HELSINKI UNIVERSITY OF TECHNOLOGY
Department of Engineering Physics and Mathematics

**Juho Kannala**

# Measuring the Shape of Sewer Pipes from Video

Master's Thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Technology.

Espoo, 10th August 2004

Supervisor: Prof. Jouko Lampinen
Instructor: Dr. Sami Brandt

TEKNILLINEN KORKEAKOULU DIPLOMITYÖN TIIVISTELMÄ


| | |
|---|---|
| **Tekijä:** | Juho Kannala |
| **Osasto:** | Teknillisen fysiikan ja matematiikan osasto |
| **Pääaine:** | Informaatiotekniikka |
| **Sivuaine:** | Laskennallinen tekniikka |


| | |
|---|---|
| **Työn nimi:** | Viemäriputken muodon määrittäminen videokuvasta |
| **Title in English:** | Measuring the Shape of Sewer Pipes from Video |
| **Professuuri:** | S-114 Laskennallinen tekniikka |
| **Työn valvoja:** | Prof. Jouko Lampinen |
| **Työn ohjaaja:** | TkT Sami Brandt |
| **Päivämäärä:** | 10.8.2004 **Sivumäärä:** 76 |


Työssä käsitellään näkymän kolmiulotteisen rakenteen määrittämistä automaattisesti kuvasarjasta. Sovelluksena on viemäriputken muodon määrittäminen videokuvasta putken sisällä liikkuvaan robottiin asennetun kameran avulla. Kamerassa on kalansilmälinssi, jonka hyvin laaja kuvakulma mahdollistaa koko putken seinämän kuvaamisen yhden läpikulun aikana. Työn keskeiset osat ovat (1) kalansilmälinssin kalibrointi ja (2) putken muodon automaattinen rekonstruointi kalibroidulla kameralla kuvatusta kuvasarjasta.

Kalansilmälinssin kalibrointiin esitetään menetelmä, joka perustuu tasomaisen kalibrointikohteen käyttöön. Työssä esitellään yleinen matemaattiseen mallintamiseen pohjautuva kameramalli, joka soveltuu paitsi erityyppisille kalansilmälinsseille myös tavanomaisille linsseille. Menetelmä kameraparametrien määräämiseksi on monitasoinen, jotta vaikeahko optimointiongelma saadaan ratkaistua. Käyttäen kalibroinnissa kohdistusmerkkeinä ympyröitä saavutettiin sekä tavanomaisella että kalansilmäkameralla alle kymmenesosapikselin suuruinen projektiovirheen keskihajonta.

Putken muodon määrittäminen kuvasarjasta pohjautuu kuvissa näkyvien piirrepisteiden jäljittämiseen peräkkäisistä kuvista. Piirrepisteillä tarkoitetaan tässä pisteitä, jotka ovat paikannettavissa putken sisäpinnan tekstuurin epätasaisuuksista. Kuvasarjan yli muodostetut vastaavuudet piirrepisteiden välillä mahdollistavat kameran liikkeen ja pisteiden kolmiulotteisten koordinaattien yhtäaikaisen ratkaisemisen. Vastaavuuksien automaattisessa määrittämisessä hyödynnetään geometrisia rajoitteita, jotka määräytyvät kuvaparien ja -kolmikoiden välille fundamentaalimatriisin ja trifokaalitensorin perusteella. Työssä esitetään geometristen rajoitteiden käyttäminen pisteiden jäljityksessä tavalla, joka yleistyy erityyppisille kalibroiduille kameroille.

Viemärivideolla tehdyt kokeilut osoittavat, että kuvasarjasta jäljitetyt ja rekonstruoidut pisteet ovat putkimaisessa muodostelmassa, josta putken todellinen muoto on määritettävissä. Saavutettavissa olevan mittaustarkkuuden sekä menetelmän soveltuvuuden arviointi erilaisille putkimateriaaleille vaatii kuitenkin jatkotutkimusta. Pitkien kuvasarjojen rekonstruointi on nykyisellä toteutuksella työlästä ja siten menetelmän soveltaminen käytännön viemäritarkastuksissa edellyttää tuntuvaa jatkokehitystä.



| | |
|---|---|
| **Avainsanat:** | tietokonenäkö, kalansilmälinssi, kameran kalibrointi, monen näkymän geometria, kolmiulotteinen rekonstruktio |
| **Hyväksytty:** | |

HELSINKI UNIVERSITY OF TECHNOLOGY ABSTRACT OF MASTER'S THESIS


| | |
|---|---|
| **Author:** | Juho Kannala |
| **Department:** | Department of Engineering Physics and Mathematics |
| **Major subject:** | Computer and Information Science |
| **Minor subject:** | Computational Engineering |


| | |
|---|---|
| **Title:** | Measuring the Shape of Sewer Pipes from Video |
| **Title in Finnish** | Viemäriputken muodon määrittäminen videokuvasta |
| **Chair:** | S-114 Computational Engineering |
| **Supervisor:** | Prof. Jouko Lampinen |
| **Instructor:** | Dr. Sami Brandt |
| **Date:** | 10.8.2004 **Pages:** 76 |


In this thesis we consider automatic 3D model acquisition from video sequences. The application problem is to recover the shape of a sewer pipe from a video sequence taken by a camera moving inside the pipe. The camera is equipped with a fish-eye lens, whose wide field of view makes it possible to obtain a scan of the whole pipe by a single pass. This thesis has two central parts: (1) calibration of a fish-eye lens camera and (2) recovery of the shape of a sewer pipe from a calibrated image sequence.

We describe a camera calibration method for fish-eye lens cameras that is based on viewing a planar calibration pattern. A general camera model is presented that is suitable for both fish-eye lens cameras and conventional cameras. The method for determining the camera parameters is hierarchical so that the optimisation problem can be solved successfully. The standard deviation of the calibration residuals was below 0.1 pixels for both a fish-eye lens camera and a conventional camera when a calibration plane with circular control points was used.

Recovering the shape of a sewer pipe from a video sequence is based on tracking interest points across successive images in the sequence. Here the interest points are points where the image intensity changes rapidly due to irregularities in the surface texture of the pipe. The established point correspondences over the image sequence allow to compute simultaneously the camera motion and the 3D coordinates of the points. To avoid false correspondences in tracking we utilise the geometric constraints between successive image pairs and image triplets. Tracking and reconstruction of points from image sequences is described in a general framework that extends to different kinds of calibrated cameras.

The experiments with real sewer videos show that the arrangement of the reconstructed points is tubular and the shape of the pipe may be estimated from the reconstruction. However, evaluating the attainable measurement accuracy and determining the validity of the approach for different kinds of sewer pipes are topics of future research. By our current implementation it is laborious to compute reconstructions from long image sequences, hence, a lot of further development is needed before the methods can be used in real sewer pipe inspections.



| | |
|---|---|
| **Keywords:** | computer vision, fish-eye lens, camera calibration, multiple view geometry, three-dimensional reconstruction |
| **Approved:** | |

# Preface

This work was carried out in the Laboratory of Computational Engineering at Helsinki University of Technology as a part of a project aiming at developing automatic methods for video-based measurements of sewer pipes. As my part in the project it became to study methods for automatic 3D model acquisition of sewer pipes from video sequences. Since I had already earlier gotten contact with geometry and computer vision I found the topic interesting and took the opportunity to work on it. I am glad for doing this since the topic turned out to be a fascinating mixture of theory and practice.

I want to express my deepest gratitude to Dr. Sami Brandt, my instructor, who has helped me with my work in so many ways and whose enthusiasm originally led me into this field. His comments on my work, his support and our numerous conversations have been invaluable.

I am grateful to Prof. Jouko Lampinen for providing me with the opportunity to work on this topic and for taking care of the project. Mr. Hannu Maula and Mr. Juhani Korkealaakso from VTT Building and Transport deserves thanks for introducing us to this interesting application area of 3D computer vision. The experiments with the sewer robot would not have been possible without Mr. Priit Uleksin, who patiently assembled and disassembled the robot and the camera always when needed, and I wish to thank him for that. I would also like to thank Mr. Jukka Laurila for the many helpful advice regarding video signal processing. Dr. Aki Vehtari as well has always kindly offered his help in practical problems.

Finally, I want to thank my family and friends for support and understanding. Sometimes my dedication to studies and work has been unfair for them.

Otaniemi, 10th August 2004,

Juho Kannala

# Contents

# Chapter 1

# Introduction

One of the most fundamental problems in computer vision is to understand the structure of a real world scene given several images of it. The problem is commonly known as the *structure from motion* problem. However, since it usually involves the simultaneous estimation of the camera motion it is also called the *structure and motion* problem.

Reconstructing the three-dimensional world from two-dimensional projections has many applications. In photogrammetry for instance, topographic maps have been constructed for a long time. In robotics, autonomous vehicles require advanced vision algorithms. Modelling of buildings and precise three-dimensional measurements of big industrial parts provide applications for the three-dimensional computer vision. The application area of this thesis, video-based measurements of sewer pipelines, may be classified to the latter category.

Finding solutions to the structure and motion problem requires knowledge of the geometric laws that describe how the different views of a scene are related. Although geometry is one of the oldest branches of mathematics, there has been some remarkable progress in the understanding of the geometry of multiple views during the past decades. The foundation for this progress is in the projective geometry which has several advantages over the usual Euclidean one. For example, in projective geometry perspective projections may be represented by linear matrix equations. The discovery of the multiple view tensors, which enclose the geometric constraints between multiple views of a single scene, is one of the recent advances in the theory of geometric computer vision.

Besides the theoretical advances, the improvements at a more practical level have also been important. In practice, geometrical transformations, like rotations and translation between the views, are estimated from image measurements which always contain noise. Often automatically extracted data, such as point correspondences between two views, also contain false measurements which are in disagreement with the geometric model. For these reasons it is not always trivial, how to convert the geometric knowledge into working algorithms. The current stage of development of theory and practice in geometric computer vision is extensively summarised in two recent books [Hartley00] and [Faugeras01].

In this thesis, we look at the reconstruction problem from the viewpoint of a video-based sewer line measurement system. The aim is to measure the shape of sewer pipes from video sequences that are acquired by a robot moving in

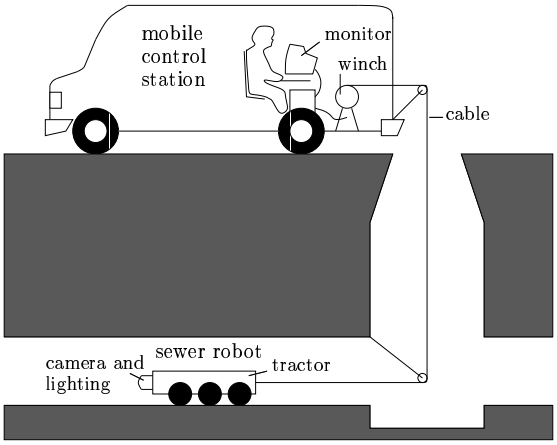



**Figure 1.1:** A typical sewer inspection system

the sewer. While the multiple view geometry and robust estimation algorithms provide the theoretical background for the solutions, the implementation of a working system for the reconstruction of sewer pipes still contains many problems. To give a better overview of the application problem we describe it in more detail in the following section.

## 1.1 Problem Setting

Typical equipment for the inspection of sewer pipelines consists of a video camera and a remote controlled tractor. The tractor is connected to a mobile control station by a cable which provides the power for the robot and transmits the video signal to the operator. Usually the camera and lighting are attached to a special pan and tilt head of the robot which enables the camera to look at different directions. The sewer robot we used had a fixed installation of the camera instead, but the camera was equipped with a fish-eye lens, which has a very wide field of view. The wide field of view makes it possible to obtain a high resolution scan of the whole pipe by a single pass. A typical sewer inspection system is schematically illustrated in Fig. 1.1.

The condition assessment of sewer pipes is usually carried out by visual inspection of the video to find defects and deformations. However, the manual inspection has a number of drawbacks such as subjectivity, varying standards, and high costs. Therefore there has been intentions to develop techniques for automatic assessment of sewer lines. The automatic measurement of deformations places the research problem for this thesis. The full three-dimensional reconstruction of the pipe would definitely solve the problem, but it is a difficult task to obtain such a reconstruction solely from the video. However, from a scientific point of view, the reconstruction problem contains many interesting subproblems while addressing them is useful, even if one would need other instruments, in addition to a camera, in order to realise a robust sewer measurement system.

## 1.2 Aims of the Thesis

The principal goal of this thesis is to develop methods for video-based shape measurement of sewer pipes. Our approach is to use a video camera as the only measuring instrument and to solve the general structure and motion problem by tracking interest points across successive frames in the video sequence. In this case the interest points are points where the image intensity changes rapidly due to irregularities in the surface texture of the pipe. If enough interest points can be tracked and reconstructed, the arrangement of the corresponding three-dimensional points should be tubular. Thus, in principle, the shape of the pipe may be estimated this way.

The advantage of the approach pursued in this thesis is its generality. The implemented methods could be used also in other applications, where there is a need to recover the scene structure or the camera motion from video sequences. However, the approach above may be computationally heavy and its validity for different kinds of pipes is unclear. Therefore we will also briefly discuss other possible solutions to the problem. For instance, the use of structured light to determine the cross-sectional shape of the pipe is considered.

Before any precise metric measurements can be done by the camera, it must be calibrated. The calibration roughly means that for each point in the image we determine the direction of the back-projected ray. The calibration is particularly important in our application because usual methods for the structure and motion problem assume the pinhole camera model, which is not a valid approximation to a fish-eye lens. The problem is that an accurate and easy-to-use calibration method has not been available for fish-eye lenses. Therefore an important partial goal of this thesis is to propose a calibration procedure for fish-eye lenses.

## 1.3 Overview of the Thesis

The organisation of the thesis is as follows. First in Chapter 2, we review some previous approaches to the structure and motion problem as well as different equipment and methods for sewer pipe inspections. Chapter 3 presents the camera model and calibration method that we propose for fish-eye lens cameras. The calibration experiments and results are described in Chapter 4. In Chapter 5, we tell how the interest points are extracted from the images and how putative point correspondences between successive frames are obtained. Chapter 6 concentrates on multiple view geometry and describes the geometric constraints between two and three views. Then, Chapter 7 illustrates how these constraints may be used for automatic and reliable tracking of points. In Chapter 7, we also describe the reconstruction procedure and show an example of a reconstructed pipe section. Finally, the results of the thesis are summarised and discussed in Chapter 8. To make this thesis self-contained, Appendix A provides a short introduction to the projective geometry.

# Chapter 2

# Related Work

The aim of this chapter is to review previous and ongoing research that is related to the subjects of the thesis. First in Section 2.1, we give an overview of the common approaches to the structure from motion problem and then in Section 2.2, we study different equipment and methods for sewer pipe inspection.

## 2.1 Structure from Motion

The problem of reconstructing a three-dimensional scene from its two-dimensional projections lies in the intersection of two disciplines. Basically, it is the case of "measuring graphically by means of light" which falls into the field of *photogrammetry* by definition [Slama80]. On the other hand, the objective of making automatic measurements from digital images has made the problem a fundamental one also in *computer vision*. Since there is a tendency towards automation of the standard photogrammetric processes, these disciplines will continue to overlap [Schenk99].

A characteristic feature of the modern geometric computer vision is that the approach is uncalibrated [Hartley00]. When the conventional way is to first calibrate the cameras and then compute a metric reconstruction from matched images, the modern way allows ignoring the values of internal camera parameters to obtain a projective reconstruction. It is even possible to update the projective reconstruction to a metric one by determining the camera parameters directly from multiple uncalibrated images without any specific calibration objects. This is called auto-calibration [Faugeras92a, Pollefeys98].

The advantage of the modern approach is that it leads to a more general theory of geometric relations between multiple views. However, the uncalibrated approach usually assumes pinhole cameras and ignores lens distortion, apart from a few exceptions [Fitzgibbon01, Mičušík03]. Often pinhole camera is not a valid assumption if the aim is to make precise measurements. In this thesis, we adopt the traditional photogrammetric principle of camera calibration prior to measurements. One reason for this is the peculiarity of the fish-eye lens and the other is the requirement of high accuracy. Although we use the calibrated approach, we also utilise novel geometric concepts, such as the multiple view tensors, originating from a more general uncalibrated framework.

The current state-of-the-art vision systems aim at fully automatic 3D model

acquisition from uncalibrated image sequences allowing even zooming of the cameras [Fitzgibbon98a, Pollefeys99, Johansson01]. Unlike some other 3D modelling systems they do not require any special equipment, such as laser radars [Leica] or structured light [ShapeCam], or manual extraction of feature correspondences [PhotoModeler]. The scope of these systems is wide since only general assumptions are made, e.g. rigid scene, pinhole camera, piecewise continuous and sufficiently textured surfaces. The techniques have also matured at the stage where first commercial products have entered the market. For example, the automatic camera tracker *boujou* [2d3], which is based on the research work done at the University of Oxford [VGG], is used by the film industry to compute camera motion from video sequences. The 3D animators need this information in adding special effects to a live-action background.

The structure from motion problem can be divided into several, more manageable subproblems. A typical automatic scene reconstruction system consists of sub-modules addressing these different subproblems. In the following, we describe the subproblems and give references where they are discussed in more detail. After that we briefly consider the implementation of our system from the viewpoint of the sub-modules.

**Feature extraction and matching** The first problem is to obtain the initial feature correspondences between successive images in the sequence. In general, matching is a difficult problem but it is simplified if one may assume a short baseline between the images. In this case, one may usually match features through *intensity cross-correlation* since the intensity neighbourhoods of corresponding features are similar in both images [Xu96]. If the appearance of a feature changes between the views, the pure translation model for the feature neighbourhoods is not adequate and one may allow affine changes of the feature windows as in [Shi94]. The matching algorithm in [Shi94] is based on the Lucas-Kanade tracker [Lucas81].

Commonly the matched features are points or lines because they are simplest to handle in later stages, in fact, concurrency and collinearity are invariant to planar projective transformations while more complex geometric primitives are not. Often the features are interest points that are extracted by the Harris corner detector [Harris88]. In the sequel, we mainly assume that the feature correspondences are points.

