

Luis Puig Morales

# Omnidirectional multi-view systems: calibration, features and 3D information

Departamento  
Informática e Ingeniería de Sistemas

Director/es  
Guerrero Campo, José Jesús

<http://zaguan.unizar.es/collection/Tesis>



**Universidad**  
Zaragoza

Tesis Doctoral

# OMNIDIRECTIONAL MULTI-VIEW SYSTEMS: CALIBRATION, FEATURES AND 3D INFORMATION

Autor

Luis Puig Morales

Director/es

Guerrero Campo, José Jesús

**UNIVERSIDAD DE ZARAGOZA**

Informática e Ingeniería de Sistemas

2011



# Omnidirectional multi-view systems: calibration, features and 3D information

Luis Puig Morales  
PhD Dissertation

Adviser: José Jesús Guerrero Campo  
Departamento de Informática e Ingeniería de Sistemas  
Escuela de Ingeniería y Arquitectura  
Universidad de Zaragoza, Spain

December 2011





# Omnidirectional multi-view systems: calibration, features and 3D information

Luis Puig Morales  
PhD Thesis

---

**PhD Committee members:**

Carlos Sagües Blázquez	Universidad de Zaragoza, Spain
José María Martínez Montiel	Universidad de Zaragoza, Spain
Patrick Rives	INRIA Sophia Antipolis, France
Pascal Vasseur	University of Rouen, France
Joaquim Salvi	Universidad de Girona, Spain

---

**European Reviewers:**

Peter Sturm	INRIA Rhône-Alpes
João Barreto	University of Coimbra, Portugal



# Acknowledgments

I would like to thank anyone who was involved directly or indirectly in the development of this thesis. First of all, I would like to thank my supervisor, Josechu Guerrero, for all these years of discussions and advising.

I would also like to thank Peter Sturm, who, during my visit to INRIA Rhône-Alpes in 2008, answered many of my early questions about catadioptric vision. I also thank Kostas Daniilidis for giving me the chance to collaborate with him at GRASP Lab at the University of Pennsylvania not only once but twice over the years.

I would like to thank the RoPeRT group at the University of Zaragoza, where I had the opportunity to develop this thesis. At the same time, I would like to thank the Spanish Ministry of Education for the FPI scholarship (BES-2007-17271), which allowed the development of this thesis, as well as for the opportunity to visit important labs around the world. More specifically the prestigious INRIA Rhône-Alpes in France and the GRASP Lab at the University of Pennsylvania, USA.

Finally, I would like to thank my family and Yujie for all the support, which even at the distance was very important in the completion of this thesis.



# Resumen

En años recientes el uso de cámaras omnidireccionales ha aumentado de manera considerable tanto en la comunidad de robótica como en la de visión por computador. La mayor ventaja de este tipo de cámaras es su gran campo de vista, lo que les permite obtener información de toda la escena en una sola imagen. Este tipo de sistemas han sido usados en una gran cantidad de aplicaciones como: videovigilancia, seguimiento de objetos, navegación visual, odometría visual, localización y SLAM, reconstrucción a partir de movimiento, visión activa, fotogrametría, redes de cámaras, etc. Entre los tipos más populares de cámaras omnidireccionales, destacan los sistemas catadióptricos, que son una combinación de cámaras convencionales y espejos, así como las lentes fisheye que se adaptan a cámaras convencionales. Para interactuar y obtener información 3D del entorno en el que se encuentra el sistema de visión omnidireccional es necesario contar con una serie de tareas básicas. Estas tareas consideran la calibración del sistema de visión, el tipo de características a utilizar como elementos relevantes en la escena, las relaciones geométricas que están presentes entre dos o más imágenes, incluyendo la combinación de diferentes tipos de cámaras. En la presente tesis tratamos todas estas tareas para los sistemas de visión omnidireccional catadióptricos. También presentamos algunas aplicaciones robóticas que usan una o más de las tareas básicas mencionadas. En particular se han tratado los siguientes problemas con sistemas catadióptricos:

- **Calibración de sistemas catadióptricos.** Existen diversos métodos de calibración de sistemas catadióptricos. Algunos requieren múltiples vistas de un patrón, otros de ciertos elementos como líneas o círculos, otros aprovechan elementos invariantes y unos más atacan el problema desde un punto de vista puramente óptico, reconstruyendo con precisión la geometría del espejo. En esta tesis presentamos el primer método lineal para calibrar sistemas catadióptricos centrales y fisheye usando correspondencias 3D-2D.
- **Características diseñadas para sistemas catadióptricos.** En la detección de características o puntos relevantes en imágenes catadióptricas, cuya geometría es más compleja que las imágenes convencionales, se siguen utilizando las técnicas desarrolladas para cámaras convencionales. Recientemente han surgido enfoques que consideran la geometría de los sistemas catadióptricos evitando problemas de aliasing y ancho de banda. En particular usan la geometría diferencial y la métrica de Riemann directamente en la imagen catadióptrica. En esta tesis presentamos un enfoque genérico para calcular el espacio de escalas de cualquier sistema de proyección central usando geometría de Riemann y el modelo genérico de la esfera.
- **Relaciones geométricas de dos vistas entre imágenes convencionales y omnidireccionales no calibradas.** El emparejamiento de imágenes de la

---

misma escena adquiridas con sensores cuya geometría es diferente ha sido un reto en los últimos años. Para dotar al emparejamiento de robustez se debe seleccionar la estrategia correcta para eliminar emparejamientos espurios. En este aspecto las restricciones geométricas han demostrado ser una buena opción. En esta tesis estudiamos en detalle tres modelos de matrices fundamentales híbridas y tres modelos de homografías híbridas.

- **Localización métrica en un ambiente de interiores.** La localización de un robot dependiendo únicamente de imágenes puede ser tratada usando estrategias de triangulación. Los requerimientos de este enfoque son parejas de imágenes con una “baseline” lo suficientemente grande que permita calcular la posición relativa de las cámaras. En esta tesis tratamos el problema de localización combinando sensores, en particular cámaras convencionales con sistemas catadióptricos, ambos no calibrados. Una cámara convencional adquiere las imágenes y previamente se ha adquirido una colección de imágenes omnidireccionales del entorno. Esto abre la oportunidad de usar estrategias usando la geometría híbrida de dos vistas.
- **Cálculo de la orientación de un sistema catadióptico.** En entornos contruidos por el hombre, la presencia de líneas rectas sobre las direcciones dominantes es un factor que se explota para calcular la orientación de un sistema de visión. Usando cámaras convencionales dicha orientación se calcula usando puntos de fuga. En imágenes omnidireccionales el mismo esquema es utilizado, pero la complejidad del cálculo de esta orientación aumenta, ya que las líneas se transforman en cónicas en este tipo de imágenes. En esta tesis atacamos el problema de la detección de proyecciones de líneas paralelas y el cálculo de los correspondientes puntos de fuga en imágenes catadiópticas.

# Abstract

Omnidirectional cameras wide field of view (FOV) is a great advantage over the conventional systems. Their usage have increased among the computer vision and robotics communities in recent years. This type of cameras allow to include the whole scene in a single view. They have been used in such different areas as surveillance, tracking, visual navigation, localization and SLAM, structure from motion, active vision, visual odometry, photogrammetry, camera networks, reconstruction of cultural heritage, etc. The more popular ones are the catadioptric systems, which are a combination of camera lenses and mirrors and the fisheye lenses which are attached to conventional cameras. When the goal of our applications is to interact and to obtain 3D information from the environment, there are several basic steps that should be performed beforehand. Among these tasks we consider the calibration of the vision system, the selection of the particular feature to extract the relevant information from the image, the two-view geometrical relations between images that can be acquired with different imaging systems. In this thesis we consider all these areas for the central catadioptric vision systems. We also deal with some robotics applications that use one or more of the previously mentioned basic tasks. In particular we analyzed the following problems involving central catadioptric systems:

- **Calibration of central catadioptric systems.** There exist several calibration methods for central catadioptric systems. Some of them require multiples views of a certain pattern or the presence of certain elements such as lines or circles in a single view. Some others take advantage of the geometric invariants as the absolute conic or compute the geometry of the mirror using a purely optical approach. In this thesis we present the first linear method to calibrate all central projection systems and fisheye cameras. This method requires 3D-2D correspondences.
- **Scale space for all central projection systems.** Nowadays the extraction of features, specially scale invariant features, from catadioptric images is performed using approaches designed for conventional cameras. Although the geometry of catadioptric cameras is more complex than the geometry of the conventional ones. Recently, some approaches which consider the geometry of the catadioptric systems have been proposed. These approaches use the Riemannian geometry to overcome the problems of bandwidth and aliasing present in the approaches based on the Fourier transform on the sphere. In this thesis we combine the Riemannian geometry with the sphere camera model to compute a generic approach to obtain the scale space of any central projection system, which is the base to compute scale space features.
- **Two-view relations between uncalibrated conventional and catadioptric cameras.** In recent years the task of matching images acquired with



---

different imaging sensors has become a challenge. In particular the combination of conventional and catadioptric images. In this thesis we explore the role of hybrid two-view relations to construct robust matching approaches. In particular we analyze three models of hybrid fundamental matrices with different dimensions. We also analyze in an analogous way three models of hybrid planar homographies.

- **Self-localization of a robot in an indoors environment.** In this thesis we tackle the problem of localizing a robot in an indoors environment using only visual information. We combine omnidirectional images and conventional images. The environment is described by a previously acquired set of omnidirectional images which is called visual memory. This visual memory can be shared with any robot. The only requirement for the robots to localize themselves on the environment is to have a conventional camera with which they can acquire images from the environment.
- **Computing the orientation of a hand-held catadioptric system.** In man-made environments the presence of straight lines associated to the dominant directions is common. This characteristic is exploited to compute the orientation of the camera with respect to the scene using the vanishing points. These vanishing points are computed from the intersection of the projections of parallel lines. In catadioptric images the same scheme is used. The problems faced with this type of cameras are more complex since the projections of straight lines are no longer lines but conics. In this thesis we deal with the problem of extracting conics from catadioptric images that represent the projections of straight lines. Then we compute the intersection of such conics from which the vanishing points are computed. Finally, the orientation of the catadioptric system is extracted from the already computed vanishing points.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Overview . . . . .	2
1.2	Contributions . . . . .	4
<b>2</b>	<b>Background</b>	<b>9</b>
2.1	Sphere Camera Model and Central Projection Systems (CPS) . . . .	9
2.1.1	Projections of Lines . . . . .	11
2.1.2	Interpretation of the Sphere Camera Model . . . . .	11
2.2	Lifted Coordinates and Matrices . . . . .	13
2.2.1	Using the $\Gamma$ Operator . . . . .	14
2.2.2	Using Symmetric Matrix Equations . . . . .	15
2.3	Generic projection to build a metric for a Riemannian manifold . . .	16
2.3.1	Paracatadioptric system: two differential metrics. . . . .	17
<b>3</b>	<b>Calibration of CPS Using a DLT-like Approach</b>	<b>21</b>
3.1	Introduction . . . . .	21
3.2	Generic Projection Matrix $P_{cata}$ . . . . .	22
3.2.1	Computation of the Generic Projection Matrix . . . . .	23
3.3	Calibration of Central Projection Systems Using $P_{cata}$ . . . . .	24
3.3.1	Other Parameters of Non-linear Calibration . . . . .	25
3.3.2	Algorithm to Compute $P_{cata}$ . . . . .	26
3.4	Theoretical and Practical Issues . . . . .	27
3.4.1	Computing Linearly $P_{cata}$ from Points on Three Planes . . .	27
3.4.2	Constraints to Estimate $P_{cata}$ from Points on Two Planes . .	28
3.5	Calibration Experiments with a Simulated Environment . . . . .	29
3.5.1	Estimation Errors for Different Camera Types . . . . .	31
3.5.2	Tilting and Distortion . . . . .	32
3.6	Experiments with Real Images using a 3D Pattern . . . . .	33
3.6.1	Intrinsic Parameters . . . . .	34
3.6.2	Extrinsic Parameters . . . . .	35
3.6.3	Structure from Motion . . . . .	35
3.7	Closure . . . . .	36

<b>4</b>	<b>Comparison of Calibration Methods for Omnidirectional Cameras</b>	<b>39</b>
4.1	Introduction . . . . .	39
4.2	Classification of Calibration Methods for Omnidirectional Cameras .	42
4.3	Calibration Methods Analyzed . . . . .	44
4.3.1	Sphere-2DPattern Approach . . . . .	44
4.3.2	Sphere-Lines Approach . . . . .	45
4.3.3	DLT-like Approach . . . . .	46
4.3.4	Distortion-2DPattern Approach . . . . .	47
4.4	Experiments . . . . .	47
4.4.1	Evaluation Criteria . . . . .	48
4.4.2	Hypercatadioptric System . . . . .	50
4.4.3	Fish-eye Lens . . . . .	53
4.4.4	Unknown-shape Catadioptric System . . . . .	55
4.4.5	Non-central Hypercatadioptric System . . . . .	57
4.4.6	Distribution of the Calibration Points in the Catadioptric Image	58
4.4.7	Discussion . . . . .	62
4.5	Closure . . . . .	64
<b>5</b>	<b>Hybrid Two-view Relations of Uncalibrated CPS</b>	<b>65</b>
5.1	Introduction . . . . .	65
5.2	Related Work . . . . .	66
5.3	Two-view Relations Between Conventional Cameras and CPS . . . .	67
5.3.1	Hybrid Fundamental Matrices . . . . .	68
5.3.2	Hybrid Homography . . . . .	71
5.4	Evaluation of the Hybrid Two-view Models . . . . .	73
5.4.1	Simulated Data . . . . .	73
5.4.2	Experiments with Real Images . . . . .	78
5.5	Automatic Matching Using Hybrid Two-view Relations . . . . .	81
5.6	Closure . . . . .	85
<b>6</b>	<b>Scale Space for CPS. Towards a Generic Camera Feature Extractor</b>	<b>87</b>
6.1	Introduction . . . . .	87
6.2	Scale Space for Central Projection Systems . . . . .	88
6.2.1	Differential Operators on Riemannian Manifolds . . . . .	88
6.2.2	Computing a Generic Metric on the Sphere . . . . .	90
6.2.3	Computing the Generic Metric on the Normalized Plane . . .	91
6.2.4	Generic Laplace-Beltrami Operator (GLBO) . . . . .	92
6.2.5	Smoothing Catadioptric Images Using the GLBO . . . . .	93
6.3	Experiments . . . . .	93
6.3.1	Smoothing of Synthetic Images . . . . .	94
6.3.2	Scale Space on Real Catadioptric Images . . . . .	96
6.3.3	Repeatability Experiment . . . . .	97
6.4	Closure . . . . .	98

## CONTENTS

---

<b>Applications of Omnidirectional Systems in 3D Localization</b>	<b>99</b>
Outline . . . . .	101
<b>7 Self-location from Monocular Uncalibrated Vision Using Reference Omniviews</b>	<b>103</b>
7.1 Introduction . . . . .	103
7.2 Our Proposal . . . . .	104
7.3 Robot location in the omnidirectional image . . . . .	105
7.4 From Omniview to Ground Location . . . . .	106
7.4.1 Uncertainty . . . . .	107
7.5 Experiments . . . . .	108
7.5.1 Epipole estimation . . . . .	108
7.5.2 Uncertainty . . . . .	109
7.5.3 Homography using real images . . . . .	109
7.6 Self-localization of the robot . . . . .	110
7.7 Closure . . . . .	111
<b>8 Orientation of a Hand-held Hypercatadioptric System in Man-made Environments</b>	<b>113</b>
8.1 Introduction . . . . .	113
8.2 Catadioptric Image Lines (CIL) Computing . . . . .	115
8.2.1 Catadioptric Image Lines Extraction . . . . .	116
8.2.2 Distance from a point to a conic . . . . .	117
8.3 Intersection of Two CILs Using the Common Self-polar Triangle . .	118
8.4 Vanishing points and Image Rectification . . . . .	119
8.4.1 Image Rectification . . . . .	120
8.5 Experiments . . . . .	120
8.6 Closure . . . . .	122
<b>9 Topological Map from Only Orientation Information</b>	<b>123</b>
9.1 Related Work . . . . .	123
9.2 Our Approach . . . . .	124
9.2.1 Computing the relative orientation . . . . .	125
9.2.2 Topological Map . . . . .	126
9.3 Experiments . . . . .	127
9.3.1 Orientation computation . . . . .	128
9.3.2 Drawbacks of the approach . . . . .	130
9.3.3 Building Topological Map . . . . .	131
9.4 Closure . . . . .	135
<b>10 Conclusions</b>	<b>137</b>
10.1 Future Work . . . . .	139



# Chapter 1

## Introduction

In recent years the use of omnidirectional cameras has widely increased among the computer vision and robotics communities. The major advantage of this type of cameras is their wide field of view (FOV) which allows them to include the whole scene in a single view. They have been used in such different areas as surveillance, tracking, visual navigation, localization and SLAM, structure from motion, active vision, visual odometry, photogrammetry, camera networks, reconstruction of cultural heritage, etc. There exist several types of omnidirectional cameras which can be classified as central and non-central. Among the non-central cameras we can find the rotating camera, which consists of a conventional camera with a mechanic system that allows it to move along a circular trajectory and to acquire images from the surroundings. Polycameras which are camera clusters of conventional cameras pointing to different directions in a particular configuration. Another type of omnidirectional systems are dioptric systems which use wide-angle lenses such as fish-eye lenses combined with conventional cameras. On the other hand, central omnidirectional cameras are those which satisfy the single-viewpoint property. This is an important property since it allows to easily calculate the directions of light rays coming into the camera. Baker and Nayar (1999) extensively studied the catadioptric systems, combinations of camera lenses and mirrors. They proved that the elliptic, parabolic and hyperbolic mirrors, combined with conventional cameras, are the only ones that ensure the single-viewpoint property, provided that the mirror is positioned appropriately relative to the camera. The two most popular of such systems are the hypercatadioptric and the paracatadioptric ones. The former is composed by a hyperbolic mirror and a perspective camera. The latter is composed by a parabolic mirror and an orthographic camera. Including fish-eye lenses, these are the three omnidirectional systems most used by the computer vision and robotics communities.

There exist several geometric and analytic models to deal with omnidirectional systems, see Sturm et al. (2011). In the case of central catadioptric systems Svoboda and Pajdla (2002) propose different models for different mirrors and give formulae for the associated epipolar geometry. Strelow et al. (2001) deal directly with the reflection properties of the rays on the mirror. A unified model was proposed by Geyer

and Daniilidis (2000), where they present the sphere camera model which allows to deal with any central catadioptric system. Later this model was extended by Barreto and Araujo (2001) and Ying and Hu (2004a). Recently, another improvement to this model, which completes the model of central catadioptric systems, was presented by Sturm and Barreto (2008). They explain the generic projection matrix, as well as the general  $15 \times 15$  fundamental matrix, and plane homographies. This model is one of the most used models in current days, since it provides important information about the mirror shape. This information can be used in different tasks such as calibration, line detection, computing geometrical invariants, computing the scale space for a particular mirror, etc. With respect to the slightly non-central systems, in particular for fish-eye lenses we can find the following approaches. Swaminathan and Nayar (1999) model this type of projections as a combination of three types of distortion. These distortions are the shift of the optical center, radial distortion and decentering distortion. Micusik and Pajdla (2003) compute the projection of 3D points to the camera plane using trigonometric functions which are linearized through Taylor series. This is done for a particular type of camera. Kannala and Brandt (2004) propose a generic model to deal with all cameras equipped with fish-eye lenses. They consider the projections as a series of odd powers of the angle between the optical axis and the incoming ray, then they complete the model by adding radial and tangential distortion. Courbon et al. (2007) propose a generic model to calibrate any fish-eye system based on the sphere camera model. Another category that should be mentioned is the generic methods that can model any arbitrary imaging system. Grossberg and Nayar (2001) propose a method based on virtual sensing elements called raxels which describe a mapping from incoming scene rays to photo-sensitive elements on the image detector. This work has inspired many works and a list of some of them can be found in Ramalingam et al. (2005).

## 1.1 Overview

In this thesis we focus on omnidirectional multi-view systems, in particular central catadioptric systems. We go from the very early step of calibration to the high level task of 3D information recovery. We also consider a very important intermediate step, which is the development of properly adapted features for central catadioptric systems. In the following paragraphs we describe in more detail the goals reached on this thesis.

We selected the sphere camera model to deal with the central catadioptric systems since it gives information about the elements of the system than other models. In particular it gives important information about the mirror shape used in the catadioptric system. We introduce this model in detail in **Chapter 2** along with an analysis of the relation between this model and actual central catadioptric systems. We also introduce the so-called *lifted coordinates*. In Sturm and Barreto (2008) the complete general theoretic projection matrix that considers all central catadioptric systems has been presented. We use this projection matrix to construct

a new approach to calibrate any single-viewpoint catadioptric camera. **Chapter 3** presents this approach, which requires twenty 3D-2D correspondences to compute the projection matrix. The 3D points must lie on at least three different planes. The projection of 3D points on a catadioptric image is performed linearly using a  $6 \times 10$  projection matrix, which uses *lifted coordinates* for image and 3D points. From this matrix, an initial estimation of the intrinsic and extrinsic parameters of the catadioptric system is obtained. We use these parameters to initialize a non-linear optimization process. This approach is also able to calibrate slightly non-central cameras, in particular, fish-eye cameras. Since reprojection error is not sufficient to determine the accuracy of the approach, we decide to perform a 3D reconstruction from two omnidirectional images.

During the development of our proposed calibration algorithm we realize that there was a lack of deep analysis and comparison of the existing calibration methods for central catadioptric systems. Moreover, for the robotics community where most tasks require to recover information from the environment, calibration of cameras is a basic step. At the same time this step should be easy to perform and reliable. In this order we perform a classification of the existing approaches. On the other hand we select those approaches which are already available as OpenSource and which do not require a complex pattern or scene to perform a comparison using synthetic and real images, so the user could select the more convenient for its particular case. **Chapter 4** presents these methods and an analysis of their advantages and drawbacks.

In **Chapter 5** we present a deep analysis of the two-view relations of uncalibrated central catadioptric systems. We particularly pay attention to the mixture of central catadioptric systems and perspective cameras, which we call *hybrid*. The two-view geometric relations we consider are the hybrid fundamental matrix and the hybrid planar homography. These matrices contain useful geometric information. We study three different types of matrices, varying in complexity depending on their capacity to deal with a single or multiple types of central catadioptric systems. The first and simplest one is designed to deal with paracatadioptric systems, the second one and more complex, considers the combination of a perspective camera and any central catadioptric system. The last one is the complete and generic model which is able to deal with any combination of central catadioptric systems. We show that the generic and most complex model sometimes is not the best option when we deal with real images. Simpler models are not as accurate as the complete model in the ideal case, but they provide a better and more accurate behavior in presence of noise, being simpler and requiring less correspondences to be computed. Finally using the best models we present the successful matching between perspective images and hypercatadioptric images introducing geometrical constraints into a robust estimation technique.

Another basic step in vision and robotics applications is feature detection/extraction. Through the years several techniques have been developed for conventional (perspective) cameras. The SIFT proposed by Lowe (2004) has become the most used feature extraction approach. This scale invariant approach is based on the approximation to the Laplacian of Gaussians (LoG) through the difference of Gaussians (DoG).



The Gaussian filtering in Euclidean computer vision, which is required to construct the scale space of the images can be computed in two ways: either using convolution with a Gaussian kernel or by implementing the linear heat flow. In **Chapter 6** we develop a new approach to compute the scale space of any omnidirectional image acquired with a central catadioptric system. We combine the sphere camera model and the partial differential equations framework on manifolds, to compute the Laplace-Beltrami (LB) operator which is a second order differential operator required to perform the Gaussian smoothing on catadioptric images.

In **Chapters 7, 8, 9** we present three applications we have developed using multi-view omnidirectional systems. The first one performs the self-localization of a robot using reference omnidirectional images in an indoors environment. We only need one omnidirectional image of the whole scene stored in the robot memory and a conventional uncalibrated on-board camera. We match the omnidirectional image and the conventional images captured by the on-board camera and compute the hybrid epipolar geometry using lifted coordinates and robust techniques. We map the epipole in the reference omnidirectional image to a ground plane through a homography in lifted coordinates also, giving the position of the robot in the planar ground, and its uncertainty. The second application consist of computing the orientation of a hand-held catadioptric system. We use the vanishing points which contain geometric information related to the orientation of the catadioptric system with respect to the dominant directions in man-made environments. The vanishing points are computed from the intersection of parallel lines. The 3D lines are projected in catadioptric images as conics, actually degenerate conics. We extract analytically the projected lines corresponding to straight lines in the scene by using the internal calibration and two image points that lie on the corresponding line projection. The last application instead of using a catadioptric system it uses a LadyBug<sup>1</sup> camera, which corresponds to the Polycamera previously described. The unified coordinate frame as well as the intrinsic parameters of each camera are known. We develop a new way to compute a topological map using only orientation information. Similar to the last application but with a calibrated camera we exploit the natural presence of lines in man-made environments in dominant directions. From the parallel lines we robustly compute the three dominant directions using vanishing points. With this information we align the camera with respect to the scene and we identify the turns in the trajectory, assuming a Manhattan world where the changes of heading in are related by multiples of 90 degrees. We also use geometrical image-pair constraints as a tool to identify the visual traversable nodes that compose the topological map.

## 1.2 Contributions

The most relevant contributions of the present thesis can be summarized as follows:

- A new method to calibrate central catadioptric systems using a single image of

---

<sup>1</sup><http://www.ptgrey.com>

## 1.2. Contributions

---

a 3D pattern. It computes linearly a generic projection matrix, using the well known Direct Linear Transformation (DLT) from which we extract an initial estimation to the intrinsic and extrinsic parameters.

### **Associated publications:**

1. [Puig et al. (2011)] L. Puig, Yalin Bastanlar, P. Sturm, J. J. Guerrero and J. Barreto, Calibration of Central Catadioptric Cameras Using a DLT-Like Approach, *International Journal of Computer Vision*, Volume 93, Number 1, pages 101–115 - 2011.
  2. [Bastanlar et al. (2008)] Y. Bastanlar, L. Puig, P. Sturm, J. J. Guerrero and J. Barreto, DLT-like Calibration of Central Catadioptric Cameras, *OMNIVIS 8th Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras*, Marseille, France, pp. 14. 2008.
- A deep comparison of calibration methods for central catadioptric systems, using synthetic and real images. We calibrate several central and non-central systems. This work also includes a classification of the existing approaches.

### **Associated publications:**

1. [Puig et al. (2012)] L. Puig, J. Bermúdez, P. Sturm and J. J. Guerrero, Calibration of Catadioptric Systems in Practice. A Comparison of Methods. *Computer Vision and Image Understanding*, Volume 116, Issue 1, January 2012, Pages 120–137.
- A new scale invariant feature adaptable for every particular central catadioptric system that can be described by the sphere camera model. We solve the heat diffusion equation through the Laplace-Beltrami operator.

### **Associated publications:**

1. [Puig and Guerrero (2011)] L. Puig and J. J. Guerrero, Scale Space for Central Catadioptric Systems. Towards a generic camera feature extractor, *13th International Conference on Computer Vision, ICCV*, Barcelona, Spain, November, 2011.
- A deep analysis of the two-view relation between uncalibrated central catadioptric systems. In particular we analyze the relation between central catadioptric systems and conventional cameras. It also includes an application to robustly match catadioptric views with conventional images.

### **Associated publications:**

1. L. Puig, J. J. Guerrero and P. Sturm, Homographies and Fundamental Matrices, Mixing Uncalibrated Omnidirectional and Conventional Cameras. *Machine Vision and Applications*, 2012. (Under review)
2. [Puig et al. (2008a)] L. Puig, J. J. Guerrero, P. Sturm, Hybrid Matching of Uncalibrated Omnidirectional and Perspective Images, *5th International*

Conference on Informatics in Control, Automation and Robotics, Madeira, Portugal, 125–128, May, 2008.

3. [Puig et al. (2008b)] L. Puig, J. J. Guerrero and P. Sturm, Matching of Omnidirectional and Perspective Images using the Hybrid Fundamental Matrix, OMNIVIS 8th Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras, Marseille, France, pp. 14. 2008.

- A localization application for hybrid uncalibrated systems. It combines the hybrid fundamental matrix and the hybrid homography to localize the conventional camera inside omnidirectional views stored in a robot visual memory.

**Associated publications:**

1. [Puig and Guerrero (2009)] L. Puig and J. J. Guerrero, Self-location from monocular uncalibrated vision using reference omniviews, IROS 2009: The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, pp. 5216–5221, 2009.

- Two applications to compute the orientation of omnidirectional systems in man-made environments using vanishing points. The first one use a central catadioptric system and a new way to estimate lines in non-vertical systems. The second one uses a Polycamera and a conventional way to compute the vanishing points but proposes a new way to compute a topological map using only orientation information.

**Associated publications:**

1. [Puig et al. (2010a)] L. Puig, J. Bermúdez and J. J. Guerrero, Self-orientation of a hand-held catadioptric system in man-made environments, ICRA 2010: IEEE International Conference on Robotics and Automation, Anchorage, Alaska, USA, pp. 2549-2555, 2010.
2. [Puig et al. (2010b)] L. Puig, Kostas Daniilidis and J. J. Guerrero, Topological map from only visual orientation information using omnidirectional cameras, IEEE ICRA 2010 Workshop on Omnidirectional Robot Vision, Anchorage, Alaska, USA, 2010.

I also collaborated with my supervisor in the direction of two undergraduate thesis projects, from which the following papers arise:

- [Rituerto et al. (2010a)] A. Rituerto and L. Puig and J. J. Guerrero, Comparison of omnidirectional and conventional monocular systems for visual SLAM (Best Paper Award), OMNIVIS - 10th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras, Zaragoza, Spain, 2010.
- [Rituerto et al. (2010b)] A. Rituerto, L. Puig and J. J. Guerrero, Visual SLAM with an Omnidirectional Camera, 20th International Conference on Pattern Recognition, Istanbul, Turkey, August, 2010.

## 1.2. Contributions

---

- [Bermúdez et al. (2010)] J. Bermúdez, L. Puig and J. J. Guerrero, Line extraction in central hyper-catadioptric systems, OMNIVIS - 10th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras, Zaragoza, Spain, 2010.



## Chapter 2

# Background

In this thesis we model the central catadioptric systems, which are a composition of cameras and mirrors, by using the sphere camera model, proposed originally by Geyer and Daniilidis (2000) and later improved by Barreto and Araujo (2001). Recently a general model was presented in Sturm and Barreto (2008), which proves the existence of a bilinear matching constraint for all central catadioptric cameras. In this chapter we present the projection of points and lines under the more recent and complete model. On the other hand we explain the relation between the sphere camera model and the actual central catadioptric systems with its constructive mirror, optics and camera parameters. Another useful tool to deal with the central catadioptric systems, and more specific with the non-linearities introduced by the sphere camera model, are the *lifted coordinates*. Barreto and Daniilidis (2006) use Veronese maps to create these *lifted coordinates*. They can represent in a single entity, the conic, the two image points generated by the projection of a 3D point under the sphere camera model. Moreover, because of the dual principle, they can also represent two lines using the same entity, the conic. These *lifted coordinates* are particularly useful when we explain the two-view relations between central catadioptric systems and the general projection under the sphere camera model. Another relevant tool to deal with central catadioptric systems are the partial differential equations, since the mirrors of the catadioptric systems can be seen as parametric surfaces. These equations will allow us to compute the metric of corresponding mirror which encodes its geometrical properties. These three elements are basic tools for the development of the majority of the contributions of the following thesis, so they are presented in this initial background chapter.

### 2.1 Sphere Camera Model

The sphere camera model used to explain central catadioptric systems was initially introduced by Geyer and Daniilidis (2000). All central catadioptric cameras can be modeled by a unit sphere and a perspective projection, such that the projection of 3D points can be performed in two steps (Fig. 2.1). First, one projects the

point onto the unit sphere, obtaining the intersection of the sphere and the line joining its center and the 3D point. There are two such intersection points, which are represented as  $\mathbf{s}_{\pm} \sim (Q_1, Q_2, Q_3, \pm \sqrt{Q_1^2 + Q_2^2 + Q_3^2})^T$ . These points are then projected in the second step, using a perspective projection  $\mathbf{P}$  resulting in two image points  $\mathbf{q}_{\pm} \sim \mathbf{P}\mathbf{s}_{\pm}$ , one of which is physically true. This model covers all central catadioptric cameras, encoded by  $\xi$ , which is the distance between the center of the perspective projection and the center of the sphere, and  $\psi$  which is the distance between the center of the sphere and the image plane. We have  $\xi = 0$  for perspective,  $\xi = 1$  for paracatadioptric and  $0 < \xi < 1$  for hypercatadioptric cameras.

Let the unit sphere be located at the origin and the optical center of the perspective camera, at the point  $\mathbf{C}_p = (0, 0, \xi)^T$ . The perspective camera is modeled by the projection matrix  $\mathbf{P} \sim \mathbf{A}_p \mathbf{R}_p (\mathbf{I} - \mathbf{C}_p)$ , where  $\mathbf{A}_p$  is its calibration matrix. We assume it is of the form

$$\mathbf{A}_p = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix} \quad (2.1)$$

with  $f$  the focal length and  $(c_x, c_y)$  the principal point. The rotation  $\mathbf{R}_p$  denotes a rotation of the perspective camera looking at the mirror (this rotation is usually very small, thus often neglected). Rotation about the  $z$ -axis can always be neglected since it is coupled with the rotation of the whole system about the  $z$ -axis. Since both intrinsic and extrinsic parameters of the *perspective* camera are intrinsic parameters for the *catadioptric* camera, we replace  $\mathbf{A}_p \mathbf{R}_p$  by a generic projective transformation  $\mathbf{K}$ . Note that the focal length of the perspective camera in the sphere model is different from the focal length of the physical camera looking at the mirror; its value is actually determined by the physical camera's focal length  $f_c$ , the mirror parameters  $(\xi, \psi)$  and the rotation between the camera and the mirror ( $\mathbf{R}_p$ ). More explicitly, the projection of the points on the sphere  $\mathbf{s}_{\pm}$  to points on the omnidirectional image are obtained as

$$\mathbf{q}_{\pm} \sim \mathbf{P}\mathbf{s}_{\pm}, \quad \text{where} \quad \mathbf{P} \sim \mathbf{K} \left( \mathbf{I} \mid \begin{array}{c} 0 \\ 0 \\ -\xi \end{array} \right), \quad (2.2)$$

with  $\mathbf{I}$ , a  $3 \times 3$  identity matrix. Giving the final definition of

$$\mathbf{q}_{\pm} \sim \mathbf{K} \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \pm \xi \sqrt{Q_1^2 + Q_2^2 + Q_3^2} \end{bmatrix}. \quad (2.3)$$

To simplify, it is usual to work with the intermediate image points  $\mathbf{r}_{\pm} \sim \mathbf{K}^{-1}\mathbf{q}_{\pm}$ . Explicitly defined as  $\mathbf{r}_{\pm} = (Q_1, Q_2, Q_3 \pm \xi \sqrt{Q_1^2 + Q_2^2 + Q_3^2})^T$ , before giving final results for the actual image points  $\mathbf{q}_{\pm}$ . The theoretical 2 image points  $\mathbf{q}_{\pm}$  can be represented in a single geometric object, which is the degenerate dual conic generated by the two points. This conic contains all lines incident to either one or both of these 2 points  $\Omega \sim \mathbf{q}_+ \mathbf{q}_-^T + \mathbf{q}_- \mathbf{q}_+^T$ .

## 2.1. Sphere Camera Model and Central Projection Systems (CPS)

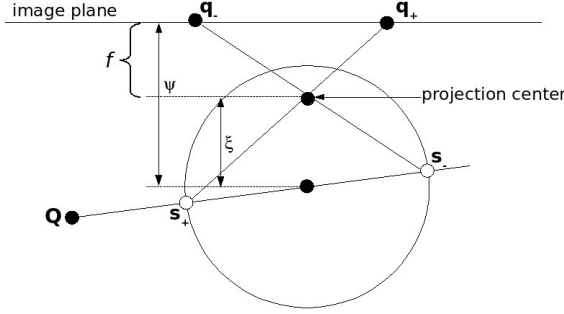


Figure 2.1: Projection of a 3D point to two image points in the sphere camera model. The z-axis of the camera coordinate system is positive upwards. The camera is looking up.

### 2.1.1 Projections of Lines

The projections of lines under the sphere camera model are explained as follows. Let  $\mathbf{\Pi} = (n_x, n_y, n_z, 0)^T$  a plane defined by a 3D line and the effective view point in the sphere camera model  $\mathbf{O}$  (see Fig. 2.2). The 2D line  $\mathbf{n}$  associated to the 3D line by  $\mathbf{P}$  can be represented as  $\mathbf{n} = (n_x, n_y, n_z)^T$ . Then, the points  $\mathbf{Q}$  lying on the 3D line are projected to points  $\mathbf{q}$ . These points satisfy  $\mathbf{n}^T \mathbf{q} = 0$  and  $\mathbf{q} = \mathbf{K}\mathbf{r}$ , so  $\mathbf{n}^T \mathbf{K}\mathbf{r} = 0$ . This equality can be written as

$$\mathbf{r}^T \mathbf{\Omega} \mathbf{r} = 0 \quad (2.4)$$

where the image conic is

$$\mathbf{\Omega} = \begin{pmatrix} n_x^2 (1 - \xi^2) - n_z^2 \xi^2 & n_x n_y (1 - \xi^2) & n_x n_z \\ n_x n_y (1 - \xi^2) & n_y^2 (1 - \xi^2) - n_z^2 \xi^2 & n_y n_z \\ n_x n_z & n_y n_z & n_z^2 \end{pmatrix} \quad (2.5)$$

and the image of the conic in the catadioptric image is

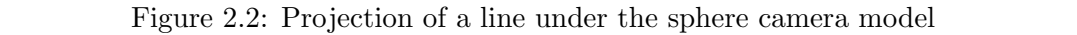
$$\hat{\mathbf{\Omega}} = \mathbf{K}^{-T} \mathbf{\Omega} \mathbf{K}^{-1}. \quad (2.6)$$

Notice that  $\mathbf{\Omega}$  is a degenerate conic when the 3D line is coplanar with the optical axis (Barreto and Araujo, 2005).

### 2.1.2 Relation Between the Real Catadioptric System and the Sphere Camera Model

Here we analyze the relation between the parameters present in a real catadioptric system and their representation in the sphere camera model. We recover the intrinsic parameters of the real catadioptric system from their counterparts in the sphere camera model. We also analyze the tilting and focal length  $f$  of the conventional camera.





The focal length in the sphere model is not the same as the focal length of the real camera, looking at the mirror. This is best seen for the paracatadioptric case, where the real camera is orthographic (infinite focal length) whereas the perspective camera in the sphere camera model has a finite focal length. The analogous is also valid for tilting parameters.

## Tilting

Focal Length  $f$

For the hypercatadioptric system, we are able to relate  $f$  with the focal length of the perspective camera in the real system,  $f_c$ . We start defining explicitly the projection matrix  $\mathbf{K}$ . Assuming image skew is zero,  $\mathbf{R}_p = \mathbf{I}$  and principal point is  $(0, 0)$ ,  $\mathbf{K}$  is given in Barreto and Araujo (2005) as

12

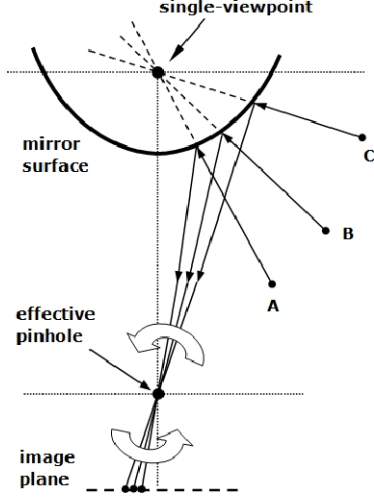


Figure 2.3: Tilt in a central catadioptric system.

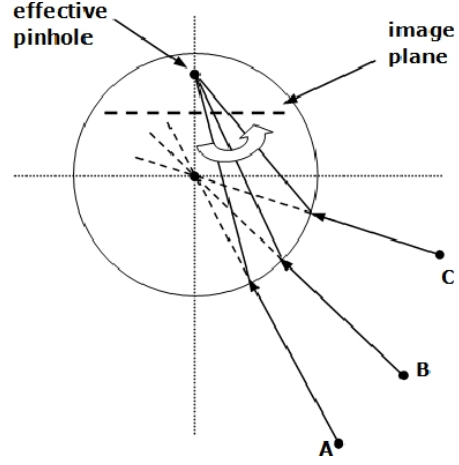


Figure 2.4: Tilt in the sphere camera model.

where  $\psi$  corresponds to the distance between the effective viewpoint and the re-projection plane (cf. Fig. 2.1). The relation between the focal lengths is  $f = (\psi - \xi)f_c$ . From the same reference Barreto and Araujo (2005) we get

$$\xi = \frac{d}{\sqrt{d^2 + 4p^2}} \quad \psi = \frac{d + 2p}{\sqrt{d^2 + 4p^2}} \quad (2.8)$$

where  $d$  is the distance between the foci of the hyperboloid and  $4p$  equals to the latus rectum. Developing the equations we obtain  $p$  in terms of  $d$  and  $\xi$ ,  $2p = d\sqrt{1 - \xi^2}/\xi$ , which is used to obtain  $\psi = \xi + \sqrt{1 - \xi^2}$ . With this final relation we can write

$$f = f_c \sqrt{1 - \xi^2} \quad (2.9)$$

from which we extract the focal length of the perspective camera in the real system

$$f_c = \frac{f}{\sqrt{1 - \xi^2}}. \quad (2.10)$$

## 2.2 Lifted Coordinates and Matrices

The derivation of (multi-)linear relations for catadioptric imagery requires the use of lifted coordinates. They allow to generalize the transformations and multiview tensors from conventional perspective images to catadioptric systems, where the projective invariant entities are quadrics instead of lines.

The Veronese map  $V_{n,d}$  of degree  $d$  maps points of  $\mathcal{P}^n$  into points of an  $m$  dimensional projective space  $\mathcal{P}^m$ , with  $m = \binom{n+d}{d} - 1$ . Consider the second order

Veronese map  $V_{2,2}$ , that embeds the projective plane into the 5D projective space, by lifting the coordinates of point  $\mathbf{q} = (q_1, q_2, q_3)^\top$  to

$$\hat{\mathbf{q}} = (q_1^2, q_1q_2, q_2^2, q_1q_3, q_2q_3, q_3^2)^\top. \quad (2.11)$$

This lifting preserves homogeneity and it is suitable to deal with quadratic functions because it discriminates the entire set of second order monomials (Barreto and Daniilidis, 2006).

As we observe, if  $\mathbf{c} = (c_1, c_2, c_3, c_4, c_5, c_6)^\top$  represents a conic, its equation  $c_1q_1^2 + c_2q_1q_2 + c_3q_2^2 + c_4q_1q_3 + c_5q_2q_3 + c_6q_3^2 = 0$ , can be written as  $\hat{\mathbf{q}}^\top \mathbf{c} = 0$ .

When the conic  $\mathbf{c}$  has the particular shape of a circle, we have  $c_2 = 0$  and  $c_1 = c_3$ . We then use the simplified lifted coordinates of a point  $\mathbf{q} = (q_1, q_2, q_3)$  in a 4-vector defined as

$$\hat{\mathbf{q}} = (q_1^2 + q_2^2, q_1q_3, q_2q_3, q_3^2)^\top. \quad (2.12)$$

There are two ways to compute this lifting, which not only involves vectors but matrices. The first one is presented by Barreto and Daniilidis (2006) and makes use of the  $\Gamma$  operator. The second one is more general and it is presented by Sturm and Barreto (2008). It makes use of symmetric matrix equations.

### 2.2.1 Lifted coordinates and matrices using the $\Gamma$ Operator

The lifting presented in (Barreto and Daniilidis, 2006) is limited to transform  $3 \times 1$  vectors into  $6 \times 1$  vectors and  $3 \times 3$  matrices into  $6 \times 6$  matrices. It makes use of the  $\Gamma$  operator, which is defined for two points  $\mathbf{q}$  and  $\bar{\mathbf{q}}$  as follows:

$$\Gamma(\mathbf{q}, \bar{\mathbf{q}}) = (q_1\bar{q}_1, \frac{q_1\bar{q}_2 + q_2\bar{q}_1}{2}, q_2\bar{q}_2, \frac{q_1\bar{q}_3 + q_3\bar{q}_1}{2}, \frac{q_2\bar{q}_3 + q_3\bar{q}_2}{2}, q_3\bar{q}_3)^\top. \quad (2.13)$$

When this operator is used with the same point we obtain the lifting of its coordinates.

$$\Gamma(\mathbf{q}, \mathbf{q}) = (q_1^2, q_1q_2, q_2^2, q_1q_3, q_2q_3, q_3^2)^\top. \quad (2.14)$$

With respect to  $3 \times 3$  matrices, the way the  $\Gamma$  operator is used to perform the lifting is the following. Matrices can be considered as linear transformations. Let us define a linear transformation  $\mathbf{L}$ , which maps points  $\mathbf{x}$  and  $\bar{\mathbf{x}}$  to points  $\mathbf{Lx}$  and  $\mathbf{L}\bar{\mathbf{x}}$ , respectively. The operator  $\Upsilon$ , that lifts the transformation  $\mathbf{L}$  from the projective plane  $\wp^2$  to the embedding space  $\wp^5$ , i.e., maps a  $3 \times 3$  matrix  $\mathbf{L}$  to a  $6 \times 6$  matrix  $\hat{\mathbf{L}}$  must satisfy

$$\Gamma(\mathbf{Lx}, \mathbf{L}\bar{\mathbf{x}}) = \Upsilon(\mathbf{L}).\Gamma(\mathbf{x}, \bar{\mathbf{x}}) \quad (2.15)$$

Such operator can be derived by algebraic manipulation. Let  $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$  be the column vectors of  $\mathbf{L}$  and  $\Gamma_{ij} = \Gamma(\mathbf{v}_i, \mathbf{v}_j)$ , the lifted representation of matrix  $\mathbf{L}$  is

$$\hat{\mathbf{L}} = \Upsilon(\mathbf{L}) = [\Gamma_{11}\Gamma_{12}\Gamma_{22}\Gamma_{13}\Gamma_{23}\Gamma_{33}]\tilde{\mathbf{D}}, \quad \text{with} \quad \tilde{\mathbf{D}} = \text{diag}\{1, 2, 1, 2, 2, 1\}. \quad (2.16)$$

## 2.2. Lifted Coordinates and Matrices

---

### 2.2.2 Lifted coordinates and matrices using symmetric matrix equations

This approach used in (Sturm and Barreto, 2008) is more general than the previously presented. It can deal with higher dimension vectors, since it is based on symmetric matrix equations. In general terms it can be described as the lower triangular part of matrix  $\mathbf{M}$ , which is obtained from the multiplication of the vector  $\mathbf{q}$  by its corresponding transpose vector  $\mathbf{q}^\top$ ,  $\mathbf{M} = \mathbf{q}\mathbf{q}^\top$ . More specifically, a vector  $\hat{\mathbf{q}}$  and matrix  $\mathbf{q}\mathbf{q}^\top$  are composed by the same elements. The former can be derived from the latter through a suitable re-arrangement of parameters. Define  $\mathbf{v}(\mathbf{U})$  as the vector obtained by stacking the columns of a generic matrix  $\mathbf{U}$  (Horn and Johnson, 1991). For the case of  $\mathbf{q}\mathbf{q}^\top$ ,  $\mathbf{v}(\mathbf{q}\mathbf{q}^\top)$  has several repeated elements because of the matrix symmetry. By left multiplication with a suitable permutation matrix  $\mathbf{S}$  that adds the repeated elements, it follows that

$$\hat{\mathbf{q}} = \mathbf{D}^{-1}\mathbf{S}\mathbf{v}(\mathbf{q}\mathbf{q}^\top), \quad (2.17)$$

with  $\mathbf{D}$  a diagonal matrix,  $D_{ii} = \sum_{j=1}^{\|\mathbf{M}\|} S_{ij}$ .

If  $\mathbf{U}$  is symmetric, then it is uniquely represented by  $\mathbf{v}_{sym}(\mathbf{U})$ , the row-wise vectorization of its lower left triangular part:

$$\mathbf{v}_{sym}(\mathbf{U}) = \mathbf{D}^{-1}\mathbf{S}\mathbf{v}(\mathbf{U}) = (U_{11}, U_{21}, U_{22}, U_{31}, \dots, U_{nn})^\top \quad (2.18)$$

Since  $\mathbf{S}$  gives us the position of the repeated elements of  $\mathbf{v}(\mathbf{U})$ , it is easy to recover  $\mathbf{v}(\mathbf{U})$  from  $\mathbf{v}_{sym}(\mathbf{U})$

$$\mathbf{v}(\mathbf{U}) = \mathbf{S}^\top \mathbf{v}_{sym}(\mathbf{U}) \quad (2.19)$$

There are two particular liftings that are useful in the present thesis. The one we explained in last section, which corresponds to the Veronese map  $V_{2,2}$ , which maps 3-vectors into 6-vectors. The second one corresponds to the Veronese map  $V_{3,2}$ , which maps 4-vectors  $\mathbf{Q} = (Q_1, Q_2, Q_3, Q_4)^\top$  into 10-vectors  $\hat{\mathbf{Q}}$ . Using the theory explained above we can define the Veronese map  $V_{2,2}$  as

$$\hat{\mathbf{q}} = \mathbf{v}_{sym}(\mathbf{q}\mathbf{q}^\top) = \left( \begin{array}{ccc|ccc} q_1^2 & q_1q_2 & q_1q_3 & & & \\ q_1q_2 & q_2^2 & q_2q_3 & & & \\ q_1q_3 & q_2q_3 & q_3^2 & & & \end{array} \right) = (q_1^2, q_1q_2, q_2^2, q_1q_3, q_2q_3, q_3^2)^\top. \quad (2.20)$$

and the Veronese map  $V_{3,2}$  as

$$\hat{\mathbf{Q}} = \mathbf{v}_{sym}(\mathbf{Q}\mathbf{Q}^T) = \begin{pmatrix} Q_1^2 & Q_1Q_2 & Q_1Q_3 & Q_1Q_4 \\ Q_1Q_2 & Q_2^2 & Q_2Q_3 & Q_2Q_4 \\ Q_1Q_3 & Q_2Q_3 & Q_3^2 & Q_3Q_4 \\ Q_1Q_4 & Q_2Q_4 & Q_3Q_4 & Q_4^2 \end{pmatrix} = \begin{pmatrix} Q_1^2 \\ Q_1Q_2 \\ Q_2^2 \\ Q_1Q_3 \\ Q_2Q_3 \\ Q_3^2 \\ Q_1Q_4 \\ Q_2Q_4 \\ Q_3Q_4 \\ Q_4^2 \end{pmatrix} \quad (2.21)$$

Let us now discuss the lifting of linear transformations (matrices), induced by lifting of points. Consider  $\mathbf{A}$  such that  $\mathbf{r} = \mathbf{A}\mathbf{q}$ . The relation  $\mathbf{r}\mathbf{r}^T = \mathbf{A}(\mathbf{q}\mathbf{q}^T)\mathbf{A}^T$  can be written as a vector mapping

$$\mathbf{v}(\mathbf{r}\mathbf{r}^T) = (\mathbf{A} \otimes \mathbf{A})\mathbf{v}(\mathbf{q}\mathbf{q}^T), \quad (2.22)$$

with  $\otimes$  denoting the Kronecker product (Horn and Johnson, 1991). Using the symmetric vectorization, we have  $\hat{\mathbf{q}} = \mathbf{v}_{sym}(\mathbf{q}\mathbf{q}^T)$  and  $\hat{\mathbf{r}} = \mathbf{v}_{sym}(\mathbf{r}\mathbf{r}^T)$ , thus, from (2.19) and (2.22):

$$\hat{\mathbf{r}} = \hat{\mathbf{A}}\hat{\mathbf{q}} = \mathbf{D}^{-1}\mathbf{S}(\mathbf{A} \otimes \mathbf{A})\mathbf{S}^T\hat{\mathbf{q}} \quad (2.23)$$

where the  $6 \times 6$  matrix  $\hat{\mathbf{A}}$ , represents the lifted linear transformation.

