction and weak Manhattan world assumption. *ViCoMoR*, 2012.

[SFS09] Davide Scaramuzza, Friedrich Fraundorfer, and Roland Siegwart. Real-time monocular visual odometry for on-road vehicles with 1-point ransac. In *2009 IEEE International Conference on Robotics and Automation, ICRA 2009, Kobe, Japan, May 12-17, 2009*, pages 4293–4299. IEEE, 2009.

[SMSH01] Dennis Strelow, Jeff Mishler, Sanjiv Singh, and Herman Herman. Omnidirectional shape-from-motion for autonomous navigation. In *Proc. of the 2001 IEEE/RSJ Int. Conf. on IROS*, 2001.

[SRD13] Natesh Srinivasan, Richard Roberts, and Frank Dellaert. High frame rate egomotion estimation. *Computer Vision Systems*, (c):183–192, 2013.

[SSC11] F. Steinbruecker, J. Sturm, and D. Cremers. Real-time visual odometry from dense rgb-d images. In *Workshop on Live Dense Reconstruction with Moving Cameras at the Intl. Conf. on Computer Vision (ICCV)*, 2011.

[SSV97] César Silva and José Santos-Victor. Robust egomotion estimation from the normal flow using search subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(9):1026–1034, 1997.

[ST94] J. Shi and C. Tomasi. Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, pages 593–600, 1994.

[Sun99] Changming Sun. Fast optical flow using cross correlation and shortest-path techniques. *Proceedings of digital image computing: techniques and Applications*, pages 143–148, 1999.

[Sze10] Richard Szeliski. *Computer Vision: Algorithms and Applications*. Springer-Verlag New York, Inc., New York, NY, USA, 1st edition, 2010.

[TLF10] Engin Tola, V. Lepetit, and P. Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(5):815–830, 2010.

[TMR+07] Nikolas Trawny, Anastasios I Mourikis, Stergios I Roumeliotis, Andrew E Johnson, and James F Montgomery. Vision-aided inertial navigation for pin-point landing using observations of mapped landmarks. *Journal of Field Robotics*, 24(5):357–378, 2007.

[TPD08] J.-P. Tardif, Y. Pavlidis, and K. Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 2531–2538, Sept 2008.

[VNH+11] R Voigt, J Nikolic, C Huerzeler, S Weiss, L Kneip, and R Siegwart. Robust embedded egomotion estimation. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011.

[WH] Yihong Wu and Zhanyi Hu. Pnp problem revisited. *Journal of Mathematical Imaging and Vision*, (1):131–141.

[WJK13] Thomas Whelan, Hordur Johannsson, and Michael Kaess. Robust real-time visual odometry for dense RGB-D mapping. *IEEE Intl. Conf. on Robotics and Automation(ICRA)*, (i), 2013.

[WKR07] B Williams, G Klein, and I Reid. Real-time {SLAM} relocalisation. In *Proc. International Conference on Computer Vision*, 2007.

[WR10] Brian Williams and Ian Reid. On combining visual SLAM and visual odometry. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 3494–3500. IEEE, 2010.

[WS03] Joachim Weickert and Christoph Schn. Lucas/Kanade Meets Horn/Schunck Combining Local and Global Optical Flow Methods. 2003.

[WU13] Michael Warren and Ben Upcroft. High altitude stereo visual odometry. In *Proceedings of Robotics: Science and Systems*, Berlin, Germany, June 2013.

[XJM12] L Xu, J Jia, and Y Matsushita. Motion detail preserving optical flow estimation. *IEEE International Journal on Pattern Analysis and Machine Intelligence*, 34(9):1744–1757, 2012.

[ZDFL95] Z Zhang, R Deriche, Oliver Faugeras, and Quang-Tuan Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Special Volume on Computer Vision*, 78(2):87 – 119, 1995.

[ZT13] Danping Zou and Ping Tan. Coslam: Collaborative visual slam in dynamic environments. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(2):354–366, 2013.

A

Appendix 1

A.1 Zero Normalized Cross Correlation

The global objective of the ZNCC method is to compare a reference subset (the correlation window sampled in the reference image) to a corresponding template in another image, see figure A.2. The method developed by Huang et al[HZP$^+$11] uses a recursive scheme for calculating the numerator of (A.1) and a global sum-table approach for the denominator, thus saving significant computation time.