**Feature tracking** The aim in this stage is to track features across several views. Although it is possible to obtain a reconstruction from just two views, the accuracy of the estimated 3D structure may be poor if the distance between camera centres, the baseline, is very small. Tracking across several views usually results to a larger effective baseline and thereby to a better reconstruction. Simultaneous robust estimation of camera motion helps also to discard the wrongly matched features.

The point correspondences obtained in the initial matching stage contain almost unavoidably some false matches. This is because the local neighbourhoods of *different* interest points may look similar. However, most of these false matches do not satisfy the geometric constraints between multiple views of a rigid scene. For two views this constraint is called the epipolar constraint and it is imposed by the fundamental matrix which is the multiple view tensor in the two-view case. Hence, estimating the fundamental matrix from the putative correspondences by some robust method allows one to discard those

matches that do not satisfy the epipolar constraint. The fundamental matrix was first introduced by Faugeras [Faugeras92b] and Hartley [Hartley92] but its counterpart for calibrated cameras, the essential matrix, appeared already in [Longuet-Higgins81].

In the three-view case the multiple view tensor is called the trifocal tensor and it imposes all the effective constraints for correspondences across three views [Hartley00]. Thus, by robustly computing the trifocal tensor for each successive image triplet one may further discard false matches and obtain correspondences practically free from false ones. The estimated three-view geometry may also be used to obtain additional matches by lowering similarity threshold for those putative correspondences which fit well to the geometry [Beardsley96].

There are several robust estimation methods that may be used in computing the multiple view tensors. Most common are RANSAC (Random Sample Consensus) [Fischler81] and LMedS (Least Median of Squares) [Rousseeuw87] and their different variants [Torr00, Zhang95]. A novel approach is the MLRE (Maximum Likelihood Robust Estimator) [Brandt02].

**Uncalibrated structure from motion** After the point correspondences are established over the successive triplets of views one must solve the structure and camera motion for the entire sequence. The optimal way is to compute the camera matrices and the 3D points in such a way that the sum of squared distances between projected and measured points is minimised. This is a nonlinear optimisation problem known as bundle adjustment and it requires a good initial guess. In the special case of affine cameras, there is a noniterative factorisation algorithm for optimal reconstruction [Tomasi92], and an iterative modification of it for situations where all the point correspondences are not visible in all the views [Brandt02]. Similar factorisation based methods have been proposed for general projective reconstruction [Sturm96, Heyden97a, Martinec02], and they can be used to compute an initial guess for the final bundle adjustment. Another approach is to proceed from triplets towards final reconstruction by hierarchical merging and bundle adjustment of sub-sequences [Fitzgibbon98b].

**Auto-calibration** The process of determining internal camera parameters directly from a sequence of images acquired by an unknown camera undergoing unknown movement is called auto-calibration. With known internal camera parameters it is possible to upgrade the projective reconstruction to metric. The first auto-calibration methods assumed constant internal parameters, i.e., the whole sequence is taken by the same camera with fixed focal length and focus [Faugeras92a, Heyden96]. However, auto-calibration is possible also under less restrictive constraints [Heyden97b, Pollefeys98]. For example, Pollefeys *et al* [Pollefeys98] showed that the assumption of zero-skew (i.e. orthogonal pixel coordinate system) alone is sufficient for auto-calibration. Pollefeys also experimented different auto-calibration techniques for zooming cameras [Pollefeys99].

**Dense stereo matching** The tracked features are scattered around the scene and typically there are by far too few of them to obtain a dense reconstruction. Small details and even some important scene features may be missing from the reconstruction even if some kind of interpolation between the feature points is used. One solution to this problem is the dense stereo matching which is possible *after* solving the motion and calibration of the camera. The stereo matching algorithms have been developed for calibrated stereo rigs and they utilise the

known epipolar constraint to reduce the search space of correspondences to one dimension (along the epipolar lines). Pollefeys uses a variant of Cox's stereo algorithm [Cox96] to compute dense depth maps between adjacent image pairs and then fuses these maps together in order to reduce uncertainty and to detect outliers [Pollefeys99].

**Model building** Building the final, texture-mapped model is the last stage in automatic scene reconstruction system. The dense or sparse depth map must be approximated by a 3D surface model for visualisation. A simple way to do this is to generate a triangular wire-frame model by performing a 2D Delaynay triangulation in one of the images (e.g. the middle image in the sequence) and projecting this into 3D, but also more sophisticated methods exist [Johansson01, Morris00]. Finally the texture extracted from the images is mapped on the facets of the mesh.

Above we have described the modules of a general purpose system for automatic scene reconstruction from image sequences. Depending on the requirements of different applications, all modules might not be necessary. Our approach to the sewer reconstruction problem follows the general framework above, but there are certain differences too. The first difference is the precalibration of the camera that will be described in Chapters 3 and 4. Thus, in the tracking stage we compute calibrated multiple view tensors and the reconstruction obtained is directly metric. Naturally, the auto-calibration stage is skipped. Furthermore, the pipe may be directly modelled by fitting a cylindrical surface to the 3D points and dense matching is not needed.

## 2.2 Sewer Survey

Sewerage systems are an important part of modern infrastructure and their proper functioning is essential. However, in many countries sewer networks are deteriorating due to their high age [Kuntze98, Cooper98, Chae01]. Deteriorated sewer systems are threatening to contaminate ground water and soil, in addition to causing traffic disruptions and loss of property. Since the restoration and maintenance of sewer systems require huge investments, a great effort has been put in developing new pipe inspection methods. In the following section, we review different equipments and methods suggested for sewer pipe inspection. After that, we focus on the DigiSewer-system which was used to scan our test videos.

### 2.2.1 Review of Equipments and Methods

Traditional and widely used technique for sewer pipeline inspection is the closed-circuit television (CCTV) survey. The data acquired from this kind of survey consists of a videotape, photographs of specific defects and a record produced by a technician. During the inspection, the CCTV camera provides a real time frontal view of the pipe. When the operator notices defects, he is able to turn the camera head and take a closer look. Diagnosis of defects depends heavily on the experience of the operator which makes the evaluation error prone. Later access to video images of particular sections of pipe is also difficult and time consuming.

Several approaches for automation of sewer surveys have been suggested. Xu, Luxmoore and Davies [Xu98] investigated video images of clay and concrete pipes and observed that the structural changes in pipes are associated with diametric changes of their mortar joints. They proposed a method for automatic detection of pipe-joints and their shape analysis. Ruiz-del-Solar and Köppen concentrated on automatic detection of pipe sockets [Ruiz-del-Solar96]. The limitation of these approaches is that they ignore the parts of pipes between sockets, so their practical applicability is a bit unclear. Cooper, Pridmore and Taylor presented an idea of a system recovering the three-dimensional shape of a surveyed pipe from survey videos [Cooper98]. They also recovered the pose of the camera relative to the central axis of the pipe, but unfortunately their method is restricted to brick sewers with visible mortar lines [Cooper01].

There are also multisensoric pipe inspection systems, where the sewer robot is equipped with additional sensors besides the camera. For example, the robot by Gooch, Clarke and Ellis [Gooch96] has a cylindrically scanning range camera employing a laser based optical triangulation scheme. The range camera measures the cross-sectional shape of the pipe while the robot moves forward. The range measurements are reported to have a very good accuracy, varying between 60 and 600 $\mu$m depending on the diameter of the pipe.

Another multisensor sewer robot is the German KARO [Kuntze98] which, in addition to the usual video camera, has an 3D-optical, ultrasonic and microwave sensors. The 3D-optical sensor is based on optical triangulation and it consists of a circular pattern projected onto the pipe wall and a high resolution infrared camera. The sensor is used to measure pipe deformations. The ultrasonic sensors measure the pipe wall thickness as well as coarse cracks and deformations. The microwave transmitters and receivers are used to observe damages (e.g. water leakages) behind the pipe wall. The many sensors of multisensoric robots provide much information, but they naturally also lead to a more complex and expensive construction.

Researchers at Fraunhofer Institute in Germany [Fraunhofer AIS] have additionally developed autonomous sewer robots, which have batteries and manage without a cable. An on-board processor evaluates the data acquired by different sensors and decides on operations needed for execution of a given mission. Autonomous robots open up many interesting problems from autonomous navigation and motion control to power saving [Hertzberg96, Kolesnik02]. Nevertheless, this kind of robots are not yet ready for extensive use in sewer measurements.

### 2.2.2 DigiSewer-system

This thesis is a part of a larger project that aims at developing automatic image analysis methods for digital sewer images. The project is done in co-operation with VTT Building and Transport and corporate partner is Painehuuhtelu Oy PTV. A platform for the development work is the DigiSewer-system, which was also used to scan the test videos. DigiSewer name is registered by Painehuuhtelu Oy PTV but the manufacturer of the measurement equipment is OYO Corporation from Japan. In the USA, similar scanning technology is marketed as SSET (Sewer Scanner and Evaluation Technology) by Blackhawk-PAS Inc. which is a subsidiary of OYO.

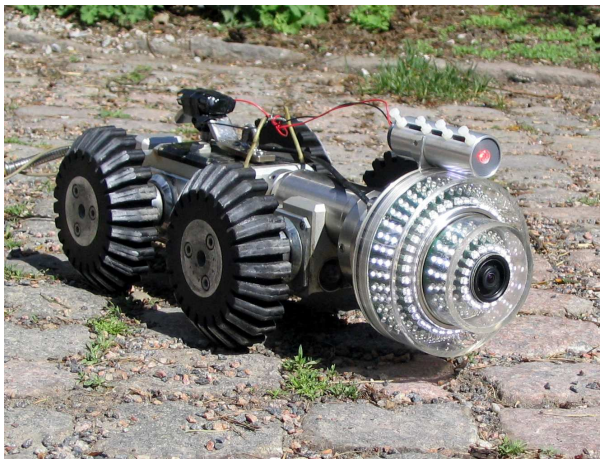


**Figure 2.1:** DigiSewer robot equipped with an additional laser unit (the cylinder with red light on top of the LEDs).

A special feature of the DigiSewer-equipment is the fish-eye lens, which provides a hemispherical frontal view of the pipe. The fish-eye lens camera and the illuminating LEDs are installed onto a separate probe, which can be attached to different sized tractors depending on the diameter of the pipe. The probe is also equipped with a precision dual-axis inclinometer. The probe does not contain any moving parts, like rotating mirrors, cameras etc., that would require constant maintenance. The robot shown in Figure 2.1 is a prototype version with an additional laser unit (see Section 2.2.3) on top of the probe. Some versions of the robot also have a gyroscope which is used together with the inclinometer to map horizontal and vertical profiles of pipes. Without the gyroscope only the vertical or slope profile is computed.

The scanning methodology of the DigiSewer-system is illustrated in Fig. 2.2 and 2.3. The robot moves at a constant speed in the pipe and the video camera captures the hemispherical frontal view. A single frame from the video is shown in Fig. 2.2. The portion of the moving image that passes through the annular zone indicated by the yellow dashed line gets digitally scanned in. The ring-shaped image zones are cut off, flattened and concatenated to form the unfolded image shown in Fig. 2.3. The zones are always cut off at the lowest point, which is determined by the inclinometer. Hence, the top of the pipe is always on the centre line of the unfolded image, even when the robot leans. The concatenation is done with an electronic distance counter which measures the speed of the robot from the cable. The maximum scanning speed is about 4 m/min.

The digital side scan image is convenient in sewer surveys for several reasons. First, one is able to gain a quick overview of long pipe sections by having a look at the concatenated side scan images. Secondly, the compressed digital images can be easily stored and archived. The side scan image is also a good starting point for the development of automatic condition assessment methods. For example, Pantsar [Pantsar00] and Chae and Abraham [Chae01] have investigated methods for automatic detection and classification of cracks and pipe joints. This is also another topic in our ongoing research project.

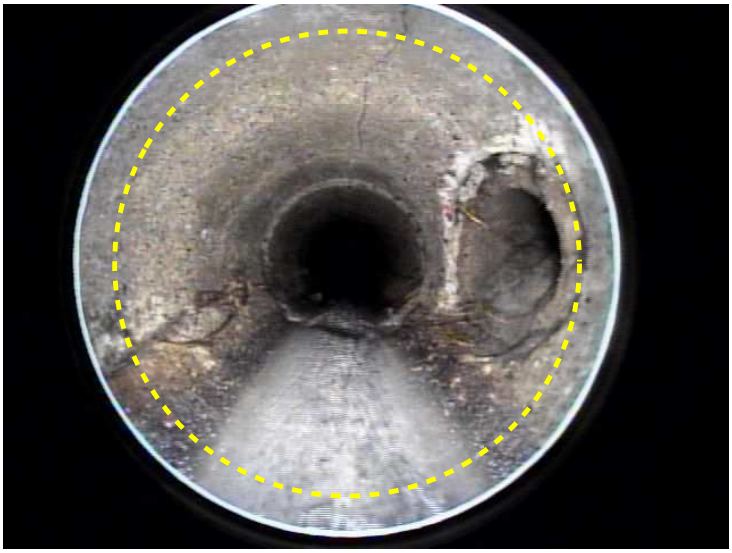


**Figure 2.2:** A hemispherical frontal view of sewer pipe. The annular zone under the yellow dashed line is the scan area for the side scan image.

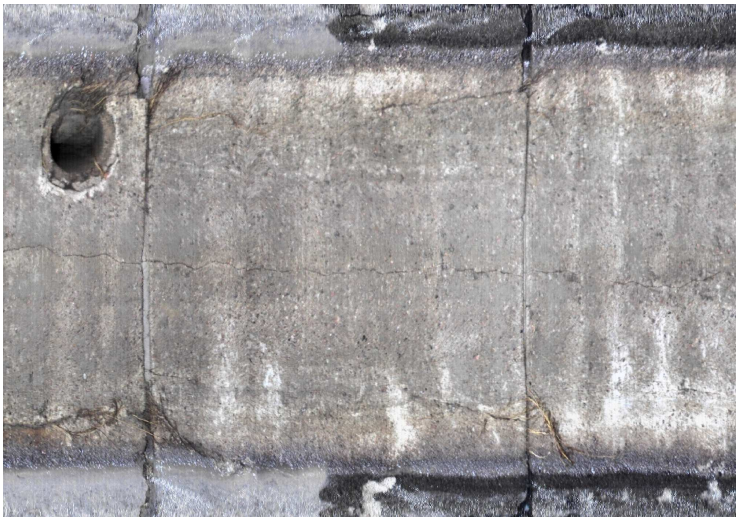


**Figure 2.3:** Unfolded side scan image of the pipe. The roof of the pipe is in the middle. The lateral joint shown on the right in Fig. 2.2 is here in the upper left corner.

### 2.2.3 Prototype with Structured Light

Incorporating shape measurements to the DigiSewer-system is an important field of development. Although the slope profile gives some information of the longitudinal shape of the pipe, the cross-sectional shape is mostly unknown. In the following chapters, we consider solutions for obtaining a three-dimensional reconstruction of the pipe which is the contribution of this thesis. However, if there is not any significant texture on the inner wall of the pipe, the structure from motion approach does not work. Therefore we have also considered alternative approaches. Here we discuss briefly the possibility of using structured light to determine the shape of the pipe.

A prototype version of the robot was equipped with a laser light source and a beam splitter. The beam splitter was designed in the University of Joensuu and fabricated by electron beam lithography. The resulting light pattern after beam splitting is illustrated in Fig. 2.4. The pattern consists of concentric circles each containing eight principal rays of high intensity. Since it is possible to precalibrate the system and find out the direction of each ray with respect to the optical axis of the camera, the 3D position of lighted dots (with respect to camera) is solved from a single image. Thus, if one assumes that the optical axis is collinear with the central axis of the pipe one may estimate the cross-sectional shape by fitting an ellipse to the eight lighted dots.

However, the schema above is not yet implemented and some practical problems also exists. For example, in the situation of Fig. 2.5 one laser dot is displaced due to the lateral joint. The elliptical shape model is an approximation, which is needed due to the small number of points (eight). In this respect a projected circular pattern, as in KARO, would be better. The localisation of the laser dots from the images is a subject of future research too.

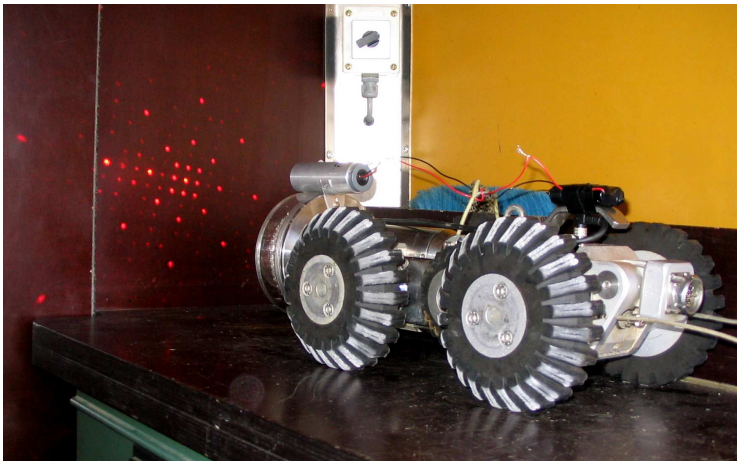


**Figure 2.4:** The pattern formed by the splitted laser beam.

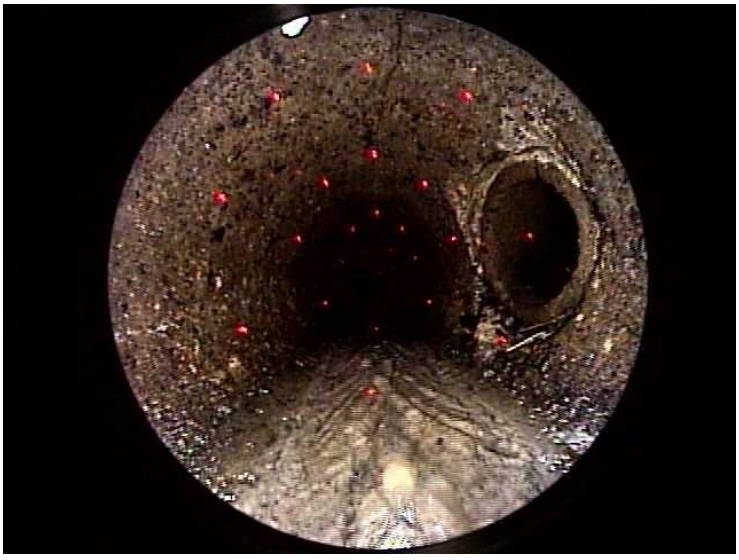


**Figure 2.5:** The laser dots on the inner wall of a sewer pipe.

# Chapter 3

# Camera Models and Calibration

The purpose of this chapter is to propose a generic camera model for cameras equipped with fish-eye lenses and a method for calibration of such cameras.[1] Moreover, it will be shown that the proposed camera model is also valid for conventional cameras with narrow-angle lenses. In this chapter, we describe the generic camera model and the calibration procedure, but the related experimental results are postponed to Chapter 4. But first, we begin by introducing the conventional pinhole camera model and show its limitations.