Finally, let us note that useful properties of the lifting of transformations are given in (Horn and Johnson, 1985, 1991):

$$\widehat{\mathbf{A}\mathbf{B}} = \hat{\mathbf{A}}\hat{\mathbf{B}} \quad \widehat{\mathbf{A}^{-1}} = \hat{\mathbf{A}}^{-1} \quad \widehat{\mathbf{A}^T} = \mathbf{D}^{-1}\hat{\mathbf{A}}^T\mathbf{D}. \quad (2.24)$$

## 2.3 Generic projection to build a metric for a Riemannian manifold

The mirrors of central catadioptric systems, which are composed of cameras and mirrors, can be seen as parametric surfaces  $\mathcal{M}$  in  $\mathbb{R}^3$ . Their geometrical properties are encoded in their induced Riemannian metric  $g_{ij}$ , for which the partial differential equations are a very versatile tool that is well suited to cope with the computation of such metric.

In the catadioptric systems, any ray incident in one of the foci of the mirror is projected to the other focus. In the case of the paracatadioptric system this focus is located at the infinity. A pixel is created where the reflected light ray intersects the camera plane.

Let  $(q_1, q_2)$  and  $(Q_1, Q_2, Q_3)$  be coordinates in the image plane, which we consider an open subset  $\Xi \subseteq \mathbb{R}^2$ , and on the mirror surface, respectively. The image formation

### 2.3. Generic projection to build a metric for a Riemannian manifold

---

process previously described induces a mapping between manifolds from the surface of the mirror to the camera plane

$$\begin{aligned}\Psi : \mathcal{M} &\rightarrow \Xi \\ (Q_1, Q_2, Q_3) &\rightarrow (q_1, q_2)\end{aligned}\tag{2.25}$$

This mapping allows to transport the metric  $g_{ij}(Q_1, Q_2, Q_3)$  and to provide the image with a pullback metric  $h_{ij}(q_1, q_2)$

$$h_{kl} = \frac{\partial}{\partial q_k} Q_r \frac{\partial}{\partial q_l} Q_s g_{rs}, \quad k, l \in \{1, 2\}, r, s \in \{0, 1, 2\}, \tag{2.26}$$

the Einstein's convention summation is used. Notice that the image plane  $(\Xi, h)$  carries a very specific geometry encoded in its metric  $h_{ij}$ . This geometry is inherited from that of the mirror  $(\mathcal{M}, g)$ , but it uses a more conventional domain  $\Xi \subseteq \mathbb{R}^2$ . Therefore with this simpler and geometrically accurate parameterization we are able to perform computations on  $\mathcal{M}$  easily.

Omnidirectional images can be treated as scalar fields on parametric manifolds. With this idea Riemannian geometry is used to derive generalizations of concepts of smoothing and scale-space for this particular type of images. A general framework is derived based on energy minimization and partial differential equations. The parameterization is used to compute efficiently these operations *directly from the camera plane* and at the same time respecting the complex geometry of the manifold.

#### 2.3.1 Paracatadioptric system: two differential metrics.

Here we show two ways to obtain the differential metric representing the parabolic mirror. The first one represents the ray reflected in the mirror surface and then projected to the image plane through an orthographic projection. The second one uses the inverse stereographic projection which reconstructs a perfect viewpoint at the focus of the mirror Geyer and Daniilidis (2000). The same sensor image is endowed with different geometries, depending on which information one wishes to process.

##### On the parabolic mirror

The parabolic mirror is defined by the paraboloid  $\mathcal{P}^2$  (Fig. 2.5(a)), which is a quadratic surface and it is represented as  $Q_1 = Q_2^2 + Q_3^2$ . Defining this paraboloid in polar coordinates  $(r, \varphi)$  we have

$$\begin{aligned}Q_1 &= r^2 \\ Q_2 &= r \cos \varphi, \quad 0 \leq \varphi \leq 2\pi, \quad r \geq 0 \\ Q_3 &= r \sin \varphi\end{aligned}\tag{2.27}$$

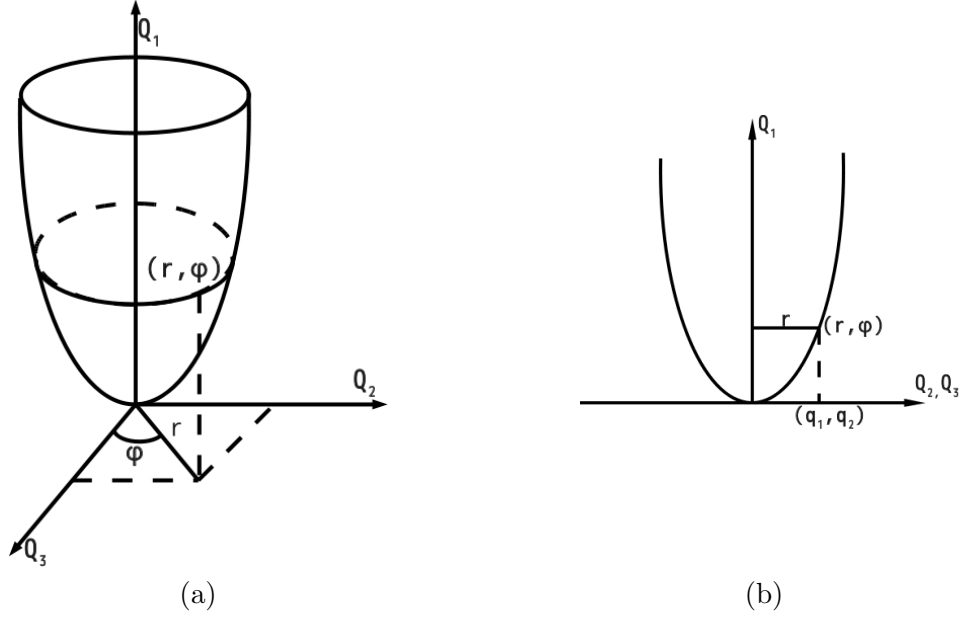


Figure 2.5: Geometry of the paraboloid  $\mathcal{P}^2$  embedded in  $\mathbb{R}^3$ . (a) The original coordinates  $(Q_1, Q_2, Q_3)$  expressed in polar coordinates  $(r, \phi)$  are mapped to coordinates  $(q_1, q_2)$  after focal projection. (b) The focal projection coincides with a simple orthographic projection in this case.

To construct the metric of the surface we need to define the Euclidean line element, which for the paraboloid in cartesian and polar coordinates is

$$dl^2 = dQ_1^2 + dQ_2^2 + dQ_3^2 = (1 + 4r^2)dr^2 + r^2d\phi^2 \quad (2.28)$$

from which we extract the metric on  $\mathcal{P}^2$  as

$$g_{ij}(r, \phi) = \begin{pmatrix} 1 + 4r^2 & 0 \\ 0 & r^2 \end{pmatrix} \quad (2.29)$$

Since the focal projection  $\Phi$  from the mirror  $\mathcal{M}$  to the camera plane is simply the orthographic projection (see Fig.2.5(b)), i.e.,  $(q_1, q_2) = (Q_2, Q_3)$ . The induced metric on the camera plane in  $(q_1, q_2)$  coordinates is needed. Since  $r^2 = q_1^2 + q_2^2$  and  $\phi = \arctan \frac{q_2}{q_1}$ . The 2D Euclidean line element is

$$dl^2 = (1 + 4q_1^2)dq_1^2 + 8q_1q_2dq_1dq_2 + (1 + 4q_2^2)dq_2^2 \quad (2.30)$$

which corresponds to the following metric

$$h_{ij}(q_1, q_2) = \begin{pmatrix} 1 + 4q_1^2 & 4q_1q_2 \\ 4q_1q_2 & 1 + 4q_2^2 \end{pmatrix}. \quad (2.31)$$

### 2.3. Generic projection to build a metric for a Riemannian manifold

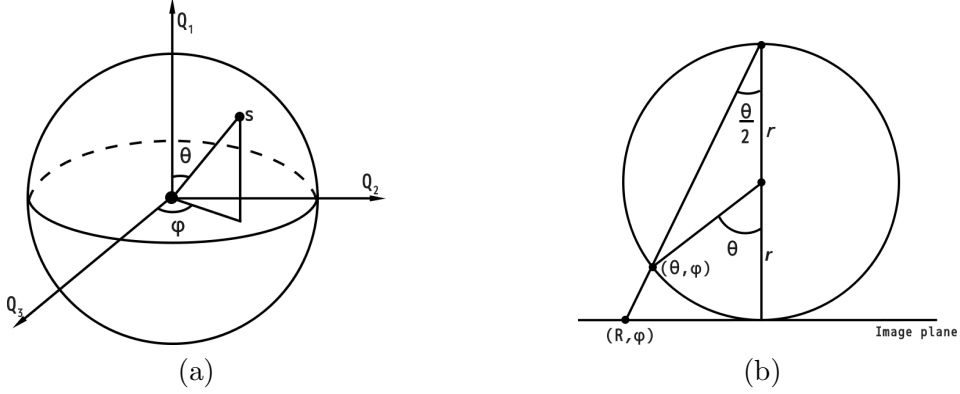


Figure 2.6: Geometry of the 2-sphere  $\mathbb{S}^2$ . (a) Spherical polar coordinates. (b) Stereographic projection.

#### Using the projection on the sphere

Geyer and Daniilidis (2001a) showed that the paracatadioptric projection is equivalent to the inverse stereographic projection. In this case we need to introduce another manifold, which corresponds to the two-dimensional sphere  $\mathbb{S}^2$  in  $\mathbb{R}^3$ . Notice that using this approach not only the paracatadioptric systems can be modeled through a projection on the sphere, but all central projection systems.

We now require to link the geometry of  $\mathbb{S}^2$  to the one of the sensor image. This will allow to process the spherical image directly using the sensor output. Consider a sphere of radius  $r$  (Fig. 2.6(a)). A point  $s$  on  $\mathbb{S}^2$  is defined in cartesian and polar coordinates as

$$(s_1, s_2, s_3) = r(\cos \theta, \sin \theta \sin \varphi, \sin \theta \cos \varphi), \quad \text{with } \theta \in [0, \pi), \varphi \in [0, 2\pi) \quad (2.32)$$

The Euclidean line element can be represented in cartesian and polar coordinates

$$dl^2 = ds_1^2 + ds_2^2 + ds_3^2 = r^2(d\theta^2 + \sin^2 \theta d\varphi^2) \quad (2.33)$$

The stereographic projection (see Fig. 2.6(b)) sends a point  $(\theta, \varphi)$  on the sphere to the point with polar coordinates  $(R, \varphi)$  in the plane, with  $\varphi = \varphi$  and  $R = 2r \tan \frac{\theta}{2}$ . Then the terms in (2.33) are

$$d\theta^2 = \frac{16r^2}{(4R^2 + r^2)^2} dR^2, \quad \sin^2(\theta) = \frac{16r^2}{(R^2 + 4r^2)^2} R^2 \quad (2.34)$$

simplifying

$$dl^2 = \frac{16r^4}{(R^2 + 4r^2)^2} (dR^2 + R^2 d\varphi^2) \quad (2.35)$$



Let  $(q_1, q_2) \in \mathbb{R}^2$  on the sensor plane define cartesian coordinates, where  $R^2 = q_1^2 + q_2^2$ ,  $\varphi = \arctan \frac{q_2}{q_1}$  and  $r = 1$ . The line element becomes

$$dl^2 = \frac{16}{(4 + q_1^2 + q_2^2)^2} (dq_1^2 + dq_2^2) \quad (2.36)$$

giving the Riemannian metric  $h_{ij}$

$$h_{ij}(q_1, q_2) = \begin{pmatrix} \frac{16}{(4+q_1^2+q_2^2)^2} & 0 \\ 0 & \frac{16}{(4+q_1^2+q_2^2)^2} \end{pmatrix} \quad (2.37)$$

and its corresponding inverse  $h^{ij}$

$$h^{ij}(q_1, q_2) = \begin{pmatrix} \frac{(4+q_1^2+q_2^2)^2}{16} & 0 \\ 0 & \frac{(4+q_1^2+q_2^2)^2}{16} \end{pmatrix} \quad (2.38)$$

Metrics that differ only by a multiplicative factor are conformal equivalent. The stereographic projection endows the plane with a metric conformal to the regular Euclidean metric.

Observe that metrics (2.31) and (2.37) are different. This is because the first one corresponds to light intensity on the parabolic mirror while the second one corresponds to light intensity perceived by an observer at the focus of the mirror. The same sensor image is thus endowed with different geometries.

## Chapter 3

# Calibration of Central Projection Systems Using a DLT-like Approach

*In this chapter, we present a new calibration technique that is valid for all single-viewpoint catadioptric cameras. We are able to represent the projection of 3D points on a catadioptric image linearly with a  $6 \times 10$  projection matrix, which uses lifted coordinates for image and 3D points. This projection matrix can be linearly computed from 3D-2D correspondences (minimum 20 points distributed in three different planes). We show how to decompose it to obtain intrinsic and extrinsic parameters. Moreover, we use this parameter estimation followed by a non-linear optimization to calibrate various types of cameras. Our results are based on the sphere camera model. We test our method both with simulations and real images, and we analyze the results performing a 3D reconstruction from two omnidirectional images.*

### 3.1 Introduction

Since their introduction to the computer vision community, catadioptric omnidirectional cameras have been utilized in many application areas such as surveillance, tracking, tele-presence, visual navigation, localization and SLAM, structure from motion, active vision, visual odometry, photogrammetry, camera networks, reconstruction of cultural heritage, among others.

Camera calibration is essential when we want to extract metric information from images. It establishes a relationship between the 3D rays and their corresponding pixels in the image. This relationship makes possible to measure distances in a real world from their projections on the images Faugeras (1993). Camera calibration is basically composed of two steps. The first step consists of modeling the physical and optical behavior of the sensor through a geometric-mathematical model. There exist several approaches that propose different models to deal with central catadioptric systems Kang (2000); Svoboda and Pajdla (2002); Scaramuzza et al. (2006); Toepfer

and Ehlgen (2007); Geyer and Daniilidis (2000). The second step consists of estimating the parameters that compose this model using direct or iterative methods. These parameters are of two types, intrinsic and extrinsic. The intrinsic parameters basically consider how the light is projected through the mirror and the lens onto the image plane of the sensor. The extrinsic parameters describe the position and orientation of the catadioptric system with respect to a world coordinate system.

Several methods have been proposed for calibration of catadioptric systems. Some of them consider estimating the parameters of the parabolic (Geyer and Daniilidis, 2002a; Kang, 2000), hyperbolic (Orghidan et al., 2003) and conical (Cauchois et al., 1999) mirrors together with the camera parameters. Some others separate the geometry of the mirror from the calibration of the conventional camera (Svoboda and Pajdla, 2002; Morel and Fofi, 2007). Calibration of outgoing rays based on a radial distortion model is another approach. Kannala and Brandt (2004) used this approach to calibrate fisheye cameras. Scaramuzza et al. (2006) and Tardif et al. (2006) extended the approach to include central catadioptric cameras as well. Mei and Rives (2007), on the other hand, developed another Matlab calibration toolbox that estimates the parameters of the sphere camera model. Parameter initialization is done by user input, namely, the location of the principal point and depiction of a real world straight line in the omnidirectional image (for focal length estimation).

In this chapter a new method to calibrate central catadioptric systems is proposed. An ongoing proposal was presented at the 8th OMNIVIS, Bastanlar et al. (2008). The improved final version Puig et al. (2011), which also includes: the studying of the use only two planes and additional constraints to perform the calibration; the relation between the intrinsic parameters of the sphere camera model and the actual camera; and a 3D reconstruction experiment to show the effectiveness of the approach; was published on the International Journal of Computer Vision. In this work the calibration theory of central cameras proposed by Sturm and Barreto (2008) is put into practice. We compute the generic projection matrix,  $P_{cata}$ , with 3D-2D correspondences, using a straightforward DLT-like (Direct Linear Transform Abdel-Aziz and Karara (1971)) approach, i.e. by solving a linear equation system. Then, we decompose  $P_{cata}$  to estimate intrinsic and extrinsic parameters. Having these estimates as initial values of system parameters, we optimize the parameters based on minimizing the reprojection error. A software version of our method is available at my Web page<sup>1</sup>. When compared with alternate techniques our approach has the advantage of not requiring input for parameter initialization and being able to calibrate perspective cameras as well. Although it only requires a single catadioptric image, it must be of a 3D calibration object.

### 3.2 Generic Projection Matrix $P_{cata}$

As explained in section 2.1, a 3D point is mathematically projected to two image points. Sturm and Barreto (2008) represented these two 2D points via the degenerate

<sup>1</sup><http://webdiis.unizar.es/~lpuig/DLT0mniCalibration/Toolbox.tar.gz>

### 3.2. Generic Projection Matrix $P_{cata}$

dual conic generated by them, i.e. the dual conic containing exactly the lines going through at least one of the two points. Let the two image points be  $\mathbf{q}_+$  and  $\mathbf{q}_-$ ; the dual conic is then given by

$$\Omega \sim \mathbf{q}_+ \mathbf{q}_+^\top + \mathbf{q}_- \mathbf{q}_-^\top \quad (3.1)$$

The vectorized matrix of the conic can be computed as shown below using the lifted 3D point coordinates, intrinsic and extrinsic parameters.

$$\mathbf{v}_{sym}(\Omega) \sim \hat{\mathbf{K}}_{6 \times 6} \mathbf{X}_\xi \hat{\mathbf{R}}_{6 \times 6} (\mathbf{I}_6 \quad \mathbf{T}_{6 \times 4}) \hat{\mathbf{Q}}_{10} \quad (3.2)$$

Here,  $\mathbf{R}$  represents the rotation of the catadioptric camera.  $\mathbf{X}_\xi$  and  $\mathbf{T}_{6 \times 4}$  depend only on the sphere model parameter  $\xi$  and the position of the catadioptric camera  $\mathbf{C} = (t_x, t_y, t_z)$  respectively, as shown here:

$$\mathbf{X}_\xi = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ -\xi^2 & 0 & -\xi^2 & 0 & 0 & 1 - \xi^2 \end{pmatrix} \quad (3.3)$$

$$\mathbf{T}_{6 \times 4} = \begin{pmatrix} -2t_x & 0 & 0 & t_x^2 \\ -t_y & -t_x & 0 & t_x t_y \\ 0 & -2t_y & 0 & t_y^2 \\ -t_z & 0 & -t_x & t_x t_z \\ 0 & -t_z & -t_y & t_y t_z \\ 0 & 0 & -2t_z & t_z^2 \end{pmatrix} \quad (3.4)$$

Thus, a  $6 \times 10$  **catadioptric projection matrix**,  $P_{cata}$ , can be expressed by its intrinsic and extrinsic parameters, like the projection matrix of a perspective camera.

$$P_{cata} = \underbrace{\hat{\mathbf{K}} \mathbf{X}_\xi}_{\mathbf{A}_{cata}} \underbrace{\hat{\mathbf{R}}_{6 \times 6} (\mathbf{I}_6 \quad \mathbf{T}_{6 \times 4})}_{\mathbf{T}_{cata}} \quad (3.5)$$

#### 3.2.1 Computation of the Generic Projection Matrix

Here we show the way used to compose the equations using 3D-2D correspondences to compute  $P_{cata}$ . Analogous to the perspective case ( $[\mathbf{q}]_\times \mathbf{P} \mathbf{Q} = \mathbf{0}$ ), we write the constraint based on the lifted coordinates Sturm and Barreto (2008):

$$[\widehat{\mathbf{q}}]_\times P_{cata} \hat{\mathbf{Q}} = \mathbf{0} \quad (3.6)$$

This is a set of 6 linear homogeneous equations in the coefficients of  $P_{cata}$ . Using the Kronecker product, this can be written in terms of the 60-vector  $\mathbf{p}_{cata}$  containing the 60 coefficients of  $P_{cata}$ :

$$(\hat{\mathbf{Q}}^\top \otimes [\widehat{\mathbf{q}}]_\times) \mathbf{p}_{cata} = \mathbf{0}_6 \quad (3.7)$$

Stacking these equations for  $n$  3D-2D correspondences gives a system of equations of size  $6n \times 60$ , which can be solved by linear least squares, e.g. using the SVD (Singular Value Decomposition). Note that the minimum number of required correspondences is 20: a  $3 \times 3$  skew symmetric matrix has rank 2, its lifted counterpart rank 3. Therefore, each correspondence provides only 3 independent linear constraints.

### 3.3 Generic Projection Matrix and Calibration

The calibration process consists of getting the intrinsic and extrinsic parameters of a camera. Once  $\mathbf{P}_{cata}$  has been computed from point correspondences, our purpose is to decompose  $\mathbf{P}_{cata}$  as in (5.10). Consider first the leftmost  $6 \times 6$  submatrix of  $\mathbf{P}_{cata}$ :

$$\mathbf{P}_s \sim \hat{\mathbf{K}}\mathbf{X}_\xi\hat{\mathbf{R}} \quad (3.8)$$

Let us define  $\mathbf{M} = \mathbf{P}_s\mathbf{D}^{-1}\mathbf{P}_s^T$ . Using the properties given in (2.24) and knowing that for a rotation matrix  $\mathbf{R}^{-1} = \mathbf{R}^T$ , we can write  $\hat{\mathbf{R}}^{-1} = \mathbf{D}^{-1}\hat{\mathbf{R}}^T\mathbf{D}$ . And from that we obtain  $\mathbf{D}^{-1} = \hat{\mathbf{R}}\mathbf{D}^{-1}\hat{\mathbf{R}}^T$  which we use to eliminate the rotation parameters:

$$\mathbf{M} \sim \hat{\mathbf{K}}\mathbf{X}_\xi\hat{\mathbf{R}}\mathbf{D}^{-1}\hat{\mathbf{R}}^T\mathbf{X}_\xi^T\hat{\mathbf{K}}^T = \hat{\mathbf{K}}\mathbf{X}_\xi\mathbf{D}^{-1}\mathbf{X}_\xi^T\hat{\mathbf{K}}^T \quad (3.9)$$

Equation (3.9) holds up to scale, i.e. there is a  $\lambda$  with  $\mathbf{M} = \lambda\hat{\mathbf{K}}\mathbf{X}_\xi\mathbf{D}^{-1}\mathbf{X}_\xi^T\hat{\mathbf{K}}^T$ . For initialization we assume that the camera is well aligned with the mirror axis, i.e. assume that  $\mathbf{R}_p = \mathbf{I}$ , thus  $\mathbf{K} = \mathbf{A}_p = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}$ .

We then use some elements of  $\mathbf{M}$  to extract the intrinsic parameters:

$$\begin{aligned} \mathbf{M}_{16} &= \lambda \left( -(f^2\xi^2) + c_x^2(\xi^4 + c_x(1 - \xi^2)^2) \right) \\ \mathbf{M}_{44} &= \lambda \left( \frac{f^2}{2} + c_x^2(2\xi^4 + (1 - \xi^2)^2) \right) \\ \mathbf{M}_{46} &= \lambda c_x(2\xi^4 + (1 - \xi^2)^2) \\ \mathbf{M}_{56} &= \lambda c_y(2\xi^4 + (1 - \xi^2)^2) \\ \mathbf{M}_{66} &= \lambda (2\xi^4 + (1 - \xi^2)^2) \end{aligned} \quad (3.10)$$

The intrinsic parameters are computed as follows:

$$\begin{aligned} c_x &= \frac{\mathbf{M}_{46}}{\mathbf{M}_{66}} & c_y &= \frac{\mathbf{M}_{56}}{\mathbf{M}_{66}} & \xi &= \sqrt{\frac{\frac{\mathbf{M}_{16}}{\mathbf{M}_{66}} - c_x^2}{-2(\frac{\mathbf{M}_{44}}{\mathbf{M}_{66}} - c_x^2)}} \\ f &= \sqrt{2(2\xi^4 + (1 - \xi^2)^2) \left( \frac{\mathbf{M}_{44}}{\mathbf{M}_{66}} - c_x^2 \right)} \end{aligned} \quad (3.11)$$

After extracting the intrinsic part  $\mathbf{A}_{cata}$  of the projection matrix, we are able to obtain the  $6 \times 10$  extrinsic part  $\mathbf{T}_{cata}$  by multiplying  $\mathbf{P}_{cata}$  with the inverse of  $\mathbf{A}_{cata}$ :

$$\mathbf{T}_{cata} = \hat{\mathbf{R}}_{6 \times 6}(\mathbf{I}_6\mathbf{T}_{6 \times 4}) \sim (\hat{\mathbf{K}}\mathbf{X}_\xi)^{-1}\mathbf{P}_{cata} \quad (3.12)$$

Hence, the leftmost  $6 \times 6$  part of  $T_{cata}$  will be the estimate of the lifted rotation matrix  $\hat{R}_{est}$ . If we multiply the inverse of this matrix with the rightmost  $6 \times 4$  part of  $T_{cata}$ , we obtain an estimate for the translation ( $T_{6 \times 4}$ ). This translation should have an ideal form as given in (3.4) and we are able to identify translation vector elements  $(t_x, t_y, t_z)$  from it straightforwardly.

We finally have to handle the fact that the estimated  $\hat{R}_{est}$  will not, in general, be an exact lifted rotation matrix. This lifted rotation matrix in particular is oversized since it considers the lifting of a full rotation matrix  $\hat{R} = \hat{R}_z(\gamma)\hat{R}_y(\beta)\hat{R}_x(\alpha)$ . For illustration in (3.13) we show the lifting of a rotation matrix around the  $x$ -axis.

$$\hat{R}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \cos \alpha & 0 & -\sin \alpha & 0 & 0 \\ 0 & 0 & \cos^2 \alpha & 0 & -2 \cos \alpha \sin \alpha & \sin^2 \alpha \\ 0 & \sin \alpha & 0 & \cos \alpha & 0 & 0 \\ 0 & 0 & \cos \alpha \sin \alpha & 0 & \cos^2 \alpha - \sin^2 \alpha & -\cos \alpha \sin \alpha \\ 0 & 0 & \sin^2 \alpha & 0 & 2 \cos \alpha \sin \alpha & \cos^2 \alpha \end{pmatrix} \quad (3.13)$$

Since  $P_{cata}$  has been estimated up to scale it is impossible to extract the rotation components from single elements of  $\hat{R}_{est}$ . To deal with this problem we algebraically manipulate the ratios between the elements of this lifted matrix and we extract the angles one by one. First, we recover the rotation angle around the  $z$  axis,  $\gamma = \tan^{-1} \left( \frac{\hat{R}_{est,51}}{\hat{R}_{est,41}} \right)$ . Then,  $\hat{R}_{est}$  is modified by being multiplied by the inverse of the rotation around the  $z$  axis,  $\hat{R}_{est} = \hat{R}_z^{-1}(\gamma)\hat{R}_{est}$ . Then, the rotation angle around the  $y$  axis,  $\beta$ , is estimated and  $\hat{R}_{est}$  is modified  $\beta = \tan^{-1} \left( \frac{-\hat{R}_{est,52}}{\hat{R}_{est,22}} \right)$ ,  $\hat{R}_{est} = \hat{R}_y^{-1}(\beta) \hat{R}_{est}$ . Finally, the rotation angle around the  $x$  axis,  $\alpha$ , is estimated as  $\alpha = \tan^{-1} \left( \frac{\hat{R}_{est,42}}{\hat{R}_{est,22}} \right)$ .

#### 3.3.1 Other Parameters of Non-linear Calibration

The intrinsic and extrinsic parameters extracted in closed-form in Section 3.3 are not always adequate to model a real camera. Extra parameters are needed to correctly model the catadioptric system, namely, tilting and lens distortions.

As mentioned before  $\hat{K} = \hat{A}_p \hat{R}_p = \hat{A}_p \hat{R}_p$  where  $R_p$  is the rotation between camera and mirror coordinate systems, i.e. tilting. Tilting has only  $R_x$  and  $R_y$  components, because rotation around the optical axis,  $R_z$ , is coupled with the external rotation around the  $z$  axis of the entire catadioptric system. Note that tilting angles of the sphere camera model are not equivalent to the tilting angles of the actual perspective camera looking at the mirror.

As is well known, imperfections due to lenses are modeled as distortions for camera calibration. Radial distortion models contraction or expansion with respect to the image center and tangential distortion models lateral effects. To add these distortion effects to our calibration algorithm, we employed the approach of Heikkila and Silven (1997).

Radial distortion:

$$\begin{aligned}\Delta x &= x(k_1 r^2 + k_2 r^4 + k_3 r^6 + \dots) \\ \Delta y &= y(k_1 r^2 + k_2 r^4 + k_3 r^6 + \dots)\end{aligned}\tag{3.14}$$

where  $r = \sqrt{x^2 + y^2}$  and  $k_1, k_2, \dots$  are the radial distortion parameters. We observe that estimating two parameters is enough for an adequate estimation.

Tangential distortion:

$$\begin{aligned}\Delta x &= 2p_1 xy + p_2(r^2 + 2x^2) \\ \Delta y &= p_1(r^2 + 2y^2) + 2p_2 xy\end{aligned}\tag{3.15}$$

where  $r = \sqrt{x^2 + y^2}$  and  $p_1, p_2$  are the tangential distortion parameters.

Once we have identified all the parameters to be estimated we perform a non-linear optimization to compute the whole model. We use the Levenberg-Marquardt method (LM)<sup>2</sup>. The minimization criterion is the root mean square (RMS) of distance between a measured image point and its reprojected correspondence. Since the projection equations we use map 3D points to dual image conics, we have to extract the two potential image points from it. The one closer to the measured point is selected and then the reprojection error measured. We take as initial values the parameters obtained from  $P_{cata}$  and initialize the additional 4 distortion parameters and the tilt angles in  $R_p$ , by zero.

### 3.3.2 Algorithm to Compute $P_{cata}$

Here we summarize the algorithm used to compute the generic projection matrix  $P_{cata}$ .

1. **Linear Solution.** Using 3D-2D correspondences we compute  $P_{cata}$  by a DLT-like approach.
2. **Intrinsic/Extrinsic Parameter Extraction.** Assuming that the perspective camera is perfectly aligned with the mirror axis, i.e. there is no tilting and that the images are not distorted. We extract from the linear solution, the intrinsic  $(\xi, f, c_x, c_y)$  and extrinsic  $(\alpha, \beta, \gamma, t_x, t_y, t_z)$  parameters in closed-form.
3. **Initialization Vector.** An initialization vector is constructed with the extracted parameters. Two parameters are added to consider the tilting angles  $(r_x, r_y)$  and four more corresponding to the radial  $(k_1, k_2)$  and tangential  $(p_1, p_2)$  distortion.
4. **Non-linear Optimization Process.** Using this vector as an initialization vector, we perform a non-linear optimization process using the LM algorithm. The minimization criterion is the reprojection error.

---

<sup>2</sup>Method provided by the function `lsqnonlin` in Matlab

### 3.4 Theoretical and Practical Issues

In the last section we explained that twenty 3D-2D correspondences are enough to compute the calibration of the central catadioptric systems. In principle these twenty correspondences can be located anywhere inside the FOV of the catadioptric system. Since we want to construct a feasible calibration system based on planar patterns we restrict the 3D points to be located in planes. From simulations we observed that the minimum number of planes where the 3D points should be located is three in the general case. In particular, two planes can be used to compute  $P_{cata}$  if several constraints are imposed, but the simplicity of using linear equations is lost.

Since we restrict the calibration points to lie on planes (planar grid-based calibration) some degeneracies can appear if the calibration points are located in a particular configuration. Something similar to the pinhole camera case with the twisted cubic (Buchanan, 1988), for which calibration fails even if the points lie on more than two planes. However, a complete analysis of such degeneracies is out of the scope of this thesis.

In this section we present a proof that points lying in three different planes are required to linearly and uniquely compute the generic projection matrix  $P_{cata}$ . We also show that under several assumptions we can compute  $P_{cata}$  from points lying in just two planes.

#### 3.4.1 Three Planes Are Needed to Compute $P_{cata}$ Using Linear Equations.

Here we show that in order to compute  $P_{cata}$ , the 3D calibration points must lie in at least 3 different planes. We first prove that two planes are not sufficient. Let  $\Pi_1$  and  $\Pi_2$  be the two planes. Hence, each calibration point  $\mathbf{Q}$  satisfies  $(\Pi_1^T \mathbf{Q}) (\Pi_2^T \mathbf{Q}) = 0$ . This can be written as a linear constraint on the lifted calibration points:  $\mathbf{p}^T \tilde{\mathbf{Q}} = 0$ , where the 10-vector  $\mathbf{p}$  depends exactly on the two planes. Thus, if  $P_{cata}$  is the true  $6 \times 10$  projection matrix, then adding some multiple of  $\mathbf{p}^T$  to any row of  $P_{cata}$  gives another  $6 \times 10$  projection matrix,  $\bar{P}_{cata}$ , which maps the calibration points to the same image entities as the true projection matrix. We may write the ambiguity as

$$\bar{P}_{cata} = P_{cata} + \mathbf{v} \mathbf{p}^T \quad (3.16)$$

where  $\mathbf{v}$  is a 6-vector and represents the six degrees of freedom (DoF) on  $P_{cata}$  that can not be recovered using only linear projection equations and calibration points located in only two planes. This is not the case for perspective cameras, where two planes are enough to compute the  $3 \times 4$  perspective projection matrix.

For three planes, there is no linear equation as above that holds for all calibration points. Hence, also supported by our experiments, it seems plausible that three planes are sufficient for uniquely computing the projection matrix. Note that by planes we do not mean that calibration grids have to be composed of three or more planar grids. The planes can be virtual: whenever it is possible to fit the two planes to the whole set of 3D points,  $P_{cata}$  can not be computed.



### 3.4.2 Adding Constraints to Estimate the Projection Matrix from Points on Two Planes Only

In the last section we observe that to compute  $P_{cata}$  linearly and uniquely, 3D points must be sufficiently well distributed, such that no two planes contain all of them. In this section we analyze what prior information allows nevertheless to compute the calibration parameters using two planes. We know by (3.16) that the true projection matrix is related to any other solution by

$$P_{cata} = \bar{P}_{cata} - \mathbf{v}\mathbf{p}^T \quad (3.17)$$

Consider the equation to eliminate the extrinsic parameters:

$$\mathbf{M} \sim \mathbf{P}_s \mathbf{D}^{-1} \mathbf{P}_s^T \quad (3.18)$$

where  $\mathbf{P}_s$  is the leftmost  $6 \times 6$  submatrix of  $\mathbf{P}_{cata}$ . Now we redefine it as follows:

$$\mathbf{M} \sim (\bar{\mathbf{P}}_s - \mathbf{v}\mathbf{p}_s^T) \mathbf{D}^{-1} (\bar{\mathbf{P}}_s - \mathbf{v}\mathbf{p}_s^T)^T \quad (3.19)$$

where  $\bar{\mathbf{P}}_s$  is the leftmost  $6 \times 6$  submatrix of  $\bar{\mathbf{P}}_{cata}$  and  $\mathbf{p}_s$  is the first 6 elements of the 10-vector  $\mathbf{p}$ . Assuming that the two planes are perpendicular to each other, we can write  $\mathbf{\Pi}_1 = [1, 0, 0, 0]^T$  and  $\mathbf{\Pi}_2 = [0, 1, 0, 0]^T$  which gives us  $\mathbf{p}_s = [0, 1, 0, 0, 0, 0]^T$  (we obtain  $\mathbf{p}$  by  $\mathbf{v}_{sym}(\mathbf{\Pi}_1 \mathbf{\Pi}_2^T + \mathbf{\Pi}_2 \mathbf{\Pi}_1^T)$  since  $\mathbf{\Pi}_1 \mathbf{\Pi}_2^T$  represents a degenerate dual conic on which all  $\mathbf{Q}$  lie).

Let us develop (3.19):

$$\mathbf{M} \sim \underbrace{\bar{\mathbf{P}}_s \mathbf{D}^{-1} \bar{\mathbf{P}}_s^T}_{\bar{\mathbf{M}}} - \underbrace{\bar{\mathbf{P}}_s \mathbf{D}^{-1} \mathbf{p}_s}_{\mathbf{b}} \mathbf{v}^T - \mathbf{v} \underbrace{\mathbf{p}_s^T \mathbf{D}^{-1} \bar{\mathbf{P}}_s}_{\mathbf{b}^T} + \mathbf{v} \underbrace{\mathbf{p}_s^T \mathbf{D}^{-1} \mathbf{p}_s}_{\rho} \mathbf{v}^T \quad (3.20)$$

$$\mathbf{M} \sim \bar{\mathbf{M}} - \mathbf{b}\mathbf{v}^T - \mathbf{v}\mathbf{b}^T + \rho\mathbf{v}\mathbf{v}^T \quad (3.21)$$

We can compute  $\rho$ , it is  $\frac{1}{2}(D_{22} = 2)$ . So we just need to obtain elements of  $\mathbf{v}$  to recover  $\mathbf{P}_{cata}$ . The principal point can be computed using different approaches, one of these is shown in (Mei and Rives, 2007), which requires the user interaction. Let us suppose we know the principal point  $(c_x, c_y)$ , and we put the origin of the image reference system on it ( $c_x = 0, c_y = 0$ ). Then we have:

$$\mathbf{M} = \begin{pmatrix} f^4 & 0 & 0 & 0 & 0 & -f^2\xi^2 \\ 0 & \frac{f^4}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & f^4 & 0 & 0 & -f^2\xi^2 \\ 0 & 0 & 0 & \frac{f^2}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{f^2}{2} & 0 \\ -f^2\xi^2 & 0 & -f^2\xi^2 & 0 & 0 & 2\xi^4 + (1 - \xi^2)^2 \end{pmatrix} \quad (3.22)$$

From this matrix we can extract 6 equations to solve for the elements of  $\mathbf{v}$ . For example:  $M_{11} - M_{33} = 0$ ,  $M_{11} - 2M_{22} = 0$ ,  $M_{44} - M_{55} = 0$ ,  $M_{13} = 0$ ,  $M_{35} = 0$ ,  $M_{56} = 0$ .

### 3.5. Calibration Experiments with a Simulated Environment

---

We test the case where  $f_x = f_y$  using simulated data with perfect 3D-2D correspondences. We observe that as explained in theory, the only modified column is the second one, described by the vector  $\mathbf{p}_s = [0, 1, 0, 0, 0, 0]^T$ . In this case we are able to obtain the correct  $\mathbf{P}_{cata}$ . However, when we added Gaussian noise to the 3D-2D correspondences, more than one column is modified making very difficult to recover the real projection matrix. Therefore, we conclude that the approach using points lying in just two planes is not suitable to compute the generic projection matrix in real situations. We continue our experiments with calibration grids having three planes.

## 3.5 Calibration Experiments with a Simulated Environment

We use a simulated calibration object having 3 planar faces which are perpendicular to each other. The size of a face is 50x50cm. There are a total of 363 points, since each face has 11x11 points and the distance between points is 5cm. The omnidirectional image fits in a 1 Megapixel square image. To represent the real world points we expressed the coordinates in meters, so they are normalized in a sense. This is important because we observed that using large numerical values causes bad estimations with noisy data in the DLT algorithm. Normalization of image coordinates is also performed since we observed a positive effect both on estimation accuracy and the convergence time. Therefore, in the presented experiments, 3D point coordinates are in meters and image coordinates are normalized to be in the same order of magnitude, this is performed by dividing the image coordinates by a constant.

We performed experiments for different settings of intrinsic parameters and varying position of the 3D calibration grid. We especially tested the accuracy of calibration to variations in the intrinsic parameters ( $\xi$  and  $f$ ), the distance between the camera and the grid and the orientation of the grid w.r.t. the camera. In all these cases, we measure the errors in final estimates of  $\xi$  and  $f$ , the main parameters of the sphere camera model. Errors are depicted in Fig. 3.1, where an individual graph is plotted for each case for clarity. In all experiments, Gaussian noise with  $\sigma = 1$  pixel is added to the actual coordinates of grid corners. The plotted errors are  $err_\xi = 100 \cdot |\xi_{nonlin} - \xi_{real}| / \xi_{real}$  and  $err_f = 100 \cdot |f_{nonlin} - f_{real}| / f_{real}$ . For all the nodes in the graphs, the experiment was repeated 100 times and the mean value of estimates is plotted.

Fig. 3.1a shows the effect of changing distance between the camera and the grid. From left to right in the graph distance-to-grid increases and distance values are selected randomly within the given ranges. When the distance is small, we reach an “optimal” position, such that the grid fills the image well. As the grid moves away from the omnidirectional camera, its image gets smaller and smaller. Examples of the omnidirectional images generated are shown in Fig. 3.2. In Fig. 3.2a, distance-to-grid is 45 cm., whereas in Fig. 3.2b it is 60 cm. The quality of parameter estimation

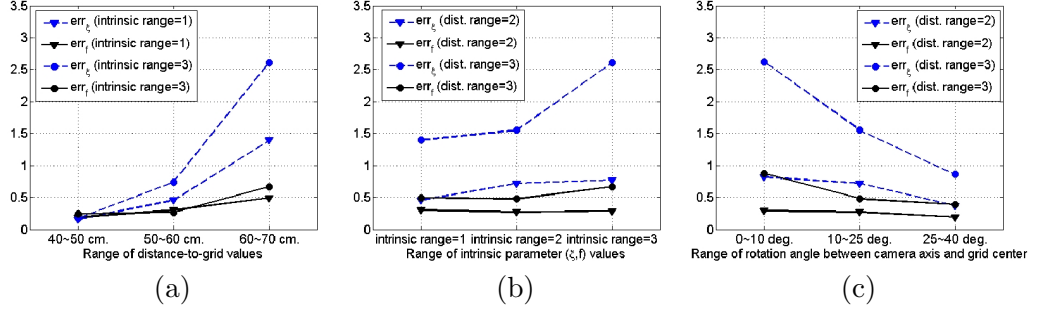


Figure 3.1: Relative errors for  $\xi$  and  $f$  after non-linear optimization (in percent) for varying intrinsic parameters and varying position of the 3D calibration grid. For all the nodes in the graphs, the experiment was repeated 100 times and the mean value of estimates is plotted. Real intrinsics, distance and orientation values are selected randomly from the ranges given in  $x$ -axis. Intrinsic parameters range 1:  $(\xi, f) = [(0.96, 360) (0.84, 300)]$ , range 2:  $(\xi, f) = [(0.84, 300) (0.72, 250)]$ , range 3:  $(\xi, f) = [(0.72, 250) (0.60, 210)]$ . Distance-to-grid (in cm.) range 1: [40 50], range 2: [50 60], range 3: [60 70]. In (a), (b) and (c), errors depicted versus increasing distance-to-grid, decreasing  $(\xi, f)$  pairs and increasing rotation angle respectively.

decreases with increasing distance. Since the grid covers a smaller area, the same amount of noise (in pixels) affects the non-linear optimization more and errors in non-linear results increase as can be expected. We observe the importance of a good placement of the calibration grid, i.e. such that it fills the image as much as possible.

Fig. 3.1b shows the effect of real  $\xi$  and  $f$  values on the estimation error (for two different distance-to-grid value ranges). From left to right in the graph,  $\xi$  and  $f$  values decrease. They decrease in parallel, otherwise decreasing  $\xi$  with fixed  $f$  would cause grid to get smaller in the image. We truncated  $(\xi, f)$  pairs at  $\xi = 0.6$  since even smaller  $\xi$  values are unlikely for omnidirectional cameras. We observe that larger  $(\xi, f)$  values produce slightly better results especially for increased distances. This observation can also be made in Fig. 3.1a since the errors are depicted with two different ranges of intrinsic parameter values. The reason is that for fixed distance-to-grid values, higher  $(\xi, f)$  spreads the grid points to a larger area in the image, which decreases the effect of noise. Observe Fig. 3.2b with Fig. 3.2c, where distance-to-grid values are equal but Fig. 3.2b has higher  $(\xi, f)$ .

Fig. 3.1c shows the effect of changing orientation of the grid w.r.t. the camera. This is expressed in terms of the angle between the optical axis of the omnidirectional camera and the grid center. The grid is not rotated independently from the camera axis because camera (mirror) has to see the inside of the 3D grid always. Fig. 3.2d shows the case when the grid is rotated so that the angle between its center and camera optical axis is  $40^\circ$ . Compare with Fig. 3.2b, where the intersection of the three planes of the grid is at the image center. We observe improvement with rotation specially for increased distance-to-grid since grid points are more spread and effect of noise decreases.

### 3.5. Calibration Experiments with a Simulated Environment

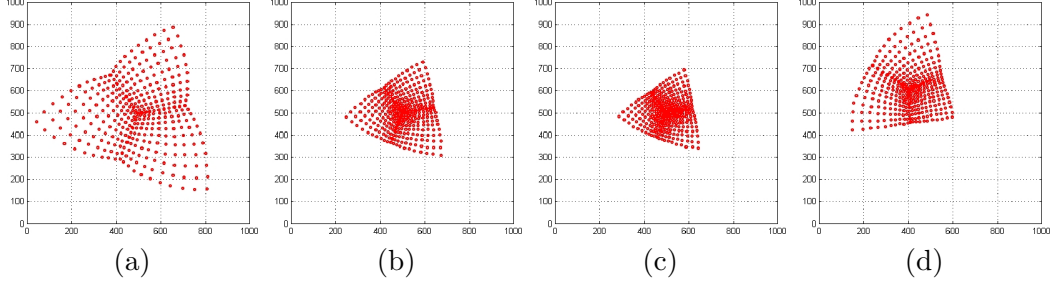


Figure 3.2: Omnidirectional images generated with varying intrinsics, distance-to-grid and orientation. (a)  $(\xi, f)=(0.96, 360)$ , distance=45cm., no rotation. (b)  $(\xi, f)=(0.96, 360)$ , distance=60cm., no rotation. (c)  $(\xi, f)=(0.76, 270)$ , distance=60cm., no rotation. (d)  $(\xi, f)=(0.96, 360)$ , distance=60cm., rotated by  $40^\circ$ .

	Distance-to-grid			
	45 cm.		60 cm.	
$\xi_{real}$	0.96	0.8	0.96	0.80
$f_{real}$	360	270	360	270
$\xi_{DLT}$	0.54	0.40	0.04	0.03
$f_{DLT}$	361	268	243	190
$\xi_{nonlin}$	0.96	0.80	0.98	0.78
$f_{nonlin}$	360	270	365	266
$err_\xi$	0.0	0.0	2.1	2.5
$err_f$	0.0	0.1	1.4	1.5

Table 3.1: Initial and optimized estimates with different intrinsics and distance-to-grid values. Amount of noise:  $\sigma = 1$  pixel.  $\xi_{DLT}, f_{DLT}$  and  $\xi_{nonlin}, f_{nonlin}$  are the results of the DLT algorithm and non-linear optimization respectively,  $err_\xi$  and  $err_f$  are the relative errors, in percent after non-linear optimization.

In Table 3.1, we list the results of the algorithm after linear (DLT) and non-linear steps for a few cases. Our main observation is that the errors in linear estimates,  $\xi_{DLT}$  and  $f_{DLT}$ , are biased (values are smaller than they should be). For all the cases, however, the true intrinsic parameters are reached after non-linear optimization, modulo errors due to noise.

#### 3.5.1 Estimation Errors for Different Camera Types

Here we discuss the intrinsic and extrinsic parameter estimation for the two most common catadioptric systems: hypercatadioptric and paracatadioptric, with hyperbolic and parabolic mirrors respectively. We also discuss calibration results for perspective cameras.

	Real values	$\sigma = 0.5$		$\sigma = 1$	
		Initial	Estimated	Initial	Estimated
$f$	360	361	360	354	360
$c_x$	500	503	500	505	500
$c_y$	500	498	500	509	500
$\xi$	0.96	0.84	0.96	0.53	0.96
$R_x(\alpha)$	-0.62	-0.60	-0.62	-0.40	-0.62
$R_y(\beta)$	0.62	0.62	0.62	0.65	0.62
$R_z(\gamma)$	0.17	0.15	0.17	0.18	0.17
$t_x$	0.30	0.38	0.30	0.45	0.30
$t_y$	0.30	0.40	0.30	0.44	0.30
$t_z$	0.20	0.05	0.20	0.01	0.20
RMSE			0.70		1.42

Table 3.2: Non-linear optimization results for a hypercatadioptric system, 10 parameters (rotation, translation and intrinsic) are optimized. Distance-to-grid is 45 cm. and grid center coincides with camera optical axis (no rotation).

### Hypercatadioptric System.

Table 3.2 shows non-linear optimization results including the rotation and translation parameters for fixed intrinsic parameters which corresponds to a hypercatadioptric system. 3D pattern is used at the “optimal” grid position, i.e. it fills the omnidirectional image like Fig. 3.2a. Results are in accordance with Table 3.1 and Fig. 3.1.

### Paracatadioptric System.

Here  $\xi = 1$ , which has a potential to disturb the estimations because  $\mathbf{X}_\xi$  becomes a singular matrix. We observe that the results of the DLT algorithm are not as close to the real values when compared to the hypercatadioptric system (cf. initial values in Table 3.2). However, the non-linear optimization estimates the parameters as successful as the hypercatadioptric examples given in Table 3.2.