In summary, the method has two distinctive parts one for calculating ZNCC numerator and other for the denominator calculation. The ZNCC equation (4.2) can be described in the following form.

$$C_{x,y}(u,v) = \frac{P(x,y;u,v) - Q(x,y;u,v)}{\sqrt{F(x,y)}\sqrt{G(x,y;u,v)}} \tag{A.1}$$

where the numerator term can be calculated using the following equations:

$$P(x,y;u,v) = \sum_{x=x-N_x}^{x+N_x} \sum_{y=y-N_y}^{y+N_y} [f(x,y) \times g(x+u, y+v)]. \tag{A.2}$$

$$\begin{aligned} Q(x,y;u,v) = \frac{1}{(2N_x+1)(2N_y+1)} \left[\sum_{x=x-N_x}^{x+N_x} \sum_{y=y-N_y}^{y+N_y} f(x,y) \right] \\ \times \left[\sum_{x=x-N_x+u}^{x+N_x+u} \sum_{y=y-N_y+v}^{y+N_y+v} g(x,y) \right] \end{aligned} \tag{A.3}$$

On the other hand, although Q$(x,y;u,v)$ can be calculated using a sum-table approach, the term P$(x,y;u,v)$ involves cross correlation terms between both images and cannot be calculated recurring to a sum-table approach, since (u,v) are sliding window parameters.

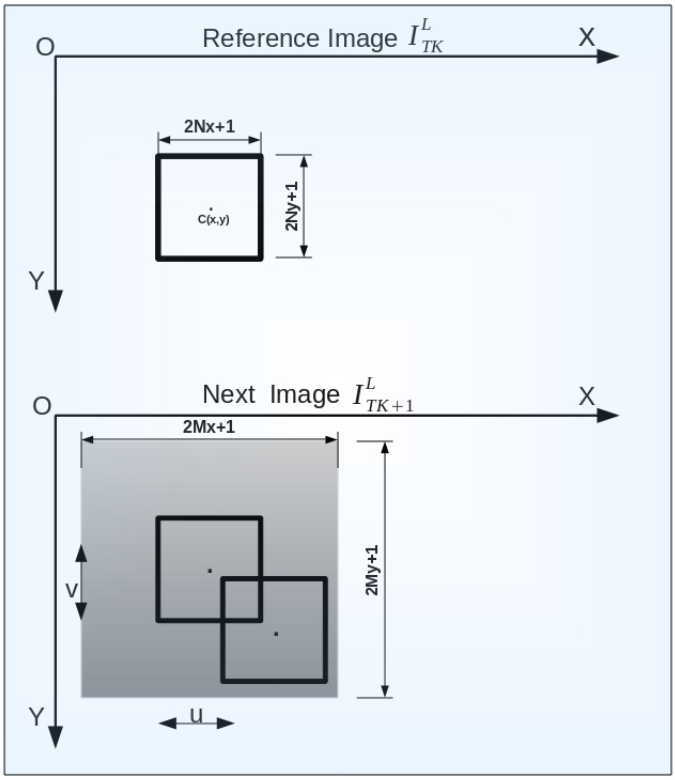

Figure A.1: ZNCC reference template matching

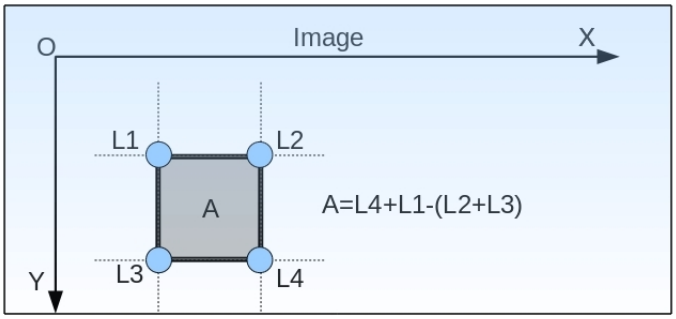

Figure A.2: Integral Window calculation

For the denominator calculation a global sum-table approach can be used:

$$F(x,y) = \sum_{x=x-N_x}^{x+N_x} \sum_{y=y-N_y}^{y+N_y} f^2(x,y) - \frac{1}{(2N_x+1)(2N_y+1)} \times \left[\sum_{x=x-N_x}^{x+N_x} \sum_{y=y-N_y}^{y+N_y} f(x,y) \right]^2 \tag{A.4}$$

$$G(x,y;u,v) = \sum_{x=x-N_x+u}^{x+N_x+u} \sum_{y=y-N_y+v}^{y+N_y+v} g^2(x,y) - \frac{1}{(2N_x+1)(2N_y+1)} \times \left[\sum_{x=x-N_x+u}^{x+N_x+u} \sum_{y=y-N_y+v}^{y+N_y+v} g(x,y) \right]^2 \tag{A.5}$$

where the four global sum schemes can be calculated as an integral window approach, see figure A.2.