## 3.1 Pinhole Model

Traditional film cameras as well as modern CCD cameras are usually modelled with the pinhole camera model, which is just a perspective projection followed by an affine transformation in the image plane. The pinhole camera geometry is illustrated in Fig. 3.1. The centre of projection is called the *camera centre*, $\mathbf{C}$, and its distance from the image plane is the *focal length*, $f$. By similar triangles, it may be seen from Fig. 3.1 that the point $(X_c, Y_c, Z_c)^\top$ in camera coordinate frame is projected to the point $(fX_c/Z_c, fY_c/Z_c)^\top$ in normalised image coordinate frame. In terms of homogeneous coordinates, see Appendix A, this perspective projection is expressed by a $3 \times 4$ homogeneous projection matrix,

$$\mathbf{x} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{X}_c \ .$$

However, instead of the normalised image coordinates $(x, y)^\top$ one usually uses pixel coordinates $(u, v)^\top$ which are obtained by the affine transformation

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{bmatrix} m_u & s \\ 0 & m_v \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} u_0 \\ v_0 \end{pmatrix}, \tag{3.1}$$

where $(u_0, v_0)^\top$ is the principal point and $m_u$ and $m_v$ give the number of pixels

[1] The content of this and the following chapter is partly published in [Kannala04].

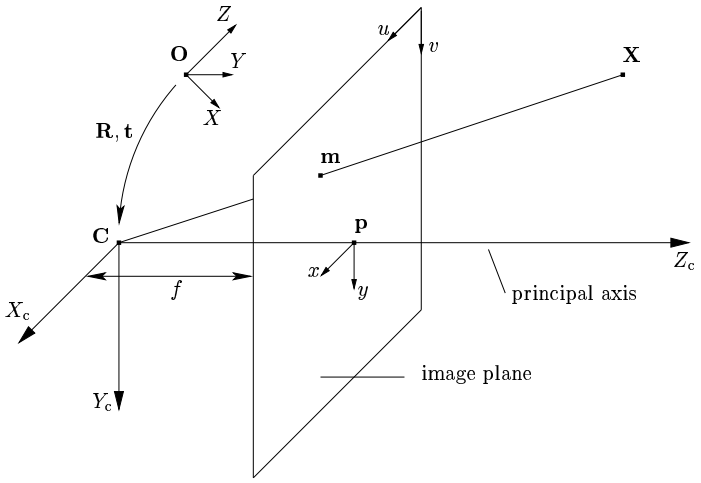



**Figure 3.1:** Pinhole camera model. **C** is the camera centre and the origin of the camera coordinate frame. The principal point **p** is the origin of the normalised image coordinate system $(x, y)$. The pixel image coordinate system is $(u, v)$. Sometimes the pinhole model is illustrated by placing the image plane behind the camera centre, but the resulting model is the same (cf. Fig. 3.3(b)).

per unit distance in $u$ and $v$ directions, respectively. The skew parameter $s$ is zero in the conventional case of orthogonal pixel coordinate axes.

In general, points in space are not expressed in the camera coordinate frame but in a different Euclidean coordinate frame, known as the *world coordinate frame*. The representations of points in the two coordinate frames are related via a rotation and a translation:

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = \mathbf{R} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \mathbf{t}. \tag{3.2}$$

Now, from the previous equations, one obtains the relation between the homogeneous world point $\mathbf{X}$ and its image $\mathbf{m}$ in pixel coordinates, i.e.,

$$\begin{aligned} \mathbf{m} &= \begin{bmatrix} m_u & s & u_0 \\ 0 & m_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \mathbf{X} \\ &= \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \mathbf{X} \\ &= \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \mathbf{X}, \end{aligned} \tag{3.3}$$

where products $fm_u$ and $fm_v$ have been replaced by $\alpha_u$ and $\alpha_v$ because a change in the focal length and a change in the pixel units are indistinguishable. The upper triangular matrix $\mathbf{K}$ is the *camera calibration matrix* and contains the *internal camera parameters*. The rotation and translation parameters, $\mathbf{R}$ and $\mathbf{t}$, are called the *external camera parameters*.

It follows from (3.3) that a general pinhole camera may be represented by a

homogeneous $3 \times 4$ matrix

$$\mathbf{P} = \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \tag{3.4}$$

which is called the *camera projection matrix*. If the left hand submatrix $\mathbf{KR}$ is non-singular, the camera $\mathbf{P}$ is called a *finite projective camera*. [2] A camera represented by an *arbitrary* homogeneous $3 \times 4$ matrix of rank 3 is called a *general projective camera*.

### 3.1.1 Calibration

Calibration of a pinhole camera refers to the determination of the calibration matrix $\mathbf{K}$. Calibration is possible by viewing a calibration object which contains control points in known positions. In the following, we describe two ways to determine $\mathbf{K}$, the first one is via the projection matrix $\mathbf{P}$ and the second is a direct way.

If the projection matrix $\mathbf{P}$ (3.4) is given, the calibration matrix $\mathbf{K}$ may be extracted from its left hand $3 \times 3$ submatrix by the RQ-decomposition. There is a linear algorithm for computation of the camera projection matrix from at least six correspondences $\mathbf{m}^i \leftrightarrow \mathbf{X}^i$ in general position [Hartley00]. The points $\mathbf{X}^i$ are not allowed to be coplanar, hence, a three-dimensional calibration object is required. If the measurement errors are Gaussian, the estimate of $\mathbf{P}$ given by the linear algorithm should be refined by minimising the sum of squared distances between the measured and projected control points, i.e.,

$$\sum_i d(\mathbf{m}^i, \mathbf{P}\mathbf{X}^i)^2 \ . \tag{3.5}$$

This is a non-linear minimisation problem with respect to the 12 parameters of $\mathbf{P}$ and requires iterative techniques, such as Levenberg-Marquardt.

The direct method for solving $\mathbf{K}$ [Sturm99, Zhang00] is based on viewing a planar calibration object at different orientations. The mapping between a scene plane and its perspective image is a planar homography. Since one may assume that the calibration plane is the plane $Z = 0$, the homography is derived as follows:

$$\mathbf{m} = \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} = \mathbf{H} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}, \tag{3.6}$$

where the $3 \times 3$ homography

$$\mathbf{H} = \mathbf{K} \begin{bmatrix} \mathbf{r}^1 & \mathbf{r}^2 & \mathbf{t} \end{bmatrix} \tag{3.7}$$

is formed by dropping the last column of the rotation matrix. The outline of the calibration method is to first determine the homographies for each view and then use (3.7) to derive constraints for determination of $\mathbf{K}$. In the following, we describe these constraints in more detail. Methods for determining a homography from point correspondences are described in [Hartley00], for example.

[2] Every non-singular square matrix has a unique decomposition into a product of an upper-triangular and orthogonal matrix where the diagonal elements of the upper-triangular matrix are positive (RQ-decomposition, see e.g. [Hartley00]).

Denoting the columns of $\mathbf{H}$ by $\mathbf{h}^i$ and using the knowledge that $\mathbf{r}^1$ and $\mathbf{r}^2$ are orthonormal one obtains from (3.7) that

$$\mathbf{h}^{1\top}\mathbf{K}^{-\top}\mathbf{K}^{-1}\mathbf{h}^2 = 0, \tag{3.8}$$

$$\mathbf{h}^{1\top}\mathbf{K}^{-\top}\mathbf{K}^{-1}\mathbf{h}^1 = \mathbf{h}^{2\top}\mathbf{K}^{-\top}\mathbf{K}^{-1}\mathbf{h}^2. \tag{3.9}$$

Thus, each homography provides two constraints on the intrinsic parameters, and the constraints above may be written as linear equations on the elements of the homogeneous *symmetric* matrix $\boldsymbol{\omega} = \mathbf{K}^{-\top}\mathbf{K}^{-1}$. The equation system is of the form $\mathbf{A}\mathbf{v} = \mathbf{0}$, where the vector of unknowns $\mathbf{v} = (\omega_{11}, \omega_{12}, \omega_{13}, \omega_{22}, \omega_{23}, \omega_{33})^\top$. Matrix $\mathbf{A}$ has $2N$ rows, where $N$ is the number of views. Given three or more views, the solution vector $\mathbf{v}$ is the right singular vector of $\mathbf{A}$ corresponding to the smallest singular value. Under the general skew-zero assumption two views are enough (see [Sturm99]). When $\boldsymbol{\omega}$ is solved (up to scale) one may compute the calibration matrix $\mathbf{K}$ by Cholesky-factorisation [Golub96]. Still, the solution should be refined by minimising error (3.5) in all views. The external parameters for the projection matrices may be retrieved from (3.7) given $\mathbf{H}$ and $\mathbf{K}$.

### 3.1.2 Lens Distortion

The above calibration techniques assume that the pinhole model is an accurate model of the imaging process. This is true for most long focal length lenses of hiqh quality. However, when the focal length and price of the lens decrease, the deviations from the pinhole model increase. The most important deviation is radial distortion which causes an inward or outward displacement of a given image point from its ideal location, as Fig. 3.2 illustrates. Decentering of lens elements causes additional distortion that has also tangential components.

A commonly used approach for correcting lens distortion contains models for radial and decentering distortion [Brown71, Slama80]. The corrected coor-

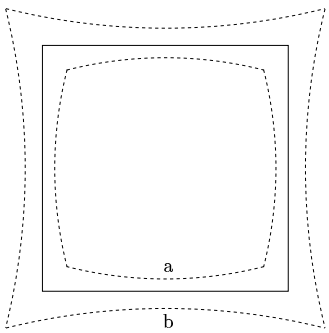



**Figure 3.2:** Effect of radial distortion on the image of a square. a) barrel distortion, b) pincushion distortion.

dinates $x'$,$y'$ are obtained from

$$\begin{aligned} x' &= x + \bar{x}\left(K_1 r^2 + K_2 r^4 + K_3 r^6 + \ldots\right) \\ &\quad + \left(P_1(r^2 + 2\bar{x}^2) + 2P_2\bar{x}\bar{y}\right)\left(1 + P_3 r^2 + \ldots\right) \\ y' &= y + \bar{y}\left(K_1 r^2 + K_2 r^4 + K_3 r^6 + \ldots\right) \\ &\quad + \left(2P_1\bar{x}\bar{y} + P_2(r^2 + 2\bar{y}^2)\right)\left(1 + P_3 r^2 + \ldots\right), \end{aligned} \tag{3.10}$$

where $x$ and $y$ are the measured coordinates, and

$$\begin{aligned} \bar{x} &= x - x_p \\ \bar{y} &= y - y_p \\ r &= \sqrt{(x - x_p)^2 + (y - y_p)^2}. \end{aligned}$$

The centre of distortion $(x_p, y_p)$ is also a free parameter in addition to the radial distortion coefficients $K_i$ and decentering distortion coefficients $P_i$. The values for the distortion parameters are computed by least-squares adjustment by requiring that images of straight lines are straight [Brown71]. The problem with the formulation above is that not only the distortion coefficients but also the other camera parameters are normally unknown.The formulation (3.10) requires that the scales in both coordinate directions are equal that is not the case with pixel coordinates unless the pixels are square. In addition, (3.10) corrects the *noisy* measurements, which may deteriorate the calibration result.

To cope with the problems above, slightly different models have been proposed [Zhang98, Heikkilä00b]. They are of the form

$$\mathbf{m} = \mathcal{P}(\mathbf{X}) = \mathbf{PX} + \mathcal{D}(\mathbf{PX}), \tag{3.11}$$

where $\mathcal{P}$ is a general imaging function of the camera. $\mathcal{D}$ is some nonlinear distortion function whose parameters are estimated together with the other camera parameters by minimising the error

$$\sum_{j=1}^{N}\sum_{i=1}^{M} d(\mathbf{m}_j^i, \mathcal{P}_j(\mathbf{X}^i))^2, \tag{3.12}$$

where $N$ is the number of views and $M$ is the number of control points. In [Zhang98], only radial distortion is modelled but the model in [Heikkilä00b] also contains decentering distortion coefficients, as derived from (3.10).

## 3.2 Generic Model

### 3.2.1 Fish-Eye Lenses

Although the pinhole model accompanied with lens distortion models is a fair approximation for most conventional cameras, it is not suitable for fish-eye lens cameras. The fish-eye lens is designed to cover the whole hemispherical field in front of the camera, hence, the angle of view is very large, about 180°. Because it is impossible to project the hemispherical field of view on a finite image plane by a perspective projection, fish-eye lenses are designed to obey some other

projection model. Therefore the inherent distortion of a fish-eye lens should not be considered only as a deviation from the pinhole model [Miyamoto64].

There have been some efforts to model the radially symmetric distortion of fish-eye lenses with different models [Basu95, Devernay01, Bräuer-Burchardt01]. The idea of these approaches is to transform the original fish-eye image to follow the pinhole model. In [Devernay01] and [Bräuer-Burchardt01], the parameters of the distortion model are estimated by forcing that straight lines are straight after the transformation but the problem is that the methods do not give the full calibration. They can be used to "correct" the images to follow the pinhole model but their applicability is limited when one needs to know the direction of a back-projected ray corresponding to an image point. The calibration procedures in [Shah96] and [Bakstein02] instead aim at calibrating fish-eye lenses generally. However, their methods are slightly cumbersome in practise because a laser beam or a cylindrical calibration object is required.

In the following two sections, we describe a general camera model which is also valid for fish-eye lens cameras, and in Section 3.3, we propose methods for estimating the parameters of the model.

### 3.2.2 Radially Symmetric Model

The perspective projection of a pinhole camera can be described by the following formula

$$r = f \tan\theta \qquad \text{(i. perspective projection)}, \tag{3.13}$$

where $\theta$ is the angle between the principal axis and the incoming ray, $r$ is the distance between the image point and the principal point and $f$ is the focal length. Fish-eye lenses instead are usually designed to obey one of the following projections:

$$r = 2f \tan(\theta/2) \qquad \text{(ii. stereographic projection)}, \tag{3.14}$$

$$r = f\theta \qquad \text{(iii. equidistance projection)}, \tag{3.15}$$

$$r = 2f \sin(\theta/2) \qquad \text{(iv. equisolid angle projection)}. \tag{3.16}$$

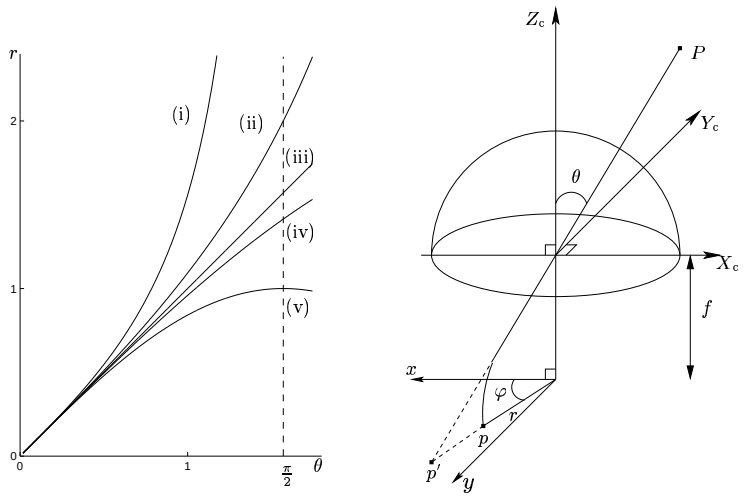
Perhaps the most common model is the equidistance projection. Sometimes lenses obeying orthogonal projection,

$$r = f \sin\theta \qquad \text{(v. orthogonal projection)}, \tag{3.17}$$

are also categorized as fish-eye lenses. However, here we decided to treat orthogonal projection separately since the principal point is not well defined for such lenses.The behaviour of the different projections is illustrated in Fig. 3.3(a) and the difference between a pinhole camera and a fish-eye camera is shown in Fig. 3.3(b). Evidently, all the above models are radially symmetric, though the centre of symmetry is not unique for orthogonal projection.

The real lenses do not exactly follow the designed projection model. From the viewpoint of automatic calibration, it would also be useful if we had only one model for different types of lenses. Therefore we consider projections in the general form

$$r(\theta) = k_1\theta + k_2\theta^3 + k_3\theta^5 + k_4\theta^7 + \dots \quad , \tag{3.18}$$




(a) (b)

**Figure 3.3:** (a) Curves of projections (3.13)-(3.17) with $f = 1$. (b) Fish-eye camera model. The image of the point $P$ is $p$ whereas it would be $p'$ by a pinhole camera.

where, without any loss of generality, even powers have been dropped. This is due that we may extend $r$ onto the negative side as an odd function while the odd powers span the set of continuous odd functions.

For computations we need to fix the number of terms in (3.18). An important property for a projection model is that it can be analytically inverted. Therefore we choose

$$r(\theta) = k_1\theta + k_2\theta^3 \tag{3.19}$$

as the basic model. When modelling real lenses, the values of parameters $k_1$ and $k_2$ will be such that $r(\theta)$ is monotonically increasing on the interval $[0, \pi/2]$. Therefore we may solve $\theta$ from (3.19) if $r$ is given: from the three possible roots to a cubic equation we choose a real root that is between 0 and $\pi/2$. Although the model (3.19) contains only two parameters, it can approximate all the projections (3.14)-(3.17) on the interval $[0, \pi/2]$ with a moderate level of accuracy. The difference would be hardly distinguishable if the approximation nearest in the $L^2$-norm would be plotted to Fig. 3.3(a) for each projection. Moreover, also the perspective projection (3.13) can be approximated with (3.19) when $\theta$ is notably less than $\pi/2$, as it is for normal lenses.