### Perspective Camera.

In the sphere camera model,  $\xi = 0$  corresponds to the perspective camera. Our estimations in linear and non-linear steps are as successful as with the hypercatadioptric case and thus not shown in detail here.

### 3.5.2 Tilting and Distortion

It seems intuitive that small amounts of tangential distortion and tilting have a similar effect on the image. In our simulations we observed that trying to estimate both of them does not succeed. Therefore, we investigate if we can estimate tangential

### 3.6. Experiments with Real Images using a 3D Pattern

---

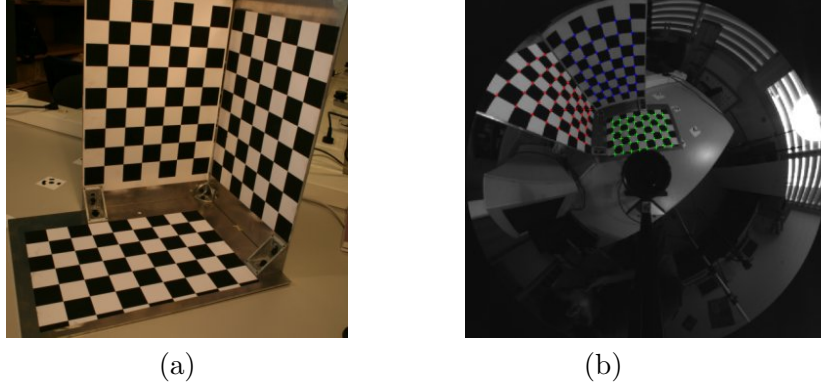


Figure 3.3: (a) 3D pattern, (b) Omnidirectional image of the 3D pattern (1024×768 pixels).

distortion of camera optics by tilt parameters, or estimate tilt in the system by tangential distortion parameters.

When there exists no tilt but tangential distortion and we try to estimate tilting parameters, we observed that the direction and amount of  $tilt_x$ ,  $tilt_y$ ,  $c_x$  and  $c_y$  changes proportionally to the tangential distortion applied and the RMSE decreases. However, the RMSE does not reach as low values as when there is no distortion. In the noiseless case, for example, the RMSE is not zero. Hence, we concluded that tilt parameters compensate the tangential distortion effect up to some extent, but not perfectly. We also investigated if tilting can be compensated by tangential distortion parameters and we had very similar results. Thus, tangential distortion parameters have the same capability to estimate tilting.

### 3.6 Experiments with Real Images using a 3D Pattern

In this section we perform experiments of camera calibration using a 3D pattern, cf. Fig. 3.3(a). The 3D pattern has been measured accurately doing a photogrammetric reconstruction by bundle adjustment. We use 6 convergent views taken with a calibrated high-resolution camera (Canon EOS 5D with 12.8Mpix.) and software PhotoModeler. The estimated accuracy of the 3D model is better than 0.1mm. The omnidirectional images were acquired using a catadioptric system with a hyperbolic mirror<sup>3</sup>. We computed the projection matrix  $P_{cata}$  from a total of 144 3D-2D correspondences and extracted the intrinsic and extrinsic parameters as explained in Section 3.2. From simulations, we observed that we have better and faster estimations if the 3D-2D correspondences are in the same order of magnitude. So 3D points are given in meters and 2D points are normalized in all the experiments. A second evaluation of the calibration accuracy is performed by a Structure from Motion experiment from two omnidirectional images.

---

<sup>3</sup>Neovision H3S with XCD-X710 SONY camera

	Real	Using distortion	Using tilting
$f$	279.84	297.24	306.11
$c_x$	531.83	528.08	552.75
$c_y$	407.98	406.28	427.89
$\xi$	0.96	0.86	0.93
RMSE	0	0.34	0.27

Table 3.3: Parameters estimated using either tangential distortion or tilting angles.

	Theoretic	$P_{cata}$ approach	[Mei and Rives,07]
$f$	279.84	297.24	298.65
$\xi$	0.96	0.86	0.72
$c_x$	531.83	528.02	528.15
$c_y$	407.98	406.28	403.39

Table 3.4: Comparison between our method and Mei's.

### 3.6.1 Intrinsic Parameters

The first experiment is focused on obtaining the intrinsic parameters from  $P_{cata}$  to get initial estimates of these values. As mentioned previously, we do not compute tilting and distortion parameters from  $P_{cata}$  but it is possible to include them in the non-linear optimization. From simulations we observed that we can compute either the tangential distortion or the tilting parameters which are coupled and can not be separated. We tested which one of these (tangential distortion and tilting) can deal better with the intrinsic parameter estimation. Table 3.3 shows a comparison of the estimations performed with these two options. The real values given in the table were computed using the calibration data of the perspective camera (previously calibrated) and the mirror parameters (provided by the manufacturer).

Catadioptric camera calibration using tilting gives a better RMSE but the intrinsic values obtained are far from the real ones. Estimation using distortion parameters increase slightly the RMSE but the intrinsic parameters are close to the real ones, except for  $\xi$  but this error can be attached to the configuration of the system (the optical center of the perspective camera may not be exactly located at the other focal point of the hyperbola describing the mirror) and not to the model.

After these results, we decided to use tangential distortion because it gives better results and depicts better the real catadioptric system.

In order to verify our approach we compare our intrinsic parameter estimates to the ones obtained by Mei and Rives (2007) (Table 3.4). As we can see neither Mei's approach nor  $P_{cata}$  approach can estimate the theoretic  $f$  and  $\xi$  parameters, but they give a good estimation to  $c_x$  and  $c_y$ . Mei computes the initial values directly from the inner circle of the omnidirectional image and using information given by the user. Our approach computes all the initial values from  $P_{cata}$  in closed form.

### 3.6. Experiments with Real Images using a 3D Pattern

---

	Experiment 1		Experiment 2		Experiment 3	
	Real	Estimated	Real	Estimated	Real	Estimated
$R_x$	-0.01	-0.02	-0.01	-0.003	-0.01	-0.002
$R_y$	0.02	0.02	0.02	0.01	0.02	0.03
$R_z$	—	—	—	—	—	—
$t_x$	0.39	0.39	0.39	0.39	0.39	0.38
$t_y$	0.21	0.21	0.33	0.33	0.23	0.23
$t_z$	-0.18	-0.18	-0.18	-0.18	-0.18	-0.18
RMSE	0.26		0.20		0.26	

Table 3.5: Rotation and translation of the camera with respect to the 3D pattern. Rotation angles are in radians. Translations are in meters. Real values were computed by the PhotoModeler software and a high resolution camera.

#### 3.6.2 Extrinsic Parameters

To obtain ground truth extrinsic parameters we have taken two additional images with the high resolution camera, observing the omnidirectional camera and the pattern. These images are added to the ones used to measure the 3D pattern. From this set of images the orientation and translation of the camera with respect to the pattern are computed. Location of the focal point was difficult since the points are not easy to identify in the images and indeed inside the mirror.

We performed experiments with 3 different camera locations. Table 3.5 shows the rotations and translations obtained from these experiments. Using PhotoModeler software we were just able to compute the direction of the  $z$ -axis but not the rotation around it. So we just show rotation estimations for the  $x$  and  $y$  axis. We can observe that the extrinsic parameter estimation is performed with a good accuracy having an average error of 0.0096 radians for rotations and 0.0022 meters for translations.

#### 3.6.3 Structure from Motion

The second experiment to evaluate the accuracy of our approach consists of obtaining the Structure and Motion (SfM) from two omnidirectional images observing the 3D pattern. Fig. 4.4(a) shows the 3D pattern with the angles between the planes composing it. Fig. 4.4(b) depicts the configuration used to perform the SfM experiment. Using the internal calibration provided by our method we compute the corresponding 3D rays from each omnidirectional image. We use these correspondences of 3D rays to compute the essential matrix  $E$  which relates them. From this matrix we compute two projection matrices  $P_1 = [I|\mathbf{0}]$  and  $P_2 = [R|\mathbf{t}]$ . Then, with these projection matrices and the 3D rays as input for a linear triangulation method Hartley and Zisserman (2000) we compute an initial 3D reconstruction. Both the 3D reconstruction and the camera location are later refined by a non-linear optimization process. We use 144 points which were extracted manually from the images. We measure the average error between the real 3D points and their estimations and the angle between the



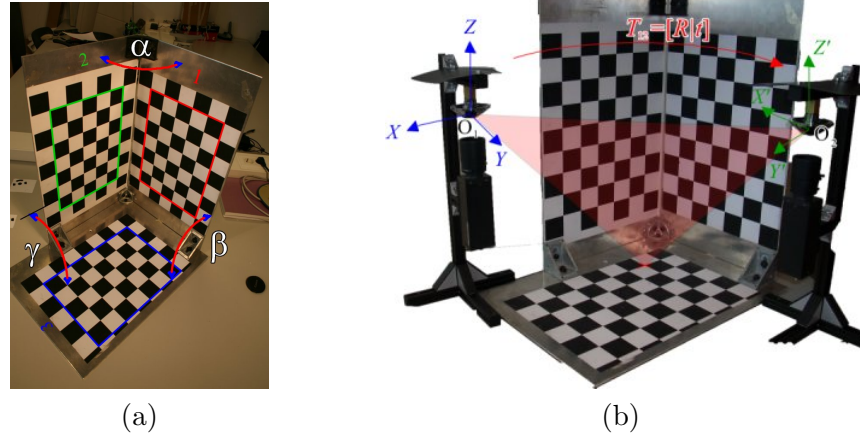


Figure 3.4: (a) 3D pattern with the angles between the planes. (b) SfM configuration.

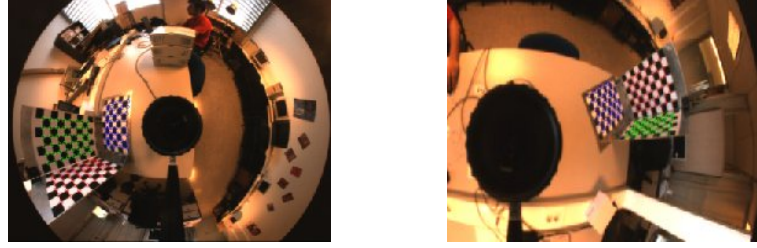


Figure 3.5: Images used in the SfM experiment.

planes. We use as ground truth the data computed by the photogrammetric software. The angles between the planes as depicted in Fig. 4.4(a) are  $\alpha = 90.06^\circ$ ,  $\beta = 89.60^\circ$  and  $\gamma = 90.54^\circ$ . The estimated values are  $\alpha = 89.22^\circ$ ,  $\beta = 90.55^\circ$  and  $\gamma = 89.73^\circ$ . We have an average error of  $0.86^\circ$ . We also measure the accuracy of the 3D points. The dimensions of the planar grids used in the 3D pattern are  $210\text{mm} \times 294\text{mm}$ . We compute the Euclidean distance between each reconstructed point and the ground truth. The average error is  $1.03\text{mm}$ .

### 3.7 Closure

In this chapter we presented a new calibration technique based on the sphere camera model which is able to represent every single-viewpoint catadioptric system. We employed a generic  $6 \times 10$  projection matrix, which uses *lifted coordinates* for image and 3D points. We estimated this projection matrix using 3D-2D correspondences. We use a single catadioptric image of a 3D calibration pattern. From the decomposition of this matrix we obtain an initial estimation of the intrinsic and extrinsic parameters of the catadioptric system. We used this parameter estimation as the initialization for a non-linear optimization process. We are able to calibrate various types of

### 3.7. Closure

---

cameras. This method was tested both with simulations and real images. Since the reprojection error is not definitive to show the good behavior of calibration approaches, we also present a Structure from Motion experiment to test the accuracy of our calibration method. For that reason we can provide error measurements in both pixels and millimeters.



## Chapter 4

# Comparison of Calibration Methods for Omnidirectional Cameras

*The number of calibration methods for central catadioptric has increased in recent years. These methods are based on different camera models and they can either consider the central catadioptric system as a whole or as a separated camera and mirror system. Many times the user requires a versatile calibration solution without spending valuable time in the implementation of a particular method. In this chapter we review the existing methods designed to calibrate any central omnivision system and analyze their advantages and drawbacks doing a deep comparison using simulated and real data. First we present a classification of such methods showing the most relevant characteristics of each particular method. Then we select the methods available as OpenSource and which do not require a complex pattern or scene. The evaluation protocol of calibration accuracy also considers 3D metric reconstruction combining omnidirectional images. Comparative results are shown and discussed in detail.*

### 4.1 Introduction

Most of the applications of computer vision techniques, such as visual navigation, localization and SLAM, structure from motion require to recover metric information from the environment. This 3D information is crucial when the omnidirectional cameras interact in real scenarios. The metric information depends entirely on the complete calibration of the omnidirectional system. For these practical applications the camera calibration is a basic step for subsequent and higher level tasks, and their final accuracy relies on the accuracy of the camera calibration. A considerable number of approaches to either calibrate central catadioptric systems or to calibrate fish-eye lens systems or both have been recently developed. Moreover, as we have observed they can use different projection models. With respect to central catadioptric systems, there exist some approaches that separate the calibration of the

perspective/orthographic camera from the computation of the mirror parameters (Svoboda and Pajdla, 2002; Morel and Fofi, 2007). However most of them deal with the catadioptric system as a whole. Some of these methods use single or multiple views of a 2D pattern (Kannala and Brandt, 2004; Scaramuzza et al., 2006; Mei and Rives, 2007; Deng et al., 2007; Frank et al., 2007; Gasparini et al., 2009), 3D pattern (Puig et al., 2011; Bastanlar et al., 2008), cylinder pattern (Toepfer and Ehlgen, 2007), some others use a single image containing features like lines (Geyer and Daniilidis, 2002a, 1999; Barreto and Araujo, 2002, 2005; Ying and Hu, 2004a; Ying and Zha, 2005; Vandeportaele et al., 2006; Wu et al., 2006; Caglioti et al., 2007; Wu et al., 2008). Finally there are other methods that perform a self-calibration of the system (Kang, 2000; Micusik and Pajdla, 2006; Ramalingam et al., 2010; Espuny and Burgos Gil, 2011).

As observed above there exist many calibration methods. They use different techniques and models to calibrate the omnidirectional systems. Some works have tried either to classify or to compare them. Ying and Hu (2004a) classify the calibration methods in three categories: (i) Known World Coordinates; (ii) Self-calibration and (iii) Projection of lines. They only consider 7 methods. On the other hand, they do not perform any comparison with any of such methods. A more specific classification is given by Deng et al. (2007) where 10 methods are grouped in five categories: (i) Self-calibration; (ii) Sphere-based calibration; (iii) Line-based calibration; (iv) Point-based calibration and (v) 2D calibration. We observe that methods based on 2D patterns have appeared emulating calibration methods for conventional cameras. The sphere-based category only contains one method, which also uses lines and it could be classified in that category. In that work there is no comparison of the proposed method to any other. Frank et al. (2007) identify three big groups of calibration methods: (i) Known World Coordinates which include those based on 2D patterns, which from our point of view should belong to different categories; (ii) Geometric Invariants which include the methods based on projections of lines and (iii) Self-calibration methods. They mention a total of 8 methods. A comparison of the proposed method with the online available methods (Scaramuzza et al., 2006; Mei and Rives, 2007) is presented. They use 4 different data sets including two fisheye, a paracatadioptric system and a system with very small distortion. Since (Mei and Rives, 2007) does not allow the manual extraction of grid points, the authors only consider those images where the grid points are extracted successfully, having a limited set of images. This situation has as consequence a poor calibration of the system. In this thesis we extract the grid points manually, allowing the methods to have the maximum data available, which permits to obtain the best performance and in consequence to perform a fair comparison of the methods. Toepfer and Ehlgen (2007) do not present a classification but a comparison of their proposed method with (Scaramuzza et al., 2006; Mei and Rives, 2007; Tsai, 1987). The performance of the methods is given considering a combination of the root mean square error (we assume the reprojection error) with the number of parameters of the method, through the principle of minimum description length. However it is not clear which catadioptric system has been calibrated neither how the method (Tsai,

1987) is adapted to work with catadioptric systems.

In this chapter we firstly present a classification of the existing approaches to calibrate omnidirectional systems. We propose five categories based on the main entity required by each method to perform the calibration of the systems. We also present in Table 4.1 the relevant information of each method according to valuable criteria: pattern or entity required, considering the minimum number of elements, number of views, analytical model, type of mirror, or if the method requires the separate calibration of the mirror and the camera.

Since the amount of calibration methods is considerable, the selection of a particular calibration method seems to be difficult and even more so if we consider the problems involved in the implementation process. Among all approaches mentioned previously, there is a group of calibration methods for omnidirectional systems (catadioptric and fish-eye) available online as OpenSource toolboxes. These methods can save time and effort when the goal is beyond the calibration itself and when the user is more interested in obtaining 3D motion and structure results than to deal with complex projection models. In this thesis, we evaluate these methods and provide an analysis with simulations and real images. Moreover, we use a Structure from Motion application with two omnidirectional images, where we become users of the calibrations provided by these approaches. This experiment shows the behavior of the approaches in a real scenario. Besides the performance, we also consider the ergonomics and ease of usage, as well as the type of features, the algorithm and the type of pattern, since they are important elements that can help the user to select the best approach. Moreover we present an up-to-date list of the calibration methods developed that consider catadioptric and fish-eye systems, allowing the reader to decide to implement a different method. These calibration methods are:

1. Mei and Rives (2007)<sup>1</sup> which uses the sphere camera model and requires several images of a 2D pattern. We will call this approach *Sphere-2DPattern*.
2. Puig et al. (2011)<sup>2</sup> which obtains a solution in closed form requiring a set of 3D-2D correspondences. It also uses the sphere camera model. We call this approach *DLT-like*.
3. Barreto and Araujo (2002)<sup>3</sup> uses also the sphere camera model and requires a single omnidirectional image containing a minimum of three lines. We call it *Sphere-Lines*.
4. Scaramuzza et al. (2006)<sup>4</sup> which models the omnidirectional images as distorted images where the parameters of distortion have to be found. We call this approach *Distortion-2DPattern*.

---

<sup>1</sup><http://www.robots.ox.ac.uk/~cmei/Toolbox.html>

<sup>2</sup><http://webdiis.unizar.es/~lpuig/DLTOmniCalibration>

<sup>3</sup><http://www.isr.uc.pt/~jpbar/CatPack/main.htm>

<sup>4</sup>[http://asl.epfl.ch/~scaramuz/research/Davide\\_Scaramuzza\\_files/Research/OcamCalib\\_Tutorial.htm](http://asl.epfl.ch/~scaramuz/research/Davide_Scaramuzza_files/Research/OcamCalib_Tutorial.htm)

## 4.2 Classification of Calibration Methods for Omnidirectional Cameras

As observed above there exist many calibration methods. They use different techniques and models to calibrate the omnidirectional systems. Some of them first calibrate the perspective camera and after that find the mirror parameters. In this section we present a review and classification of existing calibration methods. We propose five categories based on the main entity required to perform the calibration of the systems.

**Line-based calibration.** Many methods are based on the projection of lines in the omnidirectional images. The main advantage of using lines is that they are present in many environments and a special pattern is not needed. These approaches compute the image of the absolute conic from which they compute the intrinsic parameters of the catadioptric system. Geyer and Daniilidis (2002a) calibrate para-catadioptric cameras from the images of only three lines. Ying and Hu (2004a) analyze the relation of the camera intrinsic parameters and imaged sphere contours. They use the projection of lines as well as projections of the sphere. The former gives three invariants and the latter two. Vasseur and Mouaddib (2004) detect lines in the 3D scene which are later used to calculate the intrinsic parameters. This approach is valid for any central catadioptric system. Ying and Zha (2005) show that all line images from a catadioptric camera must belong to a family of conics which is called a line image family related to certain intrinsic parameters. They present a Hough transform for line image detection which ensures that all detected conics must belong to a line image family related to certain intrinsic parameters. Barreto and Araujo (2005) study the geometric properties of line images under the central catadioptric model. They give a calibration method suitable for any kind of central catadioptric system. Vandeportaele et al. (2006) slightly improves (Geyer and Daniilidis, 2002a) using a geometric distance instead of an algebraic one and they allow to deal with lines that are projected to straight lines or to circular arcs in a unified manner. Wu et al. (2006) introduce a shift from the central catadioptric model to the pinhole model from which they establish linear constraints on the intrinsic parameters. Without conic fitting they are able to calibrate para-catadioptric-like cameras. Caglioti et al. (2007) calibrate a system where the perspective camera is placed in a generic position with respect to a mirror, i.e., a non-central system. They use the image of one generic space line, from which they derive some constraints that, combined with the harmonic homology relating the apparent contours of the mirror allow them to calibrate the catadioptric system. More recently Wu et al. (2008) derive the relation between the projection on the viewing sphere of a space point and its catadioptric image. From this relation they establish linear constraints that are used to calibrate any central catadioptric camera.

**2D pattern calibration.** This kind of methods use a 2D calibration pattern with control points. These control points can be corners, dots, or any features that can be easily extracted from the images. Using iterative methods extrinsic and

## 4.2. Classification of Calibration Methods for Omnidirectional Cameras

---

intrinsic parameters can be recovered. Scaramuzza et al. (2006) propose a technique to calibrate single viewpoint omnidirectional cameras. They assume that the image projection function can be described by a Taylor series expansion whose coefficients are estimated by solving a two-step least squares linear minimization problem. Mei and Rives (2007) propose as Scaramuzza a flexible approach to calibrate single viewpoint sensors from planar grids, but based on an exact theoretical projection function -the sphere model- to which some additional radial and tangential distortion parameters are added to consider real-world errors. Deng et al. (2007) use the bounding ellipse of the catadioptric image and the field of view (FOV) to obtain the intrinsic parameters. Then, they use the relation between the central catadioptric and the pinhole model to compute the extrinsic parameters. Gasparini et al. (2009) compute the image of the absolute conic (IAC) from at least 3 homographies which are computed from images of planar grids. The intrinsic parameters of the central catadioptric systems are recovered from the IAC.

**3D Point-based calibration.** These methods require the position of 3D points observed usually in a single image. Aliaga (2001) proposes an approach to estimate camera intrinsic and extrinsic parameters, where the mirror center must be manually determined. This approach only works for para-catadioptric systems. Wu and Hu (2005) introduced the invariants of 1D/2D/3D space points and use them to compute the camera principal point with a quasi-linear method. Puig et al. (2011) present an approach based on the Direct Linear Transformation (DLT) using lifted coordinates to calibrate any central catadioptric camera. It computes a generic projection matrix valid for any central catadioptric system. From this matrix the intrinsic and extrinsic parameters are extracted in a closed form and refined by non-linear optimization afterwards. This approach requires a single omnidirectional image containing points spread in at least three different planes.

**Self-calibration.** This kind of calibration techniques uses only point correspondences in multiple views, without needing to know either the 3D location of the points or the camera locations. Kang (2000) uses the consistency of pairwise tracked point features for calibration. The method is only suitable for para-catadioptric systems. Micusik and Pajdla (2006) propose a method valid for fish-eye lenses and catadioptric systems. They show that epipolar geometry of these systems can be estimated from a small number of correspondences. They propose to use a robust estimation approach to estimate the image projection model, the epipolar geometry and to avoid outliers. Ramalingam et al. (2010) use pure translations and rotations and the image matches to calibrate central cameras from geometric constraints on the projection rays. Espuny and Burgos Gil (2011) developed a similar approach that uses two dense rotational flows produced by rotations of the camera about two unknown linearly independent axes which pass through the projection center.

**Polarization Imaging.** This method is proposed by Morel and Fofi (2007). It is based on an accurate reconstruction of the mirror by means of polarization imaging. It uses a very simple camera model which allows them to deal with any type of camera. However they observe that developing an efficient and easy-to-use calibration method is not trivial.



Method	Pattern/Entity	Number of views	Model	Mirror	W/S
Kang (2000)	10 point tracks	Multiple	Particular Central Catadioptric	Parabolic	W
Svoboda and Pajdla (2002)	—	Single	Particular Central Catadioptric	Generic	S
Caglioti et al. (2007)	1 line + mirror contours	Single	Part. Non-central Catadioptric	Generic	W
Aliaga (2001)	3D known points	Multiple	Part. Non-central Catadioptric	Parabolic	W
Toepfer and Ehlgren (2007)	3D pattern/2D pattern (multiple points)	Single/Multiple	Particular Central Catadioptric	Hyperbolic & Parabolic	W
Geyer and Daniilidis (2002a)	3 lines	Single	Sphere	Parabolic	W
Geyer and Daniilidis (1999)	2 vanishing points	Single	Sphere	Parabolic	W
Barreto and Araujo (2005)	3 lines	Single	Sphere	Generic	W
Ying and Hu (2004a)	2 lines/3 spheres	Single	Sphere	Generic	W
Ying and Zha (2005)	3 lines	Single	Sphere	Generic	W
Vandepotael et al. (2006)	3 lines	Single	Sphere	Parabolic	W
Wu et al. (2006)	lines	Single	Sphere	Parabolic	W
Wu et al. (2008)	3 lines	Single	Sphere	Generic	W
Vasseur and Mouaddib (2004)	lines (minimum no. n/a)	Single	Sphere	Generic	W
Mei and Rives (2007)	2D pattern (multiple points)	Multiple	Sphere	Generic	W
Puig et al. (2011)	3D pattern (20 3D-2D correspondences)	Single	Sphere	Generic	W
Deng et al. (2007)	2D pattern (multiple points)	Multiple	Sphere	Generic	W
Gasparini et al. (2009)	2D pattern (multiple points)	Multiple	Sphere	Generic	W
Wu and Hu (2005)	4 correspondences	Multiple	Sphere	Generic	W
Scaramuzza et al. (2006)	2D pattern (multiple points)	Multiple	Distortion	Generic	W
Frank et al. (2007)	2D pattern (multiple points)	Multiple	Distortion	Generic	W
Micusik and Pajdla (2006)	9 correspondences (epipolar geometry)	Multiple	Distortion	Generic	W
Morel and Fofi (2007)	3 polarized images	Multiple	Generic Camera*	Generic	S
Ramalingam et al. (2010)	2 Rotational & Translational Flows	Multiple	Generic Camera*	Generic	W
Espuny and Burgos Gil (2011)	2 Rotational Flows	Multiple	Generic Camera*	Generic	W

Table 4.1: Classification of the calibration methods for omnidirectional systems. W = whole system. S = separate calibration (1. camera and 2. mirror parameters). \*They use the same approach Sturm and Ramalingam (2004).

## 4.3 Calibration Methods Analyzed

In this section we summarize the four OpenSource methods used to calibrate omnidirectional systems. The purpose of this section is to show a general view of the methods that help the reader to understand the core of each method.

### 4.3.1 Sphere-2DPattern Approach

This approach (Mei and Rives, 2007) uses the sphere model explained in the last section, with the difference that it does not consider the image flip induced by  $(\psi - \xi)$ , it uses  $(\xi - \psi)$  in  $x$  and  $y$  coordinates. This approach adds to this model distortion

### 4.3. Calibration Methods Analyzed

---

parameters to consider real world errors. This method is multiview, which means that it requires several images of the same pattern containing as many points as possible. This method needs the user to provide prior information to initialize the principal point and the focal length of the catadioptric system. The principal point is computed from the mirror center and the mirror inner border. The focal length is computed from three or more collinear non-radial points. Once all the intrinsic and extrinsic parameters are initialized a non-linear process is performed. This approach is also valid for fish-eye lenses and spherical mirrors.

This approach uses a total of 17 parameters to relate a scene point to its projection in the image plane:

- Seven extrinsic parameters  $(\mathbf{q}, \mathbf{t})$  representing the relation between the camera reference system and the world reference system. A 4-vector  $\mathbf{q}$  represents the rotation as a quaternion and a 3-vector  $\mathbf{t}$  represents the translation.
- One mirror parameter  $\xi$ .
- Four distortion parameters  $Dist = [k_1, k_2, p_1, p_2]$ , two for tangential distortion and two for radial distortion (Heikkila and Silven, 1997).
- Five parameters representing a generalized camera projection  $\mathbf{P} = [\theta, \gamma_1, \gamma_2, u_0, v_0]$ .  $(\gamma_1, \gamma_2)$  are the focal lengths of the catadioptric system for  $x$  and  $y$  axis,  $\theta$  is the skew parameter, and  $(u_0, v_0)$  is the principal point.

The 2D pattern used to calibrate the camera is composed of  $m$  points  $\mathbf{X}_i$  with their associated image values  $\mathbf{x}_i$ . The solution of the calibration problem is obtained by minimizing the reprojection error using the Levenberg-Marquardt algorithm.

#### 4.3.2 Sphere-Lines Approach

This method (Barreto and Araujo, 2005) is based on computing the absolute conic  $\hat{\Omega}_\infty = \mathbf{H}_c^{-\top} \mathbf{H}_c^{-1}$  and the mirror parameter  $\xi$  under the sphere camera model. In omnidirectional images 3D lines are mapped to conics. So the first step is to fit these conics. With the information provided by these conics and the location of the principal point an intermediate step is performed. It computes entities like polar lines, lines at infinity and circle points. From these intermediate entities and some invariant properties like collinearity, incidence and cross-ratio the mirror parameter  $\xi$  is computed. From the image of a conic in the omnidirectional images it is possible to compute two points that lie on the image of the absolute conic. Since a conic is defined by a minimum of 5 points at least three conic images are required to obtain  $\hat{\Omega}_\infty$ . Once the image of the absolute conic  $\hat{\Omega}_\infty$  is computed, from its Cholesky decomposition we obtain

$$\mathbf{H}_c = \begin{pmatrix} \gamma_x & \beta & c_x \\ 0 & \gamma_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \quad (4.1)$$

with the intrinsic parameters  $\gamma_x$  and  $\gamma_y$  (focal lengths), skew ( $\beta$ ) and principal point  $(c_x, c_y)$ .

### 4.3.3 DLT-like Approach

This approach (Puig et al., 2011) also uses the sphere camera model. To deal with the non-linearities present in this model, the lifting of vectors and matrices is used. This method computes a lifted  $6 \times 10$  projection matrix that is valid for all single-viewpoint catadioptric cameras. The required input for this method is a single image with a minimum of 20 3D-2D correspondences distributed in 3 different planes.

In this approach a 3D point  $\mathbf{X}$  is mathematically projected to two image points  $\mathbf{x}_+$ ,  $\mathbf{x}_-$ , which are represented in a single entity via a degenerate dual conic  $\Omega$ . The relation between them is  $\Omega \sim \mathbf{x}_+ \mathbf{x}_+^T + \mathbf{x}_- \mathbf{x}_-^T$ .

This conic represented as a 6-vector  $\mathbf{c} = (c_1, c_2, c_3, c_4, c_5, c_6)^T$  projected on the omnidirectional image is computed using the lifted 3D point coordinates, intrinsic and extrinsic parameters as:

$$\mathbf{c} \sim \hat{\mathbf{H}}_{6 \times 6} \mathbf{X}_\xi \hat{\mathbf{R}}_{6 \times 6} (\mathbf{I}_6 \quad \mathbf{T}_{6 \times 4}) \hat{\mathbf{X}}_{10} \quad (4.2)$$

where,  $\mathbf{R}$  represents the rotation of the catadioptric camera.  $\mathbf{X}_\xi$  a  $6 \times 6$  matrix and  $\mathbf{T}_{6 \times 4}$  depend only on the sphere model parameter  $\xi$  and position of the catadioptric camera  $\mathbf{C} = (t_x, t_y, t_z)$  respectively. Thus, a  $6 \times 10$  catadioptric projection matrix,  $\mathbf{P}_{cata}$ , is expressed by its intrinsic  $\mathbf{A}_{cata}$  and extrinsic  $\mathbf{T}_{cata}$  parameters

$$\mathbf{P}_{cata} = \underbrace{\hat{\mathbf{H}}_c \mathbf{X}_\xi}_{\mathbf{A}_{cata}} \underbrace{\hat{\mathbf{R}}_{6 \times 6} (\mathbf{I}_6 \quad \mathbf{T}_{6 \times 4})}_{\mathbf{T}_{cata}} \quad (4.3)$$

This matrix is computed from a minimum of 20 3D-2D lifted correspondences in a similar way to the perspective case (Hartley and Zisserman, 2000) using least squares

$$([\hat{\mathbf{x}}]_\times \otimes \hat{\mathbf{X}}) \mathbf{p}_{cata} = \mathbf{0}_6 \quad (4.4)$$

The 60-vector  $\mathbf{p}_{cata}$  contains the 60 coefficients of  $\mathbf{P}_{cata}$ . Manipulating this matrix algebraically the intrinsic and extrinsic parameters are extracted. These extracted values are used as an initialization to perform a non-linear process (Levenberg-Marquardt). In this process some parameters that are not included in the sphere camera model are added. These parameters are, as in Sphere-2DPattern approach, the radial and tangential distortion which are initialized to zero. This approach uses two parameters for each type of distortion. The minimization criterion is the root mean square (RMS) of distance error between a measured image point and its reprojected correspondence.

## 4.4. Experiments

---

### 4.3.4 Distortion-2DPattern Approach

In this approach (Scaramuzza et al., 2006) the projection model is different from the one previously presented. The only assumption is that the image projection function can be described by a polynomial, based on Taylor series expansion, whose coefficients are estimated by solving a two-step least squares linear minimization problem. It does not require either any a priori knowledge of the motion or a specific model of the omnidirectional sensor. So, this approach assumes that the omnidirectional image is in general a highly distorted image and we have to compute the distortion parameters to obtain such a distorted image. This approach, like Sphere-2DPattern, requires several images from different unknown positions of a 2D pattern. The accuracy depends on the number of images used and on the degree of the polynomial.

Under this model a point in the camera plane  $\mathbf{x}' = [x', y']^T$ , is related to a vector  $\mathbf{p}$  which represents a ray emanating from the viewpoint  $O$  (located at the focus of the mirror) to the scene point  $\mathbf{X}$ . This relation is encoded in the function  $\mathbf{g}$

$$\mathbf{p} = \mathbf{g}(x') = \mathbf{P}\mathbf{X} \quad (4.5)$$

where  $\mathbf{X} \in \mathbb{R}^4$  is expressed in homogeneous coordinates;  $\mathbf{P} \in \mathbb{R}^{3 \times 4}$  is a perspective projection matrix. The function  $\mathbf{g}$  has the following form

$$\mathbf{g}(\mathbf{x}') = (x', y', f(x', y'))^T, \quad (4.6)$$

and  $f$  is defined as

$$f(x', y') = a_0 + a_1\rho' + a_2\rho'^2 + \dots + a_n\rho'^n \quad (4.7)$$

where  $\rho'$  is the Euclidean distance between the image center and the point. In order to calibrate the omnidirectional camera the  $n+1$  parameters  $(a_0, a_1, a_2, \dots, a_n)$  corresponding to the coefficients of function  $f$  need to be estimated.

The camera calibration under this approach is performed in two steps. The first step computes the extrinsic parameters, i.e., the relation between each location of the planar pattern and the sensor coordinate system. Each point on the pattern gives three homogeneous equations. Only one of them is linear and it is used to compute the extrinsic parameters. In the second step, the intrinsic parameters are estimated, using the extrinsic parameters previously computed and the other two equations. The authors do not mention it, but after this linear process a non-linear optimization is performed using the Gauss-Newton algorithm<sup>5</sup>.

## 4.4 Experiments

In order to compare the different calibration methods we calibrate a hypercatadioptric system<sup>6</sup>, a fish-eye and a commercial proprietary shape mirror<sup>7</sup>, which we

---

<sup>5</sup>This algorithm is provided by the **lsqnonlin** Matlab function

<sup>6</sup>Neovision H3S with XCD-X710 SONY camera

<sup>7</sup><http://www.0-360.com>

name unknown-shape system. Additionally, we displace the perspective camera of the hypercatadioptric system far from the mirror. This allows the displacement of the optical center of the perspective camera from the other focus described by the hyperbolic mirror, leading to a non-central system. We calibrate these four systems with the four methods and compare the results with a reconstruction experiment which is explained in section 4.4.1. The set up used to calibrate the omnidirectional system for every method is explained as follows.

*Sphere-2DPattern approach.* This approach requires images of a single 2D pattern. These images have to cover most of the omnidirectional image area. This approach asks the user for the image center and for the outer mirror border to compute the principal point. Then it asks for four aligned edge points on a non-radial line to compute the focal length. With this information it asks for the four corner points of the pattern and uses a subpixel technique to extract the rest of the points present in the pattern. If the focal length is not well estimated then all points have to be given manually.

*DLT-like approach.* In the DLT-like approach a single image of a 3D pattern was used. This approach does not have an automatic extractor so all points are given manually. This method requires as input a set of points lying on at least three different planes with known relative position.

*Distortion-2D approach.* This approach also uses images coming from a 2D pattern. The last version of this method has an automatic corner detector which detects most of the corners present in a single pattern. The amount of corners given manually is minimum. Once all the points in all the images are given the calibration process starts.

*Sphere-Lines approach.* This approach is based on the projections of lines in the omnidirectional images. This method only requires one omnidirectional image containing at least three lines.

#### 4.4.1 Evaluation Criteria

Previous to the comparison of the real system we perform an experiment using simulated data. The purpose of this experiment is to observe the behavior of all approaches under optimal conditions and to measure their sensitivity to noise. We simulate two central catadioptric systems, a hypercatadioptric<sup>8</sup> with mirror parameter  $\xi = 0.7054$  and a paracatadioptric system. Firstly, we generate five synthetic images, each one containing a calibration pattern, that cover the whole FOV of the omnidirectional image, two of these images are shown in Fig. 4.1(a) and Fig. 4.1(b). These images are used by the Sphere-2DPattern and the Distortion-2D approaches. We combine the five calibration patterns in a single image (Fig. 4.1(c)) that is used by the DLT-Like approach. These three approaches use the same points. In the case of the Sphere-Lines approach four lines are present in a single image with a total of 904 points (Fig. 4.1(d)). We add Gaussian noise to the image coordinates.

---

<sup>8</sup><http://www.accowle.com>

#### 4.4. Experiments

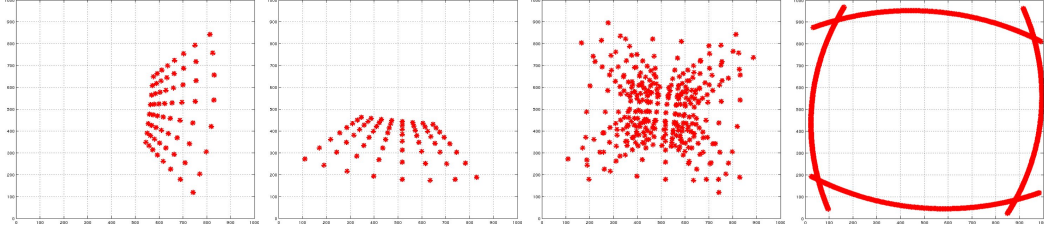


Figure 4.1: Synthetical images used by the calibration methods. Sphere-2DPattern and Distortion-2DPattern approaches use five images similar to (a) and (b). DLT-like approach uses a single image containing 5 planar patterns (c) and Sphere-Lines approach use 4 image lines (d).

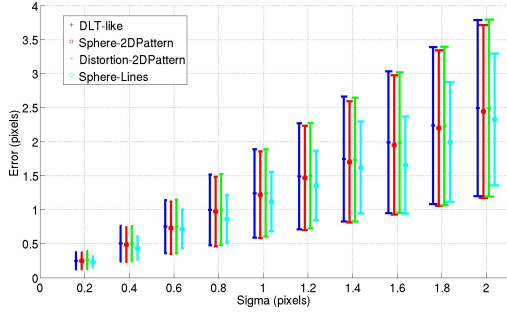


Figure 4.2: Reprojection error in pixels as a function of noise ( $\sigma$ ) corresponding to the hypercatadioptric system.

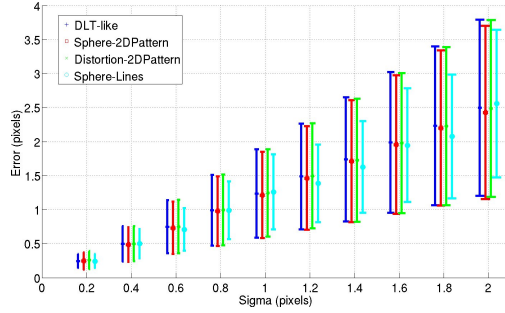


Figure 4.3: Reprojection error in pixels as a function of noise ( $\sigma$ ) corresponding to the paracatadioptric system.

For every noise level  $\sigma$  (in pixels) we repeat the experiment 100 times in order to avoid particular cases due to random noise. In Fig. 4.2 and Fig. 4.3 we show the mean and the standard deviation of the reprojection error corresponding to the hypercatadioptric and the paracatadioptric systems, respectively for the analyzed approaches.

As we observe the behavior of the four approaches is quite similar. All of them respond in the same way to the amount of noise present in the images. This experiment shows that under optimal circumstances the performance of the four approaches is quite similar. One thing we observe with respect to the Sphere-Lines approach is that four lines are enough to compute the calibration. We try to calibrate the systems using a higher number of lines which caused the approach being slower and in some occasions it did not converge. This behavior is due to the lines containing a high number of points, therefore increasing 2 or more lines means to increase hundreds of points.

We observe that the reprojection error is not sufficient to decide which method is endowed with the best performance, however it is a necessary condition to qualify an algorithm as performing well. Moreover, a disadvantage of this error measure is that we can make it smaller by just adding more parameters to a model. As

an example we can observe the reprojection error of the hypercatadioptric system given by the Sphere-2DPattern approach (see Table 4.3). The obtained reprojection error during calibration was considerably small, only 0.000005 pixels, which could be considered as zero. This approach adds five distortion parameters: three for radial distortion and two for tangential distortion. Originally, these parameters are not considered in the model. To verify the impact of these parameters in the calibration we repeated the calibration. When we only consider radial distortion (3 parameters), the reprojection error increased to 0.42 pixels. When no distortion is considered at all, the reprojection error increased to 0.64 pixels. As previously stated, the reprojection error is not definitive in showing which approach performs the best.

Instead of the reprojection error we choose to consider a Structure from Motion task from two calibrated omnidirectional images of a 3D pattern (Fig. 4.4) built in our lab. The pattern has been measured with high accuracy using photogrammetric software<sup>9</sup>. Thus, a 3D reconstruction by bundle adjustment has been made. We used 6 convergent views taken with a calibrated high-resolution camera (Canon EOS 5D with 12.8Mpix.). The estimated accuracy of the location of the 3D points is better than 0.1mm. Fig. 4.5 shows the configuration used for the SfM experiment. Using the calibration provided by each method we compute the corresponding 3D rays from each omnidirectional image. The correspondences between the two images were given manually. We use these correspondences to compute the essential matrix  $E$  which relates them. From this matrix we compute two projection matrices  $P_1 = [I|0]$  and  $P_2 = [R|t]$ . Then, with these projection matrices and the 3D rays we compute an initial 3D reconstruction using a linear triangulation method (Hartley and Zisserman, 2000) which is later refined by a bundle adjustment optimization process. The 3D reconstruction depends on the number of correspondences. We use a set of 144 points to compute the reprojection error and to evaluate the 3D reconstruction results. We choose two different criteria to measure the accuracy of each model. These criteria are:

- The average error between the real 3D points and their estimations.
- The reprojection error. We project the ground truth 3D pattern in the two cameras with the locations given by the SfM algorithm. We measure the error in pixels between the image points and the ground truth reprojection.

#### 4.4.2 Hypercatadioptric System

The hypercatadioptric system is composed by a perspective camera with a resolution of  $1024 \times 768$  and a hyperbolic mirror having a 60 mm diameter and parameters  $a = 281$  mm and  $b = 234$  mm according to manufacturer information. The mirror parameter for the sphere camera model is  $\xi = 0.9662$ . In Fig. 4.6 we observe some of the images used to calibrate this system. We use one image to calibrate the system

---

<sup>9</sup>PhotoModeler software was used.

#### 4.4. Experiments

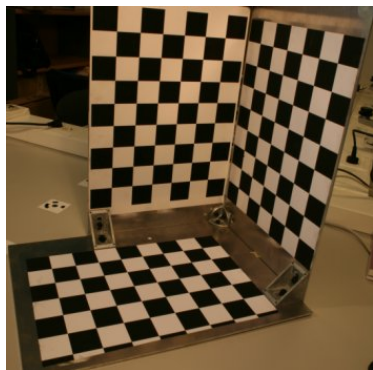


Figure 4.4: 3D pattern built at lab.

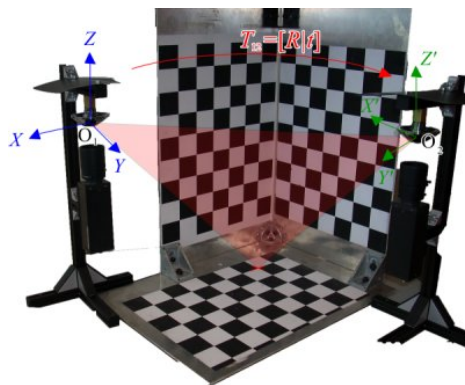


Figure 4.5: SfM configuration to test the calibrations.

	$\xi$	$(u_0, v_0)$
Manufacturer Info	0.9662	(511.88, 399.25)
Sphere-2DPattern	0.9684	(513.93, 400.76)
Sphere-Lines	1.0215	(523.82, 416.29)
DLT-like	0.9868	(509.95, 398.54)

Table 4.2: Comparison of the physical parameters given by the 3 methods based on the sphere model.

using the Sphere-Lines and the DLT-like approaches. Eleven images of the 2D pattern were used to calibrate the system using both Sphere-2DPattern and Distortion-2D approaches. In Fig. 4.7 we show the images used for the SfM experiment.

#### Mirror Parameter and Principal Point

Three of these methods are based on the sphere camera model. In Table 4.2 we present the estimations of the principal point  $(u_0, v_0)$  and the mirror parameter  $\xi$  since they are related with sphere model parameters. The Distortion-2DPattern approach does not offer information about the catadioptric system. As we can see, the best estimation of the mirror parameter is given by Sphere-2DPattern but also the DLT-like algorithm gives a good approximation. Sphere-Lines gives a value bigger than 1 which does not correspond to a hyperbolic mirror. With respect to the principal point the estimation of Sphere-2DPattern and DLT-like are close to the real one. The difference is that the Sphere-2DPattern method asks the user to give the image center and the rim of the mirror to compute the principal point and the DLT-like algorithm does not need any of this a priori information but requires known positions of the 3 planes in the pattern.

In Fig. 4.8 we show the histogram corresponding to the accuracy in millimeters of the 144 reconstructed 3D points. We observe that the Sphere-2DPattern approach



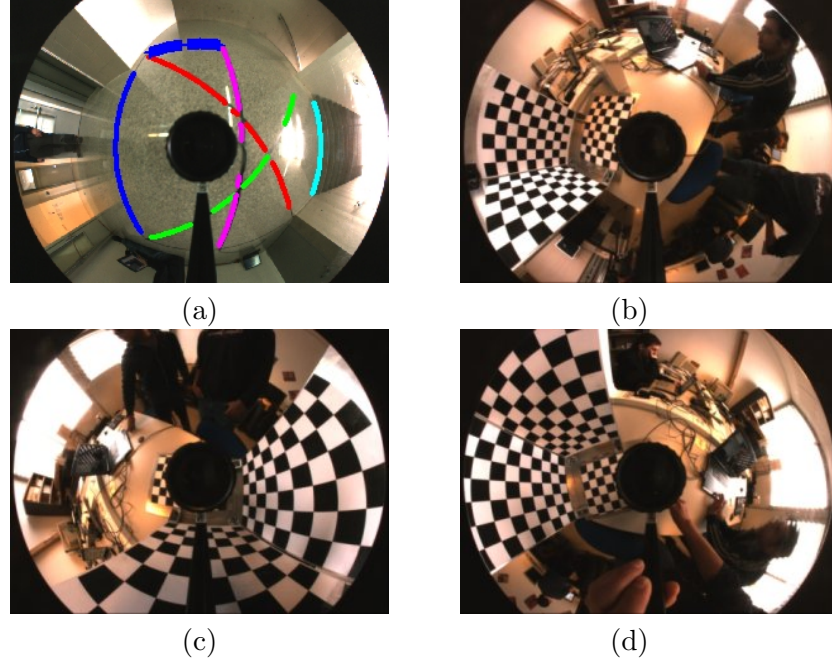


Figure 4.6: Some images used to calibrate the hypercatadioptric system. (a) Sphere-Lines. (b) DLT-like approach. (c,d) Sphere-2DPattern and Distortion-2DPattern approaches.

Method	Original paper	Calibration	Structure from Motion
Sphere-2DPattern	0.40	0.00005	0.34
Sphere-Lines	n/a	1.11*	1.19
DLT-Like	0.30	0.47	0.40
Distortion-2DPattern	1.2	0.82	0.45

Table 4.3: Reprojection error from different sources corresponding to the hypercatadioptric system. (Since the Sphere-Lines does not use the reprojection error we take it from simulations with Gaussian noise  $\sigma = 1\text{pixel}$ ).

has the highest number of reconstructed points closer to the ground truth. The highest error corresponds to Sphere-Lines and Distortion-2DPattern with one point 5 mm far from the ground truth. In Fig. 4.9 we show the reprojection error of the 288 points of the two images used in the experiment. We observe that all the approaches are below the 2 pixel error and three of them within the 1 pixel error.

The hypercatadioptric system is the more complex catadioptric system under the sphere camera model since the mirror parameter  $\xi$  is in the interval  $]0, 1[$ . In opposition, the paracatadioptric system where  $\xi = 1$  simplifies considerably the model. In this context we decide to present the reprojection error corresponding to this system from three different sources (see Table 4.3): the reprojection error

## 4.4. Experiments

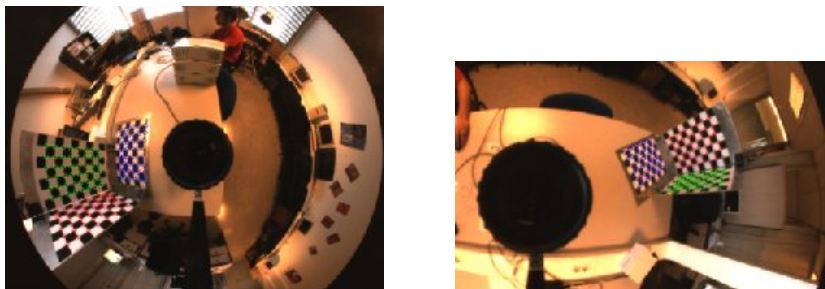


Figure 4.7: Images used in the SfM experiment with reprojected points superimposed (hypercatadioptric).

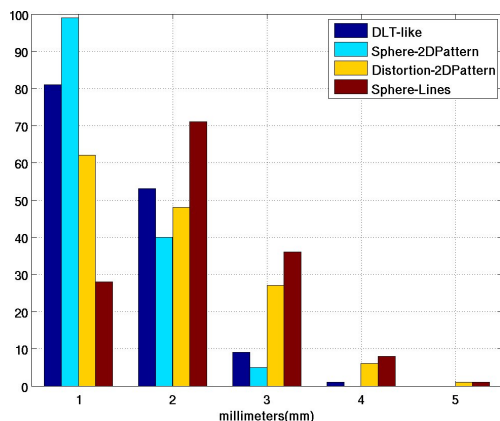


Figure 4.8: Number of reconstructed 3D points within an error in millimeters using a hypercatadioptric system.

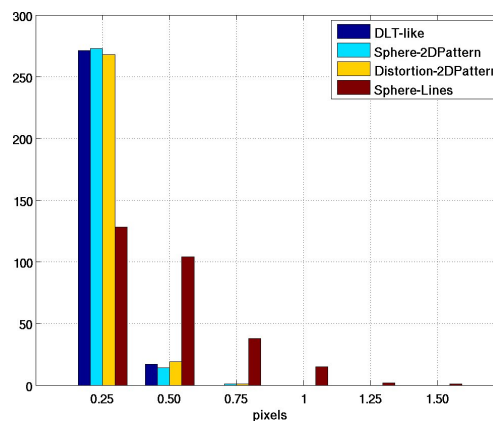


Figure 4.9: Number of reprojected points within an error distance in pixels using a hypercatadioptric system.

shown in the original paper where the approach was firstly presented (first column), the reprojection error obtained when we calibrate the system, and the reprojection error from the Structure from Motion experiment. This information provides the reader with a clear idea about the performance of all approaches under different circumstances and allows us to observe that the reprojection error given at the calibration time is less informative than the reprojection error of a Structure from Motion experiment where the calibration is an early step and all the approaches are under the same circumstances.

### 4.4.3 Fish-eye Lens

The fish-eye lens used in this experiment is a Raynox DCR-CF185PRO with a FOV of  $185^\circ$  on all directions. It is mounted on a high-resolution camera. In Fig. 4.10 we show some of the images used to calibrate this system. We use 8 images to perform the calibration with the Sphere-2DPattern and Distortion-2DPattern approaches and only one image using the DLT-like approach. We use the image of 7 lines to perform

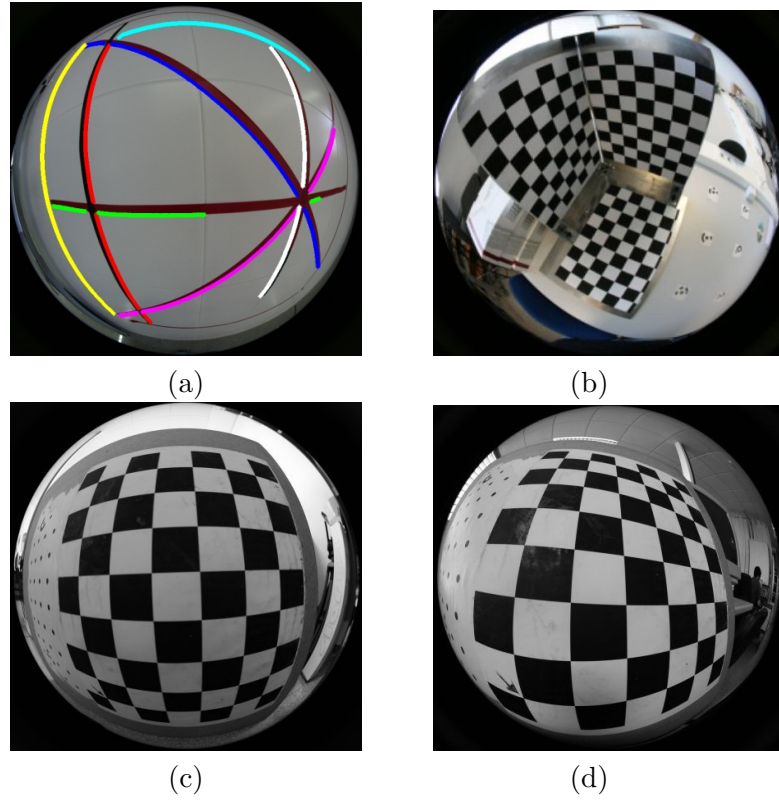


Figure 4.10: Some images used to calibrate the fish-eye system. (a) Sphere-Lines. (b) DLT-like approach. (c,d) Sphere-2DPattern and Distortion-2DPattern approaches.

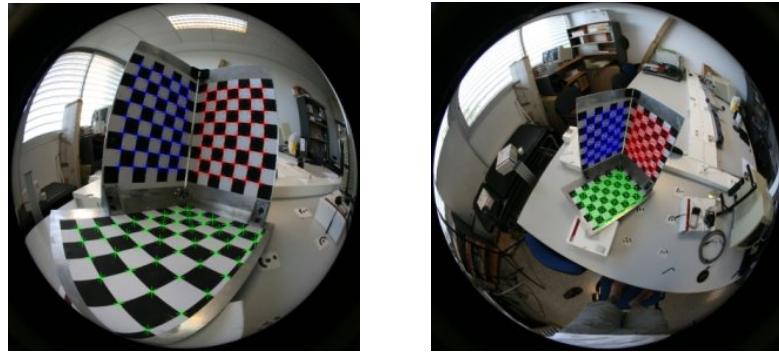


Figure 4.11: Images used in the SfM experiment with reprojected points superimposed (fish-eye).

the calibration with the Sphere-Lines approach. Since none of the methods provides any information about the system we just show the results obtained from the SfM experiment. Fig. 4.11 shows the images used for this experiment.

In Fig. 4.12 and Fig. 4.13 we show the results of this experiment. We observe

#### 4.4. Experiments

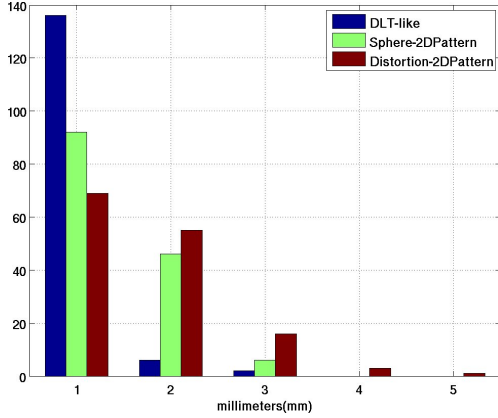


Figure 4.12: Number of reconstructed 3D points within an error in millimeters using a fish-eye lens.

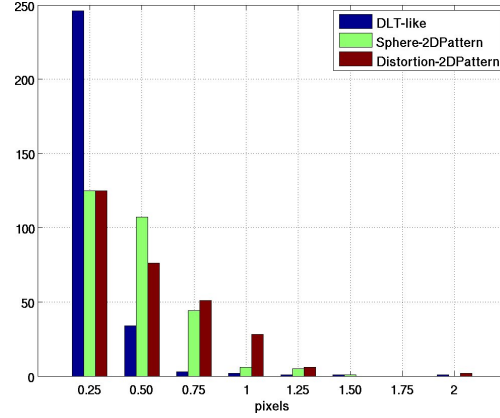


Figure 4.13: Number of reprojected points within an error distance in pixels using a fish-eye lens.

that the methods that claim to be able to calibrate the slightly non-central fish eye systems indeed give good results. This is the case of the Sphere-2DPattern, DLT-like and the Distortion-2DPattern approaches. The opposite case is the Sphere-Lines approach which is not able to correctly calibrate this camera. It gives reprojection errors close to one hundred pixels. We observe that the three valid approaches give similar results. The DLT-like gives the best results with the highest number of reconstructed points within 1 mm error, although it is not designed to handle fish-eye lenses. With respect to the reprojection error, we observe a similar behavior of the three approaches with a maximum error of 2 pixels.

##### 4.4.4 Unknown-shape Catadioptric System

This system is the combination of a commercial proprietary shape mirror and a high-resolution camera. We use 6 images of a 2D pattern to calibrate this system for the Sphere-2DPattern and Distortion-2DPattern approaches. We use an image with 4 lines to perform the calibration with the Sphere-Lines approach. We observed several difficulties when more than 4 lines were used to calibrate the system using the Sphere-Lines approach. Sometimes the calibration results contained complex numbers or there were problems of convergence. In Fig. 4.14 we observe some of the images used to calibrate this system under all the analyzed approaches. In Fig. 4.15 we show the images used to perform the SfM experiment.

We decided to use a more complex reconstruction scenario to observe the behavior of the approaches under these conditions (Fig. 4.15 (a), (b)). The results of the experiments, the 3D accuracy and the reprojection error are shown in Fig. 4.16 and Fig. 4.17, respectively. We observe that the Distortion-2DPattern approach has the highest number of points within the 5 mm error, 131 out of 144. The worst performance is given by the Sphere-Lines approach with maximum errors of 35

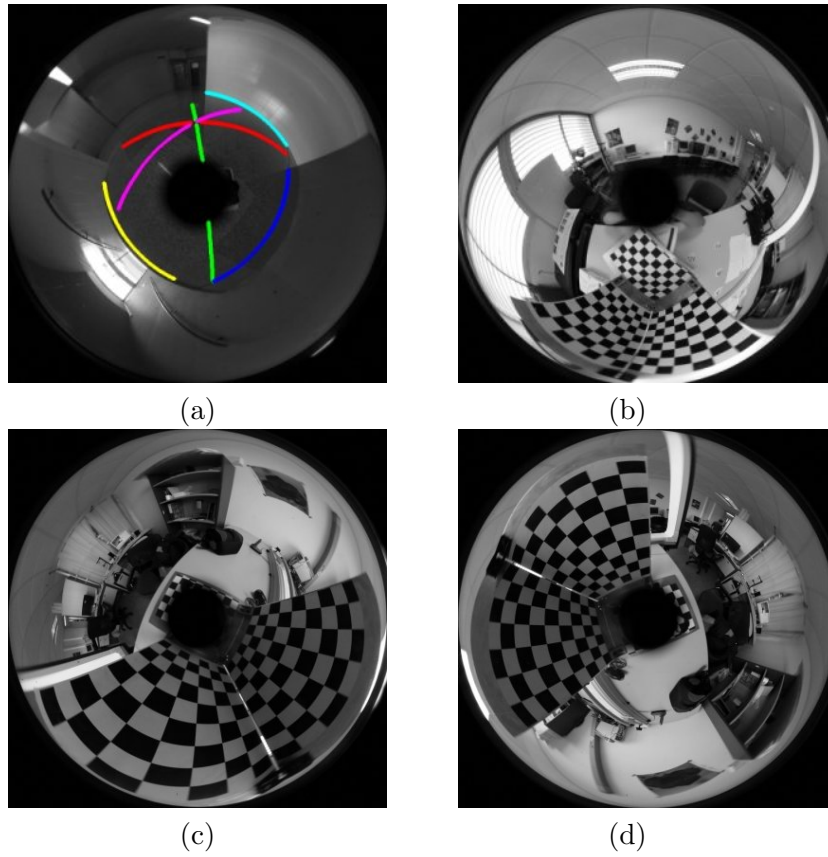


Figure 4.14: Some images used to calibrate the unknown-shape catadioptric system. (a) Sphere-Lines. (b) DLT-like approach. (c,d) Sphere-2DPattern and Distortion-2DPattern approaches.

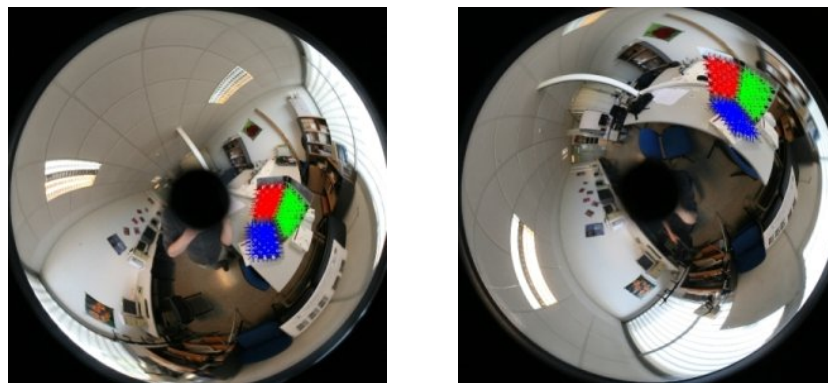


Figure 4.15: Images used in the SfM experiment with reprojected points superimposed (unknown-shape).



#### 4.4. Experiments

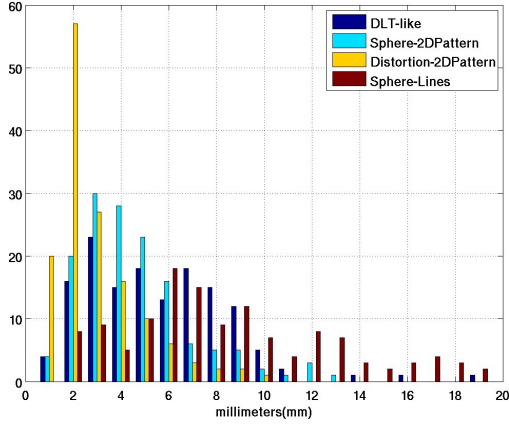


Figure 4.16: Number of reconstructed 3D points within an error in millimeters using an unknown-shape system.

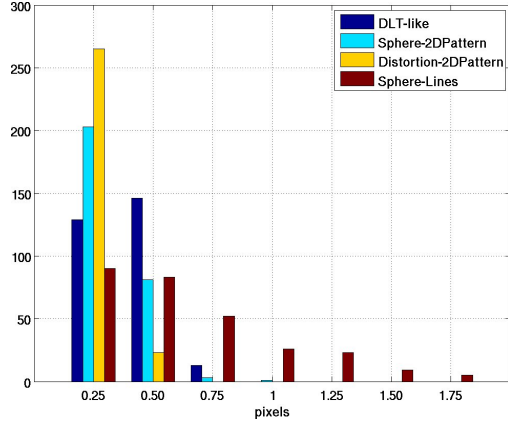


Figure 4.17: Number of reprojected points within an error distance in pixels using an unknown-shape system.

mm. The other two approaches show similar performance, with the majority of the reconstructed 3D points within the 10 mm error. The lowest reprojection error is given by the Distortion-2DPattern with all the points within the 1 pixel error and the other three approaches have a similar behavior with the majority of the reprojected points within the 3 pixel error.

#### 4.4.5 Non-central Hypercatadioptric System

This system is the one described in the hypercatadioptric experiment. The only difference is that the perspective camera is displaced as far as possible from the mirror. This causes that the optical center of the perspective camera is not located at the other focus of the hyperbolic mirror, which is the basic condition for this system to be central. Some of the images used to perform the calibration of this system under the different models are shown in Fig. 4.18. As in the hypercatadioptric case we compute the mirror parameter and the principal point. This result is shown in Table 4.4. We observe that DLT-like and Sphere-2DPattern give similar mirror hyperbolic parameter and Sphere-Lines estimates a parabolic mirror with  $\xi = 1$ .

The 3D error and the reprojection error are shown in Fig. 4.20 and Fig. 4.21, respectively. We observe that all the approaches have a similar behavior even with a non-central system. The Sphere-2DPattern has all its reconstructed 3D points within the 3 pixel error. The DLT-like and Distortion-2DPattern approaches show similar results with one 3D reconstruction error within the 5 mm error. The worst result is given by the Sphere-Lines approach with maximum reconstruction error of 8 mm. We observe that the reprojection error for all the methods is below 2 pixels.

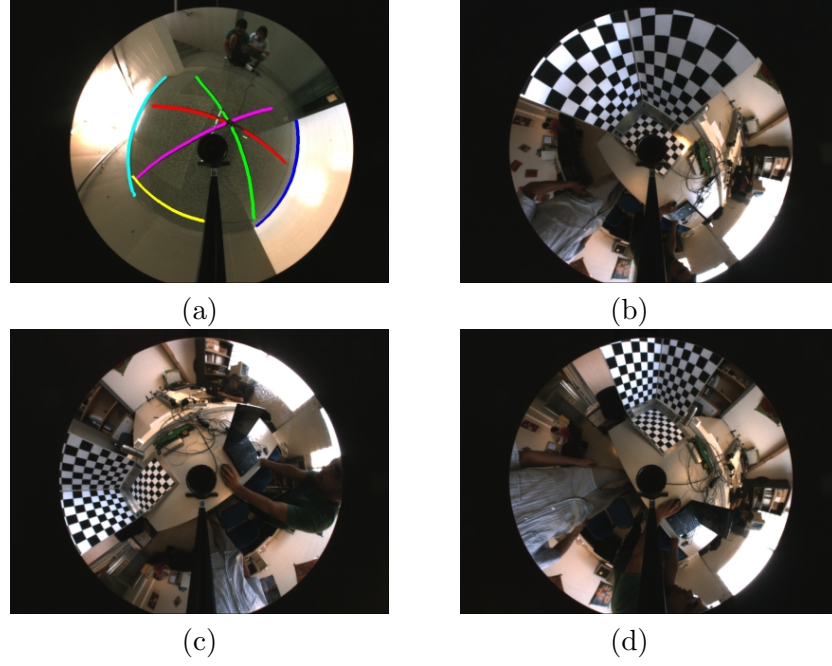


Figure 4.18: Some images used to calibrate the non-central catadioptric system. (a) Sphere-Lines. (b) DLT-like approach. (c,d) Sphere-2DPattern and Distortion-2DPattern approaches.

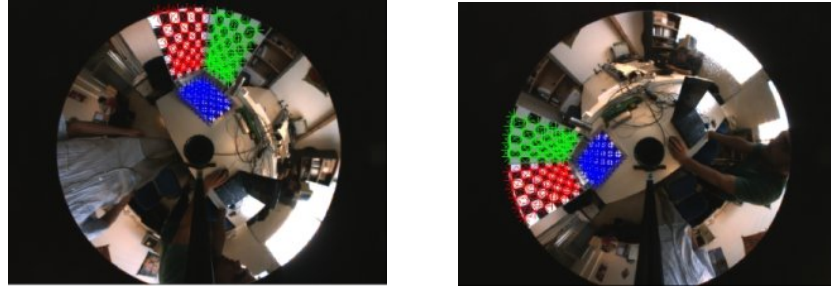


Figure 4.19: Images used in the SfM experiment with reprojected points superimposed (non-central hypercatadioptric).

#### 4.4.6 Distribution of the Calibration Points in the Catadioptric Image

The accuracy of the computed calibration relies on the area occupied by the calibration points in the calibration images. In this experiment we show the importance of the distribution of the calibration points inside the catadioptric images. We define the area to be used by selecting those points closer than  $r$  pixels from the image center. The system to calibrate is a central hypercatadioptric system. Since the images used by each approach are not the same, and also the approaches use different features

#### 4.4. Experiments

	$\xi$	$(u_0, v_0)$
Ground Truth	0.9662	(511.88, 399.25)
Sphere-2DPattern	0.8463	(519.14, 407.87)
Sphere-Lines	1.00	(537.50, 409.82)
DLT-like	0.8819	(525.05, 411.89)

Table 4.4: Comparison of the physical parameters given by the 3 methods based on the sphere model in the non-central system.

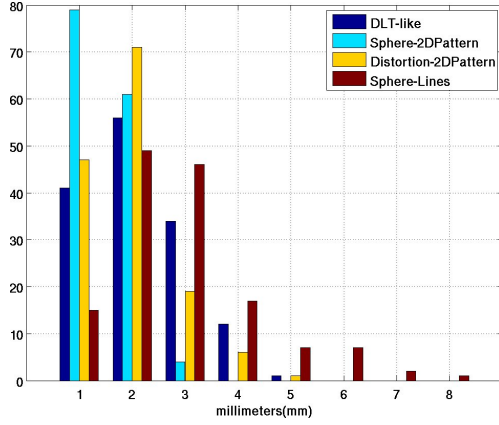


Figure 4.20: Number of reconstructed 3D points within an error distance in millimeters using a non-central hypercatadioptric system.

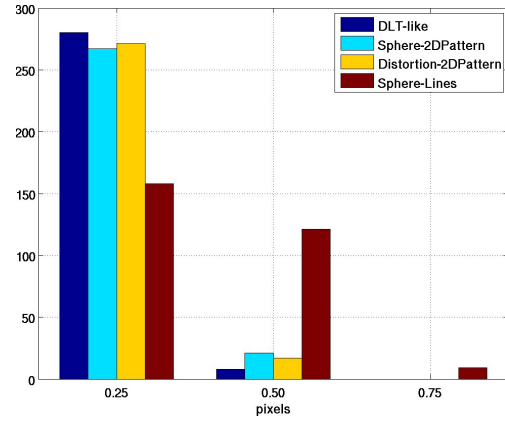


Figure 4.21: Number of reprojected points within an error distance in pixels using a non-central hypercatadioptric system.

(points and lines), a full comparison using the same distances for all the approaches is not possible. Since the approaches Distortion-2DPattern and Sphere-2DPattern share the calibration images set we present their corresponding results together. In the case of DLT-like and Sphere-2DPattern the results are shown separately. The radii  $r$  were chosen depending on the requirements of each approach to calibrate the catadioptric system.

#### Distortion-2DPattern and Sphere-2DPattern

These two approaches require several images to perform the calibration. We select the calibration points that lie closer than  $r$  pixels from the image center in all the images of the set in which these points exist. An example for two different radii  $r$  can be observed in Fig. 4.22. In Fig. 4.23 we show the mean of the reconstruction error for each calibration performed with the points within the areas described by the radii  $r$ . We observe that both approaches give similar results. When the area is small, the number of points decreases and the estimation is worse. The Distortion-2DPattern has an error of 4 mm and the Sphere-2DPattern 2 mm in



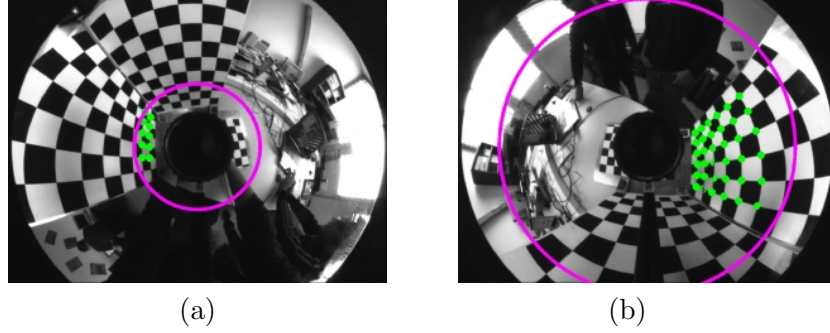


Figure 4.22: Points contained inside the areas described by (a)  $r = 150$  pixels and (b)  $r = 400$  pixels.

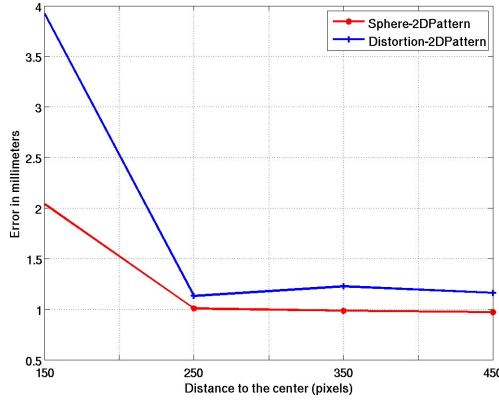


Figure 4.23: 3D reconstruction error in millimeters. The horizontal axis represents the different radii  $r$ .

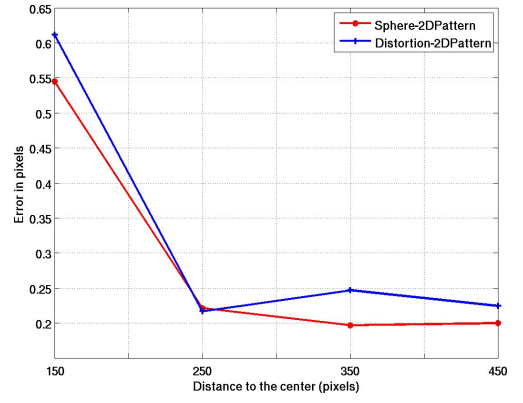


Figure 4.24: Reprojection error in pixels. The horizontal axis represents the different radii  $r$ .

the worst case. This behavior can be explained by the fact that Sphere-2DPattern estimates the image center from data given by the user and the Distortion-2DPattern does it automatically, depending more on the distribution of the points. In Fig. 4.24 we show that reprojection error that is under 1 pixel error for both approaches, even with the smallest  $r$ .

### DLT-like

Since the DLT-like approach only requires one single image, the points are selected directly using different radii  $r$ . In Fig. 4.25 we show two examples of the points selected for two different radii. In Fig. 4.26 and Fig. 4.27 we show the 3D reconstruction error and the reprojection error, respectively. We observe a similar behavior to the previous approaches using large radii. With small radii the results are worse, since with small radii only very few points of the second and third planes are considered (see Fig. 4.25(a)).

## 4.4. Experiments

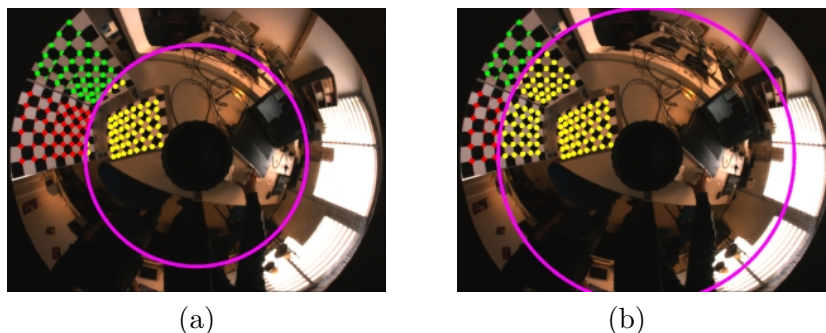


Figure 4.25: Points contained inside the areas described by (a)  $r = 300$  pixels and (b)  $r = 400$  pixels.

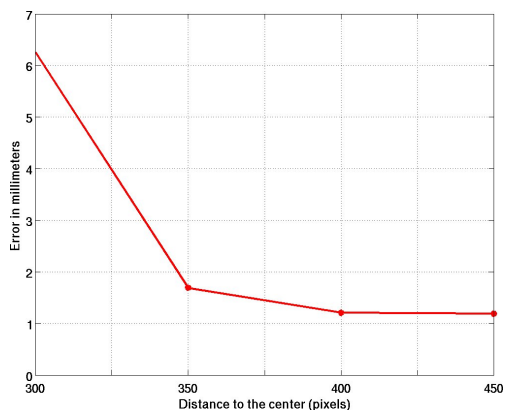


Figure 4.26: 3D reconstruction error in millimeters. The horizontal axis represents the different radii  $r$ .

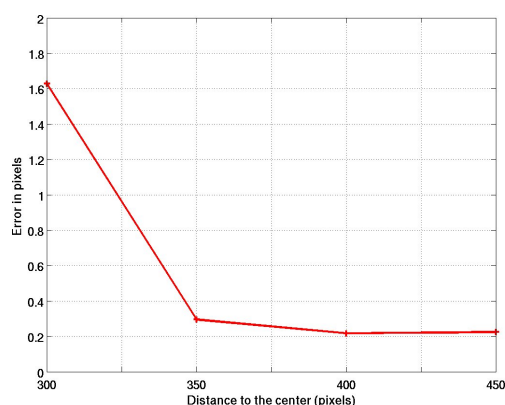


Figure 4.27: Reprojection error in pixels. The horizontal axis represents the different radii  $r$ .

### Sphere-Lines

This approach also requires a single image. The image must contain at least 3 line images. The radii limit the length of the lines used to calibrate the system. An example can be observed in Fig. 4.28. We particularly observed that this approach is quite sensitive to the length and the position of the lines. We show the results where the calibration was possible in the corresponding radii. Fig. 4.29 and Fig. 4.30 show the 3D reconstruction error and the reprojection error, respectively. We observe a similar behavior to the other approaches, but having a bigger error, both in the 3D reconstruction error and the reprojection error.

One may think that a comparison of this method using just a few representative elements, in this case lines, present in one single image, against others where hundreds of representative elements (2D points), are extracted from several images, might be unfair. In this order we tried to calibrate the central catadioptric systems using as many lines as possible, present in the same images of the 3D pattern used to

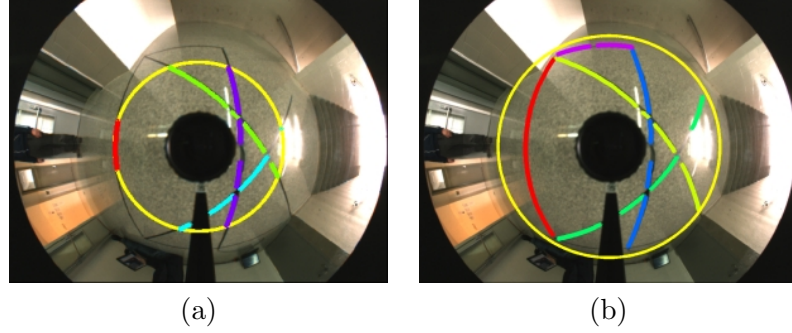


Figure 4.28: Points contained inside the areas described by (a)  $r = 230$  pixels and (b)  $r = 300$  pixels.

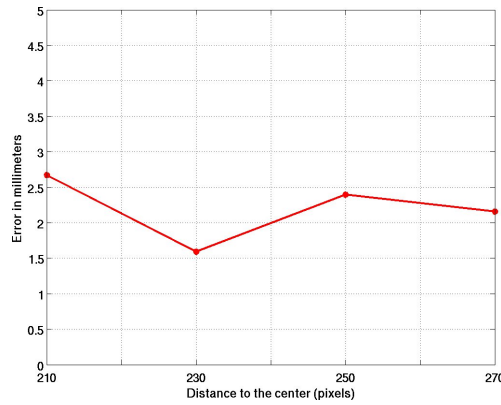


Figure 4.29: 3D reconstruction error in millimeters. The horizontal axis represents the different radii  $r$ .

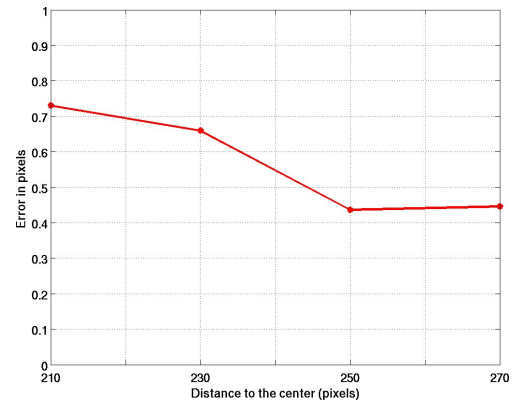


Figure 4.30: Reprojection error in pixels. The horizontal axis represents the different radii  $r$ .

calibrate the system using the other approaches. A few examples of the lines used are shown in Fig. 4.31. The results calibrating the central catadioptric systems using this method did not succeed. We obtained several types of errors and sometimes convergence problems. Because of that we calibrate the central catadioptric systems using the maximum number of lines present in one single image, different from the ones used by the other approaches.

#### 4.4.7 Discussion

After all these experiments with different systems we observe that all the approaches give good results, with the exception of the Sphere-Lines approach with the fish-eye system, basically because this approach is not designed to deal with such systems. In particular, for the fish-eye lens, the best calibration was achieved with the DLT-like approach. In the case of the Unknown-shape camera, the Distortion-2DPattern approach provided the best result. With respect to the non-central hypercatadioptric,

#### 4.4. Experiments

---

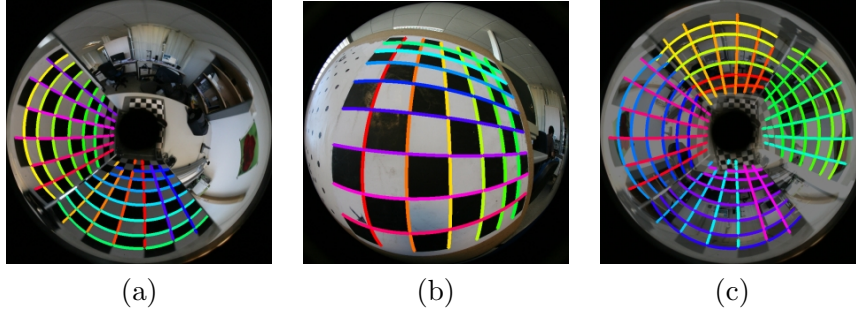


Figure 4.31: Different configurations tested to calibrate the central catadioptric systems using the Sphere-Lines approach. (a) Unknown-shape system, (b) Fish-eye lens and (c) a combination of two images to cover the whole omnidirectional image with lines.

DLT-like, Sphere-2DPattern and Distortion-2DPattern approaches all gave similar accuracy. Finally, the hypercatadioptric system was calibrated slightly better by both the DLT-like and the Sphere-2DPattern approaches. We also analyze the importance of the area occupied by the calibration elements (points, lines) in the calibration image(s). All approaches require this area to be as big as possible to compute a good calibration. In the particular case of the Sphere-Lines approach the lines must be as large as possible and must intersect far from the image center. In terms of computing performance, all these approaches perform a non-linear step after the initialization of the intrinsic and extrinsic parameters is computed. The DLT-like approach is the fastest since it estimates less parameters. Next are the Sphere-2DPattern and the Distortion-2DPattern with several extrinsic parameters corresponding to each image to compute, plus the intrinsic parameters. The slowest method is the Sphere-Lines approach, since it uses a complex geometry to compute the self-polar triangle and takes into account every single pixel contained in the line images.

We also consider the importance on what we need to make these methods to work and the problems observed at the calibration time.

- *Sphere-2DPattern, Distortion-2DPattern.* These two approaches require multiple images of a 2D pattern to perform the calibration. Both of them have automatic corner extractors but most of the times these do not work properly and the points have to be given manually. This is the most tedious part since we have a minimum of eight to ten images, each image containing 48 points giving a total of  $384 \sim 480$  points. Besides that the Sphere-2DPattern approach requires the user to indicate the image center and a minimum of three non-radial points to estimate the focal length.
- *DLT-like.* This approach does not require any prior information but one single omnidirectional image containing 3D points spread on three different planes. The inconvenient with this method is to obtain the 3D points contained in the 3D pattern. All the image points in the 3D pattern images are given manually.

We observed that depending on the image used the mirror parameter  $\xi$  is better or worse estimated. Something similar happens with the Sphere-2DPattern approach.

- *Sphere-Lines*. This approach requires the easiest setting to be constructed. It only requires one single omnidirectional image containing a minimum of 3 lines. One thing observed using this approach is that it strongly depends on the principal point estimation. If this estimation is not accurate enough the calibration is not performed properly. Also we observe some difficulties while calibrating the unknown shape catadioptric system. The number and the location of lines in the image is important to correctly calibrate the system. Sometimes using more than three lines we had convergence problems or we obtained calibrations containing non-real solutions.

Notice that each method has its own manner to extract the points from the images. In this order we should decouple the matching process from the parameter estimation process.

## 4.5 Closure

In this chapter we have presented a comparison of four methods to calibrate omnidirectional cameras available as OpenSource. Two of them require images of a 2D pattern, one requires images of lines and the last one requires one image of a 3D pattern. Three of these approaches use the sphere camera model. This model can give some information about the mirror of the omnidirectional system besides it provides a theoretical projection function. The other approach is based on a distortion function. Both models can deal with any central catadioptric system and fish-eyes. However the Sphere-Lines approach that uses the sphere camera model cannot deal with the fish-eye system. All these approaches use a non-linear step which allows them to have a reprojection error less than 1 pixel. In this chapter we perform a SfM experiment to compare the different approaches with useful criteria. This experiment showed that the calibration reached by any of these methods can give accurate reconstruction results. The distribution of the points in the omnidirectional images is important in order to have an accurate calibration. These points have to cover as much as possible of the omnidirectional image and mainly in the peripheric area.

## Chapter 5

# Hybrid Two-view Relations of Uncalibrated CPS

*In this chapter we present a deep analysis of the hybrid two-view relations combining images acquired with uncalibrated central catadioptric systems and conventional cameras. We consider both, hybrid fundamental matrices and hybrid planar homographies. These matrices contain useful geometric information. We study three different types of matrices, varying in complexity depending on their capacity to deal with a single or multiple types of central catadioptric systems. The first and simplest one is designed to deal with paracatadioptric systems, the second one and more complex, considers the combination of a perspective camera and any central catadioptric system. The last one is the complete and generic model which is able to deal with any combination of central catadioptric systems. We show that the generic and most complex model sometimes is not the best option when we deal with real images. Simpler models are not as accurate as the complete model in the ideal case, but they provide a better and more accurate behavior in presence of noise, being simpler and requiring less correspondences to be computed. Experiments with synthetic data and real images are performed. With the use of these approaches we develop the successful hybrid matching between perspective images and hypercatadioptric images using SIFT descriptors.*

### 5.1 Introduction

The combination of central catadioptric cameras with conventional ones is relevant since a single catadioptric view contains a more complete description of the scene, and the perspective image gives a more detailed description of the particular area or object we are interested in. Some areas where the combination of these cameras has an important role are localization and recognition (Menem and Pajdla, 2004), since a database of omnidirectional images would be more representative with fewer points of view and less data, and perspective images are the simplest query images. In visual surveillance (Chen et al., 2003) catadioptric views provide coarse information about locations of the targets while perspective cameras provide high resolution

images for more precise analysis. In active vision (Jankovic and Naish, 2007) this mixture is naturally implemented; an omnidirectional camera provides peripheral vision while a controllable perspective camera provides foveal vision.

We are particularly interested in the two-view relations between uncalibrated catadioptric and conventional views working directly in the raw images. In the literature very few approaches are presented to compute hybrid two-view relations mixing uncalibrated catadioptric and conventional cameras. These approaches have in common the use of lifted coordinates to deal with the non-linearities of the catadioptric projection. Most of these approaches are presented theoretically and with simple experiments. In this thesis we tackle this situation by performing a deep evaluation of the hybrid fundamental matrix and the hybrid planar homography using simulated data and real images.

To compute the two-view geometry we require pairs of corresponding points between the views. These correspondences are built from previously detected relevant features. Perhaps the most used extractor is the SIFT (Lowe, 2004). However, if SIFT features extracted in an omnidirectional image are matched to features extracted in a perspective image the results are not good, this is because the SIFT descriptor is scale invariant but not projective invariant. We observe that with a simple flip of the omnidirectional image, SIFT matching can still be useful, requiring neither image rectification nor panoramic transformation. Moreover, with the combination of a RANSAC approach with the hybrid two-view relations we are able to perform the automatic robust matching between a conventional image and a catadioptric one without requiring unwarping or any other transformation of the catadioptric image.

## 5.2 Related Work

The multiview geometry problem for central catadioptric systems has been studied in recent years. Some approaches require the calibration of the systems. Svoboda and Pajdla (2002) study the epipolar geometry for central catadioptric systems. Based on the model of image formation they propose the epipolar geometry for catadioptric systems using elliptic, hyperbolic and parabolic mirrors. Geyer and Daniilidis (2001b, 2002b) have shown the existence of a fundamental matrix for paracatadioptric cameras. This has been extended by Sturm (2002) towards fundamental matrices and trifocal tensors for mixtures of paracatadioptric and perspective images. Barreto (2006) and Barreto and Daniilidis (2006) showed that the framework can also be extended to cameras with lens distortion due to the similarities between the paracatadioptric and division models. Recently, Sturm and Barreto (2008) extended these relations to the general catadioptric camera model, which is valid for all central catadioptric cameras. They showed that the projection of a 3D point can be modeled using a projection matrix of size  $6 \times 10$ . They also show the existence of a general fundamental matrix for pairs of omnidirectional images, of size  $15 \times 15$  and plane homographies, also of size  $15 \times 15$ .

In last years some works have been developed considering the hybrid epipolar

geometry for different combinations of uncalibrated central projection systems, including the pin-hole model. To a lesser extent homographies have also been studied using uncalibrated central systems. They establish a relation between the projections on the omnidirectional images of 3D points that lie on a plane. In a seminal work Sturm (2002) proposes two models of hybrid fundamental matrices, a  $4 \times 3$  fundamental matrix to relate a paracatadioptric view and a perspective view and a  $6 \times 3$  fundamental matrix to relate a perspective view and a general central catadioptric view. He also describes the  $3 \times 4$  plane homography which represents the mapping of an image point in a paracatadioptric view to the perspective view. This mapping is unique, unlike the opposite one that maps a point in the perspective image to two points in the paracatadioptric image. He also shows how to use the homographies and fundamental matrices to self-calibrate the paracatadioptric system. All these methods use only lifted coordinates of the points in the omnidirectional image. Menem and Pajdla (2004) propose an algebraic constraint on corresponding image points in a perspective image and a circular panorama. They use a lifting from 3-vector to 6-vector to describe Plücker coordinates of projected rays. Claus and Fitzgibbon (2005) propose the lifting of image points to 6-vectors to build a general purpose model for radial distortion in wide angle and catadioptric lenses. Chen and Yang (2005) use their particular geometric projection of a parabolic mirror. They define two homographies, one for each direction of the mapping. A  $3 \times 4$  mapping from the paracatadioptric image to the perspective image, and a  $3 \times 6$  one from the perspective view to the paracatadioptric, which because of its non-linearity requires an iterative process to be computed. Barreto and Daniilidis (2006) propose a general model that relates any type of central cameras including catadioptric systems with mirrors and lenses and conventional cameras with radial distortion. They apply the lifted coordinates in both images. These lifted coordinates correspond to a map from  $\wp^2$  to  $\wp^5$  through Veronese maps, generating a  $6 \times 6$  fundamental matrix.

Recently, Sturm and Barreto (2008) presented the general catadioptric fundamental matrix, a  $15 \times 15$  matrix which uses a double lifting of the coordinates of the points in both images. In this work they also present the general omnidirectional plane homography. It corresponds to a  $15 \times 15$  matrix that relates the fourth order Veronese map of a point in the first omnidirectional or perspective image to a quartic curve in the corresponding second image. To compute the 225 elements of such a matrix, 45 correspondences of points are required since every correspondence gives 5 equations. Gasparini et al. (2009) extend this work by simplifying the catadioptric homography to a  $6 \times 6$  matrix that relates a 6-vector that corresponds to the lifting of a 3D point lying on a plane and a degenerate dual conic also represented by a 6-vector.

### 5.3 Hybrid Two-view Relations

In this section we explain the hybrid epipolar geometry between catadioptric and perspective images. We also explain the hybrid homography induced by a plane



observed in these two types of image. For both two-view relations we show and analyze three different models. F66, F36 and F34 for fundamental matrices and H66, H36 and H34 for hybrid homographies.

### 5.3.1 Hybrid Fundamental Matrices

Similar to the conventional fundamental matrix, the hybrid fundamental matrix encapsulates the intrinsic geometry between two views. In this case a perspective view and a catadioptric one. It is independent of the observed 3D scene and it can be computed from correspondences of imaged scene points. This geometry can be observed in Fig. 5.1. Under this hybrid epipolar geometry, a point in the perspective image  $\mathbf{q}_p$  is mapped to its corresponding epipolar conic in the catadioptric image  $\mathbf{c}$

$$\mathbf{c} \sim \mathbf{F}_{cp} \mathbf{q}_p. \quad (5.1)$$

The particular shape of the epipolar conics depends on the central projection system we are dealing with, the angle between the epipolar plane and the mirror symmetry axis and the orientation of the perspective/orthographic camera with respect to the mirror. In the case of a paracatadioptric system (with a perfectly orthographic camera) the shape of the epipolar conic corresponds to a circle (Svoboda and Pajdla, 2002) and can be represented as a 4-vector  $\mathbf{c}_{par} = (c_1, c_2, c_3, c_4)^\top$ . In the case of hypercatadioptric systems the epipolar conics can be ellipses, hyperbolas, parabolas, or lines. They are represented as 6-vectors  $\mathbf{c}_{hyp} = (c_1, c_2, c_3, c_4, c_5, c_6)^\top$ . The representation of such conics as matrices are

$$\Omega_{par} = \begin{pmatrix} 2c_1 & 0 & c_2 \\ 0 & 2c_1 & c_3 \\ c_2 & c_3 & 2c_4 \end{pmatrix}, \quad \Omega_{hyp} = \begin{pmatrix} 2c_1 & c_2 & c_4 \\ c_2 & 2c_3 & c_5 \\ c_4 & c_5 & 2c_6 \end{pmatrix}. \quad (5.2)$$

To verify if a point  $\mathbf{q}_c = (q_1, q_2, q_3)^\top$  belongs to a certain conic, it must satisfy the identity  $\mathbf{q}_c^\top \Omega \mathbf{q}_c = 0$ . If we develop this identity with  $\Omega = \{\Omega_{par}, \Omega_{hyp}\}$  we will obtain the expression  $\hat{\mathbf{q}}_c \mathbf{c} = 0$  with  $\mathbf{c} = \{\mathbf{c}_{par}, \mathbf{c}_{hyp}\}$  and  $\hat{\mathbf{q}}_c$  the lifted coordinates we already studied in section 2.2. In the first case as a 4-vector  $\hat{\mathbf{q}}_c = (q_1^2 + q_2^2, q_1 q_3, q_2 q_3, q_3^2)^\top$  and as a 6-vector  $\hat{\mathbf{q}}_c = (q_1^2, q_1 q_2, q_2^2, q_1 q_3, q_2 q_3, q_3^2)^\top$  in the second one.

In the opposite direction, from the catadioptric view to the perspective one, a point in the catadioptric image  $\mathbf{q}_c$  has to be lifted  $\hat{\mathbf{q}}_c$  and then mapped to its corresponding epipolar line in the perspective image

$$\mathbf{l} \sim \mathbf{F}_{cp}^\top \hat{\mathbf{q}}_c. \quad (5.3)$$

In general the relation between catadioptric and perspective images with the fundamental matrix that we call *hybrid fundamental matrix* is established by

$$\mathbf{q}_p^\top \mathbf{F}_{pc} \hat{\mathbf{q}}_c = 0. \quad (5.4)$$

Using the different lifted coordinates (cf. section 2.2) in either the catadioptric image or both the perspective and the catadioptric image, we can define different fundamental matrices.

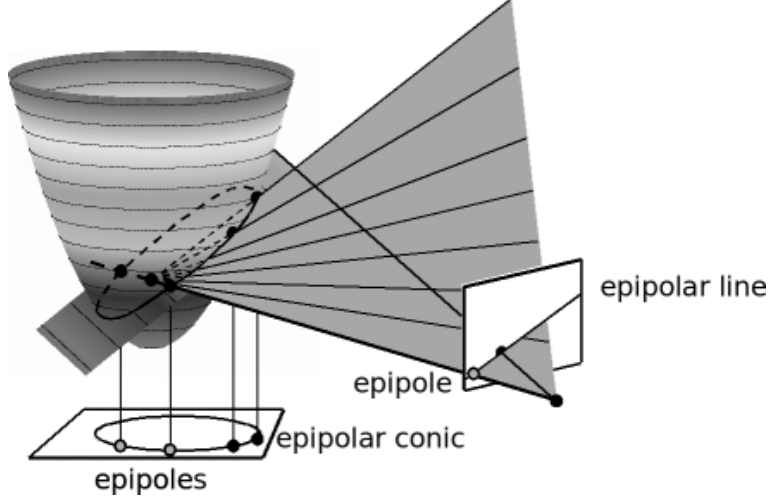


Figure 5.1: Hybrid epipolar geometry.

#### General Catadioptric System, F66 and F36

As mentioned before the generic fundamental matrix between two central projection systems (including catadioptric and perspective ones), is a  $15 \times 15$  matrix which uses a double lifting of the coordinates of the points in both the omnidirectional and the perspective image (Sturm and Barreto, 2008). This lifting represents quartic epipolar curves. Since this matrix is intractable in a practical way we prefer to refer to fundamental matrices easier to compute and that have been successfully applied.

Barreto and Daniilidis (2006) propose a  $6 \times 6$  fundamental matrix, which is able to deal with different combinations of central catadioptric systems and conventional cameras. This matrix is obtained from the lifted coordinates  $\hat{\mathbf{q}}_c$  of points in the omnidirectional and the lifted coordinates  $\hat{\mathbf{q}}_p$  of points in the perspective images.

$$\hat{\mathbf{q}}_p^T \mathbf{F}_{pc} \hat{\mathbf{q}}_c = 0 \quad (5.5)$$

This matrix establishes a bilinear relation between the two views, relating a point in the omnidirectional image to a conic in the perspective one  $\mathbf{c}_p \sim \mathbf{F}_{pc} \hat{\mathbf{q}}_c$ . This conic is composed by two lines. These lines are the forward looking epipolar line and the backward looking epipolar line. These lines can be extracted from an SVD of the epipolar conic. This  $6 \times 6$  fundamental matrix is named F66 and corresponds to the theoretically complete model.

A simpler formulation is established by Sturm (2002) where the lifted coordinates are only applied to the point in the catadioptric image:  $\hat{\mathbf{q}}_c = (q_1^2, q_1q_2, q_2^2, q_1q_3, q_2q_3, q_3^2)^T$ . This matrix establishes a relation between a perspective or affine view and a general central catadioptric view. In (Sturm, 2002) is mentioned that this matrix just works in one direction but experiments in (Puig et al., 2008b) show that it works in both directions. We name this  $3 \times 6$  fundamental matrix, F36. This matrix is an approximation to F66.

### Paracatadioptric system, F34

The paracatadioptric system is a particular catadioptric system composed by a parabolic mirror and an orthographic camera. In this case the shape of the epipolar conic is a circle and we use the simplified lifting  $\hat{\mathbf{q}}_c = (q_1^2 + q_2^2, q_1 q_3, q_2 q_3, q_3^2)^\top$  (2.12) in the coordinates of the points in the omnidirectional image. The coordinates of the points in the perspective image are normal homogeneous coordinates. We name this  $3 \times 4$  fundamental matrix, F34.