To achieve a complete camera model we need to transform the camera coordinates $(X_c, Y_c, Z_c)^\top \hat{=} (\rho, \varphi, \theta)^\top$ into the image pixel coordinates. As an intermediate, step we compute the normalised image coordinates (see Fig. 3.3(b))

$$\begin{pmatrix} x \\ y \end{pmatrix} = r(\theta) \begin{pmatrix} \cos\varphi \\ \sin\varphi \end{pmatrix}, \tag{3.20}$$

where $r(\theta)$ is obtained from (3.19). By assuming that the pixel coordinate

system is orthogonal we get the pixel coordinates $(u, v)^\top$ from

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{bmatrix} m_u & 0 \\ 0 & m_v \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} u_0 \\ v_0 \end{pmatrix}, \tag{3.21}$$

where $(u_0, v_0)^\top$ is the principal point and $m_u$ and $m_v$ give the number of pixels per unit distance in horizontal and vertical directions, respectively.

### 3.2.3 Extended Model with Distortion

The lens elements of a real fish-eye lens may deviate from precise radial symmetry and they may be inaccurately positioned causing that the projection is not exactly radially symmetric. We hence propose adding two distortion terms: one acting in the radial direction

$$\begin{aligned} \Delta_r(\theta, \varphi) = (l_1\theta + l_2\theta^3 + l_3\theta^5)(i_1 \cos\varphi + \\ i_2 \sin\varphi + i_3 \cos 2\varphi + i_4 \sin 2\varphi) \ , \end{aligned} \tag{3.22}$$

and the other in the tangential direction

$$\begin{aligned} \Delta_t(\theta, \varphi) = (m_1\theta + m_2\theta^3 + m_3\theta^5)(j_1 \cos\varphi + \\ j_2 \sin\varphi + j_3 \cos 2\varphi + j_4 \sin 2\varphi) \ . \end{aligned} \tag{3.23}$$

The distortion functions are thus separable in the variables $\theta$ and $\varphi$. Because the Fourier series of any $2\pi$-periodic continuous function converges in $L^2$-norm and any continuous odd function can be represented by a series of odd polynomials we can, in principle, model any kind of continuous distortion by simply adding more terms to (3.22) and (3.23).

With the distortion parameters we get the following formula for the normalised image coordinates

$$\begin{pmatrix} x \\ y \end{pmatrix} = (r(\theta) + \Delta_r(\theta, \varphi))\mathbf{u}_r(\varphi) + \Delta_t(\theta, \varphi)\mathbf{u}_\varphi(\varphi), \tag{3.24}$$

where $\mathbf{u}_r(\varphi)$ and $\mathbf{u}_\varphi(\varphi)$ are the unit vectors in the radial and tangential directions. Since the analytical invertibility of the model (3.20) is anyway lost in (3.24), we may also take more parameters to the radially symmetric part $r(\theta)$. Using the model (3.18) with terms up to the ninth order increases the total number of parameters to 23.

Compared to the lens distortion models in Section 3.1.2 our approach has a different philosophy. Instead of modelling different physical phenomena in the optical system we have a flexible mathematical model that is only fitted to agree with the observations. Since there are several possible sources of imperfections in the optical system we think it is not meaningful to build a separate model for all of them. For example, in addition to decentering of lenses, the image plane and individual lens elements may be tilted with respect to the principal axis. Another difference between the philosophies is that in our approach radial distortion is not considered to be a distortion at all but a feature indispensable in wide-angle imaging.

## 3.3 Calibrating the Generic Model

The basic camera model introduced in Section 3.2.2 contains the six internal camera parameters $k_1, k_2, m_u, m_v, u_0$ and $v_0$. If the extended model is used the total number of internal parameters is 23. When the external parameters $\mathbf{R}$ and $\mathbf{t}$ in (3.2) are added, the full camera model is expressed as a nonlinear imaging function $\mathcal{P}$, $\mathbf{m} = \mathcal{P}(\mathbf{X})$, defined by equations (3.2), (3.20) or (3.24) and (3.21). Depending on the projection type of the camera, we propose different methods for estimating the parameters of the model.

### 3.3.1 Projective Cameras

When calibrating perspective cameras (i.e. finite projective cameras), it is possible to use the extended model of Section 3.2.3 as an alternative to the usual distortion models complementing the pinhole model. The calibration methods of Section 3.1.1 may be used to compute an initial calibration that is used to initialise the parameters of our nonlinear model. The parameters $m_u, m_v, u_0$ and $v_0$ are directly extracted from the calibration matrix $\mathbf{K}$ (assume that $f = 1$ and set $m_u = \alpha_u$, $m_v = \alpha_v$). The coefficients $k_i$ are initialised so that (3.18) approximates (3.13) (with $f = 1$) on some suitable interval $[0, \theta_{\max}]$. The coefficients of the asymmetric distortion terms (3.22) and (3.23) may be initialised to zero. After the initialisation, the parameter values are again refined by minimising (3.12).

If the camera projection is orthographic, it is not a finite projective camera but an *affine* camera which is a special case of general projective camera and has only three internal parameters [Quan96]. Affine camera is a simplified projection model which is mainly used as an approximation to the perspective camera when the depth of an object is small compared to the viewing distance. However, sometimes the effects of lens distortion may be more significant than the perspective effects ignored in the affine approximation. Hence, the camera models of Section 3.2 might be usable even for cameras close to affine. But in this case, of course, some of the simplicity of the affine camera model would be lost.

### 3.3.2 Fish-Eye Lens Cameras

Next we present a method for calibration of fish-eye lens cameras with the generic camera model. The camera projection is assumed to be approximately either a stereographic projection, an equidistance projection or an equisolid angle projection. The calibration method is based on viewing a calibration plane which contains control points in known positions. Only one view of the plane is sufficient for calibration but more views should be used for better results.

The calibration procedure consists of four steps that are described below. We assume that $M$ control points are observed in $N$ views. For each view, there is a rotation matrix $\mathbf{R}_j$ and a translation vector $\mathbf{t}_j$ describing the position of the camera with respect to the calibration plane. We choose the calibration plane to lie in the $XY$-plane and denote the coordinates of control point $i$ with $\mathbf{X}^i = (X^i, Y^i, 0)^\top$. The corresponding homogeneous coordinates in the calibration plane are denoted by $\mathbf{x}_{\mathrm{p}}^i = (X^i, Y^i, 1)^\top$ and the observed coordinates in view $j$ by $\mathbf{m}_j^i = (u_j^i, v_j^i)^\top$. The first three steps of the calibration procedure

involve only the basic model and its six internal parameters. For these internal parameters we use a short-hand notation $\mathbf{p} = (k_1, k_2, m_u, m_v, u_0, v_0)$. If the extended model is used the additional parameters are inserted only in the final step.

**Step 1: Initialisation of internal parameters**
The initial guesses for $k_1$ and $k_2$ are obtained by fitting (3.19) to the desired projection (3.14)-(3.16) with the manufacturer's values for the nominal focal length $f_0$ and the angle of view $\theta_{\max}$. Then we also obtain the radius of the image on the sensor plane by $r_{\max} = k_1\theta_{\max} + k_2\theta_{\max}^3$.

With a circular image fish-eye lens, the actual image fills only a circular area inside the image frames. In pixel coordinates, this circle is an ellipse

$$\left(\frac{u-u_0}{a}\right)^2 + \left(\frac{v-v_0}{b}\right)^2 = 1 \ ,$$

whose parameters can be estimated. Consequently, we obtain initial guesses for the remaining unknowns $m_u$, $m_v$, $u_0$, and $v_0$ in $\mathbf{p}$, where $m_u = a/r_{\max}$ and $m_v = b/r_{\max}$. With a full-frame fish-eye lens, the best thing is probably to place the principal point to the image centre and use the reported values of the pixel dimensions to obtain initial values for $m_u$ and $m_v$.

**Step 2: Refinement of the internal parameters**
With the internal parameters $\mathbf{p}$, we transform the observed points $\mathbf{m}_j^i$ to points $\tilde{\mathbf{x}}_j^i$ that approximately follow the perspective projection for each $j$ (in Fig. 3.3(b), this corresponds to transforming the point $p$ to $p'$). Under perspective imaging, the mapping between the calibration plane and the image plane is a planar homography for which holds $\tilde{\mathbf{x}}_j^i = \mathbf{H}_j\mathbf{x}_{\mathrm{p}}^i$. The aim of this step is to iteratively search such parameter values that the mapping between $\mathbf{x}_{\mathrm{p}}^i$:s and $\tilde{\mathbf{x}}_j^i$:s is as close to a homography as possible.

In practise, we suggest the following scheme for computing the error vector $\boldsymbol{\epsilon} = \mathcal{F}(\mathbf{p})$, where $\mathcal{F} : \mathbb{R}^6 \to \mathbb{R}^{NM}$.

(i) Back-project the control points by first computing the normalised image coordinates

$$\begin{pmatrix} x_j^i \\ y_j^i \end{pmatrix} = \begin{bmatrix} 1/m_u & 0 \\ 0 & 1/m_v \end{bmatrix} \begin{pmatrix} u_j^i - u_0 \\ v_j^i - v_0 \end{pmatrix},$$

transforming them to the polar coordinates $(r_j^i, \varphi_j^i) \hat{=} (x_j^i, y_j^i)$, and finally computing $\theta_j^i$ from (3.19).

(ii) Re-project the rays $(\theta_j^i, \varphi_j^i)$ using (3.13) with $f = 1$ to obtain the points $\tilde{\mathbf{x}}_j^i$.

(iii) Compute the homography estimates $\hat{\mathbf{H}}_j$ from the correspondences $\tilde{\mathbf{x}}_j^i \leftrightarrow \mathbf{x}_{\mathrm{p}}^i$ by the linear algorithm with data normalisation [Hartley00]. Define $\hat{\mathbf{x}}_j^i$ as the exact image of $\mathbf{x}_{\mathrm{p}}^i$ under $\hat{\mathbf{H}}_j$ such that $\hat{\mathbf{x}}_j^i = \hat{\mathbf{H}}_j\mathbf{x}_{\mathrm{p}}^i$.

(iv) Compute the distances $\epsilon_j^i = d(\tilde{\mathbf{x}}_j^i, \hat{\mathbf{x}}_j^i)$, combine them to vectors $\boldsymbol{\epsilon}_j = (\epsilon_j^1, \ldots, \epsilon_j^M)$ and further to a single vector $\boldsymbol{\epsilon} = (\boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_N)$.

We then use the Levenberg–Marquardt algorithm to minimise $||\boldsymbol{\epsilon}||$ with respect to $\mathbf{p}$.

**Step 3: Initialisation of external parameters**
First we refine the homographies $\mathbf{H}_j$ by minimising the errors $||\boldsymbol{\epsilon}_j||$ while keeping $\mathbf{p}$ fixed. Then, perspective imaging of the calibration plane, with $f = 1$, gives

$$s \begin{pmatrix} \tilde{x}_j^i \\ \tilde{y}_j^i \\ 1 \end{pmatrix} = \begin{bmatrix} \mathbf{R}_j & \mathbf{t}_j \end{bmatrix} \begin{pmatrix} X^i \\ Y^i \\ 0 \\ 1 \end{pmatrix} = \begin{bmatrix} \mathbf{r}_j^1 & \mathbf{r}_j^2 & \mathbf{t}_j \end{bmatrix} \begin{pmatrix} X^i \\ Y^i \\ 1 \end{pmatrix}$$

which implies $\mathbf{H}_j = [\mathbf{r}_j^1\ \mathbf{r}_j^2\ \mathbf{t}_j]$, up to scale. Furthermore

$$\mathbf{r}_j^1 = \lambda_j \mathbf{h}_j^1, \qquad \mathbf{r}_j^2 = \lambda_j \mathbf{h}_j^2, \qquad \mathbf{r}_j^3 = \mathbf{r}_j^1 \times \mathbf{r}_j^2,$$

$$\mathbf{t}_j = \lambda_j \mathbf{h}_j^3,$$

where $\lambda_j = \text{sign}(H_j^{3,3})/||\mathbf{h}_j^1||$. Because of estimation errors, the obtained rotation matrices are not orthogonal. Thus we use the singular value decomposition to compute the closest orthogonal matrices in the sense of Frobenius norm [Zhang98] and use them as initial guess for each $\mathbf{R}_j$ .

**Step 4: Minimisation of projection error**
As we have the estimates for the internal and external camera parameters, we use (3.2), (3.20) or (3.24), and (3.21) to compute the imaging function $\mathcal{P}_j$ for each camera, where a control point is projected to $\hat{\mathbf{m}}_j^i = \mathcal{P}_j(\mathbf{X}^i)$. The camera parameters are refined by minimising the sum of squared distances between the measured and modelled control point projections

$$\sum_{j=1}^{N} \sum_{i=1}^{M} d(\mathbf{m}_i^j, \hat{\mathbf{m}}_i^j)^2 \tag{3.25}$$

using the Levenberg–Marquardt algorithm. If the extended model (3.24) is used, the additional parameters may be initialised to zero.

### 3.3.3 Modification for Circular Control Points

In order to achieve an accurate calibration, we used a calibration plane with white circles on black background since the centroids of the projected circles can be detected with a sub-pixel level of accuracy [Heikkilä96]. In this setting, however, the problem is that the centroid of the *projected* circle is not the image of the centre of the original circle. Therefore, since $\mathbf{m}_j^i$ in (3.25) is the measured centroid, we should not project the centres as points $\hat{\mathbf{m}}_j^i$.

To avoid the problem above, we propose solving the centroids of the projected circles numerically. We parameterise the interior of the circle at $(X_0, Y_0)$ with radius $R$ by $\mathbf{X}(\varrho, \alpha) = (X_0 + \varrho \sin(\alpha), Y_0 + \varrho \cos(\alpha), 0)^\top$. Given the camera parameters, we get the centroid $\hat{\mathbf{m}}$ for the circle by numerically evaluating

$$\hat{\mathbf{m}} = \frac{\int_0^R \int_0^{2\pi} \hat{\mathbf{m}}(\varrho, \alpha) |\det \mathbf{J}(\varrho, \alpha)| \, d\alpha d\varrho}{\int_0^R \int_0^{2\pi} |\det \mathbf{J}(\varrho, \alpha)| \, d\alpha d\varrho}, \tag{3.26}$$

where $\hat{\mathbf{m}}(\varrho, \alpha) = \mathcal{P}(\mathbf{X}(\varrho, \alpha))$ and $\mathbf{J}(\varrho, \alpha)$ is the Jacobian of the composite function $\mathcal{P} \circ \mathbf{X}$. The analytical solving of the Jacobian is rather a tedious task but it can be computed by mathematical software such as Maple.

### 3.3.4 Backward Model

Above we have described the calibration of the forward camera model $\mathcal{P}$. In practice, one needs to know also the backward model, which tells the direction of an incoming light ray, $(\theta, \varphi)^\top$, corresponding to a given image point $(u, v)^\top$. Computation of the backward model for our extended camera model is not entirely trivial due to the asymmetric distortion terms. Anyway, first we explain how the backward model is computed for the basic model.

The basic (forward) camera model is illustrated as follows

$$\begin{pmatrix}\theta\\ \varphi\end{pmatrix} \xrightarrow{\mathcal{F}} \begin{pmatrix}x\\ y\end{pmatrix} \xrightarrow{\mathcal{A}} \begin{pmatrix}u\\ v\end{pmatrix},$$

where the mappings $\mathcal{F}$ and $\mathcal{A}$ are defined by

$$\mathbf{x} = \mathcal{F}(\boldsymbol{\psi}) = r(\theta)\begin{pmatrix}\cos\varphi\\ \sin\varphi\end{pmatrix} = \left(\sum_i k_i\theta^{2i-1}\right)\begin{pmatrix}\cos\varphi\\ \sin\varphi\end{pmatrix}, \tag{3.27}$$

$$\mathbf{m} = \mathcal{A}(\mathbf{x}) = \begin{bmatrix}m_u & 0\\ 0 & m_v\end{bmatrix}\begin{pmatrix}x\\ y\end{pmatrix} + \begin{pmatrix}u_0\\ v_0\end{pmatrix}, \tag{3.28}$$

using notations $\mathbf{x} = (x, y)^\top$, $\boldsymbol{\psi} = (\theta, \varphi)^\top$ and $\mathbf{m} = (u, v)^\top$. The computation of the backward model is straightforward since both $\mathcal{A}$ and $\mathcal{F}$ may be inverted analytically. In practice, the cubic equation $k_2\theta^3 + k_1\theta - r = 0$ has only one real root on the interval $[0, \pi/2]$ and that is the sought solution. If higher than third order terms are used in $r(\theta)$, then the roots must be solved numerically. Again a single root should exist on the interval $[0, \pi/2]$.

The extended model (3.24) contains an additional shift $\mathbf{s}$, $\mathbf{s} = \Delta_r(\theta, \varphi)\mathbf{u}_r(\varphi) + \Delta_t(\theta, \varphi)\mathbf{u}_\varphi(\varphi)$, in the image plane. We consider that when the measured image point is $\mathbf{x}$ the unshifted point is the *corrected* point $\mathbf{x}_\mathrm{c}$. The corresponding direction $\boldsymbol{\psi}_\mathrm{c} = (\theta_\mathrm{c}, \varphi_\mathrm{c})^\top$ is the true direction of the incoming ray. The extended model is illustrated as

$$\boldsymbol{\psi}_\mathrm{c} \xrightarrow{\mathcal{F}} \mathbf{x}_\mathrm{c} \xrightarrow{I+\mathcal{D}} \mathbf{x} \xrightarrow{\mathcal{A}} \mathbf{m},$$

where $I$ is the identity mapping and the distortion function $\mathcal{D}$ gives the shift between $\mathbf{x}$ and $\mathbf{x}_\mathrm{c}$,

$$\mathbf{x} - \mathbf{x}_\mathrm{c} = \mathcal{D}(\mathbf{x}_\mathrm{c}). \tag{3.29}$$

Since $\mathcal{A}$ and $\mathcal{F}$ are easily inverted the only problem is the inversion of $(I + \mathcal{D})$.

Given a point $\mathbf{x}$, the problem is to find the shift $\mathbf{s}$ so that $\mathbf{x}_\mathrm{c} = \mathbf{x} - \mathbf{s}$. The distortion function $\mathcal{D}$ may be expressed as the composite function

$$\mathcal{D}(\mathbf{x}_\mathrm{c}) = \left(\mathcal{G} \circ \mathcal{F}^{-1}\right)(\mathbf{x}_\mathrm{c}), \tag{3.30}$$

where the function $\mathcal{G}$ is defined by

$$\mathbf{s} = \mathcal{G}(\boldsymbol{\psi}) = \Delta_r(\theta, \varphi)\mathbf{u}_r(\varphi) + \Delta_t(\theta, \varphi)\mathbf{u}_\varphi(\varphi). \tag{3.31}$$

Here $\Delta_r(\theta, \varphi)$ and $\Delta_t(\theta, \varphi)$ are the distortion terms in (3.22) and (3.23). Using the first order Taylor approximation for $\mathcal{D}$ one obtains from (3.29) that

$$\begin{aligned}\mathbf{s} &= \mathcal{D}(\mathbf{x}_\mathrm{c}) = \mathcal{D}(\mathbf{x} - \mathbf{s})\\ &= \mathcal{D}(\mathbf{x}) - \frac{\partial\mathcal{D}}{\partial\mathbf{x}}(\mathbf{x})\mathbf{s},\end{aligned}$$

and further

$$\left(I + \frac{\partial \mathcal{D}}{\partial \mathbf{x}}(\mathbf{x})\right) \mathbf{s} = \mathcal{D}(\mathbf{x})$$
$$\mathbf{s} = \left(I + \frac{\partial \mathcal{D}}{\partial \mathbf{x}}(\mathbf{x})\right)^{-1} \mathcal{D}(\mathbf{x}).$$

However, since we do not have an explicit expression for $\mathcal{D}$ it is not possible to compute the Jacobian $\partial\mathcal{D}/\partial\mathbf{x}$ analytically. But we may numerically compute $\boldsymbol{\psi} = \mathcal{F}^{-1}(\mathbf{x})$ and the chain rule gives

$$\begin{aligned}
\mathbf{s} &= \left(I + \frac{\partial \mathcal{D}}{\partial \mathbf{x}}(\mathbf{x})\right)^{-1} \mathcal{D}(\mathbf{x}) \\
&= \left(I + \frac{\partial \mathcal{G}}{\partial \boldsymbol{\psi}} \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}}\right)^{-1} \mathcal{G}(\boldsymbol{\psi}) \\
&= \left(I + \frac{\partial \mathcal{G}}{\partial \boldsymbol{\psi}} \left(\frac{\partial \mathbf{x}}{\partial \boldsymbol{\psi}}\right)^{-1}\right)^{-1} \mathcal{G}(\boldsymbol{\psi}) \\
&= \left(I + \frac{\partial \mathcal{G}}{\partial \boldsymbol{\psi}}(\boldsymbol{\psi}) \left(\frac{\partial \mathcal{F}}{\partial \boldsymbol{\psi}}(\boldsymbol{\psi})\right)^{-1}\right)^{-1} \mathcal{G}(\boldsymbol{\psi}) \ .
\end{aligned} \tag{3.32}$$

Since the Jacobians $\partial\mathcal{G}/\partial\boldsymbol{\psi}$ and $\partial\mathcal{F}/\partial\boldsymbol{\psi}$ are easily computed from (3.27) and (3.31), equation (3.32) may be used to compute $\mathbf{s}$. Then $\mathbf{x}_c$ is directly given by $\mathbf{x} - \mathbf{s}$ and finally $\boldsymbol{\psi}_c$ is computed by $\boldsymbol{\psi}_c = \mathcal{F}^{-1}(\mathbf{x}_c)$.

It seems that the first order approximation for the asymmetric distortion function $\mathcal{D}$ is tenable in practice. We experimented the backward model error for real lenses by backprojecting random image points and then reprojecting them. The mean displacement of the image points was typically several degrees smaller than the achieved calibration accuracy for the forward model. The experiments are described in detail in Chapter 4.

# Chapter 4

# Calibration Experiments

The plane-based calibration procedure for fish-eye and perspective cameras was implemented as a calibration toolbox on Matlab. In the following, we describe the implementation and structure of the toolbox in more detail. After that we present some calibration results for real cameras and compare the proposed camera models to the model used in Heikkilä's calibration toolbox [Heikkilä00a]. Finally, we summarise the results and draw some conclusions.

## 4.1 Implementation

The calibration toolbox can be divided into two modules. The first module localises control points from calibration images but the second module is the core module which computes the camera parameters from point correspondences between the calibration plane and its images.

### 4.1.1 Finding Control Points

An evident requirement for accurate camera calibration is a precisely built calibration object. Besides, the calibration pattern must be such that the control points can be localised accurately from the images. A usual choice for the calibration pattern is the Tsai grid or a checkerboard pattern [Zhang98, Bouguet04] but the-state-of-the-art calibration results in terms of accuracy have been obtained with circular control points [Heikkilä96, Heikkilä00b]. To maximise the expected accuracy, we also used circular dots in the experiments.

To make repeated calibrations easy we implemented a semi-automatic procedure for finding the centroids of the dots from calibration images. For each calibration image the toolbox user must manually pick up a polygonal image region which contains the calibration dots. Thereafter the centroids of the dots are sought automatically. The circles must be organised into a rectangural grid on the calibration plane and they may be either white on black background or black on white background.

The control point localisation begins with thresholding the grayscale images. Because the thresholded image regions are ideally black and white a suitable threshold value is easily found by fitting two normal distributions to the grayscale histogram and implementing a Bayes classifier. The centroids of the

dots can be directly measured from the thresholded binary image. Optionally one may compute the grayscale centroids which give a more accurate localisation if the illumination is uniform [Heikkilä96]. The dots are automatically organised in such a way that each of them is associated with a unique dot on the calibration plane.

### 4.1.2 Computing Camera Parameters

In the calibration toolbox, there are currently three possible choices for the camera model. The simplest one is the basic model of Section 3.2.2 with the six internal camera parameters $\mathbf{p}_6 = (k_1, k_2, m_u, m_v, u_0, v_0)$. The second model $\mathbf{p}_9$ has three additional parameters $(k_3, k_4, k_5)$ in the radially symmetric part. The most diverse model is the extended model $\mathbf{p}_{23}$ of Section 3.2.3. The number of degrees of freedom in the models is however one less than the number of parameters. This is because a scale change in the pixel units $m_u, m_v$ is compensated by an opposite change in the coefficients $k_i$.

Particular algorithms for estimating the parameters of the camera model were described in the previous chapter. The choice between the procedures depends on the assumed projection type of the camera. For perspective cameras at least two views of the calibration plane are needed, nevertheless, singular configurations should be avoided [Sturm99]. For fish-eye lenses one view of the plane is generally sufficient for calibration. However, in all the cases, several views should be always used for most reliable results. For circular control points the modification in Section 3.3.3 is implemented. In this case the radius of the original circles must be given.

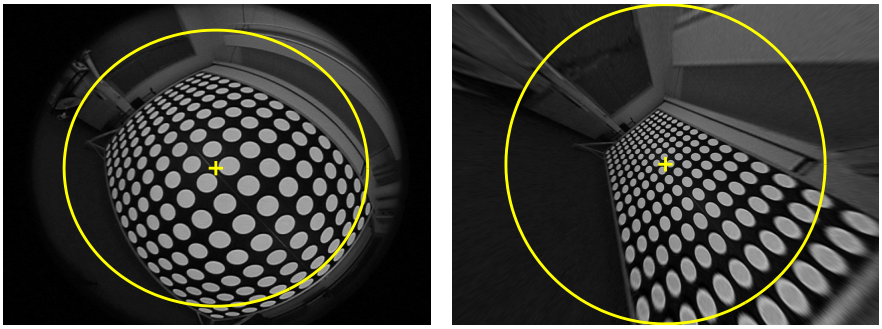
## 4.2 Results

### 4.2.1 Fish-Eye Lens Camera

In the fish-eye lens experiments, we used an equidistance lens with the nominal focal length of 1.178 mm attached to a Watec 221S CCD colour camera. The calibration object was a $2 \times 3$ m$^2$ plane containing white circles with the radius of 60 mm on the black background. The calibration images were digitised from an analog video signal to 8-bit monochrome images, whose size was 640 by 480 pixels.

When the basic model $\mathbf{p}_6$ is used, the calibration can be performed even from a single image of the planar object as Fig. 4.1 illustrates. In that example we used 60 control points for the calibration. However, for the most accurate results, the whole field of view should be covered with a large number of measurements. Therefore we experimented our method with 12 views and 740 points in total, the results are in Table 4.1. It can be seen that the centroid correction has a very important role. The extended model $\mathbf{p}_{23}$ gives the smallest deviations $\sigma_u$ and $\sigma_v$ in the $x$ and $y$ directions, respectively, but the radially symmetric model $\mathbf{p}_9$ gives almost as good results. Nevertheless, there should be no risk of over-fitting because the number of measurements is large. The estimated asymmetric distortion and the residuals are displayed in Fig. 4.2.

To demonstrate the achieved level of accuracy in another way, we approximate the size of the solid angle that projects to an ellipse with principal axes

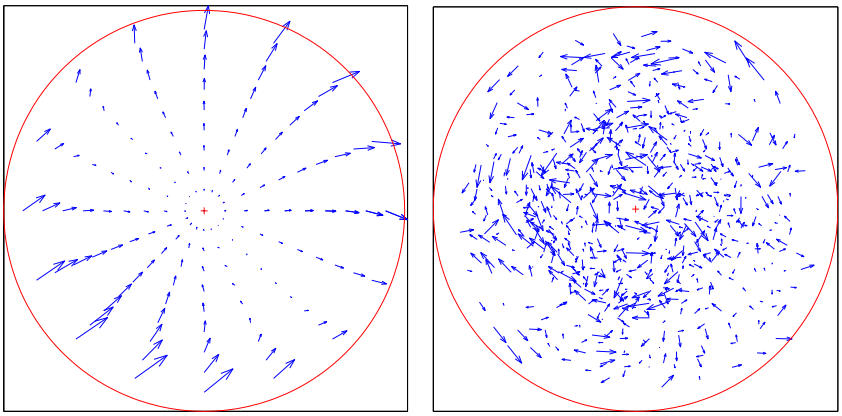


(a) (b)

**Figure 4.1:** Fish-eye lens calibration using only one view. (a) Original image where the ellipse depicts the field of view of 150°. (b) The image corrected to follow pinhole model. Straight lines are straight as they should be.

**Table 4.1:** Standard deviation of the residuals $(\mathbf{m} - \hat{\mathbf{m}})$ for the fish-eye lens camera. The star (*) indicates that the centroid correction of Sec. 3.3.3 is not used.

| | $\mathbf{p}_6^*$ | $\mathbf{p}_6$ | $\mathbf{p}_9$ | $\mathbf{p}_{23}$ |
|---|---|---|---|---|
| $\sigma_u$[pix] | 0.26 | 0.11 | 0.074 | 0.069 |
| $\sigma_v$[pix] | 0.24 | 0.10 | 0.060 | 0.058 |



(a) (b)

**Figure 4.2:** (a) The estimated asymmetric distortion $(\Delta_r \mathbf{u}_r + \Delta_t \mathbf{u}_\varphi)$ using the extended model $\mathbf{p}_{23}$. (b) The remaining residual for each control point that shows no obvious systematic error. Both plots are in normalised image coordinates and the vectors are scaled up by a factor of 150 to aid inspection.

**Table 4.2:** Standard deviation of the residuals $(\mathbf{m} - \hat{\mathbf{m}})$ for the conventional camera. The star (*) indicates that the centroid correction of Sec. 3.3.3 is not used.

| | $\mathbf{p}_6^*$ | $\mathbf{p}_6$ | $\mathbf{p}_9$ | $\mathbf{p}_{23}$ |
|---|---|---|---|---|
| $\sigma_u$[pix] | 0.100 | 0.100 | 0.091 | 0.078 |
| $\sigma_v$[pix] | 0.081 | 0.081 | 0.069 | 0.063 |

$2\sigma_u$ and $2\sigma_v$. For the equidistance projection the solid angle corresponding to the small area $dS$ in the image plane is

$$d\Omega = \frac{1}{f^2}\frac{\sin\theta}{\theta}dS \ . \tag{4.1}$$

Near the principal point ($\theta = 0$) with $f \approx k_1$=1.12 mm we have $d\Omega = 4.7 \cdot 10^{-7}$. At the distance of 500 mm from the camera centre this corresponds approximately to the area of a circle with the radius of 0.2 mm, which is in good agreement with the assumed accuracy of the calibration device.

The backward model error for $\mathbf{p}_{23}$, caused by the first order approximation of the asymmetric distortion function (see Section 3.3.4), was evaluated by backprojecting random image points and then reprojecting them. The mean and maximum displacement were $5.9 \cdot 10^{-7}$ and $9.8 \cdot 10^{-6}$ pixels, respectively. Both values are several degrees smaller than the achieved level of calibration accuracy. Therefore, it is justified to ignore this error in practice.

### 4.2.2 Conventional Camera

The calibration plane used above was also viewed by a conventional camera. The camera was Sony SNC-RZ30N with a zoom lens. The lens obeys approximately perspective projection but there is significant distortion in the peripheral regions of the image. Seven calibration images were taken at different orientations with fixed focus and zoom. The images were in compressed JPEG format and one of them is shown in Fig. 4.3. In addition, the illumination was noticeably non-uniform which, together with the compression artefacts, degrades the calibration result. The obtained results are in Table 4.2. Despite the shortcomings of the calibration images the residuals are relatively small. Naturally, the most diverse model fits best to the observations. Again we verified that the backward model error for $\mathbf{p}_{23}$ was several times smaller than the mean magnitude of the residuals.

### 4.2.3 Comparison with Heikkilä's Model

The camera models $\mathbf{p}_6$, $\mathbf{p}_9$ and $\mathbf{p}_{23}$ were compared to the camera model used in [Heikkilä00b]. Heikkilä's model is the skew-zero pinhole model accompanied with four distortion parameters and it is denoted by $\boldsymbol{\delta}_8$ in the following. The real image data for the comparison was provided by Heikkilä and is the same data as in [Heikkilä00b]. It was originally obtained by capturing a single image of a calibration object consisting of two orthogonal planes, each with 256 circular control points. The camera was a monochrome CCD camera with a 8.5 mm lens causing a distortion of several pixels near the periphery of the image. There were

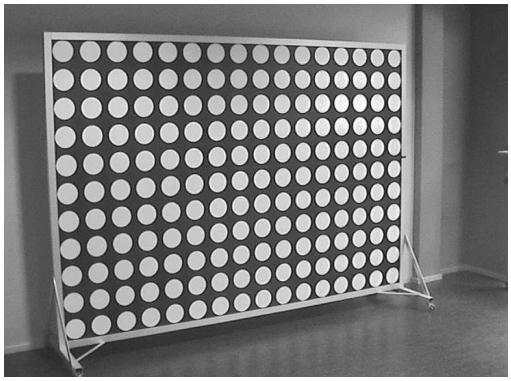


**Figure 4.3:** Calibration image captured by a conventional camera. Notice the non-uniform illumination.

**Table 4.3:** Standard deviation of the calibration residuals for Heikkilä's data.

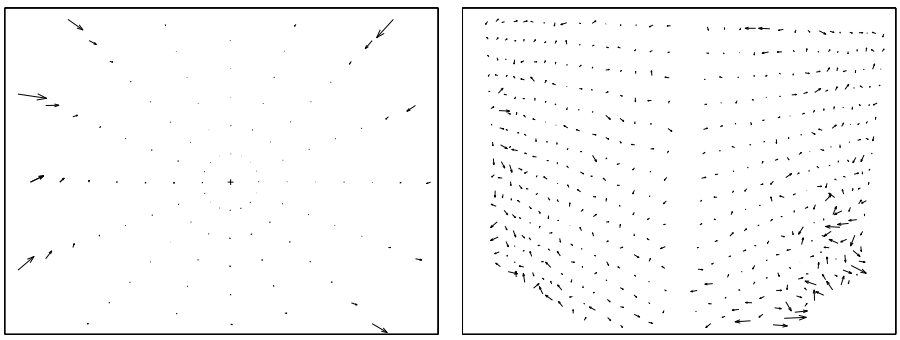
| | $\boldsymbol{\delta}_8$ | $\mathbf{p}_6$ | $\mathbf{p}_9$ | $\mathbf{p}_{23}$ |
|---|---|---|---|---|
| $\sigma_u$[pix] | 0.048 | 0.084 | 0.045 | 0.041 |
| $\sigma_v$[pix] | 0.037 | 0.066 | 0.032 | 0.032 |

491 control points that were visible in the image and were used for calibration. The control point centroids in the image were measured by ellipse boundary detection and conic fitting.

The calibration results are shown in Table 4.3. Especially interesting is the comparison between models $\boldsymbol{\delta}_8$ and $\mathbf{p}_9$ because they both have eight degrees of freedom. Model $\mathbf{p}_9$ gives slightly smaller residuals while it does not contain any tangential distortion terms. The model $\mathbf{p}_{23}$ gives again the smallest residuals but there may be a risk that it is fitted to the systematic errors of the calibration data. This is due that there are measurements only from a one plane, either one of the orthogonal planes, in each part of the image. Thus, the asymmetric distortion terms may quite easily fit to the errors of the calibration object or to the localisation errors caused by non-uniform illumination, for example. The estimated asymmetric distortion and remaining residuals for model $\mathbf{p}_{23}$ are shown in Fig. 4.4. The relatively large residuals in the lower right corner of the calibration image (Fig. 4.4(b)) may be an indication of inaccurate localisation which is partly compensated by the asymmetric distortion (Fig. 4.4(a)).