### Computation of the Hybrid Fundamental Matrix

We use a DLT-like approach (Hartley and Zisserman, 2000) to compute the hybrid fundamental matrix. It is explained as follows. Given  $n$  pairs of corresponding points  $\mathbf{q}_c \leftrightarrow \mathbf{q}_p$ , solve the equations (7.1) or (5.5) to find  $\mathbf{F}_{cp}$ . The solution is the least eigenvector of  $\mathbf{A}^\top \mathbf{A}$ , where  $\mathbf{A}^\top$  is the equation matrix

$$\mathbf{A}^\top = \begin{pmatrix} \mathbf{q}_{p_1} q_{c_{1_1}} & \cdots & \mathbf{q}_{p_1} q_{c_{1_m}} \\ \vdots & \ddots & \vdots \\ \mathbf{q}_{p_n} q_{c_{n_1}} & \cdots & \mathbf{q}_{p_n} q_{c_{n_m}} \end{pmatrix}. \quad (5.6)$$

The number of pairs of corresponding points  $n$  needed to compute the hybrid fundamental matrix depends on the number of elements of the fundamental matrix to be computed. Each pair of corresponding points gives one equation. Therefore 35, 17 and 11 correspondences are required at least to compute the F66, F36 and F34, respectively. It is recommended that  $n \gg \text{size}(\mathbf{F}_{pc})$ . The number of parameters required to compute these approaches is crucial if we take into account that in wide-baseline image pairs the correspondences are difficult to obtain, and much more in images coming from different sensor types. In this case the F34 has a clear advantage over the other two more complex approaches.

### Rank 2 constraint

The above fundamental matrices are, like for the purely perspective case, of rank 2. If the task we are interested in requires the epipoles, it is mandatory to ensure that the estimated fundamental matrix has rank 2. To impose the rank 2 constraint we have tried two options. One is to enforce this constraint minimizing the Frobenius norm using SVD as explained in (Hartley and Zisserman, 2000) which we call *direct imposition* (DI). The other option is to perform a non-linear re-estimation process minimizing the distance from points in one image to their corresponding epipolar conic or line in the other one, using the Levenberg-Marquardt (LM) algorithm. To guarantee the rank 2 we use a matrix parameterization proposed by Bartoli and Sturm (2004) which is called the *orthonormal representation* of the fundamental matrix. This approach was originally applied to  $O(3)$  matrices and we adapt it to F34, F36 and F66.

#### Computing the epipoles

The process involved in the computation of the epipoles from the three tested hybrid fundamental matrices, F34, F36 and F66 is based on the computation of their corresponding null-spaces.

The hybrid F34 matrix has a one-dimensional left null-space and one right null vector. This one corresponds to the epipole in the perspective image. The two epipoles in the omnidirectional image are extracted from the left null-space. We have to compute the left null-vectors that are valid lifted coordinates. From these, the epipoles are extracted in a closed form. The 4-vectors must satisfy the following quadratic constraint:

$$\hat{\mathbf{q}}_c \sim \begin{pmatrix} q_1^2 + q_2^2 \\ q_1 q_3 \\ q_2 q_3 \\ q_3^2 \end{pmatrix} \Leftrightarrow \hat{q}_{c1} \hat{q}_{c4} - \hat{q}_{c2}^2 - \hat{q}_{c3}^2 = 0 \quad (5.7)$$

In the case of the F36 matrix we follow a similar process. The epipole in the perspective image is also given by the right null vector and the two epipoles of the omnidirectional image are extracted from the left null-space of this matrix. In this case the valid lifted coordinates have to satisfy the following equations:

$$\hat{\mathbf{q}}_c \sim \begin{pmatrix} q_1^2 \\ q_1 q_2 \\ q_2^2 \\ q_1 q_3 \\ q_2 q_3 \\ q_3^2 \end{pmatrix} \Leftrightarrow \begin{cases} \hat{q}_{c1} \hat{q}_{c3} - \hat{q}_{c2}^2 = 0 \\ \hat{q}_{c1} \hat{q}_{c6} - \hat{q}_{c4}^2 = 0 \\ \hat{q}_{c2} \hat{q}_{c6} - \hat{q}_{c5}^2 = 0 \end{cases} \quad (5.8)$$

In the case of the F66 matrix we use the same approach as for F36 to extract the epipoles corresponding to the omnidirectional image from the left null-space of the fundamental matrix. For the epipole in the perspective one a different process is required. We extract it from the null-vector of the degenerate epipolar conic  $\Omega_p \sim F66 \hat{\mathbf{q}}_c$  projected from a point in the omnidirectional image to the perspective image. This conic contains the two points  $\mathbf{q}_+$  and  $\mathbf{q}_-$ .

#### 5.3.2 Hybrid Homography

Hybrid homographies relate the projections of points that lie on a plane on different types of images. In particular we analyze the homographies that relate omnidirectional and perspective images. Similarly as before with fundamental matrices, we consider three different models. The general model H66 and two simplified models H36 and H34.

### Generic Model, H66

From (Sturm and Barreto, 2008) the projection of a 3D point in any central catadioptric system using lifted coordinates can be described by a  $6 \times 10$  projection matrix  $P_{cata}$

$$\hat{\mathbf{q}} \sim P_{cata} \hat{\mathbf{Q}}, \quad P_{cata} = \hat{K} X_\xi \hat{R}_{6 \times 6} \begin{pmatrix} \mathbf{I}_6 & \mathbf{T}_{6 \times 4} \end{pmatrix} \quad (5.9)$$

If we assume that the 3D points lie on a plane  $z = 0$ ,  $\mathbf{Q} = (Q_1, Q_2, 0, 1)^\top$  the non-zero elements of its lifted representation is a 6-vector  $\hat{\mathbf{Q}}_c = (Q_1^2, Q_1 Q_2, Q_2^2, Q_1, Q_2, 1)^\top$  and the projection matrix reduces to size  $6 \times 6$ :

$$H66 = \hat{K} X_\xi \hat{R} (\mathbf{I}_{6 \times 3} [\mathbf{t}_1 \mathbf{t}_2 \mathbf{t}_4]) \quad (5.10)$$

where  $\mathbf{t}_i$  is the  $i$ -th column of the matrix  $\mathbf{T}$  and H66 is the  $6 \times 6$  homography matrix relating the lifting of the 2D coordinates of the points on the plane to their dual conic representation on the image plane  $\Omega$  as explained in section 2.1.

This homography can also relate the projection of these 3D points in two different images. And in particular in two images acquired with different sensors, a conventional one  $\mathbf{q}_p$  and an omnidirectional one  $\mathbf{q}_c$ .

$$\hat{\mathbf{q}}_p \sim H66 \hat{\mathbf{q}}_c \quad (5.11)$$

To compute this homography we use a DLT-like (Direct Linear Transformation) approach. As in the perspective case we need correspondences  $\mathbf{q}_p^i \leftrightarrow \mathbf{q}_c^i$  between points lying on the plane in the conventional image  $\mathbf{q}_p^i$  and in the omnidirectional one  $\mathbf{q}_c^i$ . From (5.11) we obtain

$$[\widehat{\mathbf{q}}_p]_\times H66 \hat{\mathbf{q}}_c = \mathbf{0} \quad (5.12)$$

If the  $j$ -th row of the matrix H66 is denoted by  $\mathbf{h}_j^\top$  and arranging (5.12) we have

$$[\widehat{\mathbf{q}}_p]_\times \otimes \hat{\mathbf{q}}_c \begin{pmatrix} \mathbf{h}_1^\top \\ \mathbf{h}_2^\top \\ \mathbf{h}_3^\top \\ \mathbf{h}_4^\top \\ \mathbf{h}_5^\top \\ \mathbf{h}_6^\top \end{pmatrix} = \mathbf{0} \quad (5.13)$$

These equations have the form  $\mathbf{A}^i \mathbf{h} = \mathbf{0}$ , where  $\mathbf{A}^i$  is a  $36 \times 6$  matrix, and  $\mathbf{h} = (\mathbf{h}_1^\top, \mathbf{h}_2^\top, \mathbf{h}_3^\top, \mathbf{h}_4^\top, \mathbf{h}_5^\top, \mathbf{h}_6^\top)^\top$  is a 36-vector made up of the entries of matrix H66. The matrix  $\mathbf{A}^i$  has the form

## 5.4. Evaluation of the Hybrid Two-view Models

$$A^i = \begin{pmatrix} 0 & 0 & q_3^2 \hat{\mathbf{q}}_c & 0 & -2q_3 q_2 \hat{\mathbf{q}}_c & q_2^2 \hat{\mathbf{q}}_c \\ 0 & -q_3^2 \hat{\mathbf{q}}_c & 0 & q_3 q_2 \hat{\mathbf{q}}_c & q_3 q_1 \hat{\mathbf{q}}_c & -q_2 q_1 \hat{\mathbf{q}}_c \\ q_3^2 \hat{\mathbf{q}}_c & 0 & 0 & -2q_3 q_1 \hat{\mathbf{q}}_c & 0 & q_1^2 \hat{\mathbf{q}}_c \\ 0 & q_3 q_2 \hat{\mathbf{q}}_c & -q_3 q_1 \hat{\mathbf{q}}_c & -q_2^2 \hat{\mathbf{q}}_c & q_2 q_1 \hat{\mathbf{q}}_c & 0 \\ -q_3 q_2 \hat{\mathbf{q}}_c & q_3 q_1 \hat{\mathbf{q}}_c & 0 & q_2 q_1 \hat{\mathbf{q}}_c & -q_1^2 \hat{\mathbf{q}}_c & 0 \\ q_2^2 \hat{\mathbf{q}}_c & -2q_2 q_1 \hat{\mathbf{q}}_c & q_1^2 \hat{\mathbf{q}}_c & 0 & 0 & 0 \end{pmatrix} \quad (5.14)$$

and is rank 3, so each correspondence gives 3 independent equations. Thus we need at least 12 correspondences to compute H66 (Gasparini et al., 2009).

### Simplified Homographies H34 and H36

We also consider two approximations of the hybrid homography. H34 and H36 are two hybrid homographies that map a lifted vector (2.11) or (2.12) corresponding to a point in the omnidirectional image  $\hat{\mathbf{q}}_c$  to a point in the corresponding plane  $\mathbf{q}_p$  in homogeneous coordinates. The former is related to the theoretical model of a paracatadioptric system and the latter considers any central catadioptric system. Similar to (5.13) we consider the Kronecker product of  $\mathbf{q}_p$  and  $\hat{\mathbf{q}}_c$ . Both homographies are computed using a DLT approach. Since each correspondence gives two equations we require at least 6 correspondences to compute H34 and 9 correspondences to compute H36.

## 5.4 Evaluation of the Hybrid Two-view Models

In this section we analyze the behavior of the three fundamental matrices (F66, F36, F34) and the three homographies (H66, H36, H34). We present some experiments performed with synthetic data and real images.

### 5.4.1 Simulated Data

We use a simulator which generates omnidirectional images coming from a hypercatadioptric system and perspective images from a pin-hole model. The two sensors are placed in a virtual volume of  $5 \times 2.5 \times 7$  m. width, height and depth, respectively, where points are located randomly ( $n \gg 35$ ) in the case of the fundamental matrix and in planes in the case of the homographies. The perspective camera has a resolution of  $1000 \times 1000$  pixels and is located at the origin of the coordinate system. The omnidirectional camera is located to have a good view of the whole scene. We use the sphere camera model (Barreto and Daniilidis, 2006) to generate the omnidirectional image. We consider two real hypercatadioptric systems with mirror parameters of  $\xi = 0.9662$  (m1) and  $\xi = 0.7054$  (m2), from two real hyperbolic mirrors designed by

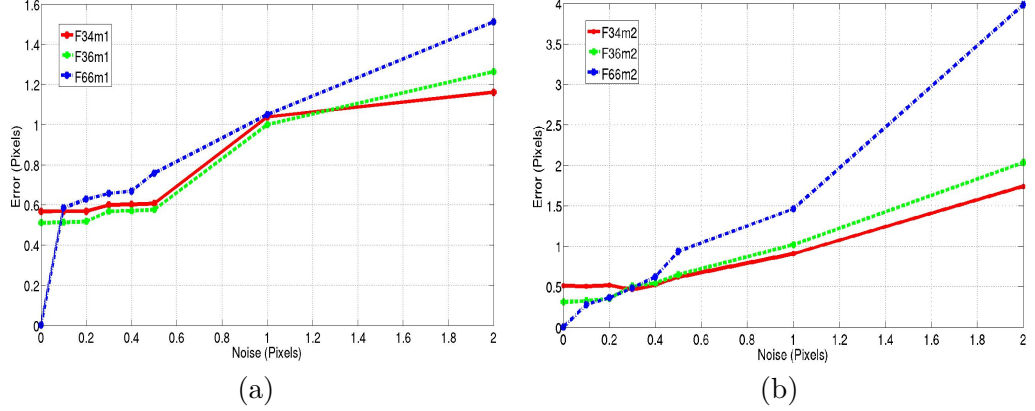


Figure 5.2: Behavior of the three fundamental matrices in function of image noise ( $\sigma$ ): RMSE of points to epipolar conics and lines using mirrors (a) m1 and (b) m2.

Neovision<sup>1</sup> and Accowle<sup>2</sup>, respectively. As a common practice and because we are using lifted coordinates we apply a normalization to the image coordinates where the origin is the image center and the width and height are 1. Once the points are projected we add Gaussian noise, described by  $\sigma$ , in both images.

### Analysis of the Fundamental Matrices

The fundamental matrices are computed using a Levenberg-Marquardt<sup>3</sup> non-linear minimization of the geometric distance from image points to epipolar lines and conics using the point to conic distance proposed by (Sturm and Gargallo, 2007). For every  $\sigma$  representing the amount of noise we repeat the experiment 10 times to avoid particular cases due to random noise. We show the mean of these iterations. Fig. 5.2 shows the distances from points to their corresponding epipolar conics and lines as a function of image noise.

From Fig. 5.2 we can observe that when there is no noise present in the image the F66 shows the best performance, which is expected since F66 is the theoretically correct model. This changes when noise increases. In this case the F34 and F36 show a better performance, being consistent with the noise present in the images. The F34 shows a better performance with the mirror m1 since this one is closer to a parabolic mirror, the one the matrix F34 was designed to deal with. The residuals of F36 are slightly larger than the ones from F34. We observe that F66 is unstable when noise is present in the images. This behavior can be caused by the over-parameterization of the model, the more the parameters the higher the sensitivity to noise; that can also explain the difference between the F36 and F34.

We also estimate the epipoles from the 3 hybrid fundamental matrices, using the

<sup>1</sup><http://www.neovision.cz>

<sup>2</sup><http://www.accowle.com>

<sup>3</sup>`lsqnonlin` function provided by Matlab

## 5.4. Evaluation of the Hybrid Two-view Models

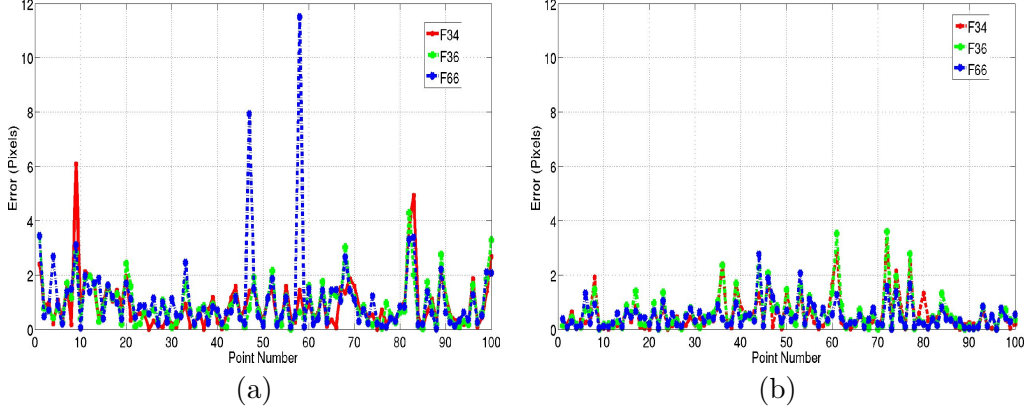


Figure 5.3: RMSE error from points to their corresponding epipolar conic and lines. (a) Using the direct imposition of the rank 2. (b) Using the LM algorithm.

	True Value	F66		F36		F34	
		DI	LM	DI	LM	DI	LM
$e_1$	(500,303.07)	(499.24,302.48)	(499.60,303.42)	(500.23,303.73)	(499.66,303.98)	(500.07,303.41)	(499.66,303.95)
$e_2$	(500,200)	(503.31,199.17)	(501.55,201.08)	(501.03,201.27)	(501.52,202.03)	(500.53,201.79)	(501.29,202.42)
RMSE	0.0	18.04	0.76	1.16	1.01	0.85	0.99

Table 5.1: Epipoles estimated by the three fundamental matrices. DI = direct imposition. LM = Levenberg-Marquardt.

m1 hypercatadioptric system. In this experiment we add  $\sigma = 1$  pixel Gaussian noise to both images. We test the two approaches DI and LM to get a rank 2 matrix (cf. section 5.3.1). We evaluate the performance of these approaches by the accuracy of the estimated epipoles and by the residual, which is the RMSE of the distances from the points used to compute the fundamental matrix to their corresponding epipolar lines and conics. In Fig. 5.3 we show the residuals for the three approaches imposing the rank 2 constraint by the *direct imposition* and by using the LM algorithm with *orthonormal representation*.

In Fig. 5.3(a) we can observe that F66 is very sensitive to the direct imposition of the rank 2 property with maximum errors of 8 and 12 pixels. This occurs because we are transforming a good solution that passes through the points in the perspective and omnidirectional images into a new matrix of rank 2 that contains the epipoles and makes all epipolar lines and conics to pass through them but far from the points in the corresponding images. This does not occur with the LM algorithm which uses the *orthonormal representation* because it imposes the rank 2 property and at the same time minimizes the distance between points and epipolar lines and conics, having a maximum error of 3 pixels.

Table 5.1 shows the epipoles from these two approaches. We can see from it that the three approaches give similar results in computing the epipole but we also observe an increment in the distance from points to conics and the minimization obtained with the LM algorithm, all this as expected. Once more F34 shows an interesting behavior giving a small distance to conics even with the DI approach.



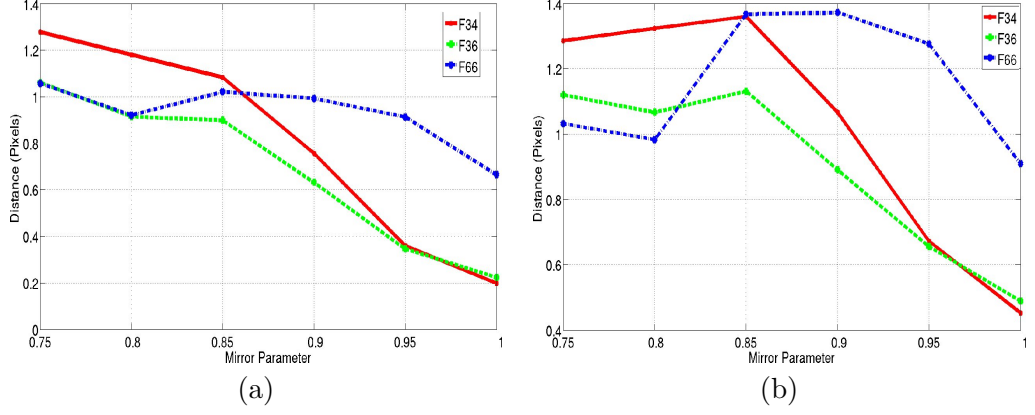


Figure 5.4: Behavior of the three fundamental matrices as a function of the mirror parameter( $\xi$ ): Mean distances from points to epipolar conics in (a) omnidirectional image, and (b) perspective image.

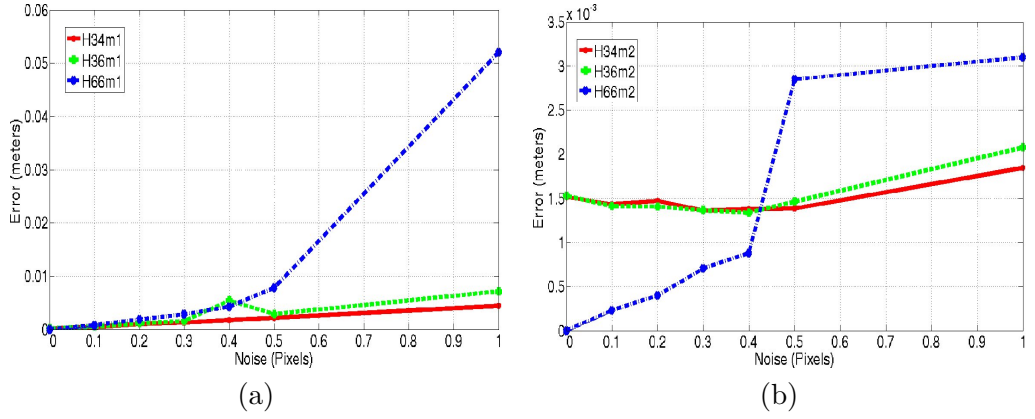


Figure 5.5: Comparison between the three approaches to compute the hybrid homography. Using mirrors (a) m1, (b) m2.

This adds another advantage to F34.

As observed from the previous experiments F34 shows a good performance dealing with images coming from a hypercatadioptric system. In order to test this behavior we designed the following experiment. We modify the mirror parameter  $\xi$  from the hyperbolic case ( $0 < \xi < 1$ ) to the parabolic case ( $\xi = 1$ ) (Barreto and Daniilidis, 2006). We add  $\sigma = 1$  pixels Gaussian noise in both images and repeat the experiment 10 times to avoid bias since we are using random noise. In Fig. 5.4 we observe that F34 can deal better with hypercatadioptric images when the mirror shape is close to a parabola ( $\xi = 1$ ) and not as good as F66 and F36 models which are designed to deal with this type of systems but still having a RMSE of 1.28 pixels with the hyperbolic mirror defined by  $\xi = 0.75$ .

### Analysis of Homographies

We perform experiments computing the hybrid homographies relating a plane in the ground and its projection in an omnidirectional image as well as the projection of a planar scene in both omnidirectional and perspective images. We use the same simulator as in the fundamental matrix case also considering the two different hyperbolic mirrors (m1) and (m2). The 3D points are distributed in a planar pattern. This pattern is composed of a squared plane with  $11 \times 11$  points and a distance between points of 40cm. The goal of the first experiment is to know the behavior of the three homography approaches in presence of noise. We add different amounts of Gaussian noise described by its standard deviation ( $\sigma$ ) to the coordinates of the points in the omnidirectional image. The DLT algorithm followed by a non-linear step using Levenberg-Marquardt minimizing errors in the image are used to compute the homographies. For every  $\sigma$  we repeat the experiment 10 times in order to avoid particular cases due to random noise. The error of the projected points in the ground plane is shown in Fig. 5.5. We observe that the three approaches have a similar behavior. When the amount of noise is low the best performance is given by H66, in fact it is the only one that has a zero error when we use noiseless data. When the amount of noise increases the performance of H66 decreases and H34 and H36 remain with smaller errors. This result shows that H66 is more sensitive to noise than the other two approaches. The difference between the errors using the different mirrors is explained because the area occupied by the plane using m2 is bigger than the area covered using m1. With the m1 mirror we have errors of 5.2cm with the H66 but with the m2 mirror this error decreases to 3mm in both cases with  $\sigma = 1$ pixel.

The next experiment maps a point from a plane in a perspective image to its projection in the omnidirectional image. In this case we added Gaussian to both perspective and omnidirectional image coordinates. We project a point from the omnidirectional to the perspective image, where the map is direct. In Fig. 5.6 we can observe the experiment using different amounts of Gaussian noise  $\sigma$ . Again H34 and H66 give better results than H66 except for the case with a very small amount of noise.

In the opposite direction, the homography maps a point in the perspective image to a conic in the omnidirectional one. Since the extraction of the corresponding point from this conic is difficult, a way to overcome this problem is to compute a different homography, which maps lifted coordinates in the perspective image to a single point in the omnidirectional one.

From the simulations we observe that the hybrid fundamental matrices and the hybrid homographies with less parameters, F34 and H34 have a good performance even dealing with hyperbolic mirrors. Also they are less sensitive to noise than the theoretically correct and more general models F66 and H66. Note also that simpler models F34 and H34 require fewer point correspondences to be computed and therefore they have advantages in practice.

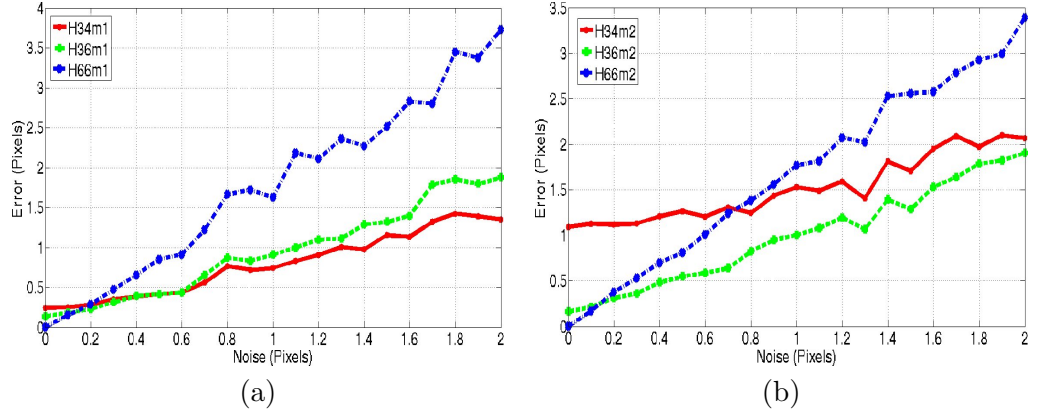


Figure 5.6: Noise sensitivity of the hybrid homographies between omnidirectional and perspective images. (a) Using m1 and (b) using m2.

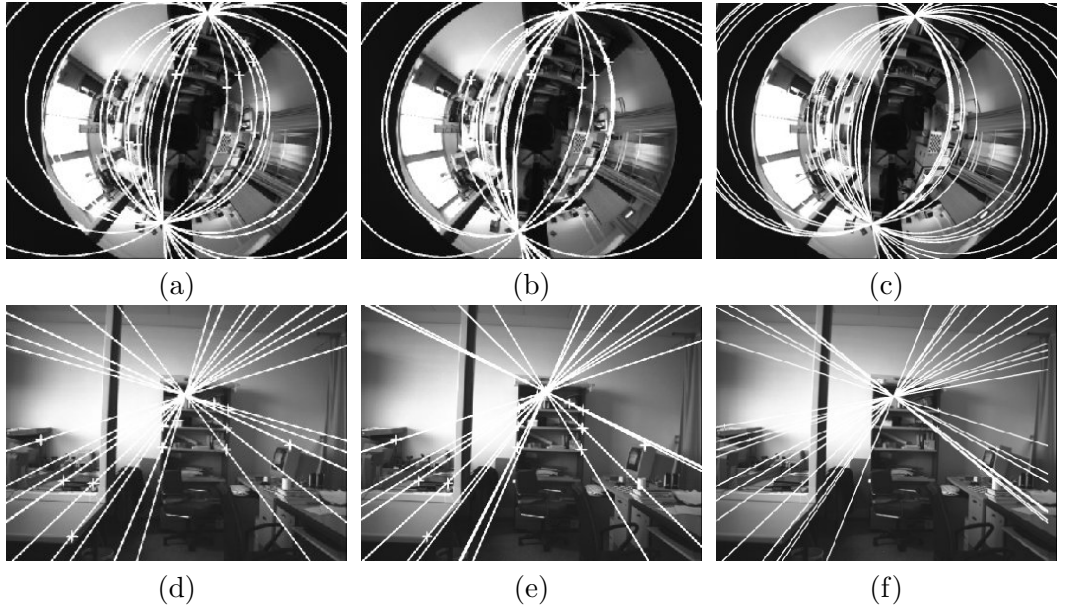


Figure 5.7: (a-c) Epipolar conics using F34, F36 and F66. (d-f) Epipolar lines using F34, F36 and F66.

#### 5.4.2 Experiments with Real Images

We also performed experiments with real images coming from a hypercatadioptric system and from a conventional camera. We compare the accuracy of the three methods to compute both the hybrid fundamental matrix and the hybrid homography.

#### 5.4. Evaluation of the Hybrid Two-view Models

	D2C			D2L		
	F34	F36	F66	F34	F36	F66
No Rank 2	0.68	0.66	0.67	1.10	1.05	0.9
DI	0.87	1.64	21.06	1.36	2.07	3.82
LM	0.71	0.70	0.69	1.13	1.07	1.01

Table 5.2: Mean of the distances to epipolar conics (D2C) and lines (D2L) for the 70 corresponding points in real images. Using Direct Imposition (DI) and Levenberg-Marquardt(LM)

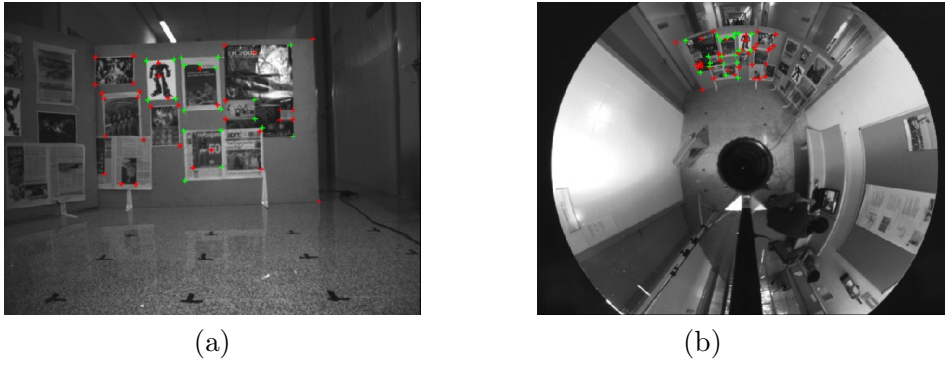


Figure 5.8: Images used to compute the hybrid homographies. (a) Perspective image. (b) Omnidirectional image. Points used to compute the homographies are in red and test points in green.

#### Hybrid Fundamental Matrix

In this case we use 70 manually selected pairs of corresponding points to compute the three approaches (F34, F36, F66). In order to measure the performance of  $F$  we calculate the root mean square error of the geometric distance from each correspondence to its corresponding epipolar conic or line. Table 5.2 shows these distances for the estimated  $F$  without imposing rank 2 and for the two ways to obtain the rank 2 fundamental matrix. We can observe that when we impose the rank 2 the error increases in particular with F66. With the *orthogonal normalization* using the LM algorithm F66 gives the best result but with very few difference with alternate models F34 and F36. When we impose the rank 2 constraint we eliminate a few degrees of freedom of the matrix that better adjusts to the data so, the residual error must be worse actually. From Fig. 5.7 we can observe the epipolar lines and conics from the three approaches. We also observed that a great number of correspondences, larger than the minimum is required to have a reasonable accuracy. Using F36 we obtain good results with 50 (three times the minimum) correspondences. This gives a good reason to use the F34 for further applications.

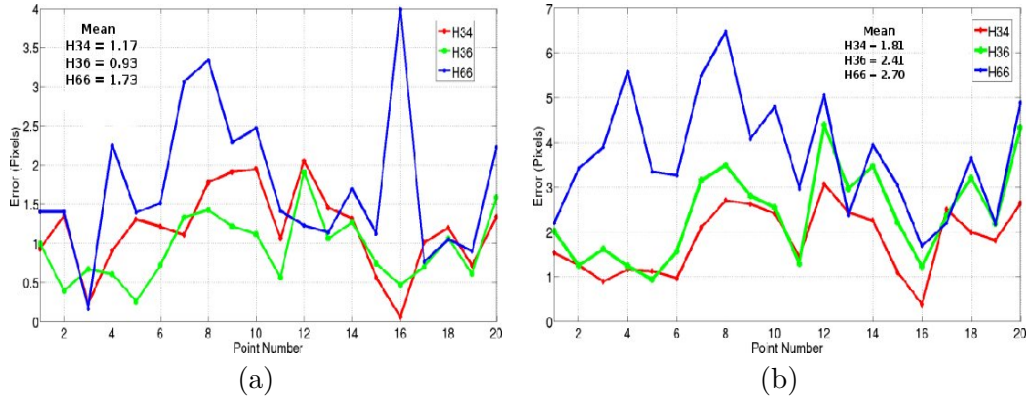


Figure 5.9: Euclidean distance from the estimated points to the test points (a) Omnidirectional image. (b) Perspective image.

	SIFT points	Matches/Inliers
Omnidirectional	2877	137/9
Unwarped image	4182	179/68
Flipped image	2867	207/76

Table 5.3: Output from the SIFT matching using the original, unwarped and flipped omnidirectional image.

## Hybrid Homographies

In this experiment we select 55 correspondences manually. From these correspondences we use 35 to compute the hybrid homographies H34, H36 and H66. We use the rest as test points. If we want to map points in the opposite direction, i.e., from the perspective image to the omnidirectional one we require the inverse mapping of matrices H34 and H36. Since these matrices are not square their computation is not possible. In this order we compute separate homographies to map points in this direction. With this computation we also avoid the extraction of the points from the corresponding conics. With respect to H66 it was shown in (Gasparini et al., 2009) that two homographies have to be computed, since the inverse matrix does not correspond to the opposite mapping. In Fig. 5.8 we show the images used to compute the hybrid homographies. In Fig. 5.9(a) we show the error corresponding to the Euclidean distance between the estimated and the test points in the omnidirectional image. Fig. 5.9(b) shows the error in the perspective image. We observe that H34 and H36 have a similar behavior. In both images we also show the corresponding means of the error. H34 has the best performance in the perspective image while H36 has it in the omnidirectional one. The worst behavior in both images corresponds to H66. All the approaches show a considerable error, which can be caused by the small area occupied by the points in the omnidirectional image.



Figure 5.10: Matching directly the SIFT points in the omnidirectional and perspective images. (a) Using the unwarped image. (b) Using the flipped omnidirectional image. The matches with the normal omnidirectional image are not shown since near by all are outliers.

## 5.5 Automatic Matching Using Hybrid Two-view Relations

In this section we present the robust automatic matching between uncalibrated hypercatadioptric images and perspective images. The first step of a matching process between wide-baseline images is to obtain an initial or putative set of pairs of corresponding features. Reasonable matching of two omnidirectional images using well-known features like SIFT, SURF or MSER has been reported (Murillo et al., 2007b; Guerrero et al., 2008; Mauthner et al., 2006). It is known that the SIFT descriptor is scale invariant but not camera invariant, making it difficult to directly match omnidirectional images with perspective images using standard SIFT features. In (Puig et al., 2008b) we observed that by unwarping the omnidirectional image we improved the number of matches between hybrid image pairs. This unwarping included a vertical flip of the omnidirectional image. In (Puig and Guerrero, 2009)

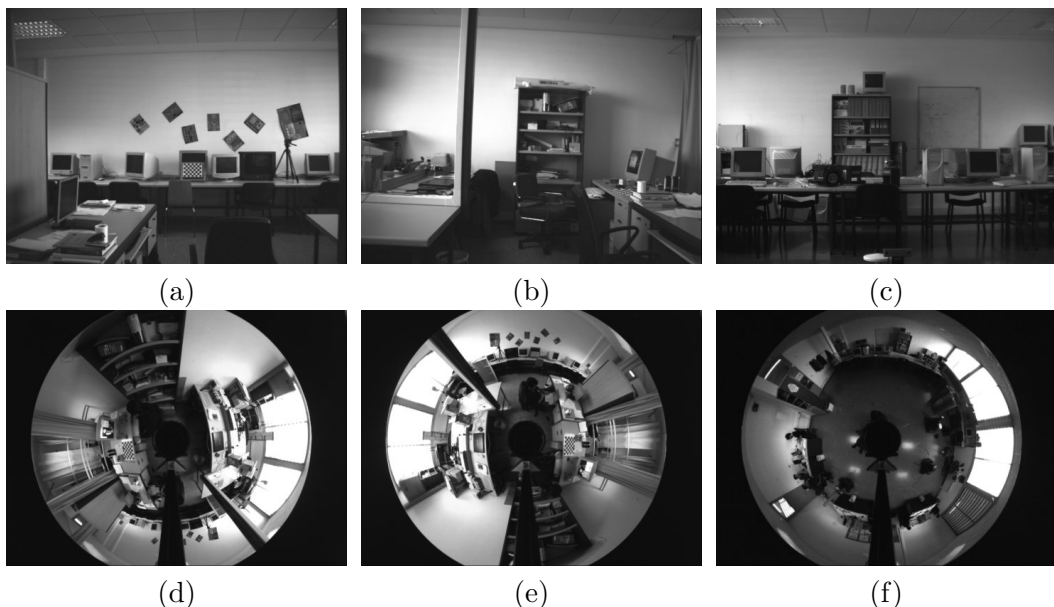


Figure 5.11: Some of the images used to test the automatic matching using the fundamental matrix.

we realized that this improvement was mainly caused by the simple flip of the omnidirectional image and not by the whole unwarping process. The SIFT descriptor is designed to be scale and rotation invariant and even camera invariant if we consider that flipping an omnidirectional image can produce good matches with a conventional image. We observe that the SIFT descriptor is not projective invariant, since the projective mirror effect is responsible for the majority of matching failures.

To evaluate that, we show in Fig. 5.10 the direct matching between SIFT points from a normal omnidirectional image and a perspective image. In this work we use the SIFT implementation by Vedaldi (2007). The inliers and outliers obtained were counted manually. Table 5.3 shows that near by all matches are wrong if the omnidirectional image is directly used. Using the unwarped and the flipped transformation of the omnidirectional image we repeat the experiment. In these cases we observe an important increment on the number of correct matches showing both similar results. More results are shown in Table 5.4.

Note that this initial matching between the perspective and the flipped omnidirectional image has a considerable amount of inliers but also many outliers. This scenario requires a robust estimation technique and a geometric model to detect the inliers and reject the outliers. Depending on the situation, either the hybrid epipolar geometry or the hybrid homography, can be used.



### Hybrid Fundamental Matrix

In a general case where the points are in any part of a 3D scene the fundamental matrix is used. The automatic process to perform the matching between an omnidirectional image and a perspective one, using the hybrid fundamental matrix as geometric constraint is as follows:

1. **Initial Matching.** Scale invariant features (SIFT) are extracted from perspective and flipped omnidirectional images and matched based on their intensity neighborhood.
2. **RANSAC robust estimation.** Repeat for  $r$  samples, where  $r$  is determined adaptively:
  - (a) Select a random sample of  $k$  corresponding points, where  $k$  depends on what model we are using (if F34,  $k = 11$ , if F36,  $k = 17$  or if F66  $k = 35$ ). Compute the hybrid fundamental matrix  $F_{cp}$  as mentioned before.
  - (b) Compute the distance  $d$  for each putative correspondence,  $d$  is the geometric distance from a point to its corresponding epipolar conic (Sturm and Gargallo, 2007).
  - (c) Compute the number of inliers consistent with  $F_{cp}$  by the number of correspondences for which  $d < t$  pixels,  $t$  being a defined threshold.

Choose the  $F_{cp}$  with the largest number of inliers.

3. **Non-linear re-estimation.** Re-estimate  $F_{cp}$  from all correspondences classified as inliers by minimizing the distance in both images to epipolar conics (Sturm and Gargallo, 2007) and epipolar lines, using a non-linear optimization process.

For the next experiments we have selected the F34 model since its performance is similar to the other models and the number of correspondences required to be computed is the smallest. In a RANSAC approach the number of parameters to estimate is important since it determines the number of iterations required. In practice, there is an agreement between the computational cost of the search in the space of solutions, and the probability of failure  $(1 - p)$ . A random selection of  $r$  samples of  $k$  matches ends up with a good solution if all the matches are correct in at least one of the subsets. Assuming a ratio  $\varepsilon$  of outliers, the number of samples to explore is  $r = \frac{\log(1-p)}{\log(1-(1-\varepsilon)^k)}$ . For example using a probability  $p = 99\%$  of not failing in the random search and 30% of outliers ( $\varepsilon$ ), 231 iterations are needed to get a result using the F34. If we use the F36, 1978 iterations are needed for the same level of confidence. In the case of the F66 the number of iterations increases to  $1.2 \times 10^6$  and becomes prohibitive for some applications.

Several omnidirectional and perspective image-pairs are used to perform the experiment of automatic matching (Fig. 5.11). We avoid the rank 2 constraint



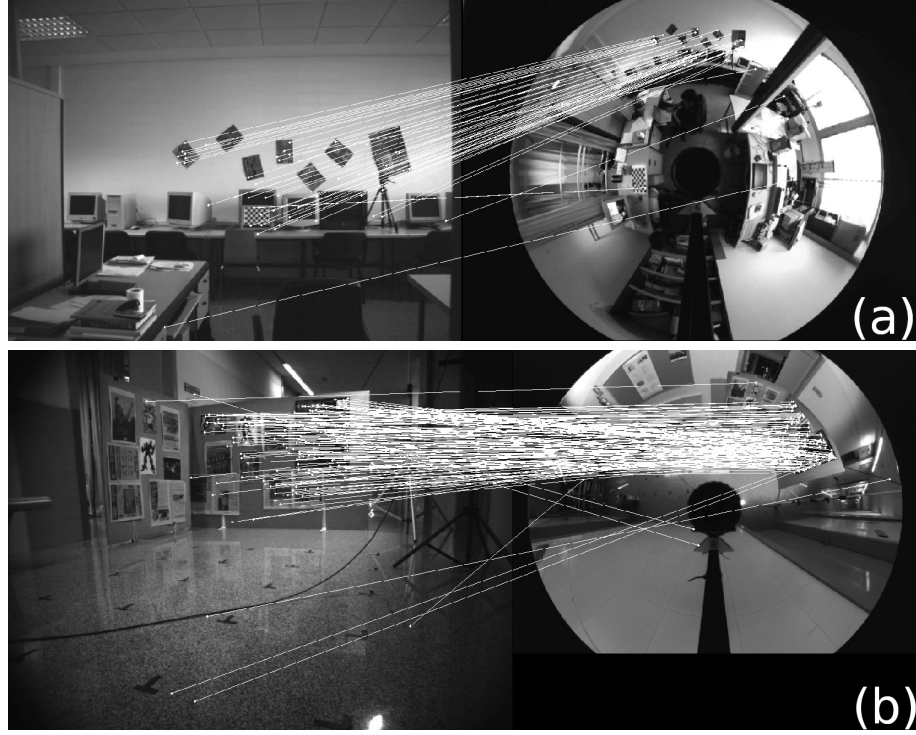


Figure 5.12: Matching between omnidirectional and perspective image using the hybrid epipolar geometry. (a) Experiment 1. (b) Experiment 4.

since we are just concerned about the matching problem. Table 5.4 summarizes the results giving the number of SIFT features extracted in the two valid versions of omnidirectional images tested (unwarped and flipped), and in the perspective one. It also shows the quantity of inliers and outliers in the initial (SIFT matching) and the robust matching (hybrid fundamental matrix), using both the unwarped and the flipped transformations. Fig. 5.12 shows two examples of the matching between omnidirectional and perspective images. In Experiment 1 we use images Fig. 5.11(a) and Fig. 5.11(e). The number of SIFT points extracted from the flipped and the unwarped images are similar. We observe that the initial matches are similar using the unwarped and flipped versions of the omnidirectional image, with a small advantage for the flipped one. These results confirm that SIFT is not projective invariant but it works well with such a distortion of catadioptric cameras. Despite the use of either of the transformed omnidirectional images the automatic matching using the hybrid epipolar geometry is able to filter most of the outliers. In Experiment 4 the increment in the SIFT features of the perspective image is caused by the resolution of the image ( $1280 \times 960$ ). The results show that the hybrid epipolar constraint eliminates most of the outliers.

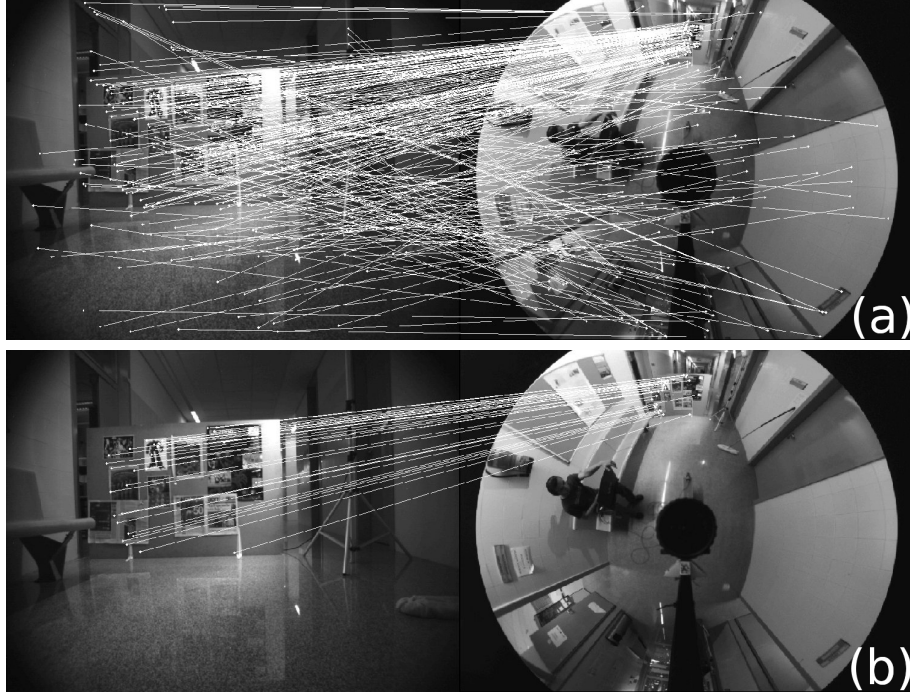


Figure 5.13: Matching between omnidirectional and perspective image using, (a) putative matches, (b) matches after the robust estimation using the hybrid homography.

	Unwarped Omni SIFT	Flipped Omni SIFT	Persp SIFT	Initial Matches (inliers/outliers)		Robust Epipolar Geometry matches(inliers/outliers)	
				Unwarped	Flipped	Unwarped	Flipped
Experiment 1	3251	2867	1735	68/111	76/131	40/8	57/4
Experiment 2	4168	4172	1528	18/68	21/71	16/7	17/7
Experiment 3	3280	2967	1682	41/101	33/112	27/9	20/9
Experiment 4	2275	2208	15658	125/322	164/360	80/5	129/22

Table 5.4: Numerical results of the hybrid matching using the set of images.

### Hybrid Homography

Analogous to the matching process using the hybrid fundamental matrix we can use the hybrid homography when most of the scene points lie in a plane. An example of the automatic matching using the hybrid homography as a geometrical constraint can be observed in Fig. 5.13. Fig. 5.13(a) shows the putative correspondences given by the SIFT matching and Fig. 5.13(b) after applying the robust matching process, computing a  $3 \times 4$  homography.

## 5.6 Closure

In this work we have presented a deep analysis of the two-view geometry combining a central catadioptric system and a conventional camera. In particular we studied

the hybrid epipolar geometry and the hybrid planar homography. We use lifted coordinates to generalize the two-view constraints, well known for perspective image pairs. We selected three approaches to compute the hybrid fundamental matrix  $F_{34}$ ,  $F_{36}$  and  $F_{66}$  and three approaches to compute the hybrid homography  $H_{34}$ ,  $H_{36}$  and  $H_{66}$ . We performed several experiments comparing the different approaches from the more complex and complete ( $F_{66}$ ,  $H_{66}$ ) to a more particular and simplified one ( $F_{34}$ ,  $H_{34}$ ), that in principle only can deal with a certain type of central catadioptric systems. From the simulation and real data experiments these simplified models obtained better results in presence of noise. We observed that the complete models can deal with any catadioptric system under ideal conditions but these approaches are more sensitive to the presence of noise. We successfully introduce the geometrical constraints in a robust matching process with initial putative matches given by SIFT points computed in the perspective image and the flipped version of the catadioptric one.

# Chapter 6

## Scale Space for Central Projection Systems. Towards a Generic Camera Feature Extractor

*In this chapter we present a new approach to compute the scale space of any omnidirectional image acquired with a central projection system. When these cameras are central they are explained using the sphere camera model, which unifies in a single model, conventional, paracatadioptric and hypercatadioptric systems. Scale space is essential in the detection and matching of interest points, in particular scale invariant points based on Laplacian of Gaussians, like the well known SIFT. We combine the sphere camera model and the partial differential equations framework on manifolds, to compute the Laplace-Beltrami (LB) operator which is a second order differential operator required to perform the Gaussian smoothing on catadioptric images. We perform experiments with synthetic and real images to validate the generalization of our approach to any central projection system.*

### 6.1 Introduction

Image processing has developed through the years techniques for conventional (perspective) cameras. Among all these techniques, feature detection/extraction is one of the most relevant, since it represents a crucial step on higher level techniques, such as matching, recognition, structure from motion, SLAM, navigation, visual localization, visual control, surveillance and many more. A particular useful property for features is to be scale-invariant. There are different approaches to detect scale-invariant features. Some of them are based on the scale-space analysis Lowe (2004); Mikolajczyk and Schmid (2004). Some others are based on the grey scale intensity Kadir and Brady (2001); Matas et al. (2002). SIFT Lowe (2004) has become the most used feature extraction approach. It has also been used directly in omnidirectional images Guerrero et al. (2008), although it is not designed to work on them. This SIFT approach has inspired different works trying to replicate its good results on different imagery systems, in particular on wide-angle cameras. In Bulow (2004) a Gaussian kernel is derived. It requires the omnidirectional image to be mapped to the sphere. Then, the spherical Fourier transform is computed and convolved with the spherical Gaussian function. In Hansen et al. (2007) the image is mapped to the sphere and obtain scale-space images as the solution to the heat diffusion equation on

the sphere which is implemented in the frequency domain using spherical harmonics. This approach introduces new drawbacks as aliasing and bandwidth selection. A complete SIFT version computed on the sphere also using the heat diffusion equation is presented by Cruz et al. (2009). In Hansen et al. (2010) an approximation to spherical diffusion using stereographic projection is proposed. It maps the omnidirectional image to the stereographic image plane through the sphere. It also maps the spherical Gaussian function to an equivalent kernel on the stereographic image plane. Then, the approximate spherical diffusion is defined as the convolution of the stereographic image with the stereographic version of the Gaussian kernel. More recently Lourenço et al. (2010) propose an improvement to the SIFT detector by introducing radial distortion into the scale-space computation. In Bogdanova et al. (2007) a framework to perform scale space analysis for omnidirectional images using partial differential equations is proposed. It leads to the implementation of the linear heat flow on manifolds through the Laplace-Beltrami operator. Omnidirectional images are treated as scalar fields on parametric manifolds. Based on this work Arican and Frossard (2010) propose a scale invariant feature detector for omnidirectional images. They deal with the paracatadioptric projection which is equivalent to the inverse of the stereographic projection. They model this projection on the sphere and obtain its corresponding metric. This metric is conformal equivalent to the Euclidean one making the computation of the Laplace-Beltrami operator straightforward. Although this approach could work with any catadioptric system, the metric describing the reflecting surface (mirror) has to be provided, which in some cases can be difficult to obtain.