## 4.3 Summary

We have proposed a novel camera calibration method for fish-eye lens cameras that is based on viewing a planar calibration pattern. The experiments verified that the method is easy-to-use and provides for a relatively high level of accuracy by using circular control points. The proposed camera model is generic, easily expandable and suitable also for conventional cameras with narrow angle lenses. The achieved level of accuracy for both conventional and fish-eye lenses is comparable to the results in [Heikkilä00b]. This is promising considering especially the aim of using fish-eye lenses in measurement purposes.



(a) (b)

**Figure 4.4:** Heikkilä's calibration data. (a) The estimated asymmetric distortion $(\Delta_r \mathbf{u}_r + \Delta_t \mathbf{u}_\varphi)$ using the extended model $\mathbf{p}_{23}$. (b) The remaining residual for each control point. The vectors are scaled up by a factor of 150.

# Chapter 5

# Interest Point Matching

Recovering scene structure from a sequence of images requires such image features that can be tracked across the sequence. The tracked features are often interest points or corners that are such locations in the image where the intensity changes two-dimensionally. Interest points are extracted by a suitable interest point detector and they are usually matched between successive views through intensity cross-correlation. In this chapter, we first introduce the interest point detector we used for video sequences of concrete sewer pipes. Then we describe a multi-resolution matching scheme which we used to compute the initial interest point correspondences between successive video frames. We also show some results with real sewer images.

## 5.1 Interest Point Detectors

Several different interest point detectors have been proposed and evaluated in the literature [Schmid00, Tissainayagam04]. The two evaluation criteria introduced in [Schmid00] are *repeatability rate* and *information content*. The repeatability rate is the percentage of the total observed points which are repeated between two images taken under varying imaging conditions (orientation, scale, illumination, camera noise). The other criterion, information content, measures the distinctiveness of the local greylevel pattern at an interest point. Evidently, both repeatability and information content are essential for image matching.

The above two criteria, repeatability rate and information content, were used in [Schmid00] to evaluate six different corner detectors. An improved version of the Harris corner detector, described in detail in Section 5.1.1, obtained the best results. Tissainayagam and Suter evaluated four different corner detectors in terms of localisation accuracy and detection stability [Tissainayagam04]. Again the Harris detector performed well. Besides the evidence of these comparative studies, the Harris corner detector is widely used and approved interest point detector in applications. Hence, it was our choice for interest point extraction.

### 5.1.1 Harris Corner Detector

The Harris corner detector [Harris88] is based on the following $2 \times 2$ symmetric matrix

$$\mathbf{M}(x,y) = \begin{bmatrix} \mathcal{W} * \mathcal{I}_x^2(x,y) & \mathcal{W} * \mathcal{I}_x\mathcal{I}_y(x,y) \\ \mathcal{W} * \mathcal{I}_x\mathcal{I}_y(x,y) & \mathcal{W} * \mathcal{I}_y^2(x,y) \end{bmatrix}, \tag{5.1}$$

which is computed at each point of the greylevel image $\mathcal{I}(x,y)$. $\mathcal{I}_x$ and $\mathcal{I}_y$ indicate the $x$ and $y$ directional derivatives of the intensity function respectively. The convolution mask $\mathcal{W}$ is a Gaussian smoothing function used to suppress the effect of noise.

The matrix $\mathbf{M}$ is related to the local auto-correlation function of the image and it captures the shape of the intensity function in the neighbourhood of the point $(x,y)$. When the both eigenvalues of $\mathbf{M}$ are small the image is approximately constant in intensity. When they are large, the point $(x,y)$ is a corner point. An edge is detected if one eigenvalue is large and the other is small.

To measure the corner quality the Harris detector uses the following corner response function

$$R(x,y) = \det \mathbf{M} - k\,(\operatorname{trace} \mathbf{M})^2\,, \tag{5.2}$$

where we used a value $k = 0.06$ [Schmid00]. In order to obtain a good corner response, i.e. high value of $R$, both of the eigenvalues of the matrix $\mathbf{M}$ must be large. The detected corners are the local maxima of the response function that exceed some selected threshold. Sub-pixel precision in localisation is achieved through a quadratic approximation of the response function around a local maximum. By using response function (5.2) the explicit eigenvalue decomposition of $\mathbf{M}$ is additionally avoided.

The original paper of Harris and Stephens [Harris88] proposed computing the directional derivatives $\mathcal{I}_x$ and $\mathcal{I}_y$ by convolution with the mask $(-1,0,1)$ and its transpose respectively. However, the improved version of the detector, proposed in [Schmid00], computes $\mathcal{I}_x$ and $\mathcal{I}_y$ by convolution with derivatives of a narrow Gaussian ($\sigma = 1$). This modification improves repeatability rates, especially when the second image is rotated for an angle of about 45°.

## 5.2 Matching

The problem of image matching is to establish correspondences between the detected interest points in two different views of the same scene. In the first matching stage we choose the match candidates by requiring that the neighbourhoods of the interest points are similar enough. Ways to measure this similarity are described next.

### 5.2.1 Cross-Correlation

Given a corner point $(x_1, y_1)$ in the first image and a corner point $(x_2, y_2)$ in the second image we select a square window of size $(2n+1) \times (2n+1)$ around both

points and compute the normalised cross-correlation between the windows,

$$\rho = \frac{\sum_{i,j=-n}^{n} \left(\mathcal{I}_1(x_1+i, y_1+j) - \overline{\mathcal{I}_1(x_1, y_1)}\right) \left(\mathcal{I}_2(x_2+i, y_2+j) - \overline{\mathcal{I}_2(x_2, y_2)}\right)}{\hat{\sigma}(\mathcal{I}_1(x_1, y_1))\, \hat{\sigma}(\mathcal{I}_2(x_2, y_2))}, \tag{5.3}$$

where

$$\hat{\sigma}(\mathcal{I}_k(x_k, y_k)) = \sqrt{\sum_{i,j=-n}^{n} \left(\mathcal{I}_k(x_k+i, y_k+j) - \overline{\mathcal{I}_k(x_k, y_k)}\right)^2}, \quad k = 1, 2 \tag{5.4}$$

and $\overline{\mathcal{I}_k(x_k, y_k)}$ is the mean intensity over the selected window. The values of the correlation score $\rho$ vary between -1 and 1. For two identical windows $\rho = 1$ and for windows that are not similar at all $\rho = 0$. If the correlation score is higher than a given threshold, the interest points $(x_1, y_1)$ and $(x_2, y_2)$ are considered as a match candidate. It is possible that a point in the first image may match with several points in the second image, and vice versa. The simplest way to solve this ambiguity is to pair the points with highest correlation score but other techniques also exist [Xu96].

Since interest points are detected with a sub-pixel level of accuracy the locations $(x_k + i, y_k + j)$ in (5.3) are not at the centres of the pixels. Therefore the intensity values should be interpolated from the original image. Of course, one could just use the greylevel values of nearby pixels (nearest neighbour interpolation) but better localisation results are obtained by bilinear or bicubic interpolation.

If there is a significant rotation about the optical axis between the views, the correlation score (5.3) can be small also for true correspondences. To deal with this problem Xu and Zhang [Xu96] computed correlations on several rotations. However, in our sewer measurement application this is not necessary since the successive frames have similar rotations.

### 5.2.2 Multi-Resolution Matching

In order to reduce the number of false matches, we use a multi-resolution matching method instead of using only one-sized correlation window. The method is similar to that proposed in [Brandt01], and it is motivated by the way humans seem to perform the matching task. Usually one first looks for similarity of fairly large areas at a lower resolution level and then focuses into details at a higher resolution. Utilising larger correlation windows is especially advantageous if the images contain repeated patterns which may give ambiguous matches. However, it is often not reasonable to compare large neighbourhoods at a high resolution level due to disparity caused by the change in camera position.

The principle of multi-resolution matching is illustrated in Fig. 5.1. The correlation windows around an interest point are shown at two resolution levels. The original high-resolution image block is of size $6 \times 6$ and the corresponding low-resolution image, obtained by decimating the original image, is of size $3 \times 3$. The size of the correlation window (denoted by crosses in Fig. 5.1) is $3 \times 3$ at both resolution levels, but the window in the low-resolution image naturally corresponds to a larger neighbourhood in the original image. The neighbourhoods are compared at the higher resolution level only if the correlation score

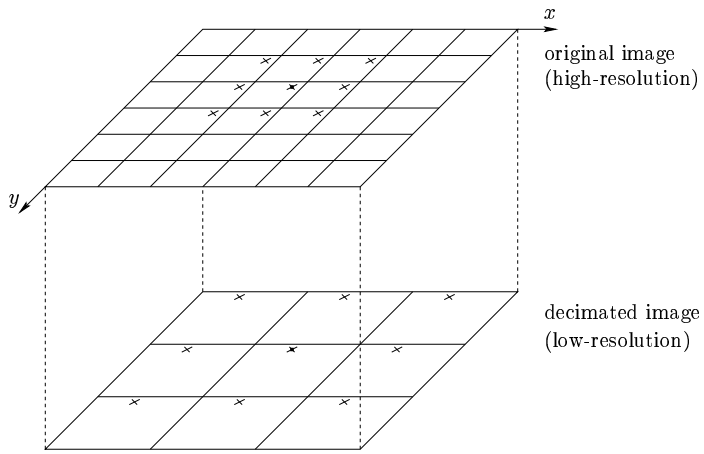



**Figure 5.1:** Correlation windows in multi-resolution matching. At both resolution levels the $3 \times 3$ window, denoted by crosses, is positioned to the detected interest point.

at the lower level exceeds a given threshold. The actual number of resolution levels used is adjustable in our implementation. One may also change the size of the correlation window and choose distinct correlation thresholds for different resolution levels.

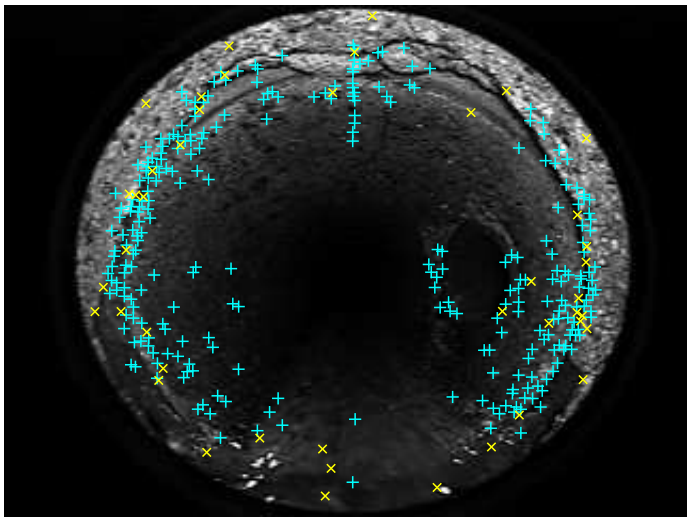
## 5.3 Sewer Videos

Next we show some matching results from experiments with real sewer videos. The test video sequence was scanned in an eroded concrete pipe using the DigiSewer robot. The uncompressed digital video was captured from an analog NTSC video signal at a resolution of $320 \times 240$ using a consumer grade capture card. The capture resolution was rather low due to the limitations of our hardware. However, capturing both fields of the NTSC signal, i.e. 480 lines instead of only 240, would cause problems related to interlacing. Interlacing here means that the camera does not capture both fields simultaneously and this causes problems when the camera moves. We avoid these problems, at the expense of losing some information, by capturing only even or odd lines.

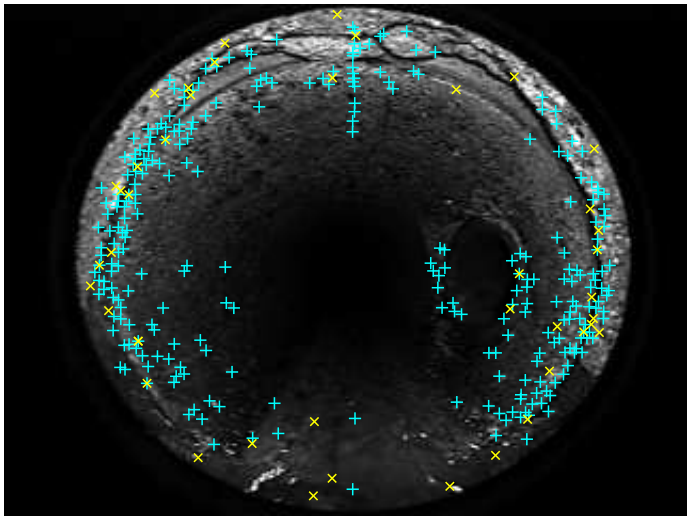
The frame rate of the test video is 30 fps. Since the robot moves relatively slow, there is very little difference between successive frames. Therefore we only process every fifth frame of the sequence. The interest points are detected with the improved Harris detector using the value 0.2 for the standard deviation of the smoothing Gaussian and the value 0.5 for the standard deviation of the Gaussian derivatives. The threshold for corner response $R$ is set adaptive so that a reasonable number of corners is detected in each quadrant of the image. The interest point correspondences are computed by the multi-resolution matching technique using two resolution levels. The size of the correlation window is $7 \times 7$ and the correlation threshold is 0.75 at both resolution levels. We also assume that the camera moves approximately in the direction of the optical axis and require that the corresponding points must lie approximately in the same sector (the difference of polar angle $\varphi$ must be below $10°$).

In Fig. 5.2 two frames from the test video sequence are shown together with

the matched Harris corners (cyan and yellow crosses). The matches denoted by yellow crosses do not satisfy the epipolar constraint which is robustly estimated from all matches (see Sections 6 and 7). Since the estimation of the two-view geometry has been successful, most of the cyan crosses denote true matches and the yellow ones are badly localised or totally false matches. In total, there were 744 corners in the first frame and 758 in the second. The number of matches is 325 of which 43 were classified false on the basis of the two-view geometry. The results above are typical for the rest of the test sequence, which consisted of hundreds of frames.



(a)



(b)

**Figure 5.2:** The matched Harris corners between two frames of the test video sequence. The matches denoted by yellow crosses were later classified erroneous on the basis of the estimated epipolar geometry.

# Chapter 6

# Multiple View Tensors

In this chapter, we introduce the multiple view tensors, which enclose the geometric constraints between multiple views of a single scene. Two- and three-view cases are considered. The tensor formulation of multiple view relations assumes projective cameras. However, also in the case of calibrated nonlinear cameras, such as fish-eye lens cameras, the multiple view tensors are useful geometric objects when solving the camera motion. This is discussed in Section 6.5.

## 6.1 Fundamental Matrix

The geometric constraint between two views of a single scene is called the *epipolar constraint* and is illustrated in Fig. 6.1. Given a point $\mathbf{m}$ in the first image its corresponding point in the second image is constrained to lie on a line called the *epipolar line* of $\mathbf{m}$, denoted by $\mathrm{l}'$. The line $\mathrm{l}'$ is the intersection of the plane $\boldsymbol{\pi}$, defined by $\mathbf{m}$, $\mathbf{C}$ and $\mathbf{C}'$, with the second image plane. Thus, the epipolar constraint states that an object point $\mathbf{X}$, its images $\mathbf{m}$, $\mathbf{m}'$ and the camera centres $\mathbf{C}$, $\mathbf{C}'$ must always lie on a single plane. Algebraically the epipolar constraint is expressed by the *fundamental matrix* which is the multiple view tensor in the two-view case. The following derivation of the fundamental matrix $\mathbf{F}$ is based on [Xu96].

Assume that the two camera projection matrices are $\mathbf{P}$ and $\mathbf{P}'$. The ray back-projected from $\mathbf{m}$ by $\mathbf{P}$ is obtained by solving $\mathbf{PX} = \mathbf{m}$. The one-parameter family of solutions is

$$\mathbf{X}(\lambda) = \mathbf{P}^{+}\mathbf{m} + \lambda\mathbf{C}, \tag{6.1}$$

where $\mathbf{P}^{+}$ is the pseudo-inverse of $\mathbf{P}$, i.e. $\mathbf{P}\mathbf{P}^{+} = \mathbf{I}$, and the camera centre $\mathbf{C}$ is the null-vector of $\mathbf{P}$. Two points on the back-projected ray are $\mathbf{C}$ and $\mathbf{P}^{+}\mathbf{m}$. Their projections in the second image are $\mathbf{P}'\mathbf{C}$ and $\mathbf{P}'\mathbf{P}^{+}\mathbf{m}$ and the epipolar line $\mathrm{l}'$ is the line joining these points, $\mathrm{l}' = (\mathbf{P}'\mathbf{C}) \times (\mathbf{P}'\mathbf{P}^{+}\mathbf{m})$. A cross-product of two 3-vectors can always be replaced by a product of a skew-symmetric matrix and a vector, i.e., $\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_{\times}\mathbf{b}$ where

$$[\mathbf{a}]_{\times} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}. \tag{6.2}$$

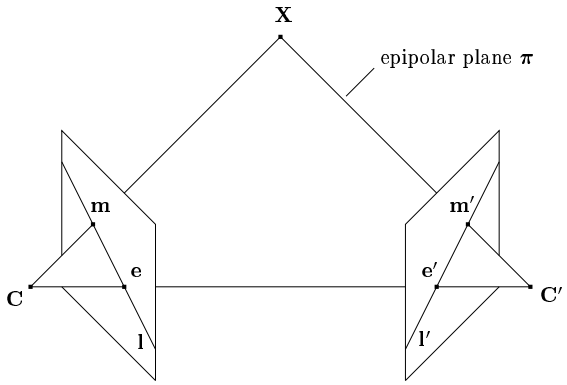



**Figure 6.1:** Epipolar geometry. Given a point $\mathbf{m}$ in the first image its corresponding point in the second image is constrained to lie on the line $\mathbf{l}'$ which is the epipolar line of $\mathbf{m}$. Correspondingly, the line $\mathbf{l}$ is the epipolar line of $\mathbf{m}'$. Points $\mathbf{e}$ and $\mathbf{e}'$ are the epipoles.

Thus, the formula above for $\mathbf{l}'$ may be written as

$$\mathbf{l}' = [\mathbf{P}'\mathbf{C}]_\times \mathbf{P}'\mathbf{P}^+\mathbf{m} = \mathbf{F}\mathbf{m}, \tag{6.3}$$

where the homogeneous $3 \times 3$ matrix $\mathbf{F}$ is the fundamental matrix. Since the point $\mathbf{m}'$ must lie on $\mathbf{l}'$, i.e. ${\mathbf{m}'}^\top \mathbf{l}' = 0$, the algebraic representation of the epipolar constraint is

$${\mathbf{m}'}^\top \mathbf{F}\mathbf{m} = 0. \tag{6.4}$$

Any pair of corresponding points $\mathbf{m} \leftrightarrow \mathbf{m}'$ in the two images must satisfy (6.4).

The rank of the fundamental matrix $\mathbf{F}$ is two and the epipoles $\mathbf{e}$ and $\mathbf{e}'$ are its left and right null-vectors, respectively. The fundamental matrix has nine elements but the rank-two constraint and indeterminate scale reduce the number of degrees of freedom to seven.

The above derivation shows that the two camera projection matrices define the fundamental matrix uniquely up to scale. On the other hand, the camera projection matrices may be retrieved from the fundamental matrix only up to a projective transformation. This is because the fundamental matrices corresponding to the pairs of camera matrices $(\mathbf{P}, \mathbf{P}')$ and $(\mathbf{P}\mathbf{H}, \mathbf{P}'\mathbf{H})$ are the same, where $\mathbf{H}$ is an arbitrary projective transformation of 3-space.

### 6.1.1 Essential Matrix

When the cameras are calibrated it is useful to write the correspondence condition (6.4) in terms of normalised image coordinates. In this case the fundamental matrix satisfies additional constraints and is called the *essential matrix*.

Finite projective cameras are of the form (3.4). Since only the relative position of the two cameras matters, we may fix the world coordinate frame at the first camera. Therefore the two camera matrices may be written as

$$\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}, \qquad \mathbf{P}' = \mathbf{K}'\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}.$$

Then

$$\mathbf{P}^{+} = \begin{bmatrix} \mathbf{K}^{-1} \\ \mathbf{0}^{\top} \end{bmatrix}, \qquad \mathbf{C} = \begin{pmatrix} \mathbf{0} \\ 1 \end{pmatrix}$$

and from (6.3) one obtains

$$\mathbf{F} = [\mathbf{P}'\mathbf{C}]_{\times}\mathbf{P}'\mathbf{P}^{+} = [\mathbf{K}'\mathbf{t}]_{\times}\mathbf{K}'\mathbf{R}\mathbf{K}^{-1} = \mathbf{K}'^{-\top}[\mathbf{t}]_{\times}\mathbf{R}\mathbf{K}^{-1}. \tag{6.5}$$

The last equality above follows from the fact that for any vector $\mathbf{a}$ and non-singular matrix $\mathbf{M}$ it holds $[\mathbf{a}]_{\times}\mathbf{M} = \mathbf{M}^{-\top}[\mathbf{M}^{-1}\mathbf{a}]_{\times}$ [Hartley00]. Substituting (6.5) into (6.4) gives

$$\mathbf{m}'^{\top}\mathbf{K}'^{-\top}[\mathbf{t}]_{\times}\mathbf{R}\mathbf{K}^{-1}\mathbf{m} = 0. \tag{6.6}$$

By denoting the normalised image coordinates by $\mathbf{x} = \mathbf{K}^{-1}\mathbf{m}$ and $\mathbf{x}' = \mathbf{K}'^{-1}\mathbf{m}'$ and defining the essential matrix by

$$\mathbf{E} = [\mathbf{t}]_{\times}\mathbf{R} \tag{6.7}$$

the correspondence condition (6.6) gets the form

$$\mathbf{x}'^{\top}\mathbf{E}\,\mathbf{x} = 0. \tag{6.8}$$

The essential matrix is determined only up to indeterminate scale by the equation (6.8). Therefore it is reasonable to consider the essential matrix as a homogeneous quantity. Then it has only five degrees of freedom, both $\mathbf{t}$ and $\mathbf{R}$ in (6.7) have three degrees of freedom but the indeterminate scale reduces the total number of degrees of freedom to five. The reduced number of degrees of freedom translates into extra constraints that are satisfied by an essential matrix, compared with a fundamental matrix. Because an essential matrix is a product of a skew-symmetric matrix and an orthogonal matrix one may prove that *a $3\times3$ matrix is an essential matrix if and only if two of its singular values are equal, and the third is zero* [Hartley00].

In general, given the fundamental matrix and a set of point correspondences one may compute a projective reconstruction for the points, i.e., the true and reconstructed scene points are related via a projective transformation. However, in the calibrated case it is possible to obtain a metric reconstruction, i.e., the scene is determined up to a similarity transformation. This is due to the fact that the rotation $\mathbf{R}$ and the *direction* of the translation $\mathbf{t}$ between the views may be retrieved from the essential matrix. Actually, there are two possible choices of $\mathbf{R}$ and two possible signs of $\mathbf{t}$ that satisfy (6.7) for a given essential matrix $\mathbf{E}$ (the scale is usually fixed so that $\|\mathbf{t}\| = 1$). This four-fold ambiguity is however solved by requiring that the reconstructed points are in front of both cameras [Hartley00].

## 6.2 Trifocal Tensor

The multiple view tensor of three-views is called the *trifocal tensor*. It has analogous properties to the fundamental matrix of two-views. Both tensors are independent of scene structure and depend only on the relations between the cameras. Also in the three-view case there is a point correspondence relation

that is linear in the elements of the tensor and is thus somewhat similar to (6.4). Like the fundamental matrix the trifocal tensor is invariant to projective transformations of the 3-space and the camera projection matrices may be retrieved from the tensor up to a common projective transformation.

The additional property of three-view geometry compared to the two-view case is the ability to transfer points and lines from two views to a third. Given a point correspondence over two views the point in the third view is determined by the trifocal tensor. This transfer property is useful when establishing correspondences over multiple views.

Above the fundamental matrix and the correspondence relation (6.4) were derived by geometric reasoning. In the following we consider multiple view relations in a more general framework of which the three-view case is a special case. The aim is to obtain a mathematical formulation of the trifocal tensor. The treatment is based on [Hartley00] and [Heyden00].

### 6.2.1 Bilinear and Trilinear Relations

Consider an object point $\mathbf{X}$ and its $n$ images,

$$\lambda_k \mathbf{m}_k = \mathbf{P}_k \mathbf{X} \quad k = 1, \ldots, n, \tag{6.9}$$

where the scale factors $\lambda_i$ are added so that the equations hold also as inhomogeneous equations. Matrix formulation of these camera equations is

$$\underbrace{\begin{bmatrix} \mathbf{P}_1 & \mathbf{m}_1 & 0 & 0 & \ldots & 0 \\ \mathbf{P}_2 & 0 & \mathbf{m}_2 & 0 & \ldots & 0 \\ \mathbf{P}_3 & 0 & 0 & \mathbf{m}_3 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_n & 0 & 0 & 0 & \ldots & \mathbf{m}_n \end{bmatrix}}_{\mathbf{G}} \begin{pmatrix} \mathbf{X} \\ -\lambda_1 \\ -\lambda_2 \\ -\lambda_3 \\ \vdots \\ -\lambda_n \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \tag{6.10}$$

which gives

$$\operatorname{rank} \mathbf{G} < n + 4.$$

This rank condition implies that all $(n+4) \times (n+4)$ minors of $\mathbf{G}$ are zero.

In the two-view case the above matrix $\mathbf{G}$ is a $6 \times 6$ square matrix and the rank condition implies

$$\det \begin{bmatrix} \mathbf{P}_1 & \mathbf{m}_1 & 0 \\ \mathbf{P}_2 & 0 & \mathbf{m}_2 \end{bmatrix} = 0, \tag{6.11}$$

which gives the bilinear constraint

$$\sum_{i,j=1}^{3} F_{ij} m_2^i m_1^j = 0, \tag{6.12}$$

where coefficients $F_{ij}$ depend only on the camera projection matrices and $m_k^i$ is the $i$:th element of $\mathbf{m}_k$. Using calculation rules of determinants, i.e., the Laplacian expansion by minors, one may verify the following expression for coefficients $F_{ij}$,

$$F_{ij} = \left(\frac{1}{4}\right) \sum_{p,q,r,s=1}^{3} \epsilon_{jpq} \epsilon_{irs} \det \begin{bmatrix} \mathbf{p}_1^p \\ \mathbf{p}_1^q \\ \mathbf{p}_2^r \\ \mathbf{p}_2^s \end{bmatrix}, \tag{6.13}$$

where $\mathbf{p}_k^i$ denotes the $i$:th row of camera matrix $\mathbf{P}_k$ and $\epsilon_{ijk}$ is the permutation tensor,

$$\epsilon_{ijk} = \begin{cases} 0 & \text{unless } i,j \text{ and } k \text{ are distinct} \\ +1 & \text{if } i,j,k \text{ is an even permutation of 1,2,3} \\ -1 & \text{if } i,j,k \text{ is an odd permutation of 1,2,3} \end{cases} . \tag{6.14}$$

Comparison of equations (6.4) and (6.12) reveals that (6.12) is just another way to express the correspondence relation (6.4) and the coefficients $F_{ij}$ are actually the elements of the fundamental matrix $\mathbf{F}$.[1] Since the fundamental matrix is defined only up to scale the factor $1/4$ in (6.13) is inessential. It is written out just to emphasise that (6.12) is equivalent expression to (6.11) including scale.

In the three-view case there are essentially two different types of $7 \times 7$ minors of matrix $\mathbf{G}$. One may choose either (i) three rows from each of two camera matrices and one row from the third or (ii) three rows from one camera matrix and two rows from each of the two others. However, it can be shown that the zero determinant condition in the case (i) just reduces to the bilinear relationships expressed by the fundamental matrices. More interesting relations are obtained by considering minors of the second type. An example of such a determinant is of the form

$$\det \begin{bmatrix} \mathbf{P}_1 & \mathbf{m}_1 & & \\ \mathbf{p}_2^i & & m_2^i & \\ \mathbf{p}_2^j & & m_2^j & \\ \mathbf{p}_3^k & & & m_3^k \\ \mathbf{p}_3^l & & & m_3^l \end{bmatrix} . \tag{6.15}$$

By expanding this determinant down the column containing $\mathbf{m}_1$ and setting it to zero one obtains a trilinear relation of the form

$$\sum_{i,j,k,r,s}^{3} T_i^{jk} m_1^i m_2^r \epsilon_{rjp} m_3^s \epsilon_{skq} = 0, \tag{6.16}$$

where the free indices $p$ and $q$ correspond to the rows omitted from the matrices $\mathbf{P}_2$ and $\mathbf{P}_3$ in (6.15). Thus, depending on the choice of $p$ and $q$, there are in total nine different trilinear constraints. However, only four of them are linearly independent [Hartley00]. The coefficients $T_i^{jk}$ in (6.16) are the elements of the trifocal tensor, denoted by $\mathbf{T}$, and are defined by

$$T_i^{jk} = \sum_{r,s}^{3} \epsilon_{irs} \det \begin{bmatrix} \mathbf{p}_1^r \\ \mathbf{p}_1^s \\ \mathbf{p}_2^j \\ \mathbf{p}_3^k \end{bmatrix} . \tag{6.17}$$

Since the first view has a special role in (6.17) the trifocal tensors corresponding to different numberings of the views are distinct. Nevertheless, they all represent the same geometric constraint and usually only one of them is considered for a given triple of views.

The trifocal tensor has 27 elements but only 18 degrees of freedom. The number of degrees of freedom may be counted by using the fact that the cameras may be retrieved from the trifocal tensor up to a projective transformation.

[1] The fundamental matrix is sometimes called the *bifocal tensor*.

The three camera matrices have 11 degrees of freedom each, giving 33 in total. Subtracting 15 degrees of freedom to account for the projective world frame leaves 18 degrees of freedom for the trifocal tensor.

### 6.2.2 Calibrated Trifocal Tensor

If the cameras are calibrated, the trifocal tensor satisfies additional constraints and has fewer degrees of freedom than in the general case. The calibrated camera matrices for a triple of views are

$$\mathbf{P} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}, \qquad \mathbf{P}' = \begin{bmatrix} \mathbf{R}' & \mathbf{t}' \end{bmatrix}, \qquad \mathbf{P}'' = \begin{bmatrix} \mathbf{R}'' & \mathbf{t}'' \end{bmatrix}, \tag{6.18}$$

where the world coordinate frame is fixed at the first camera. The rotations $\mathbf{R}'$ and $\mathbf{R}''$ and the translations $\mathbf{t}'$ and $\mathbf{t}''$ represent the positions of the second and third camera with respect to the first camera.

The trifocal tensor for cameras (6.18) may be expressed in terms of the rotations and translations,

$$T_i^{jk} = R'^j_i t''^k - t'^j R''^k_i, \tag{6.19}$$

where the subscripts correspond to columns of the rotation matrices and superscripts correspond to rows. Because the trifocal tensor is a homogeneous quantity only the relative scale of vectors $\mathbf{t}'$ and $\mathbf{t}''$ actually matters. The calibrated trifocal tensor has thus 11 degrees of freedom. Each of the rotations and translations has three degrees of freedom, giving 12 in total, but the indeterminate overall scale reduces the number to 11.

## 6.3 Estimation

Estimation of multiple view tensors is a key step in structure and motion recovery from image sequences. The bifocal or trifocal tensors for successive pairs or triplets of views provide an estimate of the (projective) camera motion through the sequence. In the following, we describe how the tensors can be computed from point correspondences.

### 6.3.1 Linear Method

Since the correspondence relations (6.12) and (6.16) are both linear in the entries of the tensors, they may be written as

$$\mathbf{a}^\top \mathbf{v} = 0, \tag{6.20}$$

where $\mathbf{v}$ consists of the elements of the tensors and $\mathbf{a}$ is determined by the correspondences. In the two-view case $\mathbf{v}$ has 9 elements, the entries $F_{ij}$ of $\mathbf{F}$, and in the three-view case 27 elements, the entries $T_i^{jk}$ of $\mathbf{T}$. A point correspondence over two views provides one linear constraint on $\mathbf{F}$, but a point correspondence over three views provides four linearly independent constraints on $\mathbf{T}$.

Given $n$ correspondences the above linear constraints form a set of linear equations,

$$\mathbf{A}\mathbf{v} = \mathbf{0}, \tag{6.21}$$

where $\mathbf{A}$ has $n$ rows in the two-view case and $4n$ rows in the three-view case. If the correspondences are exact the maximum rank of $\mathbf{A}$ is one less than the number of its columns and the sought solution $\mathbf{v}$ is the generator of the one-dimensional right null-space of $\mathbf{A}$. In practice, due to noise in the point coordinates, we seek a solution that minimises $\|\mathbf{A}\mathbf{v}\|$ subject to the condition $\|\mathbf{v}\| = 1$. This solution for $\mathbf{v}$ is the singular vector corresponding to the smallest singular value of $\mathbf{A}$ and is obtained by singular value decomposition. The required number of correspondences is at least eight in the two-view case and at least seven in the three-view case.

The linear algorithm does not consider the constraints satisfied by the bifocal and trifocal tensors. For instance, the obtained solution for $\mathbf{F}$ may not be a rank-two matrix. Therefore the solution is usually replaced by the closest singular matrix under a Frobenius norm. The constraint enforcement for a trifocal tensor is more complicated and is described for example in [Hartley00].

An important improvement to the linear estimation method is the normalisation of the image coordinates as described in [Hartley00]. The image coordinates are transformed and scaled in such a way that the stability of the least-squares problem is improved. Since the added complexity of the algorithm is insignificant, the normalisation should be always done when the linear method is used.

### 6.3.2 Minimisation of Geometric Distance

A problem with the linear method is that it does not minimise a geometrically or statistically meaningfull quantity. A common and often reasonable assumption is that the image coordinate measurement errors are normally distributed. Specifically, we assume that the noise on each image coordinate is Gaussian with zero mean and uniform standard deviation. Under this assumption the maximum likelihood estimate of the multiple view tensor, both in the two- and three-view cases, is obtained by minimising the geometric distance

$$\sum_i \sum_j d(\mathbf{m}_j^i, \hat{\mathbf{m}}_j^i)^2, \tag{6.22}$$

where $\mathbf{m}_j^i$ are the measured coordinates and $\hat{\mathbf{m}}_j^i$ are estimated noise free coordinates which exactly satisfy the correspondence relation, (6.12) or (6.16). Since the true noise free coordinates are also unknown they are estimated together with the multiple view tensor. This is done by seeking such 3D-points $\hat{\mathbf{X}}^i$ and camera matrices $\mathbf{P}_j$ that minimise (6.22) where $\hat{\mathbf{m}}_j^i = \mathbf{P}_j \hat{\mathbf{X}}^i$. The maximum likelihood estimate of the bifocal or trifocal tensor is then directly computed from the camera matrices.

Minimisation of (6.22) is a non-linear optimisation problem which can be solved using the Levenberg-Marquardt algorithm, for example. To obtain the initial values for the optimisation one must compute an initial estimate of the multiple view tensor by the linear method. The initial camera matrices are then retrieved from this tensor and the initial estimates of $\hat{\mathbf{X}}^i$ are computed by triangulation [Hartley00].

### 6.3.3 The Calibrated Case

The calibrated bifocal and trifocal tensors satisfy additional constraints that must be enforced during the estimation. Instead of (6.22) one should minimise

the distance

$$\sum_i \sum_j d(\mathbf{x}_j^i, \mathbf{P}_j \hat{\mathbf{X}}^i)^2, \tag{6.23}$$

where $\mathbf{x}_j^i$ are the normalised image coordinates and $\mathbf{P}_j$ are the *calibrated* camera matrices. Thus, the camera matrices must be parameterised in such a way that the additional constraints are satisfied. For example, in the three-view case the camera matrices have the form (6.18) and are parameterised via the rotation and translation parameters. In the following, we describe linear estimation methods that may be used to initialise the minimisation of (6.23).

The linear estimation method of the fundamental matrix, described in Section 6.3.1, is easily modified for computing the essential matrix. Also in the calibrated case the correspondence relation (6.8) leads to a set of linear equations of the form $\mathbf{A}\mathbf{v} = \mathbf{0}$, where $\mathbf{v}$ contains the elements of the essential matrix. The method differs from the computation of the fundamental matrix only in the enforcement of the constraints. Instead of choosing the closest singular matrix in Frobenius norm, one must choose a matrix whose two singular values are equal and the third is zero. This is done using the singular value decomposition as described in [Faugeras01].