In this chapter we show a new approach to compute the scale-space for any central catadioptric system. We integrate the sphere camera model (Geyer and Daniilidis, 2000; Barreto and Araujo, 2001), which describes any central catadioptric system, selecting it by one single parameter, with the partial differential equations on manifolds framework through the heat diffusion equation (Bogdanova et al., 2007; Arican and Frossard, 2010). Using this framework and the mirror parameter we compute the metric representing that particular reflecting surface. Then we use this metric to calculate the corresponding Laplace-Beltrami operator. This second order operator allow us to perform the Gaussian smoothing on omnidirectional images.

## 6.2 Scale Space for Central Projection Systems

In this section we integrate the sphere camera model (section 2.1) that models any central projection system and the techniques developed to compute the differential operators on non-Euclidean manifolds (Bertalmio et al., 2001) such as the mirror surfaces present in catadioptric systems.

### 6.2.1 Differential Operators on Riemannian Manifolds

The scale space representation  $I(x, y, t)$  is computed using the heat diffusion equation and differential operators on the non-Euclidean manifolds. It is defined as

## 6.2. Scale Space for Central Projection Systems

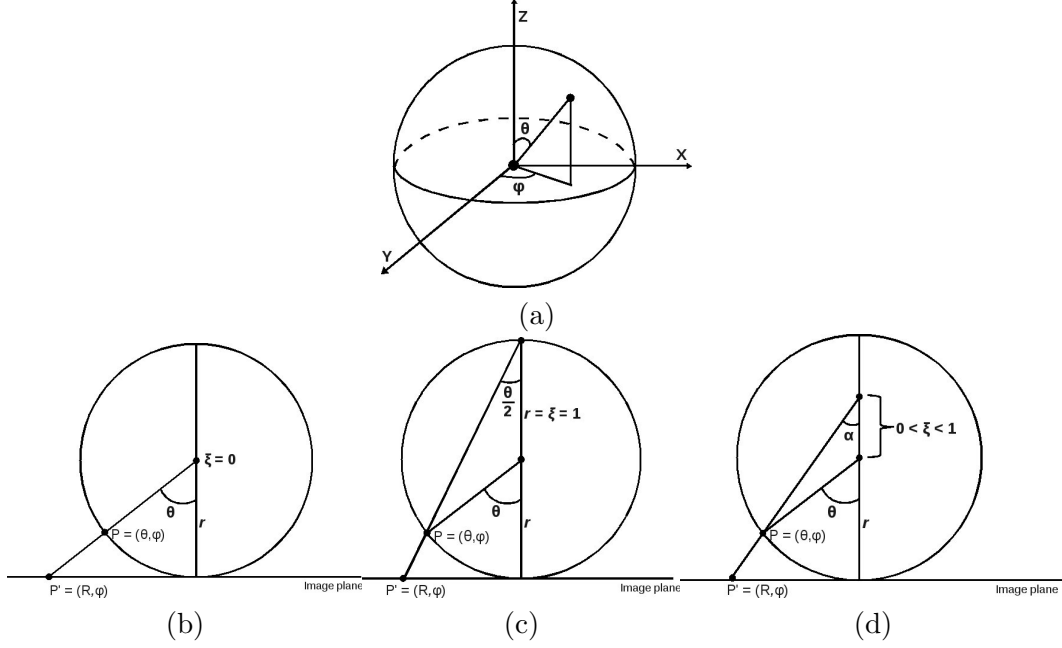


Figure 6.1: Simplified sphere camera model, showing the different radial projection functions. (a) spherical coordinates. (b) perspective projection. (c) paracatadioptric projection. (d) hypercatadioptric projection.

$$\frac{\partial I(x, y, t)}{\partial t} = \Delta I(x, y, t) \quad (6.1)$$

where  $\Delta$  is the Laplacian-Beltrami operator and  $t$  is the scale level. The initial condition is  $I(x, y, t_0) = I(x, y)$  where  $I(x, y)$  is the original image.

We briefly define the differential operators on the manifolds which make possible the computation of the Laplace-Beltrami operator. Let  $\mathcal{M}$  be a parametric surface on  $\mathbb{R}^3$  with an induced Riemannian metric  $g_{ij}$  that encodes the geometrical properties of the manifold.

In a local system of coordinates  $x^i$  on  $\mathcal{M}$ , the components of the gradient reads

$$\nabla^i = g^{ij} \frac{\partial}{\partial x^j} \quad (6.2)$$

where  $g^{ij}$  is the inverse of  $g_{ij}$ . A similar reasoning is used to obtain the expression of the divergence of a vector field  $\mathbf{X}$  on  $\mathcal{M}$

$$\text{div} \mathbf{X} = \frac{1}{\sqrt{g}} \partial_i (\mathbf{X}^i \sqrt{g}), \quad (6.3)$$

where  $g$  is the determinant of  $g^{ij}$ . Finally, combining these two operators we obtain the Laplace-Beltrami operator, which is the second order differential operator defined on scalar fields on  $\mathcal{M}$  by

$$\Delta I = \frac{1}{\sqrt{g}} \partial_j (\sqrt{g} g^{ij} \partial_i I) \quad (6.4)$$

### 6.2.2 Computing a Generic Metric on the Sphere

As explained in section 2.1 omnidirectional images are formed in two steps. The first one projects a 3D point to the unitary sphere. Then this point is projected from the unitary sphere to the image plane through a variable projection point, which is determined by the geometry of the mirror (parameter  $\xi$ ). If the system is calibrated (Puig et al., 2011), it is also possible like in any conventional camera, to map the catadioptric image to the unitary sphere.

In Bogdanova et al. (2007); Arican and Frossard (2010), the mapping from paracatadioptric images to the sphere is used for the computation of the differential operators explained before. This allows to process the spherical image directly using the image coordinates. Here we extend that approach to all central catadioptric systems.

Consider the unitary sphere  $\mathbb{S}^2$  (Fig. 6.1a). A point on  $\mathbb{S}^2$  can be represented in cartesian and polar coordinates as

$$(X, Y, Z) = (\sin \theta \cos \varphi, \sin \theta \sin \varphi, \cos \theta) \quad (6.5)$$

The Euclidean element in cartesian and polar coordinates is defined as

$$dl^2 = dX^2 + dY^2 + dZ^2 = d\theta^2 + \sin^2 \theta d\varphi^2 \quad (6.6)$$

Under the sphere camera model, a point on the sphere  $(\theta, \varphi)$ , is mapped to a point in polar coordinates  $(R, \varphi)$  in the image plane. The  $\theta$  angle depends on the central catadioptric system we are dealing with, while  $\varphi$  remain the same. For example, a conventional perspective system is described with  $\theta = \arctan(R)$  and a paracatadioptric system with  $\theta = 2 \tan(\frac{R}{2})$ . In the general case (see Fig. 6.1d) we have

$$\theta = \arctan \left( \frac{R(1+\xi + \sqrt{(1+\xi)^2 - R^2(\xi^2 - 1)})}{1+\xi - R^2\xi + \sqrt{(1+\xi)^2 - R^2(\xi^2 - 1)}} \right) \quad (6.7)$$

In terms of these new coordinates the metric becomes

$$dl^2 = \frac{(\xi + \xi^2 + \sqrt{(1+\xi)^2 - R^2(\xi^2 - 1)})^2 (R^2 d\varphi^2 + \frac{(1+\xi) dR^2}{1 - R^2(\xi - 1) + \xi})}{(R^2 + (1+\xi)^2)^2} \quad (6.8)$$

Let  $(x, y) \in \mathbb{R}^2$  on the sensor plane define cartesian coordinates, where  $R^2 = x^2 + y^2$  and  $\varphi = \arctan(\frac{y}{x})$

$$dl^2 = \lambda (4(ydx - xdy)^2 - \frac{4(1+\xi)(xdx + ydy)^2}{(x^2 + y^2)(\xi - 1) - \xi - 1}) \quad (6.9)$$

## 6.2. Scale Space for Central Projection Systems

---

where

$$\lambda = \frac{(\xi + \xi^2 + \sqrt{(1+\xi)^2 - (x^2+y^2)(\xi^2-1)})^2}{4(x^2+y^2)(x^2+y^2+(1+\xi)^2)^2}$$

To verify the correctness of this approach we substitute  $\xi$  to the values for which we already know the metric. For  $\xi = 1$  the result is the same as that one presented in (Arıcan and Frossard, 2010). In the case of  $\xi = 0$  that corresponds to the perspective case we expect a cartesian metric but what we get is the following

$$dl^2 = \frac{(1+y^2)dx^2 - 2xydxdy + (1+x^2)dy^2}{(1+x^2+y^2)^2} \quad (6.10)$$

This is explained since there is no map from the sphere to the plane being both conformal and area-preserving. However, from the generic metric we have already computed, we are able to compute the Laplace-Beltrami operator. To deal with the perspective case we only need to use the classical Laplacian.

### 6.2.3 Computing the Generic Metric on the Normalized Plane

The previous approach computes the metric on the sphere and then it is projected to the image plane through the angle  $\theta$  which encodes the radial projection function. We have observed that using the sphere camera model we are able to compute the metric directly on the image plane and at the same time take into account the geometry of the mirror, which is given by the mirror parameter  $\xi$ .

Let's observe the perspective projection in cartesian and spherical coordinates

$$\left(\frac{X}{Z}, \frac{Y}{Z}\right) = (\tan \theta \cos \varphi, \tan \theta \sin \varphi) = (x, y) \quad (6.11)$$

The radial component is  $R = \sqrt{x^2 + y^2} = \tan(\theta)$  and the Euclidean element in this case just takes into account the  $(x, y)$  coordinates but also the two spherical coordinates  $(\theta, \varphi)$

$$ds^2 = d(\tan \theta \cos \varphi)^2 + d(\tan \theta \sin \varphi)^2 = dx^2 + dy^2 \quad (6.12)$$

The same approach applied to the paracatadioptric case  $\xi = 1$  gives

$$\left(\frac{X}{1-Z}, \frac{Y}{1-Z}\right) = \left(\frac{\sin \theta \cos \varphi}{1 - \cos \theta}, \frac{\cos \theta \cos \varphi}{1 - \cos \theta}\right) = (x, y) \quad (6.13)$$

and its corresponding Euclidean element with  $\theta = 2 \arctan\left(\frac{\sqrt{x^2+y^2}}{2}\right)$  and  $\varphi = \arctan\left(\frac{y}{x}\right)$  becomes

$$dl^2 = \frac{16(dx^2 + dy^2)}{(x^2 + y^2 - 4)^2 \left(\sqrt{\frac{(x^2+y^2+4)^2}{(x^2+y^2-4)^2}} - 1\right)^2} \quad (6.14)$$

The generic metric equation for any catadioptric system is described by the following Euclidean element on the normalized plane



$$ds^2 = \frac{(\xi \cos \theta - 1)^2 d\theta^2 + (\xi - \cos \theta)^2 \sin^2 \theta d\varphi^2}{(\xi - \cos \theta)^4} \quad (6.15)$$

which allows to compute the Euclidean element on the normalized plane with the substitution of  $\theta$  and  $\varphi$  in terms of  $x$  and  $y$ .

### 6.2.4 Generic Laplace-Beltrami Operator

From (6.9) we can compute the generic metric in matrix form  $g_{ij}$  and its corresponding inverse matrix  $g^{ij}$

$$g^{ij} = \gamma_2 \begin{pmatrix} -x^2(\xi - 1) + \xi + 1 & xy(\xi - 1) \\ xy(\xi - 1) & -y^2(\xi - 1) + \xi + 1 \end{pmatrix} \quad (6.16)$$

with

$$\gamma_2 = \frac{(x^2 + y^2 + (1 + \xi)^2)^2}{(1 + \xi)(\xi + \xi^2 + \sqrt{1 - (x^2 + y^2)(\xi^2 - 1)} + 2\xi + \xi^2)^2}$$

The determinant of (6.16) is

$$\det(g^{ij}) = -\frac{(x^2 + (1 + \xi)^2 + y^2)^4 ((x^2 + y^2)(\xi - 1) - \xi - 1)}{(1 + \xi)(\xi + \xi^2 + \sqrt{1 - (x^2 + y^2)(\xi^2 - 1)} + 2\xi + \xi^2)^4} \quad (6.17)$$

With all these elements we are now able to compute the Laplace-Beltrami operator (6.4) which is represented by the differential operators

$$\Delta I = \lambda_1 \frac{\partial^2 I}{\partial x^2} + \lambda_2 \frac{\partial^2 I}{\partial y^2} + \lambda_3 \frac{\partial^2 I}{\partial xy} + \lambda_4 \frac{\partial I}{\partial x} + \lambda_5 \frac{\partial I}{\partial y} \quad (6.18)$$

$$\lambda_1 = \frac{(x^2(\xi^2 - 1) - 1)(1 + x^2 + y^2)^2}{-1 - \xi^2 + (\xi^2 - 1)(x^2 + y^2) - 2\xi\sqrt{1 - (\xi^2 - 1)(x^2 + y^2)}} \quad (6.19)$$

$$\lambda_2 = \frac{(y^2(\xi^2 - 1) - 1)(1 + x^2 + y^2)^2}{-1 - \xi^2 + (\xi^2 - 1)(x^2 + y^2) - 2\xi\sqrt{1 - (\xi^2 - 1)(x^2 + y^2)}} \quad (6.20)$$

$$\lambda_3 = \frac{2xy(\xi^2 - 1)(1 + x^2 + y^2)^2}{-1 - \xi^2 + (\xi^2 - 1)(x^2 + y^2) - 2\xi\sqrt{1 - (\xi^2 - 1)(x^2 + y^2)}} \quad (6.21)$$

$$\lambda_4 = \frac{4x(1 + x^2 + y^2)(2 - \xi^2 - 2(\xi^2 - 1)(x^2 + y^2) - \xi\sqrt{1 - (\xi^2 - 1)(x^2 + y^2)})}{-1 - \xi^2 + (\xi^2 - 1)(x^2 + y^2) - 2\xi\sqrt{1 - (\xi^2 - 1)(x^2 + y^2)}} \quad (6.22)$$

$$\lambda_5 = \frac{4y(1 + x^2 + y^2)(2 - \xi^2 - 2(\xi^2 - 1)(x^2 + y^2) - \xi\sqrt{1 - (\xi^2 - 1)(x^2 + y^2)})}{-1 - \xi^2 + (\xi^2 - 1)(x^2 + y^2) - 2\xi\sqrt{1 - (\xi^2 - 1)(x^2 + y^2)}} \quad (6.23)$$

An analogous process has been performed for the computation of the metric in the normalized plane, but the representation of the equations is not suitable to show in these pages. The code used to generate the equations can be downloaded from my website<sup>1</sup>. From now on we use the metric computed on the sphere.

<sup>1</sup>[http://webdiis.unizar.es/~lpuig/Mathematica\\_code/Mathematica\\_code\\_LB0.tar.gz](http://webdiis.unizar.es/~lpuig/Mathematica_code/Mathematica_code_LB0.tar.gz)

### 6.3. Experiments

#### Algorithm 1: Smoothing catadioptric images using the heat diffusion equation

**Input** :  $I(x, y), t, d_t, \xi, I_x, I_y, I_{xy}, I_{xx}, I_{yy}$   
**Output** :  $I(x, y, t)$

*Initialize required variables*  
 $[c_{Ix}, c_{Iy}, c_{Ixy}, c_{Ixx}, c_{Iyy}] \leftarrow \text{CDCoeff}(I(x, y), \xi)$   
 $ntimes \leftarrow t/d_t$   
 $I(x, y, t) \leftarrow I(x, y)$   
**for**  $i \leftarrow 1$  **to**  $ntimes$  **do**  
     $\text{ImIx} \leftarrow c_{Ix} * \text{Convolve}(I(x, y, t), I_x)$   
     $\text{ImIy} \leftarrow c_{Iy} * \text{Convolve}(I(x, y, t), I_y)$   
     $\text{ImIxy} \leftarrow c_{Ixy} * \text{Convolve}(I(x, y, t), I_{xy})$   
     $\text{ImIxx} \leftarrow c_{Ixx} * \text{Convolve}(I(x, y, t), I_{xx})$   
     $\text{ImIyy} \leftarrow c_{Iyy} * \text{Convolve}(I(x, y, t), I_{yy})$   
     $\text{LBO} \leftarrow \text{ImIx} + \text{ImIy} + \text{ImIxy} + \text{ImIxx} + \text{ImIyy}$   
     $I(x, y, t) \leftarrow I(x, y, t) + d_t * \text{LBO}$   
**end**

#### 6.2.5 Smoothing Catadioptric Images Using the Generic Laplace-Beltrami Operator

We compute the smoothing of the catadioptric images using the heat diffusion equation. This equation is computed at successive time steps,  $t_i = k^{2i} \sigma_o^2$  is defined in terms of the normalization and scale factors  $k$  and the base smoothing level  $\sigma_o$ .

The differentiation with respect to time in the heat diffusion equation is discretized with time intervals,  $d_t$ . To compute the discrete differential representations of the image  $\frac{\partial I}{\partial x}$ ,  $\frac{\partial I}{\partial y}$ ,  $\frac{\partial^2 I}{\partial xy}$ ,  $\frac{\partial^2 I}{\partial x^2}$  and  $\frac{\partial^2 I}{\partial y^2}$  we convolve the catadioptric image with different kernels

$$I_x = [-1 \ 1], I_y = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, I_{xx} = [1 \ -2 \ 1], I_{yy} = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}, I_{xy} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (6.24)$$

For a particular catadioptric system defined by  $\xi$  we compute the corresponding coefficients (6.19-6.23) which multiply its corresponding convolved image. Then we compute the Laplace-Beltrami operator. Finally, smoothing is performed by updating  $I(x, y, t)$  with the differences computed at previous time steps.

### 6.3 Experiments

In this section we perform several experiments using synthetic and real images to evaluate the scale space for different catadioptric systems. The synthetic catadioptric images are generated using the raytracing software POV-ray<sup>2</sup> and correspond to images from an office scene<sup>3</sup>. Two hypercatadioptric systems with mirror parameters of  $\xi = 0.9662^4$ (m1) and  $\xi = 0.7054^5$ (m2) are considered. We also consider one paracatadioptric system  $\xi = 1$  with radius  $r = 2.5\text{cm}$ . The real images are acquired using the hypercatadioptric system designed by Neovision (m1).

<sup>1</sup><http://www.povray.org>

<sup>2</sup><http://www.ignorancia.org>

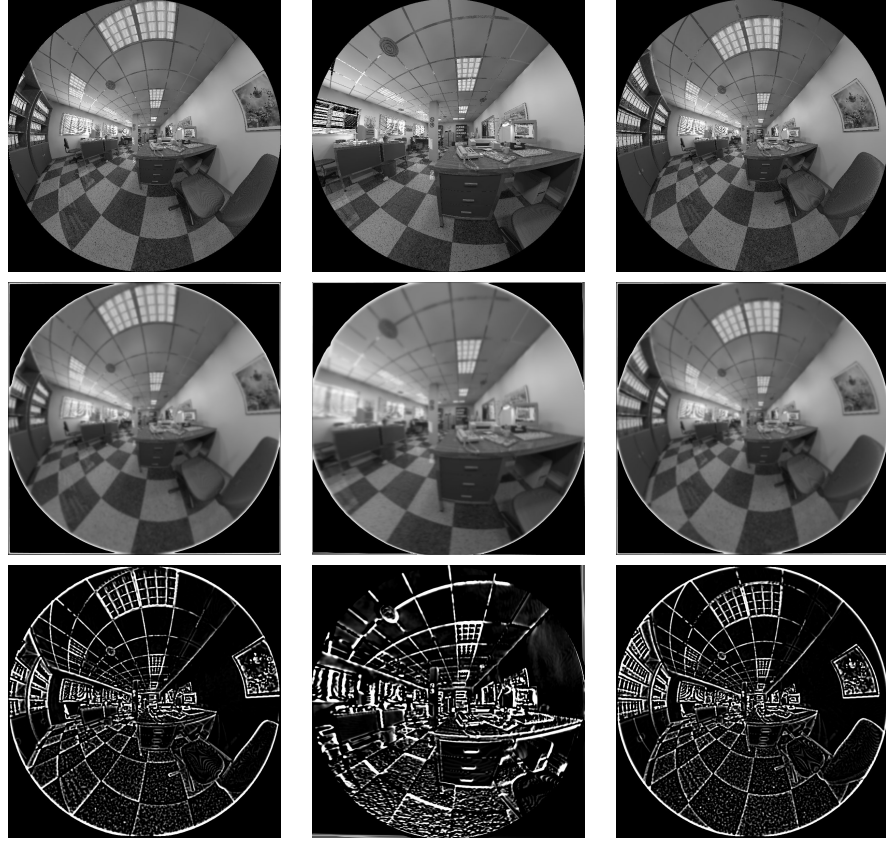


Figure 6.2: Smoothed catadioptric images with  $t = 3$ . The first column corresponds to mirror m1, the second one to mirror m2 and the third one to the paracatadioptric system. The first row represents the original images, the second represents the smoothed images and the last one the corresponding LB operators.

### 6.3.1 Smoothing of Synthetic Images

We follow the algorithm 1 with  $d_t = 0.01$ . All the coefficients  $\lambda_i$ ,  $i = 1 \dots 5$  are computed once since they only depend on the geometry of the image and not on the gray values. Fig. 6.2 shows smoothed images with different mirrors at the same scale factor  $t = 3$ . They also show the corresponding generic LB operator. We observe that the geometry of each mirror is taken into account when the smoothing is performed. The LB operator of m1 and the one corresponding to the paracatadioptric system are similar since the mirror parameter is close to 1. In the LB operators of mirrors m1 and m2, we observe how the intensities and thickness of the edges vary with respect to the distance to the image center. The differences between these LB operators are explained since the geometries of these mirrors are different.

<sup>3</sup><http://www.neovision.cz>

<sup>4</sup><http://www.accowle.com>

### 6.3. Experiments

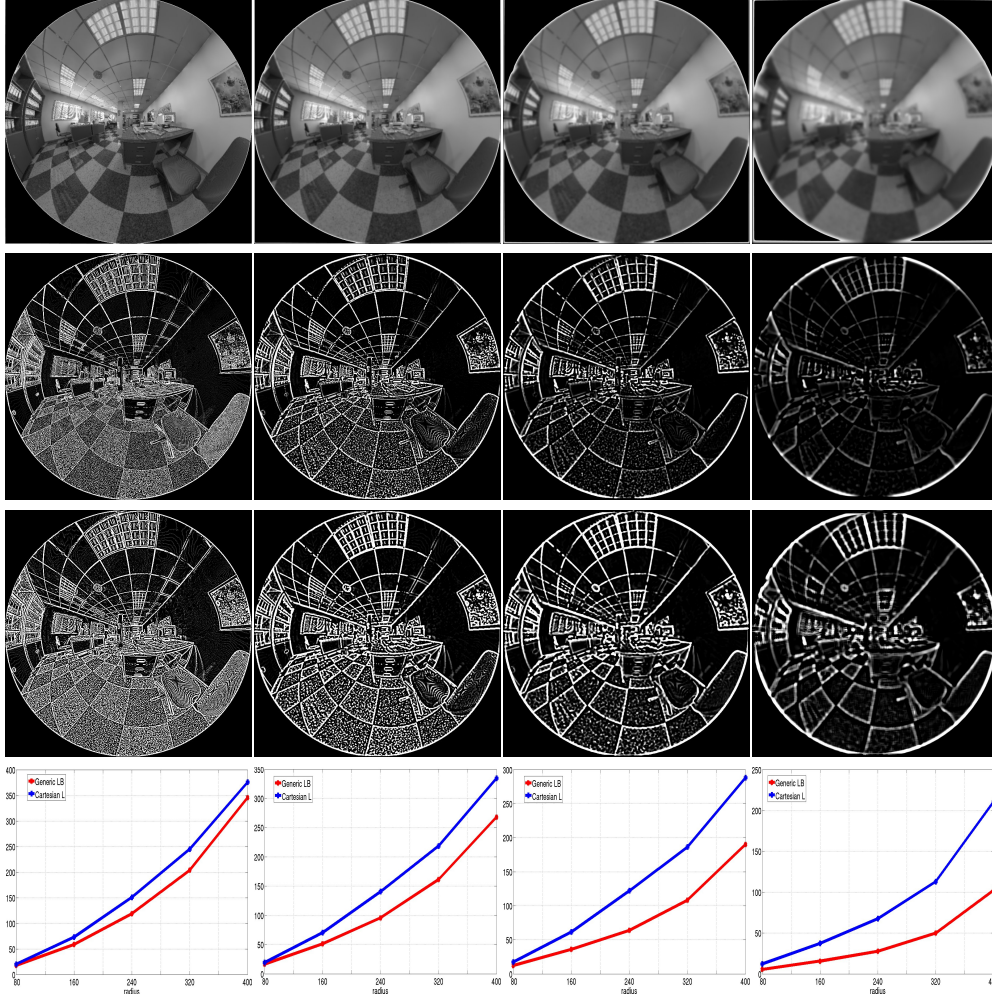


Figure 6.3: Comparison between generic LB operator and cartesian Laplacian. The first row show the smoothed images for scales  $t = \{1, 2, 3, 5\}$  using our approach. The second row shows the corresponding generic LB operators. The third row, the cartesian Laplacians. The last row presents the sum of the values (scaled for visualization purposes) of the generic LB and the cartesian Laplacian as a function of the radius of a circle with origin at the image center.

In order to verify the validity of our approach we perform an experiment where we compare the generic LB operator computed using our approach to the cartesian Laplace operator. To obtain this operator we smooth the omnidirectional image to the same scale  $t$ , using the corresponding Gaussian kernel. Then, we compute the Laplacian of this image using the cartesian operator  $I_{xy}$ . We select the hypercatadioptric (m1) image for which we compute scales  $t = \{1, 2, 3, 5\}$  and their corresponding generic LB and cartesian Laplacian. In Fig. 6.3 we show the results. We observe that the generic LB operator considers a difference between the pixels close to the center and those close to the periphery, while the cartesian Laplacian has the same effect

on all pixels, without taking into account their location with respect to the center.

To quantify the last statement we sum the values of the generic LB operator inside circles with different radii and origin at the image center. We perform the same procedure with the cartesian Laplacian. The last row of Fig. 6.3 shows the comparison of these sums for the the different scales. We observe that the pixels in the periphery have smaller values in the generic LB operator than those on the normal Laplace operator. In the center where the resolution of the catadioptric images is bigger the values of the Laplace-Beltrami operator and the normal Laplace are similar.

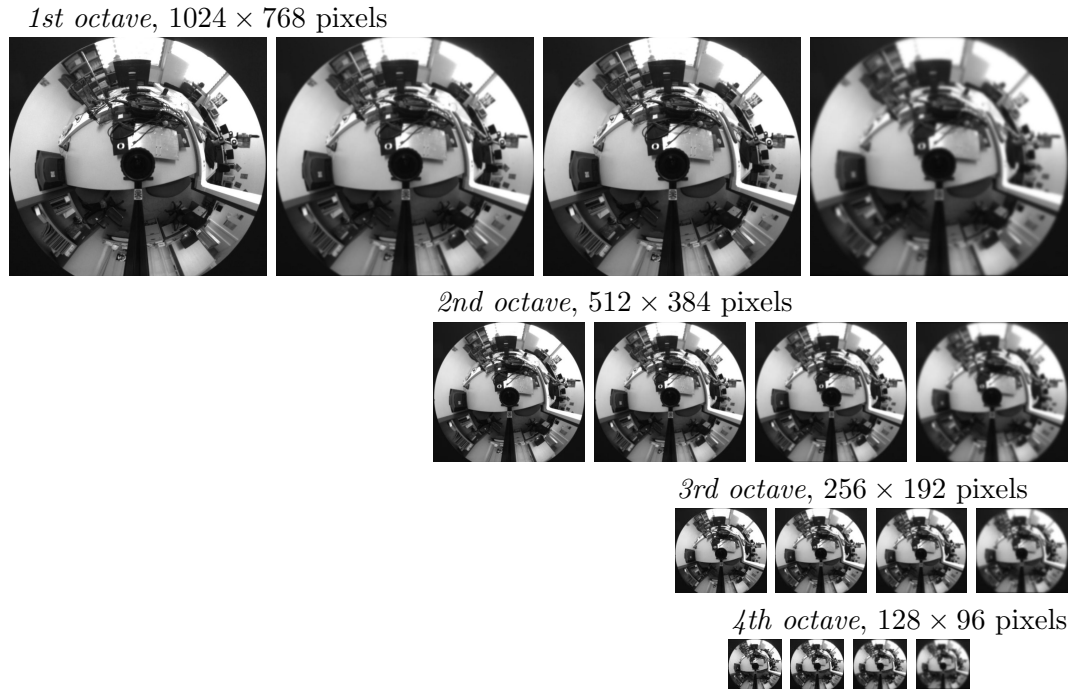


Figure 6.4: Scale Space of a real omnidirectional image, composed of four octaves and four scales per octave.

### 6.3.2 Scale Space on Real Catadioptric Images

In Fig. 6.4 we show the pyramid that compose the scale space of a hypercatadioptric image computed using our approach. We define the scale space to be composed of four octaves and four scales per octave. The initial sigma is defined as  $t = 0.8$  and the rest are computed as explained in Section 6.2.5. The value of the smoothing interval is defined as  $k = 2^{1/3}$ .

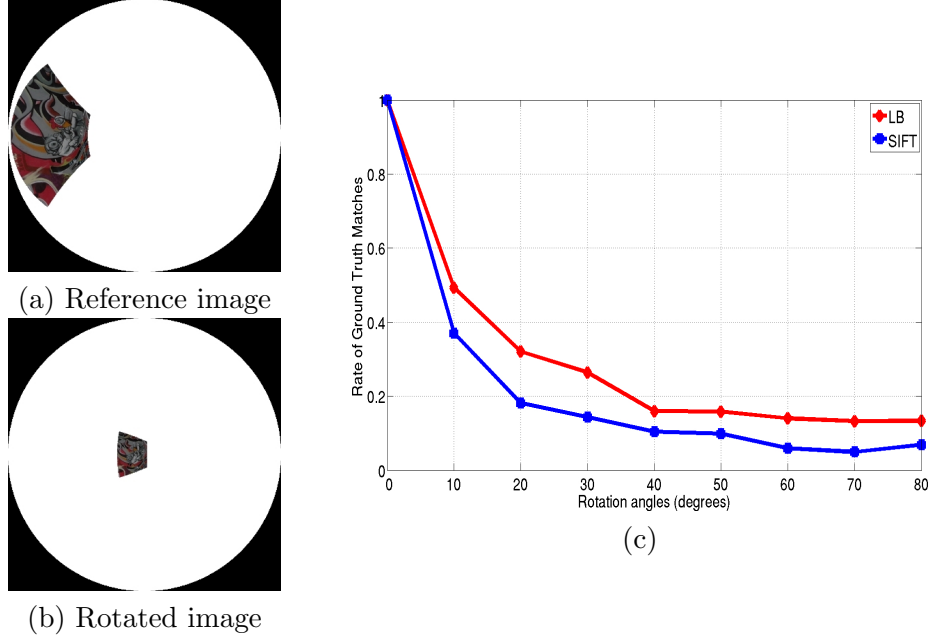


Figure 6.5: Repeatability experiment using synthetic hypercatadioptric images. (a) Reference image. (b) Image generated with the catadioptric system rotated 80° around the  $z$ -axis. (c) Percentage of correct matches through rotations of the catadioptric system around the  $z$ -axis. The  $x$ -axis represents the orientation of the catadioptric system in degrees with respect to the initial position.

#### 6.3.3 Repeatability Experiment

In this experiment we test the repeatability of the extrema points detected with the scale space computed using our approach. The extrema points are obtained from the approximation to the scale-normalized Laplacian of Gaussian through differences of Gaussians, similar to Lowe (2004). We generate nine synthetic images with rotations from zero degrees to eighty degrees around the  $z$ -axis, with steps of ten degrees between each pair of images. The two extreme images are shown in Fig. 6.5. We observe a drastic distortion on the images, this is produced by the conic shape of the mirror.

Since the rotation of the catadioptric system is known, we can map the detected features in the reference image  $\bar{\mathbf{x}}$  to the subsequent images and compute the distance between them. The matching criteria is the following

$$\|\hat{\mathbf{x}} - \bar{\mathbf{x}}\| \leq \delta_d \quad (6.25)$$

where,  $\hat{\mathbf{x}}$  is the mapped point. We use the Euclidean distance. The distance threshold  $\delta_d$  has to be adapted to capture the matches with different scale levels, which is  $\delta_d = \delta_0 \cdot t_{\bar{\mathbf{x}}}$  with  $\delta_0$  the distance threshold parameter.

We compare our approach to SIFT algorithm. In particular we use Vedaldi's

implementation Vedaldi (2007). In Fig. 6.5(c) we show the results of this experiment. We observe that the LB approach has a clear advantage over the classical SIFT approach, obtaining double repeatability than the scale space used by SIFT in extreme camera rotations. The low rate shows the difficulties of matching catadioptric images.

## 6.4 Closure

We have presented a new way to compute the scale-space of omnidirectional images. We integrate the sphere camera model which considers all central catadioptric systems with the partial differential equation framework on manifolds to compute a generic version of the second order differential operator Laplace-Beltrami. This operator is used to perform the Gaussian smoothing on catadioptric images. We perform experiments using synthetic images generated with parameters coming from actual manufactured mirrors. We observe that Laplace-Beltrami operator considers correctly the geometry of the mirror, since the pixels at the periphery have a different weight than those at the center. This situation explains the natural non-homogeneous resolution inherent to the central catadioptric systems. The near future work is to implement a complete scale-invariant feature detector also invariant to camera since the sphere camera model allows to consider all central projections.

# Applications of Omnidirectional Systems in 3D Localization





### Outline

In next three chapters we present three different applications that use omnidirectional systems to perform tasks of 3D localization and orientation for robotics applications. **Chapter 7** and **Chapter 8** use a central catadioptric system, in particular a hypercatadioptric system, while **Chapter 9** uses a different type of omnidirectional camera, a polycamera, more specific the LadyBug camera, which is composed of 6 wide angle aligned cameras.

**Chapter 7** introduces a new application to localize a robot in an indoors environment. It combines the use of a visual memory of omniviews of the environment stored in the robot memory with current images coming from a conventional camera mounted on the robot. With the aid of the hybrid two-view geometry between conventional cameras and catadioptric systems, the location of the on-board camera is computed through the location of the epipole in the memory of omniviews. Then with the use of the hybrid homography, that position in pixels is mapped to a position on the ground plane in meters.

In **Chapter 8** a new technique to compute the orientation of a hand-held catadioptric system is proposed. It makes use of geometric information embedded in man-made environments. In particular the projections of straight lines into images are useful to compute the vanishing points from projections of parallel lines. Vanishing points contain the orientation information of the camera with respect to the scene. In hypercatadioptric systems the first problem we face is the extraction of such lines from the images. They are no longer lines but conics. Then the next problem is to compute the intersection of such conics, from which the vanishing points will emerge. In this chapter we show the way we came up to solve this problem and to compute the orientation of the hypercatadioptric system using the vertical and horizontal vanishing points.

Finally in **Chapter 9** we present a new way to generate a topological map using only orientation information. We use the orientation information to identify turns, assuming continuous motion from an image sequence. The detected turns are confirmed by a traversality test. This traversality test is based on the computation of the fundamental matrix from correspondences between two images. This test is also used to identify possible loops on the image sequence and to identify the keyframes of our topological map.



## Chapter 7

# Self-location from Monocular Uncalibrated Vision Using Reference Omniviews

*In this chapter we present a novel approach to perform indoor self-localization using reference omnidirectional images. We only need omnidirectional images of the whole scene stored in the robot memory and a conventional uncalibrated on-board camera. We match the omnidirectional image and the conventional image captured by the on-board camera and compute the hybrid epipolar geometry using lifted coordinates and robust techniques, where the epipole gives the robot location in the omnidirectional image. We map the epipole in the reference omnidirectional image to a ground plane through a homography in lifted coordinates also, giving the position of the robot in the planar ground, and its uncertainty. We perform experiments with simulated and real data to show the feasibility of this new self-localization approach.*

### 7.1 Introduction

In recent years the use of catadioptric omnidirectional systems in mobile robot localization and navigation tasks and in visual surveillance applications has increased considerably. The main advantages of such systems is their wide field of view and the central single view point property. These characteristics allow overcoming the visibility constraint and help the geometrical analysis of the information captured by catadioptric cameras. Ishiguro and Tsuji (1996) describe a method for robot localization from memorized omnidirectional views, which are stored using Fourier coefficients; similarly, Pajdla and Hlavác (1999) use the image phase of a panoramic view for robot localization. Cauchois et al. (2003) present a method for robot localization by correlating real and synthesized omnidirectional images, but they can only handle small viewpoint changes. Matsumoto et al. (1999) present a similar method based on simply comparing cylindrical gray-level images. In Goedemé et al. (2007) they propose their own representation which is called *fast wide baseline feature matching* and the use of a topological map instead of a metric one. In Briggs et al. (2006) a new method for navigation and localization using a collection of 1D

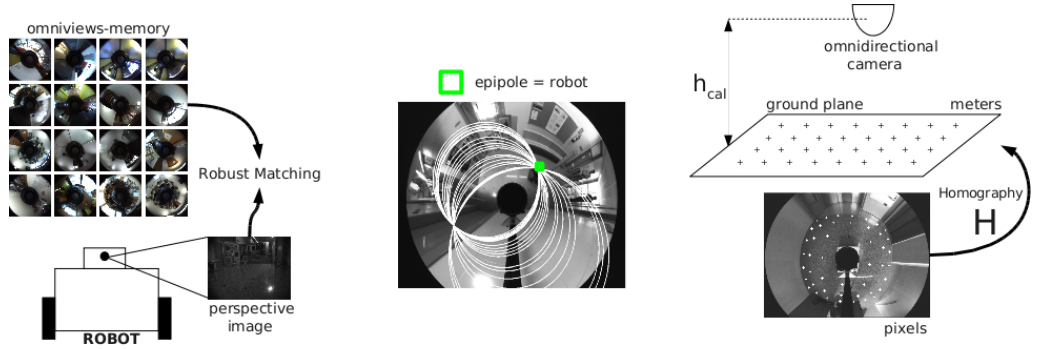


Figure 7.1: Main steps of the self-localization approach. Left: matching between a perspective and an omnidirectional image. Center: computing the epipole gives the location of the robot in the scene. Right: with a homography the position in pixels is mapped to a position in meters in the ground plane.

omnidirectional images formed by averaging the center scanlines of a cylindrical view is proposed. More recently some authors assume they have a sequence of omnidirectional images stored in memory, Courbon et al. (2008); Murillo et al. (2007a) from which they find the closest to the current frame. In Courbon et al. (2008) a fisheye lens is used. The essential matrix is computed from the current frame and the next in the visual memory, which is used to feed a control law to guide the robot to the next position. In Murillo et al. (2007a) a catadioptric system is used to perform a hierarchical localization. They go from topological localization using features matching to metric localization using the trifocal tensor.

As we have seen many authors try to deal with the problem of navigation and robot localization using omnidirectional images acquired by dioptric (fisheye) or catadioptric system mounted on the robot. Many times this type of sensors are expensive and not easily available as perspective cameras. Some other times the omnidirectional sensors are in a fixed position (surveillance) or they are used to explore a certain environment as in the Street View System<sup>1</sup>. In these cases we could use a different type of sensor to acquire the current image and to perform a matching between them.

## 7.2 Our Proposal

In this work we propose a new approach to localize a robot in an indoors environment. We depicted the main steps of our approach in Fig. 7.1. A mobile robot with a perspective camera on-board has a visual memory of omniviews stored in its memory. When it acquires a new perspective image the following steps take place:

1. **Matching.** The perspective image is matched with one of the omnidirectional images contained in the visual memory. We use a robust matching algorithm with a geometrical constraint, in this case the hybrid epipolar geometry.

<sup>1</sup><http://maps.google.com>

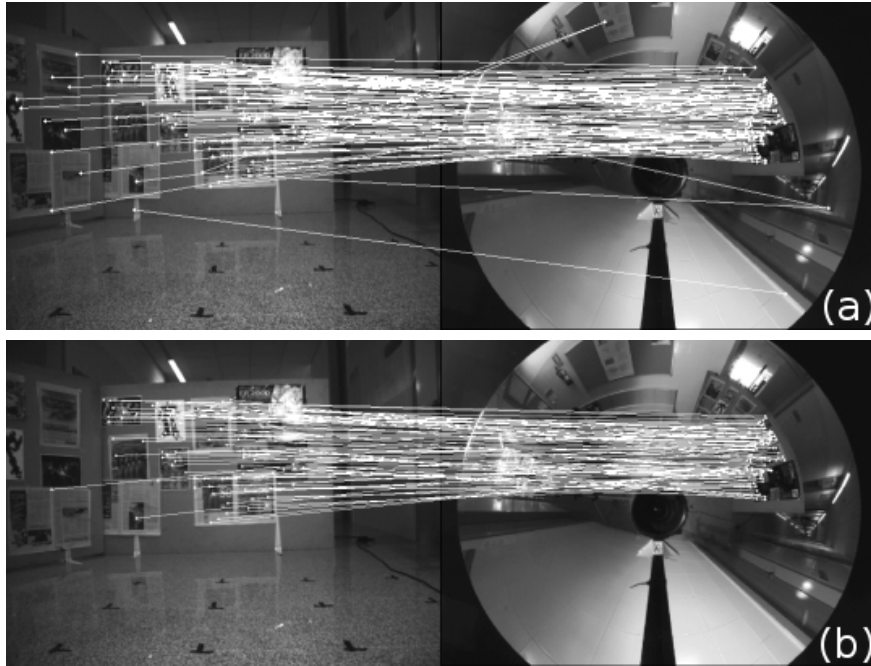


Figure 7.2: (a) Putative matching using just SIFT. (b) Matching using the hybrid fundamental matrix.

2. **Projective location of the robot camera.** We extract the epipole from the hybrid fundamental matrix, which corresponds to the current position of the on-board perspective camera in the catadioptric view.
3. **Mapping the position in the image to the ground plane.** Using the planar homography  $H_{34}$ , previously computed, we map the position of the on-board camera, represented by the epipole, to the actual ground plane. At this step we have the relative position of the on-board camera in meters with respect to the origin of the ground plane.

We also studied the uncertainty of the location of a point in the ground plane adapting the methodology developed for conventional cameras (Criminisi et al., 1999), to omnidirectional ones.

### 7.3 Robot location in the omnidirectional image

As it is well known, the epipoles are the location of the camera center in one view seeing by the other view. If we are able to extract the epipole in the omnidirectional image, we can localize the robot camera in a wide field of view. When the epipolar geometry is encapsulated by the fundamental matrix, the computation of epipoles is reduced to extract the null vectors of such matrix. This problem is more complex when we use omnidirectional images, the null vectors become null spaces. We have to

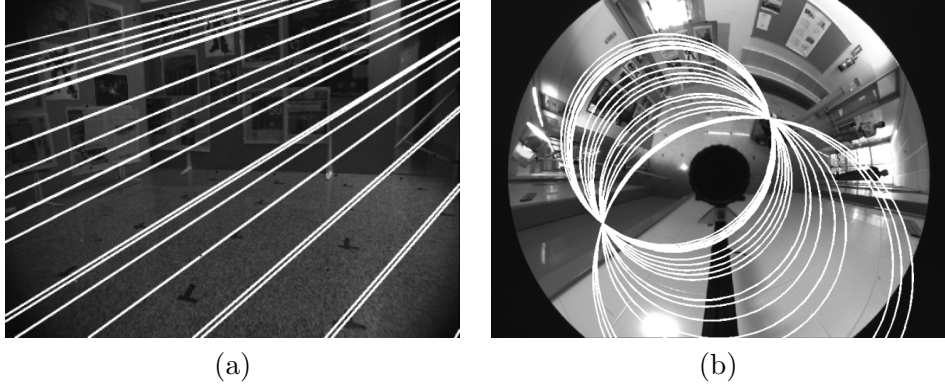


Figure 7.3: Example of epipolar lines (a) and epipolar conics (b) computed from the hybrid fundamental matrix  $F_{34}$ .

compute the null-vectors from the null space that are valid lifted coordinates. From these, the epipoles are extracted in a closed form. These process is explained in more detail in section 5.3.1.

Before computing the epipoles we have to compute the fundamental matrix between an omnidirectional image and a perspective one. In this work we use the hybrid  $F_{34}$  fundamental matrix. We choose this matrix because it requires less correspondences and its performance is good enough with the mirror shape we use in this work. The hybrid fundamental matrix is:

$$\mathbf{q}_p^T F_{34} \hat{\mathbf{q}}_c = 0 \quad (7.1)$$

where  $\hat{\mathbf{q}}_c$  is the 4-vector lifted coordinates (2.12) of a point in the catadioptric image and  $\mathbf{q}_p$  represents a point in the perspective image in homogeneous coordinates.

The automatic computing of the hybrid fundamental matrix starts with a putative matching between the SIFT points from the flipped omnidirectional images stored in the robot's memory with the SIFT points extracted from the current image acquired by the perspective on-board camera. This step can be observed in Fig. 7.2(a). Then using a RANSAC approach combined with the computation of the hybrid homography the outliers are avoided (see Fig. 7.2(b)). Finally, using only inliers a updated version of the hybrid fundamental matrix is computed. In Fig. 7.3 we can observe examples of epipolar lines and conics computed using the hybrid fundamental matrix. A more detailed explanation of the robust estimation of the hybrid fundamental matrix is given in section 5.5.

## 7.4 From Omniview to Ground Location

In this section we explain the process to map a point in the omnidirectional image to a position in the ground plane. We assume a planar robot motion. To perform this task we calibrate the omnidirectional image with respect to the ground plane using a homography with lifted coordinates. In this work we use the homography

#### 7.4. From Omniview to Ground Location

---

H34 previously studied in chapter 5. This homography has the advantages of being less sensitive to noise and to have high accuracy. The homography H34 maps a 4-vector lifted coordinates of a point in the omnidirectional image  $\hat{\mathbf{q}}_c \sim (q_1^2 + q_2^2, q_1 q_3, q_2 q_3, q_3^2)^\top$  to a point in the ground plane  $\mathbf{q}_{gp}$  in homogeneous coordinates

$$\mathbf{q}_{gp} = \mathbf{H34} \hat{\mathbf{q}}_c. \quad (7.2)$$

To compute this homography we require at least 6 correspondences  $\mathbf{q}_{gp}^i \leftrightarrow \hat{\mathbf{q}}_c^i$  between points in the ground plane  $\mathbf{q}_{gp}^i$  and lifted coordinates of omnidirectional image points  $\hat{\mathbf{q}}_c^i$ , since each correspondence gives two equations and H34 has 11 degrees of freedom.

##### 7.4.1 Uncertainty

Since the homography transformation produces a non-homogeneous uncertainty distribution we studied the error propagation from omnidirectional images to the ground plane. We analyze in particular the hybrid homography H34. We adapt the approach proposed by Criminisi et al. (1999) to omnidirectional images using lifted coordinates. We assume uncertain image points  $\mathbf{q}$  with  $\sigma_{q_1} = \sigma_{q_2} = \sigma$  the covariance in the image coordinates, represented in matrix form as

$$\Sigma = \begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{pmatrix}, \quad (7.3)$$

and we consider an exact H34. Since we are dealing with lifted coordinates in the image plane we use the Jacobian

$$\mathbf{J} = \begin{pmatrix} 2q_1 & 2q_2 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad (7.4)$$

of this lifting to translate the error into the lifted coordinates (7.5) then we propagate the error from the points in the image to the points in the plane (7.6).

$$\Lambda_{\hat{\mathbf{q}}} = \mathbf{J} \Sigma \mathbf{J}^\top \quad (7.5)$$

$$\Lambda_{\mathbf{Q}} = \mathbf{H34} \Lambda_{\hat{\mathbf{q}}} \mathbf{H34}^\top \quad (7.6)$$

As result we have a covariance matrix for homogeneous coordinates of a point in the plane  $\mathbf{Q} = (Q_1, Q_2, Q_3)$  which have to be converted into a  $2 \times 2$  covariance matrix  $\Lambda_{\mathbf{Q}}^{2 \times 2}$ :

$$\Lambda_{\mathbf{Q}}^{2 \times 2} = \nabla f \Lambda_{\mathbf{Q}} \nabla f^\top, \quad \text{where} \quad \nabla f = 1/Q_3^2 \begin{pmatrix} Q_3 & 0 & -Q_1 \\ 0 & Q_3 & -Q_2 \end{pmatrix} \quad (7.7)$$



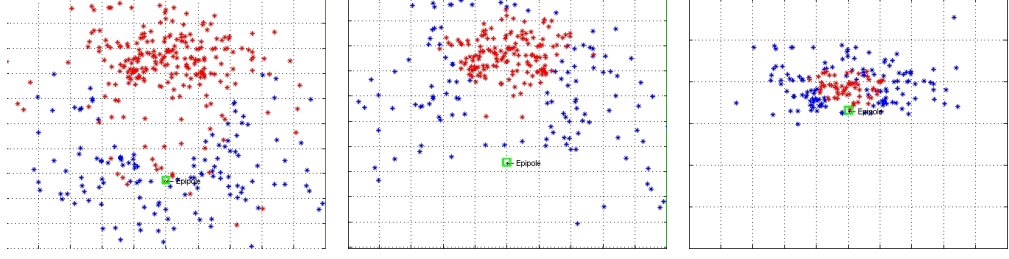


Figure 7.4: Three different CFOVs, represented by the common observable points (red points) depending on the position of the perspective camera (green square).

CFOV in radians	1.83	1.13	1.04	0.97	0.75	0.54	0.26
Baseline(m)	2.91	2.50	2.12	1.80	1.58	1.50	1.58
Number of Matches	246	219	200	176	149	119	90
Error in pixels	0.57	0.84	1.08	1.25	2.86	4.96	25.83

Table 7.1: Error in the epipole estimation depending on the CFOV, defined by the baseline and the number of matches.

## 7.5 Experiments

In order to verify the behavior of our proposal we analyze the different steps that compose it. In first place, we analyze the influence of the common field of view (CFOV) on the accuracy of the localization of the epipole. Then some experiments with the error propagation through hybrid homography are presented. Finally, the influence of the height of the catadioptric system with respect to the ground plane on the localization accuracy is tested.

### 7.5.1 Epipole estimation

We analyze different positions for the perspective camera in the scene and compute the epipole in the omnidirectional image. We move the perspective camera along the  $y$ -axis at a height of 0.5m in the same  $x$  coordinate as the omnidirectional camera. Fig. 7.4 presents three omnidirectional images showing the matched points in the scene (red points) being in the Common Field of View (CFOV) of both cameras. We naturally observe that moving the perspective camera decreases the CFOV causing a reduction in the number of matches and decreasing the accuracy of the epipole computed. We add Gaussian noise with  $\sigma = 0.5$  to points in both images. Table 7.1 shows the estimation of the epipole location depending on the size of the CFOV. We observe that even when we have enough correspondences to compute the hybrid fundamental matrix the estimation of the epipole could be 25 pixels far from the real value. This is explained because these correspondences are localized in just a small part of the omnidirectional image.

## 7.5. Experiments

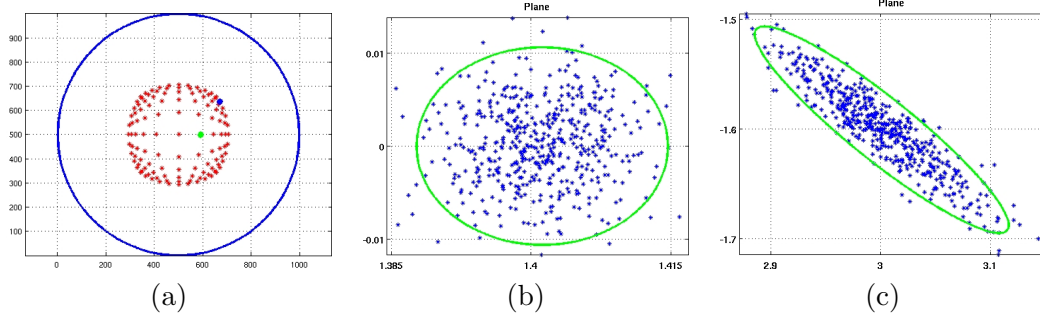


Figure 7.5: Propagation error in the plane using H34. (a) Omnidirectional image with Gaussian distributions in the central area (green) and in the periphery (blue). Theoretical ellipses of uncertainty corresponding to the points close to the center (b) and the points in the periphery (c).

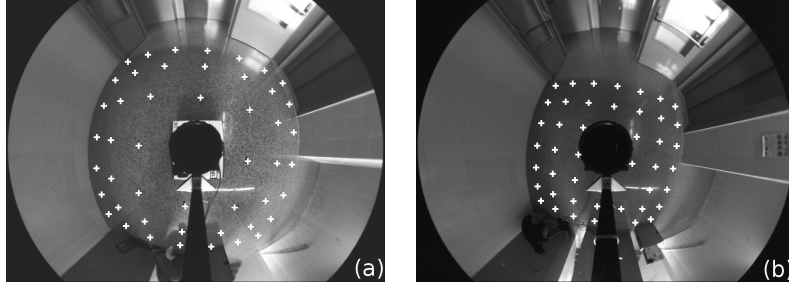


Figure 7.6: Omnidirectional images of the pattern at two different heights with the reprojection of the pattern points. (a) height = 50cm. (b) height = 100cm.

### 7.5.2 Uncertainty

Here we show the error propagation through H34. We select a point in the grid used to compute the homography. This point is used as the center for a Gaussian distribution with  $\sigma = 1$  pixel. In Fig 7.5 we observe the error propagation in the plane corresponding to two different points in the omnidirectional image, one close to the center Fig. 7.5(b) with an error of  $\pm 1.35\text{cm}$  and the other one close to the periphery Fig 7.5(c) with an error of  $\pm 14.66\text{cm}$ . We observe that the error varies depending on the position in omnidirectional image. This is because of the perspective effect, since there is high uncertainty approaching the line (conic) of the horizon of the ground plane, which corresponds with far away observations.

### 7.5.3 Homography using real images

Here we test the influence of different heights of the omnidirectional camera on the location accuracy. The plane used has  $7 \times 7$  points and the distance between them is 40cm giving a pattern of  $240 \times 240\text{cm}$ . The omnidirectional camera is located at the center of the scene. In Fig. 7.6 we can see two different omnidirectional images

	Real position	Estimated H50	Estimated H100
Point 1	(154,-114)	(155.92,-112.49)	(154.90,-110.92)
Point 2	(260,108)	(266.07,106.22)	(259.70,106.39)
Point 3	(0.0,-471)	(0.02,-469.36)	(0.01,-467.62)

Table 7.2: Estimations of position in the plane using the lifted homography

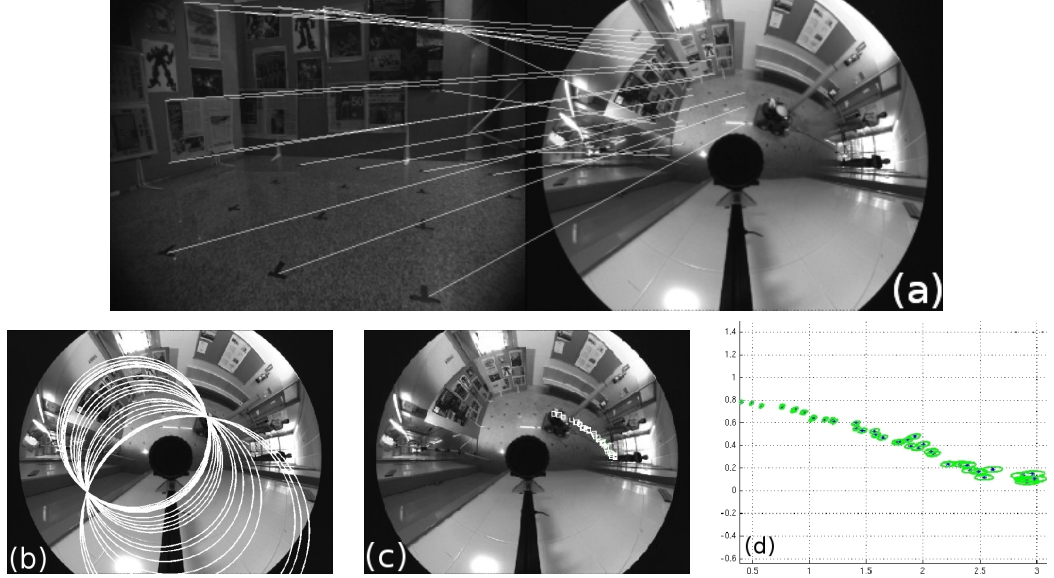


Figure 7.7: (a) Matching between omnidirectional and perspective images. (b) Epipolar conics corresponding to the matched points. (c) Epipole trajectory superimposed in one single omnidirectional image. (d) Uncertainties of the epipoles in the ground plane, units are in meters.

obtained from two different heights, 50cm (H50) and 100cm (H100), respectively. We also show the reprojection of the points used to compute the homography. To test the performance of the homographies we select three points that do not belong to the pattern. The goal is to check the accuracy for points that are far from the image center. Table 7.2 shows the location results of this experiment in centimeters. This inaccuracy is caused by the perspective effect and by the non-homogeneous resolution of the omnidirectional image, a few pixels in the periphery of the image means more distance than the same pixels close to the image center.

## 7.6 Self-localization of the robot

Here we show the behavior of the whole self-localization system (see Fig. 7.7). We have a single omnidirectional image of the scene stored in the robot memory which was acquired with a hypercatadioptric system<sup>2</sup> with a resolution of  $1024 \times 768$  pixels and we acquire perspective images from a conventional camera at a resolution of

<sup>2</sup><http://neovision.cz>

$1024 \times 768$  pixels at different positions. We match the perspective image with the omnidirectional one to compute the fundamental matrix (Fig. 7.7(a)). We observe that the features have to be distributed in a wide part of the omnidirectional image to have a good estimation of the epipole (Fig. 7.7(b)). The height of the omnidirectional camera has already been calibrated with respect to the on-board perspective camera. The robot follows a simple straight path, advancing for 3.5m. In Fig. 7.7(c) we show the estimated epipoles at different times superposed in a single image. Fig. 7.7(d) shows the uncertainty estimation of the position of the on-board camera. We observe that the estimation of the position is better when the camera is in the central area of the omnidirectional image. When the camera is in the periphery the location uncertainty increases as expected.

## 7.7 Closure

We presented a new approach to perform robot self-localization, mixing reference omnidirectional images and perspective images. We avoid the use of any catadioptric system by storing the omnidirectional images of the scene in the robot memory. The only sensor used is the on-board perspective camera installed on the robot. We propose a schema where the epipole, computed from the hybrid fundamental matrix, in the omnidirectional image is mapped to a ground plane by a hybrid homography previously computed. We observed that the accuracy of the epipole estimation depends on how well the correspondences are distributed in the perspective and omnidirectional images. We observed that the major drawbacks in practice are the hybrid matching and the computation of epipoles. In both cases the features must be well distributed in both images which is difficult when the CFOV is small. We also observed a high inhomogeneous uncertainty related to the position on the omnidirectional image, which is however well coded in our model.



## Chapter 8

# Orientation of a Hand-held Hypercatadioptric System in Man-made Environments

*In this chapter we present a new approach to extract conics from catadioptric images, in particular those which represent the projection of 3D lines in the scene. Using the internal calibration and two image points we are able to compute the catadioptric image lines analytically. The presence of parallel lines in man-made environments is exploited to compute the dominant vanishing points in the omnidirectional image. The vanishing points are extracted from the intersection of two conics in the catadioptric image, which represent parallel lines in the scene. This intersection is performed via the common self-polar triangle associated to this pair. From the information contained in the vanishing points we obtain the orientation of a hand-held hypercatadioptric system. This approach is tested by performing vertical and full rectifications in real sequences of images.*

### 8.1 Introduction

In robotics when a catadioptric system is used it is commonly observed that it has a vertical orientation. This is because most robotic platforms used are wheel-based. Under this configuration planar-motion and/or 1D image geometry is assumed which reduces the degrees of freedom (DOF) of the problem (Guerrero et al., 2008). Moreover, in applications where line tracking or line matching is performed this assumption is useful (Mezouar et al., 2004; Murillo et al., 2007b). Besides that there exist robot platforms where the vertical assumption is not satisfied and they require the development of new algorithms to interact with the environment. One of this algorithms can be a self-orientation system to be used for stabilization of a biologically-inspired humanoid robot platform (Miyauchi et al., 2007). One of the advantages of the non-vertical configuration is that both the horizontal and vertical vanishing points are present in the image and can be computed by the intersection of parallel lines. In man-made environments we can observe sets of parallel and

orthogonal lines and planes that can be exploited to compute the orientation of the system (Kosecka and Zhang, 2002). However, the extraction of lines in catadioptric images becomes extraction of conics. Five collinear image points are required to extract them in the uncalibrated case. However, two points are enough if we take advantage of the internal calibration of the catadioptric system. We call these lines, catadioptric image lines (CILs). Some works have been proposed to deal with this problem. In (Vasseur and Mouaddib, 2004), the space of the equivalent sphere which is the unified domain of central catadioptric sensors combined with the Hough transform is used. In (Ying and Hu, 2004b) they also use the Hough transform and two parameters on the Gaussian sphere to detect the image lines. The accuracy on the detection of these two approaches depends on the resolution of the Hough transform. The higher the accuracy the more difficult to compute the CILs. In (Mei and Malis, 2006) the randomized Hough transform is used to overcome the singularity present in (Vasseur and Mouaddib, 2004; Ying and Hu, 2004b) and to speed up the extraction of the conics. This scheme is compared in converge mapping to a RANSAC approach. In (Bazin et al., 2007) an scheme of split and merge is proposed to extract the CILs present in a connected component. These connected components, as in our case, are computed in two steps. The first step consist of detecting the edges using the Canny operator. The second step is a process of chaining which builds the connected components. In contrast to (Vasseur and Mouaddib, 2004; Ying and Hu, 2004b; Mei and Malis, 2006) our approach does not use the Hough transform, instead we compute the CIL directly from two image points present in a connected component. Then a RANSAC approach is used to identify the points that belong to this conic. As opposed to (Bazin et al., 2007) we use an estimation of the geometric distance from a point to a conic instead of an algebraic distance. Notice that a connected component can contain more than one CIL and the process has to be repeated until all CILs are extracted.

Once we have extracted the lines in the catadioptric images (CILs) we need to compute the intersection of parallel CILs to extract the vanishing points. In this paper we propose a modification to the computation of the common self-polar triangle (Barreto, 2003) in order to compute the intersection between a pair of CILs. Instead of having four intersections points between two general conics we have just two in the case of CILs. When this intersection corresponds to parallel CILs these points are the vanishing points. We compute all the intersection between the CILs present in the image. Then with a voting approach we robustly determine which ones are the vanishing points. The first vanishing point to compute is the vertical vanishing point (VVP) from which we are able to perform a rectification of the omnidirectional image. With this rectification we obtain an omnidirectional image fitting the vertical assumption and the applications designed with this constraint can be used. Using an analogous process we compute the horizontal vanishing point (HVP). From this HVP we compute the last angle that gives the whole orientation of the catadioptric system.

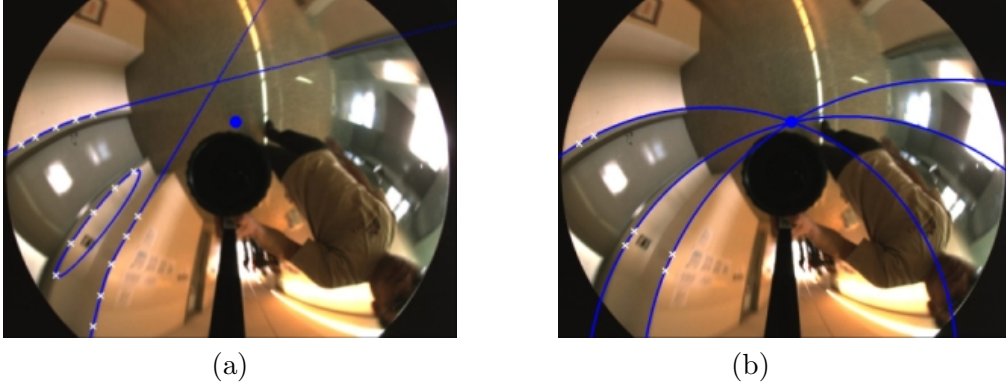


Figure 8.1: Computing a CIL with (a) using the five point approach. (b) using our approach with only two close points. The central blue point corresponds to the vertical vanishing point.

## 8.2 Catadioptric Image Lines (CIL) Computing

In this section we explain the method used to extract the CILs from two image points. As mentioned before in the case of uncalibrated systems we require five points to describe a conic. If these points are not distributed in the whole conic, the estimation is not correctly computed. Another disadvantage of a 5-point approach is the number of parameters. When a robust technique is used, like RANSAC this is quite important, because the number of iterations required hardly increases with the number of parameters of the model. Our approach overcomes these problems since two points are enough. As we assume the calibrated camera we can describe the conics using only two parameters and the calibration parameters, which allows to extract the CIL from 2 points. We compute the points in the normalized plane  $\mathbf{r} = (sx \ sy \ s)^T = (x \ y \ 1)^T$  using the inverse of matrix  $\mathbf{K}$

$$\mathbf{r} = \mathbf{K}^{-1}\mathbf{q}. \quad (8.1)$$

As previously presented, a 2D line  $\mathbf{n}$  in the normalized plane, which corresponds to the projection of a 3D line under the sphere camera model, can be represented as  $\mathbf{n} = (n_x, n_y, n_z)^T$ . The points  $\mathbf{r}$  in the normalized plane lying on this line satisfy  $\mathbf{r}^T \Omega \mathbf{r} = 0$ , where  $\Omega$  is defined as

$$\Omega = \begin{pmatrix} n_x^2(1-\xi^2) - n_z^2\xi^2 & n_x n_y(1-\xi^2) & n_x n_z \\ n_x n_y(1-\xi^2) & n_y^2(1-\xi^2) - n_z^2\xi^2 & n_y n_z \\ n_x n_z & n_y n_z & n_z^2 \end{pmatrix} \quad (8.2)$$

Developing the relation  $\mathbf{r}^T \Omega \mathbf{r} = 0$  and after some algebraic manipulation we obtain

$$(1 - \xi^2)(n_x x + n_y y)^2 + 2n_z(n_x x + n_y y) + n_z^2(1 - \xi^2(x^2 + y^2)) = 0 \quad (8.3)$$



simplifying

$$(1 - \xi^2 r^2) \beta^2 + 2\beta + (1 - \xi^2) = 0 \quad (8.4)$$

where a change of variable to  $\beta = \frac{n_z}{n_x x + n_y y}$  and  $r^2 = x^2 + y^2$  is performed. We can compute  $\beta$  by solving the quadratic equation

$$\beta = -\frac{1}{1 - \xi^2 r^2} \pm \frac{\xi}{1 - \xi^2 r^2} \sqrt{1 + r^2 (1 - \xi^2)} \quad (8.5)$$

After solving this quadratic equation we compute the normal  $\mathbf{n}$ . Consider two points in the normalized plane  $\mathbf{x}_1 = (x_1, y_1, 1)^\top$  and  $\mathbf{x}_2 = (x_2, y_2, 1)^\top$ . From (8.5) we compute the corresponding  $\beta_1$  and  $\beta_2$ . Notice that there exist two solutions for  $\beta$  and just one has a physical meaning<sup>1</sup>. Using these parameters we obtain the linear system

$$\begin{pmatrix} x_1 & y_1 & -\frac{1}{\beta_1} \\ x_2 & y_2 & -\frac{1}{\beta_2} \end{pmatrix} \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (8.6)$$

Since  $\mathbf{n}$  is orthonormal  $n_x^2 + n_y^2 + n_z^2 = 1$ . Solving for  $n_x$ ,  $n_y$  and  $n_z$  we have

$$n_x = \frac{y_1/\beta_2 - y_2/\beta_1}{\nu}, \quad n_y = \frac{x_2/\beta_1 - x_1/\beta_2}{\nu}, \quad n_z = \frac{x_2 y_1 - x_1 y_2}{\nu}, \quad (8.7)$$

with  $\nu = \sqrt{(x_2 y_1 - x_1 y_2)^2 + (y_1/\beta_2 - y_2/\beta_1)^2 + (x_2/\beta_1 - x_1/\beta_2)^2}$ .

Notice that we have analytically computed the normal  $\mathbf{n}$  that defines the projection plane of the 3D line. In Fig. 8.1 we show a comparison of the computing of a image line in the uncalibrated case using five points, and the calibrated case using our approach with only two points. In this figure we can observe that our approach obtains a better estimation even with two very close points. We also observe that the distance of the conic to the vanishing point using our 2-point approach is much better than the general 5-point approach.

### 8.2.1 Catadioptric Image Lines Extraction

Our line extraction proposal can be explained as follows. First we detect the edges using the Canny algorithm. Then the connected pixels are stored in components. For each component we perform a RANSAC approach to detect the CILs present into this component. Two points from the connected component are chosen randomly and the corresponding CIL is computed. The distance from the rest of the points to this CIL is computed. The points with a distance smaller than some threshold vote for this CIL. The process stops when the number of points that has not voted for any conic and the number of points in the component are smaller than a threshold. In Fig. 8.2 we can observe the three main steps to extract the CILs.

<sup>1</sup>We have observed that the negative solution is the correct one.

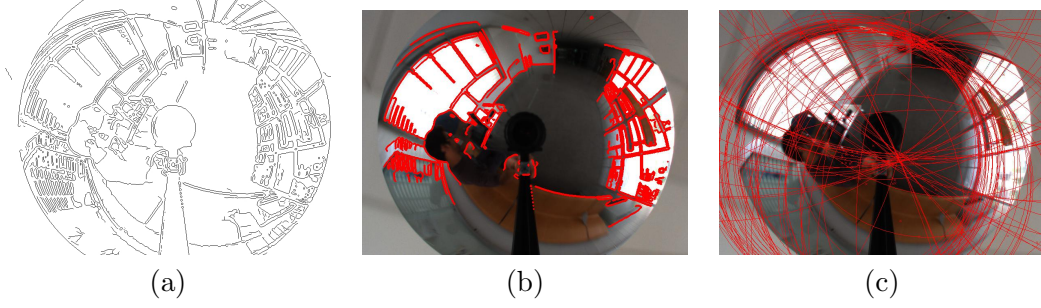


Figure 8.2: Extraction of image lines (CILs). (a) Canny edge detector result, (b) connected components and (c) CILs extracted.

### 8.2.2 Distance from a point to a conic

In order to know if a point  $\mathbf{x}$  lies on a conic  $\mathbf{C}$  we need to compute the distance from a point to a conic. Two distances are commonly used to this purpose. The algebraic distance defined by (8.8) which just gives an scalar value and the geometric distance, which gives the distance from this point to the closest point on the conic. This distance is more difficult to calculate and its computing is time consuming. We propose an estimation to this distance replacing the point-to-conic distance by a point-to-point distance (see Fig. 8.3). Our proposal is based on the gradient of the algebraic distance from a point  $\mathbf{x}_c$  to a conic represented as a 6-vector  $\mathbf{C} = (c_1, c_2, c_3, c_4, c_5, c_6)$

$$d_{alg} = c_1x^2 + c_2xy + c_3y^2 + c_4x + c_5y + c_6. \quad (8.8)$$

We define the perpendicular line to a point that lies on the conic  $\mathbf{C}$  as

$$\ell_{\perp} = \mathbf{x}_c + \lambda \tilde{\mathbf{n}}(\mathbf{x}_c), \quad \text{where} \quad \tilde{\mathbf{n}}(\mathbf{x}_c) = \frac{\nabla d_{alg}}{\|\nabla d_{alg}\|} \quad (8.9)$$

The normal vector  $\tilde{\mathbf{n}}$  is computed from the gradient of the algebraic distance.

$$\nabla d_{alg} = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} = \begin{pmatrix} 2c_1x + c_2y + c_4 \\ c_2x + 2c_3y + c_5 \end{pmatrix} \quad (8.10)$$

When a point does not lie on the conic  $\mathbf{x}_o$  we can compute an estimation to its corresponding perpendicular line using the property that  $\tilde{\mathbf{n}}(\mathbf{x}_c) = \tilde{\mathbf{n}}(\mathbf{x}_o + \Delta\mathbf{x}) \approx \tilde{\mathbf{n}}(\mathbf{x}_o)$

$$\ell_{est} = \mathbf{x}_o + \lambda_{est} \tilde{\mathbf{n}}(\mathbf{x}_o) = \begin{pmatrix} x_o + \lambda_{est} \tilde{n}_x(\mathbf{x}_o) \\ y_o + \lambda_{est} \tilde{n}_y(\mathbf{x}_o) \end{pmatrix} \quad (8.11)$$

To compute  $\lambda_{est}$  we substitute  $x$  by  $x_o + \lambda_{est} \tilde{n}_x(\mathbf{x}_o)$  and  $y$  by  $y_o + \lambda_{est} \tilde{n}_y(\mathbf{x}_o)$  in (8.8), giving a quadratic equation

$$\lambda_{est}^2 \underbrace{(c_1 \tilde{n}_x^2 + c_2 \tilde{n}_x \tilde{n}_y + c_3 \tilde{n}_y^2)}_a + \lambda_{est} \underbrace{(2c_1 \tilde{n}_x + 2c_3 \tilde{n}_y + c_2(x_o \tilde{n}_y + y_o \tilde{n}_x))}_b + \underbrace{(c_1 x_o^2 + c_2 x_o y_o + c_3 y_o^2 + c_4 x_o + c_5 y_o + c_6)}_c = 0 \quad (8.12)$$

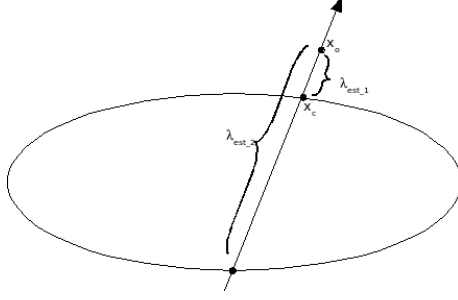


Figure 8.3: Approximation to the distance from a point to a conic.

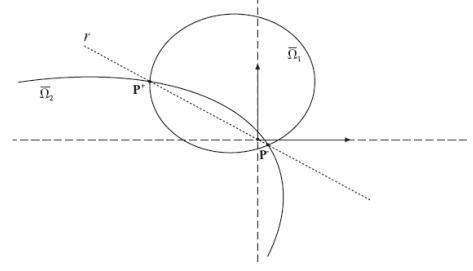


Figure 8.4: Intersection of two CILs in the normalized plane.

We observe that  $\lambda_{est}$  gives the two distances that intersect the conic so, we choose the closest to  $\mathbf{x}_o$  as the distance from that point to the conic  $d = \|\mathbf{x}_o - \mathbf{x}_c\| = \lambda_{est}$ .

### 8.3 Intersection of Two CILs

In a general configuration, two conics intersect in four points. The intersection of these points define three distinct pair of lines. The intersection of these lines represent the vertices of the self-polar triangle common to a pair of conics (Barreto, 2003). We study the particular case where two CILs intersect, which is a degenerate configuration with just two intersection points. As we observe in Fig. 8.4, there exist a line  $r$  that intersects these two points and the origin of the normalized plane. Our goal is to compute this line and from it to extract the two intersections of the conics that correspond to points  $\mathbf{P}^+$  and  $\mathbf{P}^-$ .

Let  $\mathbf{n}_1 = (n_{x1}, n_{y1}, n_{z1})^T$  and  $\mathbf{n}_2 = (n_{x2}, n_{y2}, n_{z2})^T$  two normal vectors representing the projection of two lines in the scene and  $\Omega_1$  and  $\Omega_2$  two conics representing the image lines in the normalized plane. The vertices of the self-polar triangle associated to the pencil  $\Omega(\lambda) = \Omega_1 + \lambda\Omega_2$  satisfy

$$\det(\Omega_1 + \lambda\Omega_2) = 0. \quad (8.13)$$

If we develop this constraint we obtain a third order polynomial where just one of the solutions is real and it corresponds to  $\lambda_1 = -n_{z1}^2/n_{z2}^2$ . So, the null-space of  $\Omega(\lambda_1) = \Omega_1 + \lambda_1\Omega_2$  is the line  $r$ , expressed in a parametric way as

$$r = \mu \cdot \mathbf{v} = \mu \begin{pmatrix} v_x \\ v_y \end{pmatrix} = \mu \begin{pmatrix} n_{z2}^2 n_{y1} n_{z1} - n_{z1}^2 n_{y2} n_{z2} \\ n_{z1}^2 n_{x2} n_{z2} - n_{z2}^2 n_{x1} n_{z1} \end{pmatrix}. \quad (8.14)$$

The intersection of this line to both  $\Omega_1$  and  $\Omega_2$  gives the two points  $\mathbf{P}^+$  and  $\mathbf{P}^-$ . To obtain them we solve for  $\mu$  in (8.15) and substitute in (8.14).

$$\mu^2(c_1 v_x^2 + c_2 v_x v_y + c_3 v_y^2) + \mu(c_4 v_x + c_5 v_y) + c_6 = 0. \quad (8.15)$$

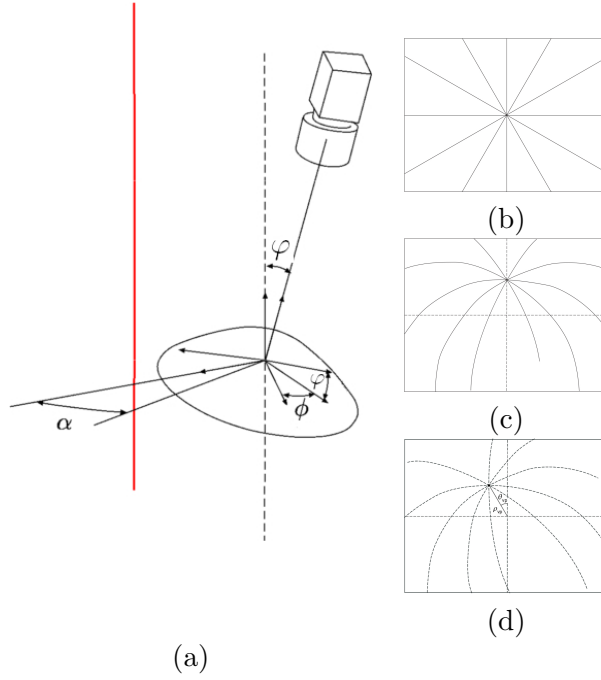


Figure 8.5: (a) Configuration of the catadioptric system in a hand-held situation. (b) The VVP is in the center of the image. (c) The VVP moves in the vertical axis when the camera rotates around the x-axis. (d) The VVP rotates around the image center when the camera rotates around the z-axis.

## 8.4 Vanishing points and Image Rectification

The vanishing points indicate the intersection of image lines corresponding to parallel lines in the scene. In vertical aligned catadioptric systems, vertical lines are radial lines in the image representation. Their intersection point, the vertical vanishing point (VVP), is located at the image center. When the camera is not vertically aligned, the radial lines become conic curves. In this case the VVP moves from the image center and its new location contains important information about the orientation of the camera with respect to the scene.

We use a classic algorithm to detect the VVP. Let  $m$  be the number of putative vertical CILs detected in the omnidirectional image and let  $\mathbf{n}_i$  their corresponding representation in the normalized plane. For every pair of CILs (there is a total of  $m(m-1)/2$  pairs), we compute their intersection as explained above. Then for each line  $\mathbf{n}_i$  we compute the distance to these points. If the line is parallel to that pair of CILs the distance is smaller than a threshold and then that line votes that possible VVP. The most voted point is considered the VVP.

### 8.4.1 Image Rectification

Here we explain the relation between the VVP computed in the normalized plane and the orientation of the catadioptric system. Writing the VVP in polar coordinates  $\mathbf{x}_{vp} = (\rho_{vp}, \theta_{vp})^T$  (see Fig. 8.5(d)) we observe that there exist a relation between the angle  $\theta_{vp}$  and the angle  $\phi$  representing the rotation of the catadioptric system around the z-axis (8.16). The negative angle is produced by the mirror effect which inverts the catadioptric image.

$$\phi = -\theta_{vp} \quad (8.16)$$

We observed that the component  $\rho_{vp}$  is intrinsically related to the rotation angle  $\varphi$  and the mirror parameter  $\xi$  of the catadioptric system. Since angles  $\varphi$  and  $\phi$  are independent, we consider the case where  $\phi = 0$  (see Fig. 8.5(c)). Using (8.14) and (8.15) with a pair of parallel CILs in polar coordinates we compute the following relationship

$$\rho_{vp} = -\frac{\sin \varphi}{\cos \varphi \pm \xi}. \quad (8.17)$$

A lookup table can be built to speed up the computing of the vertical orientation. An analogous process is performed to detect the horizontal vanishing point. With the information provided by this point we are able to compute the full orientation of the catadioptric system.

## 8.5 Experiments

In this section we present some experiments rectifying real images. We acquire an image sequence with a calibrated hand-held hypercatadioptric system<sup>2</sup>. The calibration was performed using (Mei and Rives, 2007). The process to extract the vanishing points and to perform the rectification can be summarized as follows: i) the edges are detected by the Canny operator; ii) the process to construct the connected components is performed; iii) a RANSAC approach is performed for each connected component to extract all CILs present on it; iv) all CIL intersections are computed and the vanishing points are estimated; v) the vertical correction is performed using the orientation information obtained from the VVP; vi) The full rectification is performed using the orientation information contained in the HVP.

In Fig. 8.6 we show an example of full rectification using our approach for a single frame of the image sequence. We observe how the VVP and the HVP are computed. Once the rectification angles are computed we align the image to the reference system given by the scene itself. This allows to see how the only movement present in the sequence is translation (Fig. 8.7).

---

<sup>2</sup><http://www.neovision.cz/>

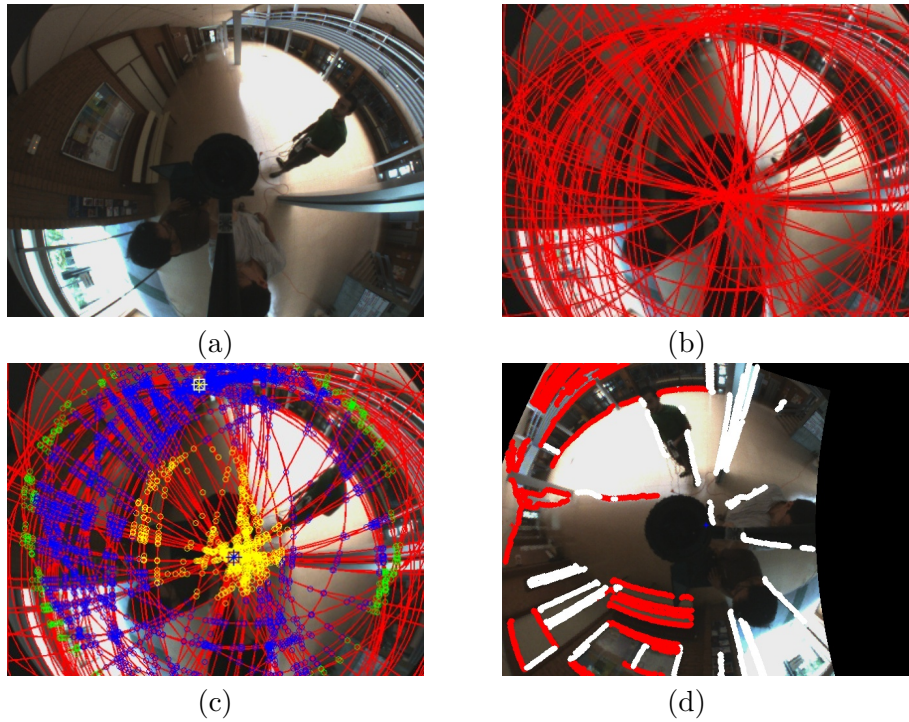


Figure 8.6: Example of full image rectification. (a) Frame 1 of the image sequence. (b) Conic extraction using our approach. (c) Putative vertical and horizontal vanishing points. Yellow circles represent putative vertical vanishing points. The blue ones represent the putative horizontal vanishing points and the green ones are the intersections points that cannot be consider either vertical or horizontal vanishing points. The white square is the estimated HVP and the black one is the VVP. (d) Full-rectified image. The vertical CILs are shown in white and the horizontal ones in red.

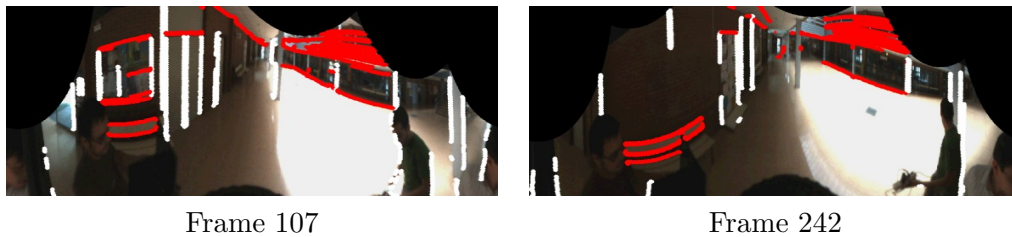


Figure 8.7: Panoramic representation of two full-rectified frames. Vertical lines are shown in white and horizontal ones in red. The horizontal vanishing point is aligned to the image center.

## 8.6 Closure

In this chapter we presented a new way to extract lines in omnidirectional images generated by a calibrated catadioptric system. We only use two points lying on the CIL and an approximation to the geometric distance from a point to a conic. We also showed how to compute the intersection of two image lines based on the common self-polar triangle. From the intersection of CILs we compute the vertical and horizontal vanishing points, which contain the orientation information of the catadioptric system with respect to the scene. To show the effectiveness of this approach we perform experiments with real images. We compute the orientation of a hand-held hyper-catadioptric system through a video sequence.

## Chapter 9

# Topological Map from Only Orientation Information

*In this chapter we present a new way to compute a topological map using only orientation information. We exploit the natural presence of lines in man-made environments in dominant directions. We extract all the image lines present in the scene acquired by an omnidirectional system composed of 6 aligned wide angle cameras. From the parallel lines we robustly compute the three dominant directions using vanishing points. With this information we are able to align the camera with respect to the scene and to identify the turns in the trajectory, assuming a Manhattan world where the changes of heading in the navigation are related by multiples of 90 degrees. We also use geometrical image-pair constraints as a tool to identify the visual traversable nodes that compose our topological map.*

### 9.1 Related Work

In the field of robotics the representation of the space plays a very important role. In general this representation allows the robot to perform different tasks to interact with the environment. Among these tasks we can mention localization, path planning, navigation, etc. Along the years several representations have been proposed but Kuipers (1999) proposes the Spatial Semantic Hierarchy where four levels are considered. The two most used in the literature are the metric and the topological maps. The metric maps are quantitative representations of the environment. This representation usually uses raw data or lines and has some disadvantages. It requires accurate determination of the robot position, it is inefficient for planning, the resolution does not depend on the complexity of the environment (Thrun and Bcklen, 1998). On the other hand the topological maps are purely qualitative, and many of its benefits are independent of the accuracy or even the existence of quantitative knowledge of the environment. These characteristics make the topological maps robust to poor odometry and position errors. Topological approaches represent the environment using a graph structure where nodes represent different places



in the world and edges denote traversable paths between them (Goedemé et al., 2006). Furthermore, the elements of the topological map are strongly related to the semantics of the environments.

There exist several approaches that deal with the automatic generation of topological maps. The difference between them depends on the method and the sensors used. Tapus et al. (2004) use two laser range finders and one omnidirectional camera. They propose the concept of fingerprints to characterize the places visited by the robot and the partially observable Markov decision processes (POMDP) for global localization. Goedemé et al. (2006) try to identify the loop closings. This happens when the robot revisits a place. More specifically, an equivalent sensor reading occurs twice in the sequence. They use a single omnidirectional sensor and the Dempster-Shafer theory to model the uncertainty. Rybski et al. (2003) use a panoramic sensor and two different types of features, the Kanade-Lukas-Tomasi (KLT) and 3D color histograms. The map is modeled as a physics-based mass and spring system. More recently Booij et al. (2007) propose an appearance based topological map. They also use an omnidirectional system combined with the planar motion constraint and the epipolar geometry. They use the number of inliers and outliers to define a similarity value that links different locations. These are the edges of the topological map.

## 9.2 Our Approach

In man-made environments the presence of straight lines is common (Murillo et al., 2007a). Moreover, these lines are aligned with the principal orthogonal directions of the world coordinate frame (Kosecka and Zhang, 2002). From the images of parallel lines we can compute the orientation of the camera with respect to the scene. With the relative orientation of the camera and the guaranty that frames are reachable to each other and the temporal ordering constraints given by a video sequence, we decide to explore the creation of a topological map using only orientation information. With the use of lines we overcome some problems of feature-based approaches, such as lack of texture in the scene or the similarity of features, which is commonly observed in long corridors.

We propose to combine the orientation information with the epipolar geometry to build a topological map in an indoor environment. We extract the orientation of each frame with respect to the scene using the vanishing points. The changes of direction are detected when two consecutive frames have a drastic change on their orientation. We use a feature matching, particularly SIFT Lowe (2004) features, between these frames to identify the direction of the turn. Finally, the epipolar geometry is used to identify the frames that represent the nodes in the topological map. We compute the essential matrix between every  $n$  frames from which we extract the rotation component and compare to that obtained from the visual compass. If they are coherent we recompute the essential matrix with the next frame. The process stops when either the rotations are not coherent or the essential matrix computation fails. We choose the last frame as a new node in the topological map and the process

## 9.2. Our Approach

---

restarts.

### 9.2.1 Computing the relative orientation

In this work we use a polycamera which is composed of 6 aligned wide-angle cameras, having a unified reference system for the device as a whole. The position of each camera with respect to this reference systems is given by the manufacturer, who also provides the calibration of each camera. With the calibration and the alignment information we are able to transform any point in the images to its corresponding ray on the unitary sphere.

We compute the relative orientation of the camera with respect to the scene based on the computation of the vanishing points. The vanishing points are computed using a voting scheme. The first step consist of extracting the line segments  $\mathbf{l}_i$  present in every single image. The Canny extractor is used to extract the edges, then a link function is used to compute the connected components. The two endpoints of one of these connected components are  $\mathbf{q}_1, \mathbf{q}_2$  in the image plane. Using the calibration and the global reference system we transform such points into  $\mathbf{s}_1$  and  $\mathbf{s}_2$  lying on the unitary sphere. The line segment passing through the two endpoints is represented by a plane normal of a plane passing through the center of projection and intersecting the sphere in a line  $\mathbf{l}$ , such that  $\mathbf{l} = \mathbf{s}_1 \times \mathbf{s}_2$ . The unit vectors corresponding to the plane normals  $\mathbf{l}_i$  can be viewed as points on a unit sphere. The vectors  $\mathbf{l}_i$  corresponding to parallel lines in 3D world all lie in one plane. The vanishing direction then corresponds to the plane normal where all these lines lie. Given two lines the common normal is determined by  $\mathbf{v}_m = \mathbf{l}_i \times \mathbf{l}_j$ . Then, for every pair of plane normals we compute putative vanishing points  $\mathbf{v}_m$ . The number of total putative vanishing points corresponding to  $n$  plane normals are  $(n(n-1)/2)$ . The third step consist of computing the distance between the putative vanishing points and all the plane normals. Since in the noise free case  $\mathbf{l}_i^T \mathbf{v}_m = 0$ , we use this product as a measure of error of the line  $\mathbf{l}_i$  pointing in the dominant vanishing direction  $\mathbf{v}_m$ . The plane normals with a distance smaller than some threshold vote for the corresponding vanishing point. The most voted is chosen as the first vanishing point  $\mathbf{v}_i$ . Then we look for the second most voted and orthogonal to the first vanishing point  $\mathbf{v}_j$ . Then we look for the third most voted vanishing point  $\mathbf{v}_k$  and orthogonal to the other two. If just two vanishing points are computed, the third one is obtained by computing the cross product of the other two  $\mathbf{v}_k = \mathbf{v}_i \times \mathbf{v}_j$ . Since the vanishing points are projections of the vectors associated with three orthogonal directions  $i, j, k$ , they depend on rotation only. In particular we can write that

$$\mathbf{v}_i = R\mathbf{e}_i, \quad \mathbf{v}_j = R\mathbf{e}_j, \quad \mathbf{v}_k = R\mathbf{e}_k. \quad (9.1)$$

From matrix  $R$  we can extract the orientation angles corresponding to the three main axis  $(\alpha, \beta, \gamma)$ .

### 9.2.2 Topological Map

Topological maps try to simplify the representation of the environment by modelling space using graphs. This representation is suitable when we just need to know if a new place is reachable from our current position. In this case the metric information of the environment is not needed. Depending on the task, topological or metric maps are required to solve different problems. In this section we explain how to construct a topological map from orientation only information. In this work we represent the frames as nodes and edges meaning visual traversality between such frames.

#### Keyframes

The computation of the keyframes is performed in two steps. The first step is to identify the keyframes from only orientation information. The frame where a change of direction, i.e. a turn, is detected and added to the topological map automatically. The change of direction of the motion axis is detected when a change of approximately  $\pi/2$  is observed in the magnitude of the angle of the main horizontal axis in two consecutive frames. However, this change can also be observed in other circumstances. To avoid false detection of turns we compute the visual traversality test between the fifth frame prior to and the fifth frame after the possible turn frame. The visual traversality test is explained in detail below. If these two frames are not connected, the turn has been performed and the turn frame is added to the topological map. This vanishing points based approach gives us the existence of a turn of  $90^\circ$  but the projective information does not give the direction of turn. In order to identify the direction of the turn ( $+90^\circ$  or  $-90^\circ$ ) we measure the average motion in pixels of the features in the two analyzed frames. If the features move to the left, the motion performed by the camera is to the right and vice versa. When a turn is performed, the direction of motion is assigned to the new one that is detected. This means that the two horizontal axis are switched, an event that is recorded only in the topological map.

In the second step we compute the visually traversable nodes of the topological map. To decide if two frames are connected (visually traversable test), we compute the essential matrix  $\mathbf{E} = [\mathbf{t}]_{\times} \mathbf{R}$  between the frames  $f_1$  and  $f_2$  using a correspondence-based approach. SIFT points are the most used local and robust features and we use them in this work. From the essential matrix we extract the rotation matrix  $\mathbf{R}$  and translation vector  $\mathbf{t}$  as explained in Hartley and Zisserman (2000). Then we compare the rotation matrix  $\mathbf{R}$  to the one obtained from the combination of the rotation matrix of  $f_1$  and  $f_2$  with respect to the scene,  $\mathbf{R}_c = \mathbf{R}_1 \mathbf{R}_2^T$ , respectively. It is graphically presented in Fig. 9.1. If both matrices are congruent and the number of SIFT correspondences is bigger than a threshold, the two frames are connected. In the real world, the interpretation is that the space between the two frames is visually traversable. In this case we increment  $f_2$  by  $n$  frames and compute again the essential matrix. When the computing of the essential matrix is not possible, that means the space between  $f_1$  and  $f_2$  is not visually traversable. In this case we

### 9.3. Experiments

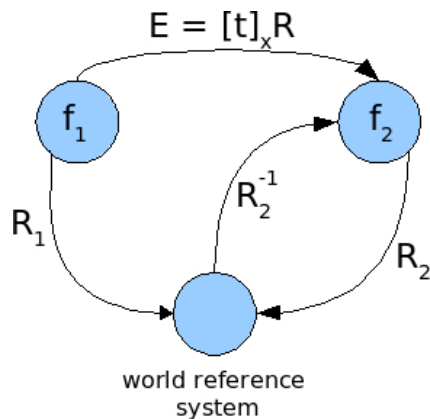


Figure 9.1: Relation between the rotation matrices from Essential matrix and from vanishing points.

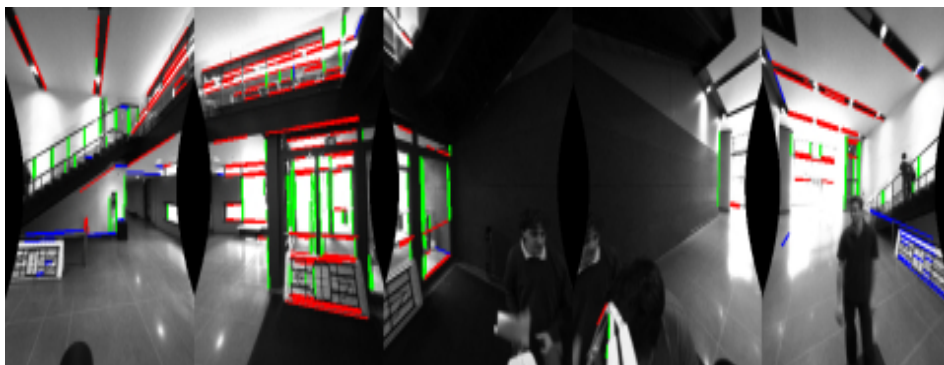


Figure 9.2: Lines corresponding to the computed vanishing points.

add the frame  $f_2$  as a visually traversable node to the topological map. We restart the computation of the essential matrix with the node  $f_2$  as the initial image and look for the next visually traversable node as explained before.

### 9.3 Experiments

In this section we present the experiments using an indoor sequence acquired by an omnidirectional camera composed of 6 perspective cameras<sup>1</sup>. We use a total of 17,000 frames (3400 by each camera, the top sensor is not used). The trajectory performed involves several turns inside a building. The calibration of the individual cameras and the alignment matrices of the cameras with respect to the camera head coordinate frame are given by the manufacturer.

<sup>1</sup>Ladybug 2 <http://www.ptgrey.com/>

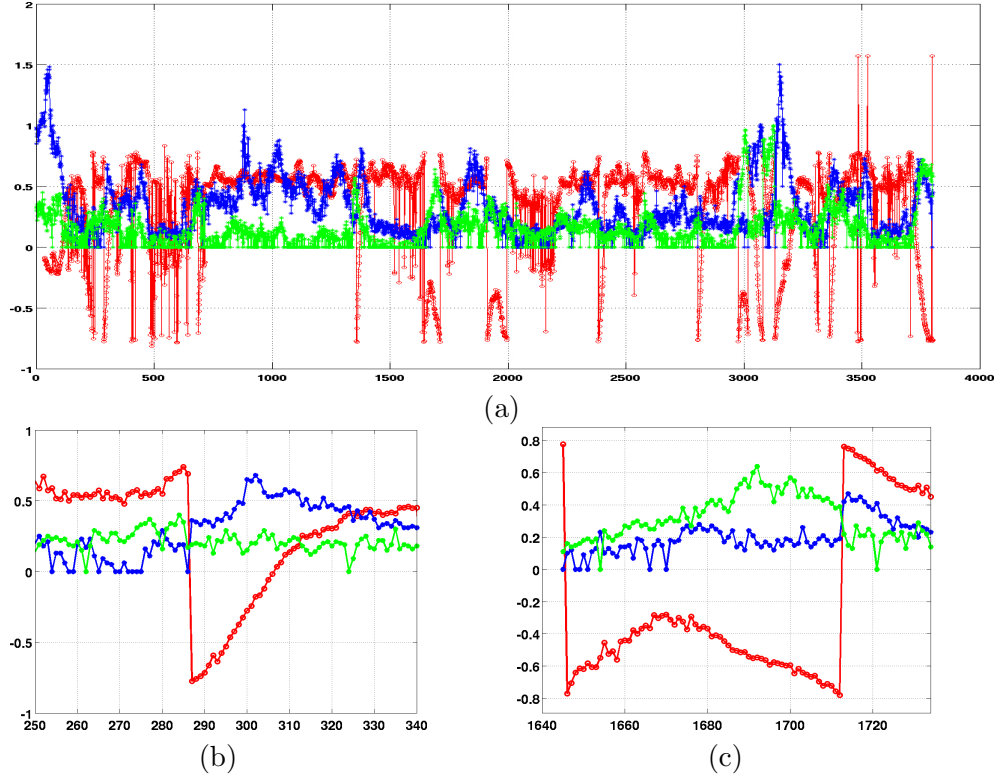


Figure 9.3: Rotation angle around  $x$ -axis in radians corresponding to the to the frame number indicated in the horizontal axis of the plot (red). In blue, number of lines supporting the  $x$ -axis vanishing direction. In green, number of lines supporting the  $y$ -axis vanishing direction. Both numbers are scaled by 0.01. (a) Whole sequence. (b) Turn performed in the trajectory. (c) Change of orientation with respect to the scene.

### 9.3.1 Orientation computation

As we mentioned before, the first step of our approach is to compute the vanishing points. In Fig. 9.2 we observe the extraction and classification of the lines from the cameras used by the omnidirectional system. Each color represents a direction in the scene. From the three vanishing points obtained we extract the rotation angles for each axis. As the camera moves on a platform an almost planar motion is performed. The motion around the  $x$ -axis is the main motion observed through the trajectory.

In Fig. 9.3(a) we show the computation of the rotation angle corresponding to the main axis of the omnidirectional camera and the number of lines (scaled by 100) supporting the two main directions of this system. We observe changes of magnitude of  $\pi/2$  in consecutive frames. This displacement indicates a possible change on the axis of motion. A few examples of these possible turns are the frames 287, 686, 1357, 3018 and 3081. We analyze two particular cases, given by frames 287 and 1645. In

### 9.3. Experiments

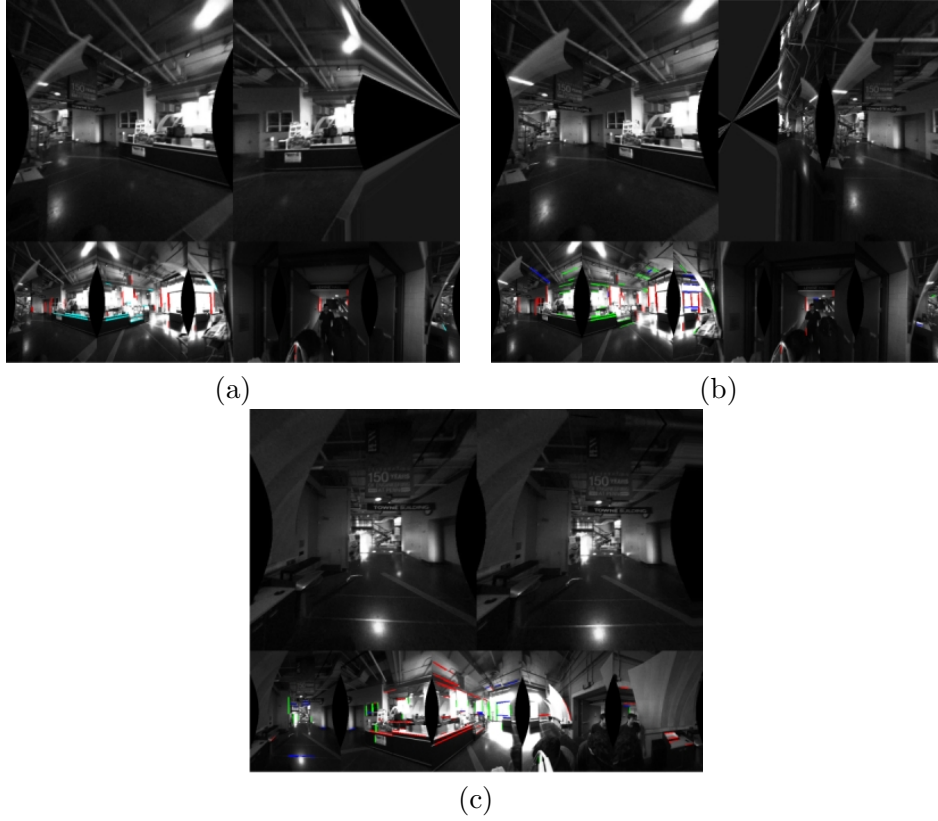


Figure 9.4: Change of direction corresponding to a turn detected in two consecutive frames (a) and (b). (c) frame aligned to the new scene, the turn has been completed (0.064 radians).

the first case (Fig. 9.3(b)), we observe an orientation change of 1.46 radians. In this case a turn has been performed, from left to right. We observe that the number of lines supporting the main axis decreases (blue) while the number of lines supporting the other non-vertical orthogonal axis increases (green) and at the frame where the turn is detected a switch is performed. In Fig. 9.4 we show the two frames where the turn was detected and a frame where the camera is aligned to the new position in the scene, i. e. when the orientation angle of this frame with respect to the scene is close to zero. In this particular case the turn has been performed in approximately 20 frames. To detect the turns performed in the opposite direction (right-to-left), we observe the changes of orientation of magnitude close to  $\pi/2$ . From negative to positive values.

The second case analyzed corresponds to the 1645 frame (Fig. 9.3(c)) where a change of 1.53 radians has been observed. In this case the radical change on the orientation does not correspond to a turn. The angle between the main axis of the camera reference system and the motion axis is bigger than  $\pi/4$ . In this case, the camera is pointing to a different direction from where the motion is performed. A

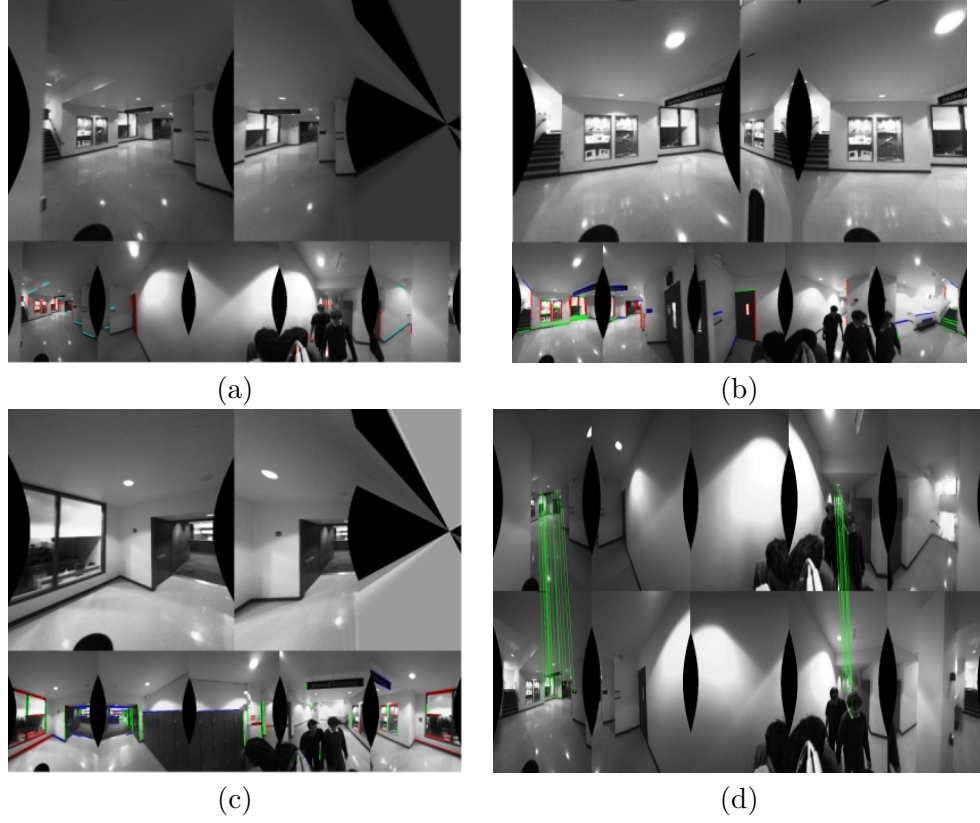


Figure 9.5: Change of direction that does not correspond to a turn. (a) frame 1645. (b) frame 1672. (c) frame 1713. (d) Visual traversability test between frames 1640 and 1650.

characteristic of this behavior is that the camera is not aligned to the scene, i.e. its orientation is not close to zero. The end of this phase is identified when there is a change in the orientation from a negative value to a positive one with a magnitude close to  $\pi/2$  (see Fig. 9.3(c) frame 1713). In this case we also observe how the number of lines supporting the two non-vertical directions switches when the radical change of orientation is observed. The start, middle and ending steps of this phase can be observed in Fig. 9.5(a-c). Fig. 9.5(d) shows the SIFT correspondences between the frames 1640 and 1650. The number of correspondences indicates that the two frames are connected. Therefore this change on the orientation does not correspond to a turn.

### 9.3.2 Drawbacks of the approach

It is possible that the lines present in the scene are not aligned with the principal orthogonal directions of the world coordinate frame. In this case the estimation of the vanishing points is not reliable. We can observe this situation in Fig. 9.3(a),



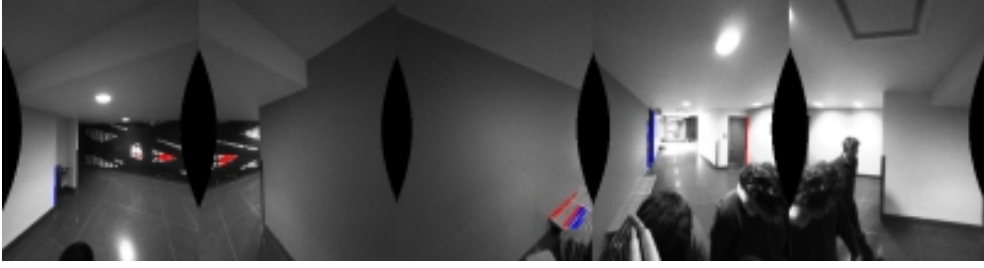


Figure 9.6: Bad estimation of the vanishing points in frame 200.

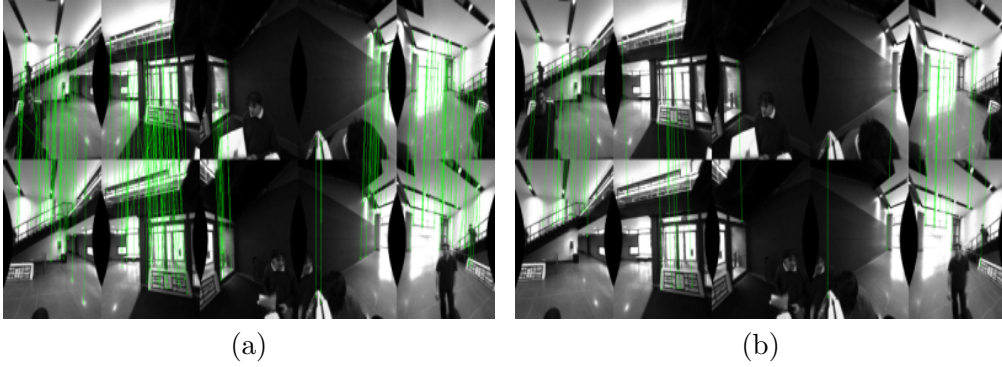


Figure 9.7: Connected frames. Frames 35 and 45. (a) Putative matches. (b) Robust matches.

from frame 200 to frame 283. An example of the lines detected in this situation can be observed in Fig. 9.6.

#### 9.3.3 Building Topological Map

In order to build a topological map we need to identify the keyframes that represent the nodes. Above we explain how to identify the turn frames. These frames are added to the topological map. The rest of keyframes are computed using an essential matrix to identify the visual traversable frames. The essential matrix is computed using a feature based approach. We extract the SIFT points from two frames. To have a better distribution of the points we use a bucketing in every image. We match these well distributed points to have the first putative correspondences (see Fig. 9.7(a)). The presence of outliers is inevitable. In that order we use a robust approach with a geometric constraint to avoid the outliers Fig. 9.7(b). With eight correspondences we compute the essential matrix. From which we extract the rotation matrix.

From the rotation matrices corresponding to the two frames we extract the rotation angles corresponding to each axis. If the difference between the angles is greater than a predetermined threshold or the number of matches is less than 10, these frames are not connected. In Fig. 9.8 we observe two non-connected frames. In this case the second frame is selected as a keyframe and it is added to the topological



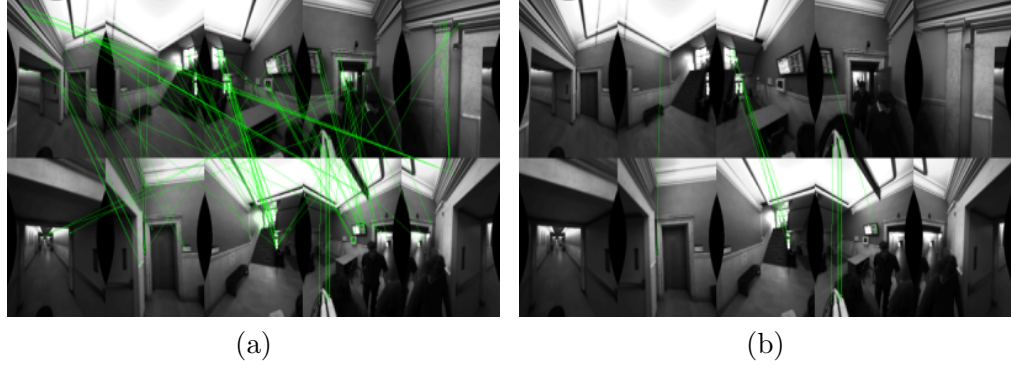


Figure 9.8: Non-Connected frames. Frames 675 and 685. (a) Putative matches. (b) Robust matches.

map. When the frames are connected we recompute the essential matrix with the next frame and repeat this action until we find a couple of non-connected frames. Fig. 9.7 shows two frames that are connected.

As mentioned before we detect the turns in the trajectory (Fig. 9.4). To identify the direction of the turn we verify the motion of the features between the two frames where the turn was detected. An example of the topological map created by our approach can be observed in Fig. 9.9. The red circles are the detected turns and the green ones are the visually traversable frames computed by the epipolar geometry. The distance between the nodes is related to the number of frames that separates them. We also observe the corresponding orientation of the frame with respect to the scene. We take the orientation of the first frame as the world reference system. The trajectory corresponds to a loop. The initial and final keyframes are not close since we do not have the correct scale to plot the map. We observe that the initial and final frames have a similar orientation. The blue circles represent areas where a change of orientation is detected but it does not correspond to a turn (see Fig. 9.3(c)). As we observe the direction of the motion is not changed in these cases.

### Loop-Closure detection

In order to detect the loop-closure we compute the essential matrix between the actual keyframe and all the previous keyframes stored in the topological map. A loop-closure is detected when the two keyframes have the same orientation and the number of correspondences between the two frames is bigger than a previously defined threshold. In the sequence used just one loop is present. In Fig. 9.10 we observe frame 287 ( $-0.77$  radians) and frame 3365 ( $-0.74$  radians) that besides having a similar orientation they have 46 correspondences validated by the epipolar geometry.

### 9.3. Experiments

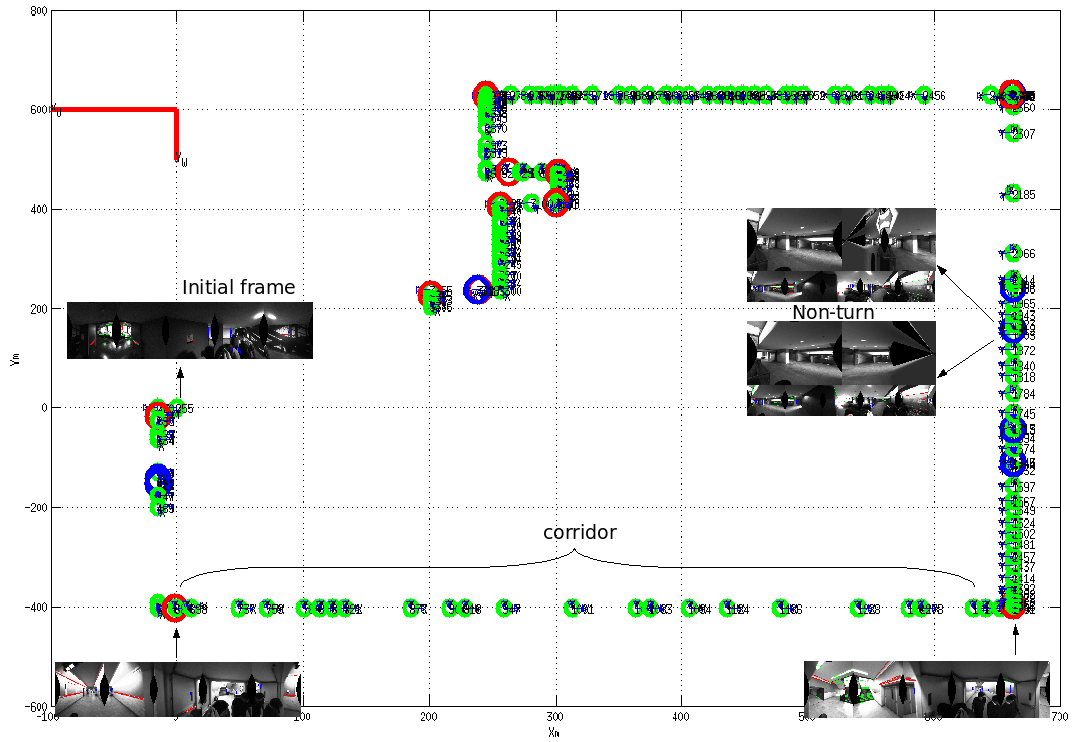


Figure 9.9: Topological map of the indoor sequence.

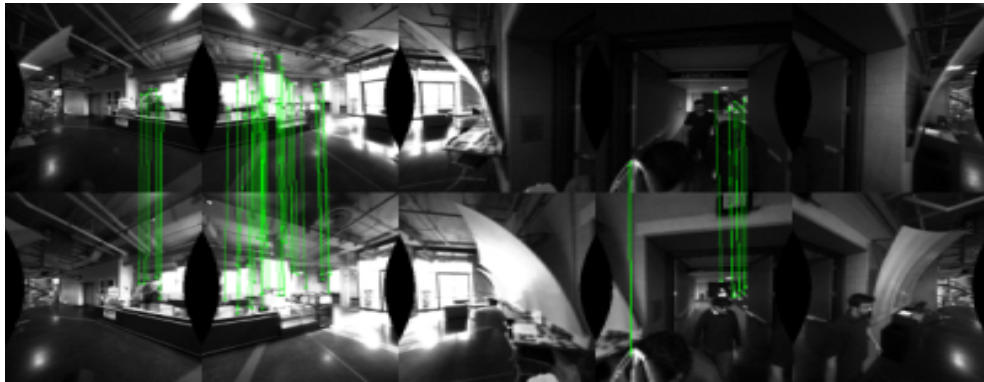


Figure 9.10: Frames 287 and 3365 where the loop-closure is detected.

#### Corridor detection

To detect a corridor there are some premises that must be satisfied: i) a continuous motion from the image sequence is assumed; ii) turns are not detected, meaning the orientation of the omnidirectional system is within a certain range; and iii) the number of lines supporting the non-vertical dominant direction is clearly bigger than the rest. With this information we can infer that the camera is traversing a corridor.

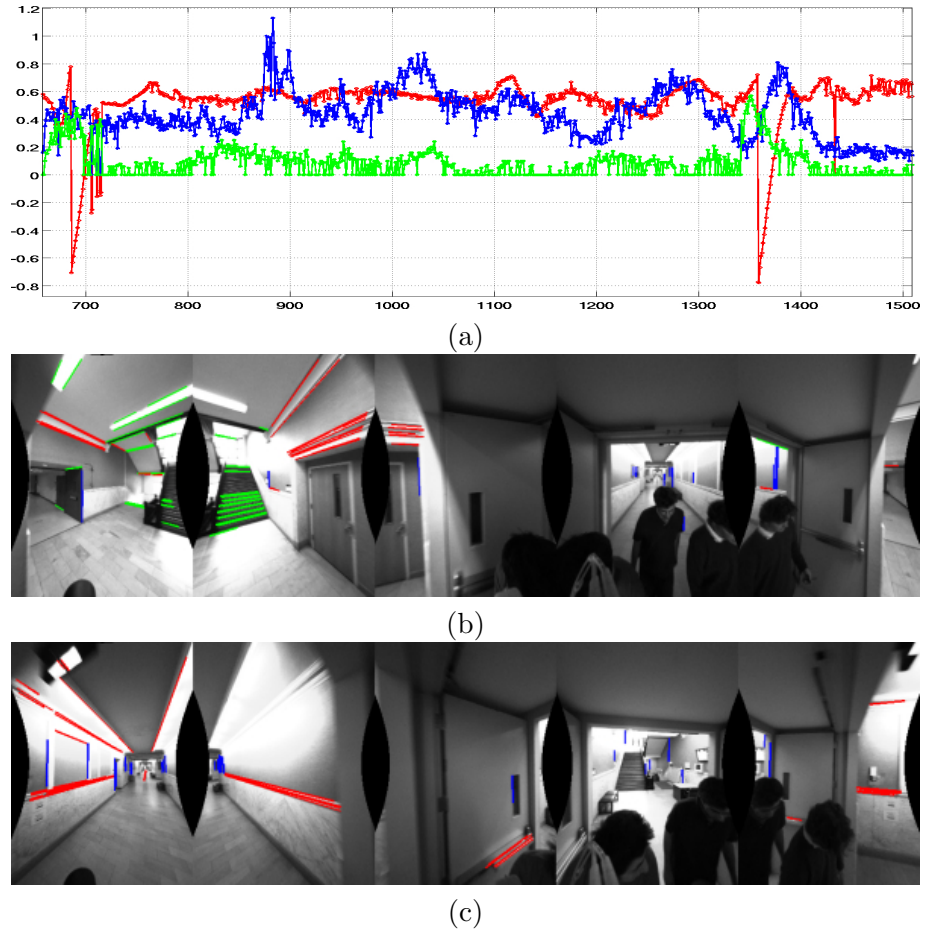


Figure 9.11: Corridor detection. (a) Detection in the sequence showing the number of lines scaled by 0.01, supporting the two main directions of the camera.  $x$ -axis (blue) and  $y$ -axis (green). (b) Initial frame. (c) Final frame.

In Fig. 9.11(a) we observe how the angle (in red) is inside the range  $(0.4, 0.7)$  radians. It starts at frame 716 and finishes at frame 1357. The number of lines supporting this dominant direction (blue line) is clearly superior to the rest (green line). In Fig. 9.11(b,c) we show the initial and the final frames where the corridor is detected.

### Traversable area detection

In order to analyze the situation of the keyframes with respect to the scene we compute the position of the lines present on the scene. We observe that the position of lines perpendicular to the main axis of the omnidirectional camera indicate if it is possible to move in that direction. We divided the space in four big areas corresponding to the main directions, up, down, left and right. We count the number of lines inside each area for each dominant direction. In Fig. 9.12 we show an example where the number of perpendicular lines (green) in the left cell indicates that it

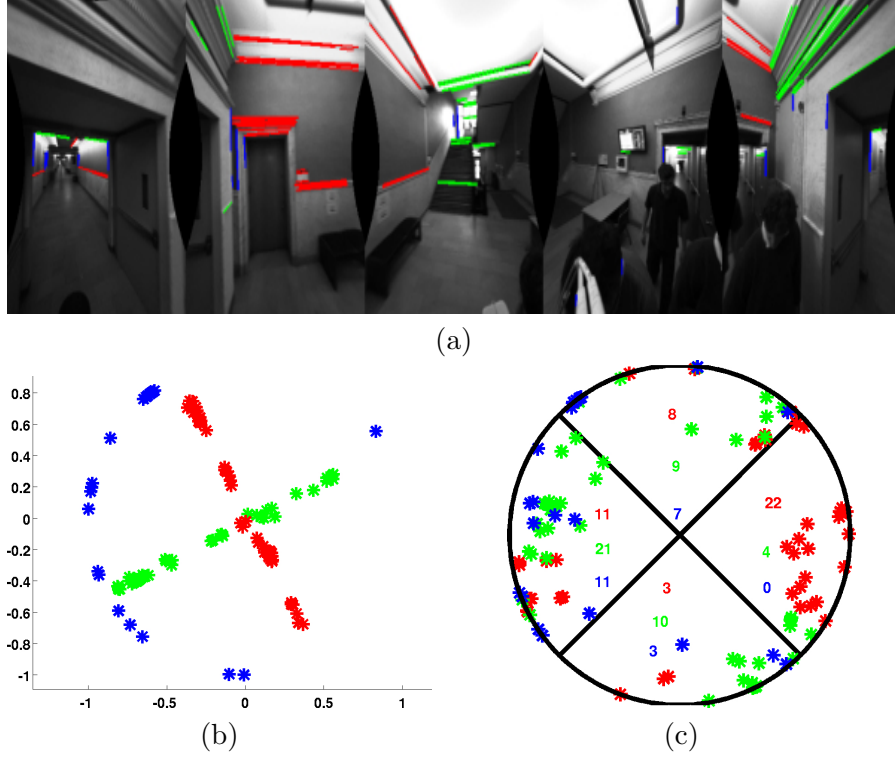


Figure 9.12: Identification of traversable areas. (a) Frame 690 with lines supporting the dominant directions  $x$ -axis (red),  $y$ -axis (green) and  $z$ -axis (blue). (b) Space of lines represented by their normals and point to the corresponding vanishing points. (c) Lines spread on the four displacement areas (up, right, down and left).

is possible to move into that direction. As we can see in Fig. 9.12(a), those lines are pointing to the corridor where the camera come from. Therefore, it is actually possible to move in that direction.

## 9.4 Closure

We have presented a new approach to compute a topological map in an indoors environment using an omnidirectional camera. We exploit the presence of straight lines to compute the vanishing points, from which we estimate the orientation of the camera with respect to the scene. From only orientation information we are able to detect turns in the trajectory and with an initial reference system we are able to construct a coherent topological map. We use geometrical constraints, particularly the essential matrix, to select the keyframes that represent the nodes in the topological map. The nodes are visually traversable to each other. We performed experiments with a sequence of real images in an indoor environment. We observe that the use of only orientation information and the epipolar geometry as a tool give a coherent construction of a topological map.



## Chapter 10

# Conclusions

In this thesis we have studied omnidirectional cameras, in particular central catadioptric systems. We analyzed from the very basic step of calibration, where we propose a new method that is able to calibrate all central catadioptric systems including the pin-hole camera. In practice this method showed that it could go further, being able to calibrate non-central catadioptric systems, such as the popular wide-angle fish-eye cameras. We also performed a deep analysis of the available calibration methods for central catadioptric systems, considering also those used to calibrate fish-eye cameras. The analysis is theoretical and practical. First, we present a classification of such methods and give the most relevant information about them. For the practical part we select those methods available on-line as OpenSource, since in the computer vision and robotics communities, these methods can save time and effort when the goal is beyond the calibration itself. The new calibration method and the analysis of calibration methods were presented in **Chapter 3** and **Chapter 4**, respectively.

In **Chapter 5** we studied in great detail the two-view relations of central catadioptric systems, in particular the combination of conventional cameras and catadioptric systems. We select this combination since a single catadioptric view contains a more complete description of the scene, and the perspective image gives a more detailed description of the particular area or object we are interested in. We perform a deep analysis on the behavior of three approaches to compute the hybrid fundamental matrix and the hybrid planar homography. From the simplest, specific for a particular mirror shape, to the generic ones, which consider all central catadioptric systems. We show how to extract relevant information from the fundamental matrix, more specific the extraction of the epipoles from the corresponding hybrid fundamental matrix. We also show how to impose the rank 2 property to this matrix, which is required to obtain the epipoles. We observed that the generic models are more accurate under ideal conditions, but at the same time are more sensitive to perturbations (image noise) than the simpler models. With the more reliable and simplest models we successfully match perspective images with catadioptric views robustly.

We also studied in this thesis the scale-space for catadioptric systems. The scale-space computation is the most important step on the extraction of scale invariant

features. We presented a method which is able to compute this scale-space for all central catadioptric systems. We propose to combine the sphere camera model and the partial differential equations framework on manifolds. With this combination we compute the Laplace-Beltrami (LB) operator related to any parametric surface, in this case, the manifold which represents the mirror of the catadioptric system. The LB operator is used to smooth the catadioptric images in order to construct the scale-space. Once the scale-space is obtained the approximation to the Laplacian of Gaussians (LoG) through the Difference of Gaussians (DoG) is performed and the extrema points are detected. This approach and some results with synthetic and real images are presented in **Chapter 6**.

In the second part of this thesis, which corresponds to **chapters 7, 8 and 9**, we presented three different robotic applications using omnidirectional systems. In the first application we show that with the use of the two-view relations, the hybrid fundamental matrix and the hybrid planar homography it is possible to construct a localization system for a robot. This robot is equipped with an on-board perspective camera and a visual memory of omnidirectional images of the environment stored in its memory. It robustly matches the current image acquired with the perspective to the closest omniview stored in the visual memory, with the geometric constraint of the hybrid fundamental matrix. From this matrix the epipole, given the position of the on-board camera in the scene, is computed. Then this point (in pixels) is mapped to the ground plane (in meters) through the hybrid planar homography.

In **chapter 8** we compute the orientation of a hypercatadioptric system in a man-made environment. We take advantage of the presence of straight line elements in such environments. These elements are aligned to the dominant directions and their projections give the orientation of the system with respect to that reference system. In this chapter we propose different solutions to the different steps comprised computing the orientation of the system. In first place, the extraction of the projections of straight lines in the scene from the catadioptric image becomes more complex, since these projections are no longer lines but conics. We propose a new strategy to extract catadioptric image lines. This approach requires the calibration of the system and only 2 points lying on the catadioptric image lines we want to extract. To decide if a point belongs or not to a certain conic we propose a point-to-point distance that approximates the distance from a point to a conic. With the lines extracted the next step is to compute the intersection between projections of parallel lines. In this order we propose the use of the common self-polar triangle adapted for catadioptric image lines, which instead of having four intersections as in the general case, it has only two and its computation is simplified to a solution of a quadratic equation. Finally, a classic voting scheme is used to identify the intersections of parallel lines with more supporting lines, which represent the vertical and the horizontal vanishing point. From the computed vanishing points the extraction of the orientation of the catadioptric system with respect to the absolute reference system given by the dominant directions is straightforward.

The last application proposed in this part of the thesis uses a polycamera, the popular LadyBug camera, in an indoors environment. We showed that with only

orientation information and the assumption of continuous motion from an image sequence we were able to construct a topological map. From the projections of lines present in the scene we compute the vanishing points, which as mentioned before, contain the orientation information of the camera system with respect to the reference system given by the dominant directions. To identify the keyframes that compose our topological map we propose the use of a traversality test. With this test we are able to identify turns, intermediate keyframes in long corridors and also loops in the camera trajectory. This traversality test is based on the computation of the robust matching between camera frames using local features.

### 10.1 Future Work

As we can observe from this thesis, omnidirectional images are facing the same problems as the conventional ones with a more complex geometry. The development of omnidirectional vision is at the beginning if compared to the development of the conventional one. Because of this situation all areas where omnidirectional sensors could be used must be explored. In particular we would like to highlight the following:

- In the case of catadioptric systems where the information of the mirror is required to take into account its geometry for further applications. This situation opens an opportunity where automatic methods to extract this information from the environment should be developed, instead of obtaining the information from an off-line calibration process.
- The development of the omnidirectional vision should consider models that can deal with the majority of projection systems. Facing at the same time the problems present in the conventional images and the more complex omnidirectional ones.
- From conventional vision we observe that the first goal is to find reliable and robust methods to solve the problems. The second goal is to speed up this methods or to design faster ones based on the previously developed ones. This should be the way to lead the development of the omnidirectional vision.





# Bibliography

- Abdel-Aziz, Y. and Karara, H. (1971). Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In *Symposium on Close-Range Photogrammetry*, pages 1–18.
- Aliaga, D. (2001). Accurate catadioptric calibration for real-time pose estimation in room-size environments. *Proceedings of the 8th IEEE International Conference on Computer Vision, ICCV 2001.*, 1:127–134 vol.1.
- Arican, Z. and Frossard, P. (2010). OmniSIFT: Scale Invariant Features in Omnidirectional Images. In *Proceedings of the International Conference on Image Processing (ICIP)*.
- Baker, S. and Nayar, S. (1999). A theory of single-viewpoint catadioptric image formation. *International Journal of Computer Vision*, 35(2):175–196.
- Barreto, J. (2003). *General Central Projection Systems: Modeling, Calibration and Visual Servoing*. PhD thesis, University of Coimbra.
- Barreto, J. (2006). A unifying geometric representation for central projection systems. *Computer Vision and Image Understanding*, 103(3):208–217.
- Barreto, J. and Daniilidis, K. (2006). Epipolar geometry of central projection systems using veronese maps. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006*, volume 1, pages 1258–1265.
- Barreto, J. P. and Araujo, H. (2001). Issues on the geometry of central catadioptric image formation. *Proceedings of the Conference on Computer Vision and Pattern Recognition, CVPR 2001.*, 2:422–427.
- Barreto, J. P. and Araujo, H. (2002). Geometric properties of central catadioptric line images. In *Proceedings of the 7th European Conference on Computer Vision-Part IV, ECCV '02*, pages 237–251, London, UK.
- Barreto, J. P. and Araujo, H. (2005). Geometric properties of central catadioptric line images and their application in calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1327–1333.

- Bartoli, A. and Sturm, P. (2004). Non-linear estimation of the fundamental matrix with minimal parameters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(4):426–432.
- Bastanlar, Y., Puig, L., Sturm, P., Guerrero, J. J., and Barreto, J. (2008). Dlt-like calibration of central catadioptric cameras. In *Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras, Marseille, France*.
- Bazin, J. C., Demonceaux, C., and Vasseur, P. (2007). Fast central catadioptric line extraction. In *IbPRIA '07: Proceedings of the 3rd Iberian conference on Pattern Recognition and Image Analysis, Part II*, pages 25–32.
- Bermúdez, J., Puig, L., and Guerrero, J. (2010). Line extraction in central hyper-catadioptric systems. In *OMNIVIS - 10th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras, Zaragoza, Spain*.
- Bertalmío, M., Cheng, L.-T., Osher, S., and Sapiro, G. (2001). Variational problems and partial differential equations on implicit surfaces. *Journal of Computational Physics*, 174(2):759 – 780.
- Bogdanova, I., Bresson, X., Thiran, J.-P., and Vanderghenst, P. (2007). Scale space analysis and active contours for omnidirectional images. *IEEE Transactions on Image Processing*, 16(7):1888 –1901.
- Booi, O., Terwijn, B., Zivkovic, Z., and Krose, B. (2007). Navigation using an appearance based topological map. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 3927–3932.
- Briggs, A., Yunpeng, L., Scharstein, D., and Wilder, M. (2006). Robot navigation using 1d panoramic images. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2679 –2685.
- Buchanan, T. (1988). The twisted cubic and camera calibration. *Computer Vision, Graphics and Image Processing*, 42(1):130–132.
- Bulow, T. (2004). Spherical diffusion for 3d surface smoothing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(12):1650–1654.
- Caglioti, V., Taddei, P., Boracchi, G., Gasparini, S., and Giusti, A. (2007). Single-image calibration of off-axis catadioptric cameras using lines. *Proceedings of the 11th IEEE International Conference on Computer Vision, ICCV 2007.*, pages 1–6.
- Cauchois, C., Brassart, E., Delahoche, L., and Clerentin, A. (2003). 3d localization with conical vision. In *Computer Vision and Pattern Recognition Workshop, 2003. CVPRW '03. Conference on*, volume 7, page 81.
- Cauchois, C., Brassart, E., Drocourt, C., and Vasseur, P. (1999). Calibration of the omnidirectional vision sensor: Syclop. In *IEEE International Conference on*

## BIBLIOGRAPHY

---

- Robotics and Automation, 1999. Proceedings. 1999*, volume 2, pages 1287–1292 vol.2.
- Chen, D. and Yang, J. (2005). Image registration with uncalibrated cameras in hybrid vision systems. In *WACV/MOTION*, pages 427–432.
- Chen, X., Yang, J., and Waibel, A. (2003). Calibration of a hybrid camera network. In *Proceedings. Ninth IEEE International Conference on Computer Vision*, pages 150 –155 vol.1.
- Claus, D. and Fitzgibbon, A. W. (2005). A rational function lens distortion model for general cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 213–219.
- Courbon, J., Mezouar, Y., Eck, L., and Martinet, P. (2007). A generic fisheye camera model for robotic applications. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 1683–1688.
- Courbon, J., Mezouar, Y., Eck, L., and Martinet, P. (2008). Efficient visual memory based navigation of indoor robot with a wide-field of view camera. In *ICARCV 2008. 10th International Conference on Control, Automation, Robotics and Vision*, pages 268 –273.
- Criminisi, A., Reid, I., and Zisserman, A. (1999). A plane measuring device. *Image and Vision Computing*, 17(8):625–634.
- Cruz, J., Bogdanova, I., Paquier, B., Bierlaire, M., and Thiran, J.-P. (2009). Scale Invariant Feature Transform on the Sphere: Theory and Applications. Technical report, EPFL.
- Deng, X.-M., Wu, F.-C., and Wu, Y.-H. (2007). An easy calibration method for central catadioptric cameras. *Acta Automatica Sinica*, 33(8):801 – 808.
- Espuny, F. and Burgos Gil, J. (2011). Generic self-calibration of central cameras from two rotational flows. *International Journal of Computer Vision*, 91:131–145. 10.1007/s11263-010-0335-9.
- Faugeras, O. (1993). *Three-Dimensional Computer Vision (Artificial Intelligence)*. The MIT Press.
- Frank, O., Katz, R., Tisse, C., and Durrant Whyte, H. (2007). Camera calibration for miniature, low-cost, wide-angle imaging systems. In *British Machine Vision Conference*.
- Gasparini, S., Sturm, P., and Barreto, J. P. (2009). Plane-based calibration of central catadioptric cameras. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1195–1202.

- Geyer, C. and Daniilidis, K. (1999). Catadioptric camera calibration. In *Proceedings of the 7th IEEE International Conference on Computer Vision*, volume 1, pages 398–404 vol.1.
- Geyer, C. and Daniilidis, K. (2000). A unifying theory for central panoramic systems and practical applications. In *Proceedings of the 6th European Conference on Computer Vision-Part II*, pages 445–461, London, UK.
- Geyer, C. and Daniilidis, K. (2001a). Catadioptric projective geometry. *International Journal of Computer Vision*, 45:223–243.
- Geyer, C. and Daniilidis, K. (2001b). Structure and motion from uncalibrated catadioptric views. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. CVPR 2001.*, volume 1, pages I–279–I–286 vol.1.
- Geyer, C. and Daniilidis, K. (2002a). Paracatadioptric camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):687–695.
- Geyer, C. and Daniilidis, K. (2002b). Properties of the catadioptric fundamental matrix. In *Proceedings of the 7th European Conference on Computer Vision*, pages 140–154.
- Goedemé, T., Nuttin, M., Tuytelaars, T., and Van Gool, L. (2007). Omnidirectional vision based topological navigation. *International Journal of Computer Vision*, 74:219–236. 10.1007/s11263-006-0025-9.
- Goedemé, T., Tuytelaars, T., and Gool, L. J. V. (2006). Visual topological map building in self-similar environments. In *ICINCO-RA*, pages 3–9.
- Grossberg, M. and Nayar, S. (2001). A general imaging model and a method for finding its parameters. In *Proceedings of the Eighth IEEE International Conference on Computer Vision, ICCV 2001.*, volume 2, pages 108–115 vol.2.
- Guerrero, J. J., Murillo, A. C., and Sagüés, C. (2008). Localization and matching using the planar trifocal tensor with bearing-only data. *IEEE Transactions on Robotics*, 24(2):pp. 494–501.
- Hansen, P., Corke, P., and Boles, W. (2010). Wide-angle visual feature matching for outdoor localization. *The International Journal of Robotics Research*, 29:267–297.
- Hansen, P., Corke, P., Boles, W., and Daniilidis, K. (2007). Scale-invariant features on the sphere. In *IEEE 11th International Conference on Computer Vision*, pages 1–8.
- Hartley, R. I. and Zisserman, A. (2000). *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049.

## BIBLIOGRAPHY

---

- Heikkila, J. and Silven, O. (1997). A four-step camera calibration procedure with implicit image correction. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997.*, pages 1106–1112.
- Horn, R. and Johnson, C. (1985). *Matrix Analysis*. Cambridge Univ. Press.
- Horn, R. and Johnson, C. (1991). *Topics in Matrix Analysis*. Cambridge Univ. Press, 2nd edition.
- Ishiguro, H. and Tsuji, S. (1996). Image-based memory of environment. In *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 634–639.
- Jankovic, N. and Naish, M. (15-17 April 2007). A centralized omnidirectional multi-camera system with peripherally-guided active vision and depth perception. *Networking, Sensing and Control, 2007 IEEE International Conference on*, pages 662–667.
- Kadir, T. and Brady, M. (2001). Saliency, scale and image description. *International Journal of Computer Vision*, 45:83–105.
- Kang, S. B. (2000). Catadioptric self-calibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 201–207.
- Kannala, J. and Brandt, S. (2004). A generic camera calibration method for fish-eye lenses. In *Proceedings of the 17th International Conference on Pattern Recognition. ICPR 2004*, volume 1, pages 10–13 Vol.1.
- Kosecka, J. and Zhang, W. (2002). Video compass. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part IV*, pages 476–490, London, UK. Springer-Verlag.
- Kuipers, B. (1999). The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233.
- Lourenço, M., Barreto, J., and Malti, A. (2010). Feature detection and matching in images with radial distortion. In *IEEE International Conference on Robotics and Automation*, pages 1028 –1034.
- Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110.
- Matas, J., Chum, O., Urban, M., and Pajdla, T. (2002). Robust wide baseline stereo from maximally stable extremal regions. In *British Machine Vision Conference*.
- Matsumoto, Y., Ikeda, K., Inaba, M., and Inoue, H. (1999). Visual navigation using omnidirectional view sequence. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 317 –322 vol.1.

- Mauthner, T., Fraundorfer, F., and Bischof, H. (2006). Region matching for omnidirectional images using virtual camera planes. In *Proc. of the 11th Computer Vision Winter Workshop 2006, Telc, Czech Republic*, pages 93–98.
- Mei, C. and Malis, E. (2006). Fast central catadioptric line extraction, estimation, tracking and structure from motion. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 4774–4779.
- Mei, C. and Rives, P. (2007). Single view point omnidirectional camera calibration from planar grids. In *IEEE International Conference on Robotics and Automation*, pages 3945–3950.
- Menem, M. and Pajdla, T. (2004). Constraints on perspective images and circular panoramas. In Andreas, H., Barman, S., and Ellis, T., editors, *BMVC 2004: Proceedings of the 15th British Machine Vision Conference*, London, UK. BMVA, British Machine Vision Association.
- Mezouar, Y., Abdelkader, H., Martinet, P., and Chaumette, F. (2004). Central catadioptric visual servoing from 3d straight lines. In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 1, pages 343–348.
- Micusik, B. and Pajdla, T. (2003). Estimation of omnidirectional camera model from epipolar geometry. In *Computer Vision and Pattern Recognition.*, volume 1, pages I-485 – I-490 vol.1.
- Micusik, B. and Pajdla, T. (2006). Structure from motion with wide circular field of view cameras. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(7):1135–1149.
- Mikolajczyk, K. and Schmid, C. (2004). Scale & affine invariant interest point detectors. *International Journal of Computer Vision*, 60:63–86.
- Miyauchi, R., Shiroma, N., and Matsuno, F. (2007). Development of omni-directional image stabilization system using camera posture information. In *Robotics and Biomimetics, 2007. ROBIO 2007. IEEE International Conference on*, pages 920–925.
- Morel, O. and Fofi, D. (2007). Calibration of catadioptric sensors by polarization imaging. *IEEE International Conference on Robotics and Automation.*, pages 3939–3944.
- Murillo, A., Sagüés, C., Guerrero, J. J., T. Goedemé, T. T., and Gool, L. V. (2007a). From omnidirectional images to hierarchical localization. *Robotics and Autonomous Systems*, 55(5):372–382.
- Murillo, A. C., Guerrero, J. J., and Sagüés, C. (2007b). Surf features for efficient robot localization with omnidirectional images. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3901–3907, Rome-Italy.

## BIBLIOGRAPHY

---

- Orghidan, R., Salvi, J., and Mouaddib, E. M. (2003). Calibration of a structured light-based stereo catadioptric sensor. In *Conference on Computer Vision and Pattern Recognition Workshop, 2003. CVPRW '03.*, volume 7, pages 70–70.
- Pajdla, T. and Hlavác, V. (1999). Zero phase representation of panoramic images for image based localization. In *Proceedings of the 8th International Conference on Computer Analysis of Images and Patterns*, pages 550–557. Springer-Verlag.
- Puig, L., Bastanlar, Y., Sturm, P., Guerrero, J. J., and Barreto, J. (2011). Calibration of central catadioptric cameras using a dlt-like approach. *International Journal of Computer Vision*, 93:101–114.
- Puig, L., Bermúdez, J., and Guerrero, J. J. (2010a). Self-orientation of a hand-held catadioptric system in man-made environments. In *IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, USA.
- Puig, L., Bermúdez, J., Sturm, P., and Guerrero, J. J. (2012). Calibration of omnidirectional cameras in practice: A comparison of methods. *Computer Vision and Image Understanding*, 116(1):120 – 137.
- Puig, L., Daniilidis, K., and Guerrero, J. J. (2010b). Topological map from only visual orientation information using omnidirectional cameras. In *IEEE ICRA 2010 Workshop on Omnidirectional Robot Vision*, Anchorage, Alaska, USA.
- Puig, L. and Guerrero, J. J. (2009). Self-location from monocular uncalibrated vision using reference omniviews. In *IROS 2009: The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, USA.
- Puig, L. and Guerrero, J. J. (2011). Scale space for central catadioptric systems. towards a generic camera feature extractor. In *IEEE 13th International Conference on Computer Vision*, Barcelona, Spain.
- Puig, L., Guerrero, J. J., and Sturm, P. (2008a). Hybrid matching of uncalibrated omnidirectional and perspective images. In *ICINCO-RA (2)*, pages 125–128.
- Puig, L., Guerrero, J. J., and Sturm, P. (2008b). Matching of omindirectional and perspective images using the hybrid fundamental matrix. In *Proceedings of the Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras, Marseille, France*.
- Ramalingam, S., Sturm, P., and Lodha, S. (2005). Towards complete generic camera calibration. In *Proceedings of the Conference on Computer Vision and Pattern Recognition. CVPR 2005.*, volume 1, pages 1093–1098 vol. 1.
- Ramalingam, S., Sturm, P., and Lodha, S. K. (2010). Generic self-calibration of central cameras. *Computer Vision and Image Understanding*, 114(2):210 – 219. Special issue on Omnidirectional Vision, Camera Networks and Non-conventional Cameras.



- Rituerto, A., Puig, L., and Guerrero, J. (2010a). Comparison of omnidirectional and conventional monocular systems for visual slam. In *10th OMNIVIS, held with Robotics Science and Systems (RSS) (Best Paper Award)*.
- Rituerto, A., Puig, L., and Guerrero, J. (2010b). Visual slam with an omnidirectional camera. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 348–351.
- Rybicki, P. E., Zacharias, F., Lett, J.-F., Masoud, O., Gini, M., and Papanikolopoulos, N. (2003). Using visual features to build topological maps of indoor environments. In *In Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, pages 850–855.
- Scaramuzza, D., Martinelli, A., and Siegwart, R. (2006). A flexible technique for accurate omnidirectional camera calibration and structure from motion. In *Proceedings of IEEE International Conference on Computer Vision Systems*.
- Strelow, D., Mishler, J., Koes, D., and Singh, S. (2001). Precise omnidirectional camera calibration. *Proceedings of the Conference on Computer Vision and Pattern Recognition, CVPR 2001*, 1:I-689–I-694 vol.1.
- Sturm, P. (2002). Mixing catadioptric and perspective cameras. In *Workshop on Omnidirectional Vision, Copenhagen, Denmark*, pages 37–44.
- Sturm, P. and Barreto, J. (2008). General imaging geometry for central catadioptric cameras. In *Proceedings of the 10th European Conference on Computer Vision*, volume 4, pages 609–622. Springer.
- Sturm, P. and Gargallo, P. (2007). Conic fitting using the geometric distance. In *Proceedings of the Asian Conference on Computer Vision, Tokyo, Japan*. Springer.
- Sturm, P. and Ramalingam, S. (2004). A generic concept for camera calibration. In Pajdla, T. and Matas, J., editors, *Computer Vision - ECCV 2004*, volume 3022 of *Lecture Notes in Computer Science*, pages 1–13. Springer Berlin / Heidelberg.
- Sturm, P., Ramalingam, S., Tardif, J.-P., Gasparini, S., and Barreto, J. (2011). Camera models and fundamental concepts used in geometric computer vision. *Foundations and Trends in Computer Graphics and Vision*, 6(1-2):1–183.
- Svoboda, T. and Pajdla, T. (2002). Epipolar geometry for central catadioptric cameras. *International Journal of Computer Vision*, 49(1):23–37.
- Swaminathan, R. and Nayar, S. (1999). Non-metric calibration of wide-angle lenses and polycameras. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on*, volume 22, pages 1172–1178.
- Tapus, A., Tomatis, N., and Siegwart, R. (2004). Topological global localization and mapping with fingerprints and uncertainty. In *ISER*, pages 99–111.

## BIBLIOGRAPHY

---

- Tardif, J.-P., Sturm, P., and Roy, S. (2006). Self-calibration of a general radially symmetric distortion model. In *Proceedings of the 9th European Conference on Computer Vision*, volume 4 of *Lecture Notes in Computer Science*, pages 186–199. Springer.
- Thrun, S. and Bcken, A. (1998). Learning maps for indoor mobile robot navigation. *Artificial Intelligence*, 99:21–71.
- Toepfer, C. and Ehlgen, T. (2007). A unifying omnidirectional camera model and its applications. In *Proceedings of the 11th International Conference on Computer Vision, ICCV 2007.*, pages 1–5.
- Tsai, R. (1987). A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *Robotics and Automation, IEEE Journal of*, 3(4):323–344.
- Vandeportaele, B., Cattoen, M., Marthon, P., and Gurdjos, P. (2006). A new linear calibration method for paracatadioptric cameras. *Proceedings of the 18th International Conference on Pattern Recognition. ICPR 2006.*, 4:647–651.
- Vasseur, P. and Mouaddib, E. M. (2004). Central catadioptric line detection. In *British Machine Vision Conference*.
- Vedaldi, A. (2007). An open implementation of the SIFT detector and descriptor. Technical Report 070012, UCLA CSD.
- Wu, F., Duan, F., Hu, Z., and Wu, Y. (2008). A new linear algorithm for calibrating central catadioptric cameras. *Pattern Recognition*, 41(10):3166–3172.
- Wu, Y. and Hu, Z. (2005). Geometric invariants and applications under catadioptric camera model. *IEEE International Conference on Computer Vision*, 2:1547–1554.
- Wu, Y., Li, Y., and Hu, Z. (2006). Easy calibration for para-catadioptric-like camera. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5719–5724.
- Ying, X. and Hu, Z. (2004a). Catadioptric camera calibration using geometric invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10):1260–1271.
- Ying, X. and Hu, Z. (2004b). Catadioptric line features detection using hough transform. In *ICPR 2004. Proceedings of the 17th International Conference on*, volume 4, pages 839–842.
- Ying, X. and Zha, H. (2005). Simultaneously calibrating catadioptric camera and detecting line features using hough transform. *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005, IROS 2005.*, pages 412–417.