When there are three views the calibrated camera matrices may be written as in (6.18). One may compute the essential matrices for view-pairs (1,2) and (1,3) by the linear method. From these essential matrices one obtains estimates of the rotations $\mathbf{R}'$, $\mathbf{R}''$ and the directions of the translations $\mathbf{t}'$, $\mathbf{t}''$. However, the ratio of the magnitudes of the translations is left undetermined. In the following, we propose solving also this ratio linearly when the rotations and translation directions are known.

By denoting $\mathbf{t}' = s'\mathbf{t}'/\|\mathbf{t}'\| = s'\hat{\mathbf{t}}'$ and correspondingly $\mathbf{t}'' = s''\hat{\mathbf{t}}''$ we may write (6.19) in the form

$$T_i^{jk} = s''R'^j_i \hat{t}''^k - s'\hat{t}'^j R''^k_i. \tag{6.24}$$

By substituting this into the correspondence relation (6.16), which is written in terms of the normalised coordinates in the calibrated case, one obtains a relation of the form

$$\mathbf{a}^\top \begin{pmatrix} s' \\ s'' \end{pmatrix} = 0. \tag{6.25}$$

Thus, one correspondence over three views is enough to solve the ratio $s'/s''$. When more correspondences are used a set of linear equations is formed and the solution is obtained through the singular value decomposition.

## 6.4 Uncertainty of the Epipolar Geometry

Due to noise in the measured image coordinates the estimated fundamental matrix is not exact. Because the estimate of $\mathbf{F}$ is uncertain and the measured points are noisy the correspondences do not satisfy (6.4) precisely. Thus, given a point $\mathbf{m}$ in the first image the corresponding point in the second image does not lie exactly on the epipolar line $\mathbf{F}\mathbf{m}$ but probably in a narrow region on either side of the line. This region is bounded by the *epipolar envelope* and it is determined by the covariance matrix of the fundamental matrix. The

estimation of the covariance of the fundamental matrix is discussed in [Csurka97] and [Hartley00], and in the robust case in [Brandt02].

When the epipolar lines are defined by the normalised equation

$$\mathbf{l} = \frac{\mathbf{F}\mathbf{m}}{\|\mathbf{F}\mathbf{m}\|}, \tag{6.26}$$

the first-order covariance approximation is given by

$$\mathbf{\Lambda_l} = \frac{\partial \mathbf{l}}{\partial \mathbf{F}} \mathbf{\Lambda_F} \frac{\partial \mathbf{l}}{\partial \mathbf{F}}^\top + \frac{\partial \mathbf{l}}{\partial \mathbf{m}} \mathbf{\Lambda_m} \frac{\partial \mathbf{l}}{\partial \mathbf{m}}^\top, \tag{6.27}$$

where the two terms account for the uncertainty of $\mathbf{F}$ and $\mathbf{m}$, respectively. $\mathbf{\Lambda_F}$ is the $9 \times 9$ covariance matrix of $\mathbf{F}$ and $\partial \mathbf{l}/\partial \mathbf{F}$ is the Jacobian of (6.26) with respect to $\mathbf{F}$, which is here regarded as a vector of 9 elements. The covariance of $\mathbf{m}$ is often assumed to have the following simple form

$$\mathbf{\Lambda_m} = \begin{bmatrix} \sigma^2 & 0 & 0 \\ 0 & \sigma^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{6.28}$$

Due to the constraint $\|\mathbf{l}\| = 1$ implied by (6.26) the covariance matrix $\mathbf{\Lambda_l}$ has rank 2. It can be shown [Hartley00] that if $\mathbf{l}$ is a random line obeying a Gaussian distribution with mean $\bar{\mathbf{l}}$ and covariance matrix $\mathbf{\Lambda_l}$ of rank 2, then the plane conic

$$\mathbf{C} = \bar{\mathbf{l}}\bar{\mathbf{l}}^\top - k^2 \mathbf{\Lambda_l} \tag{6.29}$$

represents an equal-likelihood contour bounding some fraction of all instances of $\mathbf{l}$. If $F_2(k^2)$ represents the cumulative $\chi^2_2$ distribution and $k^2$ is chosen such that $F_2^{-1}(k^2) = \alpha$, then a fraction $\alpha$ of all lines lie within the region bounded by $\mathbf{C}$. Normally the conic $\mathbf{C}$ is a hyperbola, whose branches lie on different sides of the mean epipolar line, and it forms the envelope of the epipolar lines. For example the 95 % envelope is obtained when $k^2 = 5.99$ and it defines the region within which the epipolar line lies with a probability of 95 %, according to the above first-order Gaussian approximation.

## 6.5 General Calibrated Cameras

Next we consider the case where we have two or three views taken by a general calibrated camera. For example, the camera may be a fish-eye lens camera that is calibrated using the methods of Chapter 3. Also in this case the rotations and translations between the views may be estimated by first computing the calibrated multiple view tensor.

Assume that we have established a set of point correspondences over the views. Point $i$ in view $j$ is denoted by $\mathbf{m}_j^i$. Because the camera is calibrated we may transform the points $\mathbf{m}_j^i$ to points $\tilde{\mathbf{x}}_j^i$ that follow the perspective projection. The transformation is done by first back-projecting the points and then perspectively re-projecting them. The points $\tilde{\mathbf{x}}_j^i$ are considered as correspondences over the *transformed* views and they are images of some object points $\mathbf{X}^i$ so that

$$\tilde{\mathbf{x}}_j^i = \begin{bmatrix} \mathbf{R}_j & \mathbf{t}_j \end{bmatrix} \mathbf{X}^i, \tag{6.30}$$

where the rotations and translations are the same as between the *original* views. The world frame may be fixed to the first camera by setting $\mathbf{R}_1 = \mathbf{I}$ and $\mathbf{t}_1 = \mathbf{0}$.

Due to noise, there are no such object points that (6.30) would hold exactly. However, one may still compute the points $\tilde{\mathbf{x}}_j^i$ and use the linear methods of Section 6.3.3 to compute the calibrated multiple view tensor between the transformed views. This way one obtains estimates of $\mathbf{R}_j$ and $\mathbf{t}_j$. Estimates $\hat{\mathbf{X}}^i$ for the object points may then be computed by triangulation.

Again, the optimal way of estimating the camera motion is to minimise a geometric distance in the original images where the measurements are done, i.e.,

$$\min \sum_i \sum_j d(\mathbf{m}_j^i, \hat{\mathbf{m}}_j^i)^2, \tag{6.31}$$

where $\hat{\mathbf{m}}_j^i$ are the estimated exact correspondences,

$$\hat{\mathbf{m}}_j^i = \mathcal{P}_j(\hat{\mathbf{X}}^i). \tag{6.32}$$

Here $\mathcal{P}_j$ is the imaging function of the general camera in view $j$. The cost (6.31) is minimised over the external camera parameters of $\mathcal{P}_j$ and the 3D-points $\hat{\mathbf{X}}^i$.

### 6.5.1 Essential Matrix and Epipolar Envelopes

In the two-view case, there is a way to compute an approximation to the geometric distance (6.31) without estimating the optimal 3D-points $\hat{\mathbf{X}}^i$. Since $n$ 3D-points has $3n$ parameters, the number of parameters in the minimisation problem is reduced from $5 + 3n$ to 5. These five parameters are the parameters of the essential matrix: three for the rotation and two for the direction of the translation. We describe this method in the following because it also provides a simple way to estimate the covariance of the essential matrix. The covariance of the essential matrix is used to compute the epipolar envelopes.

The transformation that "corrects" the original image to a perspective one is denoted by $\mathcal{T}$, i.e. $\tilde{\mathbf{x}}_j^i = \mathcal{T}(\mathbf{m}_j^i)$. Given the essential matrix $\mathbf{E}$ and the transformed correspondences, $\tilde{\mathbf{x}}_1^i \leftrightarrow \tilde{\mathbf{x}}_2^i$, there is a non-iterative algorithm [Hartley00] for computing the points $\hat{\mathbf{x}}_1^i$ and $\hat{\mathbf{x}}_2^i$ that minimise the geometric distance

$$\sum_i d(\tilde{\mathbf{x}}_1^i, \hat{\mathbf{x}}_1^i)^2 + d(\tilde{\mathbf{x}}_2^i, \hat{\mathbf{x}}_2^i)^2 \tag{6.33}$$

in the transformed image plane subject to the constraint

$$\hat{\mathbf{x}}_2^{i\top} \mathbf{E}\, \hat{\mathbf{x}}_1^i = 0.$$

By transforming the points $\hat{\mathbf{x}}_j^i$ to the original image, one obtains points

$$\hat{\mathbf{m}}_j^i = \mathcal{T}^{-1}(\hat{\mathbf{x}}_j^i), \tag{6.34}$$

which may be used as approximations to the optimal exact correspondences that minimise (6.31). Hence, one may write the minimisation problem (6.31) in the form

$$\min_{\mathbf{z}} \sum_i C_i(\mathbf{y}_i, \mathbf{z})^2, \tag{6.35}$$

where vectors $\mathbf{y}_i$ contain the measured correspondences in both views and $\mathbf{z}$ is the 5-vector containing the parameters of the essential matrix. The cost function in (6.35) has such a form that as a by-product of the minimisation one can compute an estimate for the covariance of the parameters $\mathbf{z}$. This is possible by making some simplifying assumptions as described in detail in [Csurka97].

The epipolar envelopes in the transformed image are determined by (6.29) but the covariance matrix of the epipolar line is now approximated by

$$\mathbf{\Lambda_l} = \left(\frac{\partial \mathrm{l}}{\partial \mathbf{E}}\frac{\partial \mathbf{E}}{\partial \mathbf{z}}\right)\mathbf{\Lambda_z}\left(\frac{\partial \mathrm{l}}{\partial \mathbf{E}}\frac{\partial \mathbf{E}}{\partial \mathbf{z}}\right)^{\top} + \frac{\partial \mathrm{l}}{\partial \mathbf{m}}\mathbf{\Lambda_m}\frac{\partial \mathrm{l}}{\partial \mathbf{m}}^{\top}, \tag{6.36}$$

where the Jacobians are computed from

$$\mathrm{l} = \frac{\mathbf{E}(\mathbf{z})\mathcal{T}(\mathbf{m})}{\|\mathbf{E}(\mathbf{z})\mathcal{T}(\mathbf{m})\|}. \tag{6.37}$$

The explicit form of the Jacobian $\partial \mathrm{l}/\partial \mathbf{m}$ depends on the transformation $\mathcal{T}$. For our fish-eye lens camera we used the extended camera model of Section 3.2.3 with which it is possible to compute $\partial \mathrm{l}/\partial \mathbf{m}$ analytically.

# Chapter 7

# Tracking and Reconstruction

In this chapter we show how the theory and methods of the previous chapter can be applied to acquire 3D models from video sequences. Sewer videos scanned by the DigiSewer robot are experimented.

## 7.1 Computation of the Multiple View Tensors

In practice, the initial match candidates over successive video frames always contain false matches, as illustrated in Fig. 5.2. The erroneous correspondences are often referred to *outliers* since they usually do not satisfy the geometric constraints between the views. Hence, when estimating the bifocal or trifocal tensors from initial correspondences we need robust estimation methods that are tolerant to false correspondences. The RANSAC (Random Sample Consensus) algorithm [Fischler81] is a most commonly used robust estimation method in geometric computer vision. We implemented the RANSAC algorithm for the estimation of two- and three-view relations between views that are taken by a calibrated fish-eye lens camera. For the most part the implementation follows the recommendations in [Hartley00], but the application to the fish-eye case is our own. In the following, we outline the procedure.

The objective of robust estimation is to fit a model to a data set which contains outliers. Assume that we have a data set $S$ containing $n$ data points of which some are outliers. Furthermore, assume that a minimum of $s$ data points are required to instantiate the free parameters of the model. The idea of the RANSAC algorithm is to randomly select samples of $s$ data points from $S$ and compute the model from each sample subset. For each instance of the model we determine the set of data points which are within a distance threshold of the model. This subset defines the inliers of $S$. The model with largest number of inliers is selected and re-estimated using all the inliers. The estimate should be close to the true model if enough samples were drawn so that at least one of them is free from outliers.

When estimating the calibrated two-view geometry, the model is defined by (6.8), the parameters of the model are those of the essential matrix and the data points are the measured correspondences between the views. The size of

the minimal data sample is eight if the linear algorithm of Section 6.3.1 is used to instantiate the model. Nevertheless, there is an algorithm for solving the fundamental matrix from seven point correspondences [Torr00]. This algorithm can be also used in the calibrated case by finally enforcing the additional constraints of the essential matrix. We used the seven point algorithm in the experiments because smaller sample sets have a higher probability to be free from outliers.

For the fish-eye images, the essential matrix is computed from the transformed correspondences $\tilde{\mathbf{x}} \leftrightarrow \tilde{\mathbf{x}}'$, as described in Section 6.5. However, the distance of a correspondence from the model is measured in the original image by

$$d(\mathbf{m}, \hat{\mathbf{m}})^2 + d(\mathbf{m}', \hat{\mathbf{m}}')^2, \tag{7.1}$$

where $\mathbf{m}$ and $\mathbf{m}'$ are the observed points in the fish-eye images and the points $\hat{\mathbf{m}}$ and $\hat{\mathbf{m}}'$ are computed by (6.34) and satisfy the two-view constraint as described in Section 6.5.1. The distance threshold for the inliers, $t^2$, is determined by $t^2 = F_1^{-1}(0.95)\,\sigma^2$, where $F_1$ represents the cumulative $\chi_1^2$ distribution and $\sigma$ is the estimated standard deviation of the measurement errors in the original fish-eye coordinates [Hartley00]. A robust estimate of $\sigma$ can be computed as explained in [Xu96].

In the three-view case, we first robustly estimate the essential matrices for view pairs (1,2) and (1,3). Then the RANSAC procedure is used to determine the relative scale of the two translations from the three-view correspondences. At minimum only one correspondence is required, which implies that only one random sample needs to be drawn. The distance measure used for the three-view correspondences is

$$d(\mathbf{m}, \hat{\mathbf{m}})^2 + d(\mathbf{m}', \hat{\mathbf{m}}')^2 + d(\mathbf{m}'', \hat{\mathbf{m}}'')^2, \tag{7.2}$$

where $\hat{\mathbf{m}}$ and $\hat{\mathbf{m}}'$ are computed exactly as in (7.1) and $\hat{\mathbf{m}}''$ is the point that is obtained by transferring the correspondence $\hat{\mathbf{m}} \leftrightarrow \hat{\mathbf{m}}'$ to the third view using the transfer property of the trifocal tensor. Now the distance threshold is $t^2 = F_3^{-1}(0.95)\,\sigma^2$ because the codimension of the model is 3 in the three-view case [Hartley00].

Nevertheless, the final estimate of the camera motion over each triple of views is refined by minimising (6.31) over both the motion parameters and the 3D coordinates of the inliers. Furthermore, we iterate between (i) optimal fit to inliers and (ii) re-classification of inliers; until the number of inliers converges. The sub-optimal distance measures (7.1) and (7.2) in the RANSAC are used for computational efficiency.

## 7.2 Tracking with Geometric Constraints

After we have successfully estimated the two- or three-view geometry for the successive images in a sequence, we may discard those point correspondences that are not consistent with the geometry. However, we may also use the estimated multiple view geometry to guide the matching. A weaker similarity threshold can be employed because the geometric constraints discriminate the false matches. In the following sections, we illustrate this with the sewer video sequences.

### 7.2.1 Two-View Geometry

In Fig. 5.2 we showed the initial point correspondences between two frames in a sewer video sequence. In Fig. 7.1, the two-view geometry of the frames is illustrated by choosing two points from the first image (denoted by yellow crosses) and plotting the corresponding epipolar curves to the second image (the magenta curves). The epipolar curves correspond to the epipolar lines in the transformed image and are determined by (6.37). The curves were plotted into Fig. 7.1 by transforming the epipolar lines back to the original image.

In Fig. 7.1, the epipoles are the cyan crosses near the centre of the images. The yellow curves are the envelopes of the epipolar curves and they were computed by using the estimated uncertainty of the essential matrix. The envelope of the horizontal curve is broad because a very large value of $k^2 = 1000$ was chosen in (6.29) in order to better illustrate the error bounds. The narrow envelope of the vertical curve is the 95 % envelope which we used to define the search region for the correspondence. The estimated value for the standard deviation of measured points was $\sigma = 0.25$. The narrow error bounds show that the search region for the correspondence is very strictly limited by the epipolar constraint.

The yellow crosses in the second image correspond to the narrowest point of the envelope, i.e., the centre of the hyperbola in the transformed image. It appears that the narrow point is often close to the true correspondence [Hartley00, Brandt02]. Hence, if there are several match candidates in the search region that exceed the similarity threshold, we choose the candidate that is closest to the narrowest point. In the guided matching stage, the threshold for the correlation score was set to 0.3 at both resolution levels of the multi-resolution method.

### 7.2.2 Three-View Geometry

The point matches that are consistent with the estimated two-view geometry are used as initial correspondences when estimating the three-view geometry. In Fig. 7.2, we show the initial three-view correspondences over three frames. Again, the correspondences that are denoted by yellow crosses were classified outliers. Since the three-view constraint is much stronger than the two-view constraint false matches are very improbable after imposing the constraint.

The matches that survive through several successive image triplets are extended over the sequence as follows. If a match in image triplet (1,2,3) and another match in the successive triplet (2,3,4) correspond to the same interest point in the overlapping views, images 2 and 3, they are combined into a single match over four images. Continuing this way, some interest points may be tracked across several images.

In Fig. 7.3 we illustrate the tracking results for a sewer image sequence that consists of 159 images. The found interest points are on the vertical axis and the horizontal line segments indicate the visibility of the track. The total number of found interest points is 6864. Only points that could be tracked through at least three successive images were accepted. Hence, due to the three-view constraint practically all established correspondences are true. Because the camera moves all the time forward the correspondence chains are short on average as illustrated in the length histogram of the correspondence chains shown in Fig. 7.4.

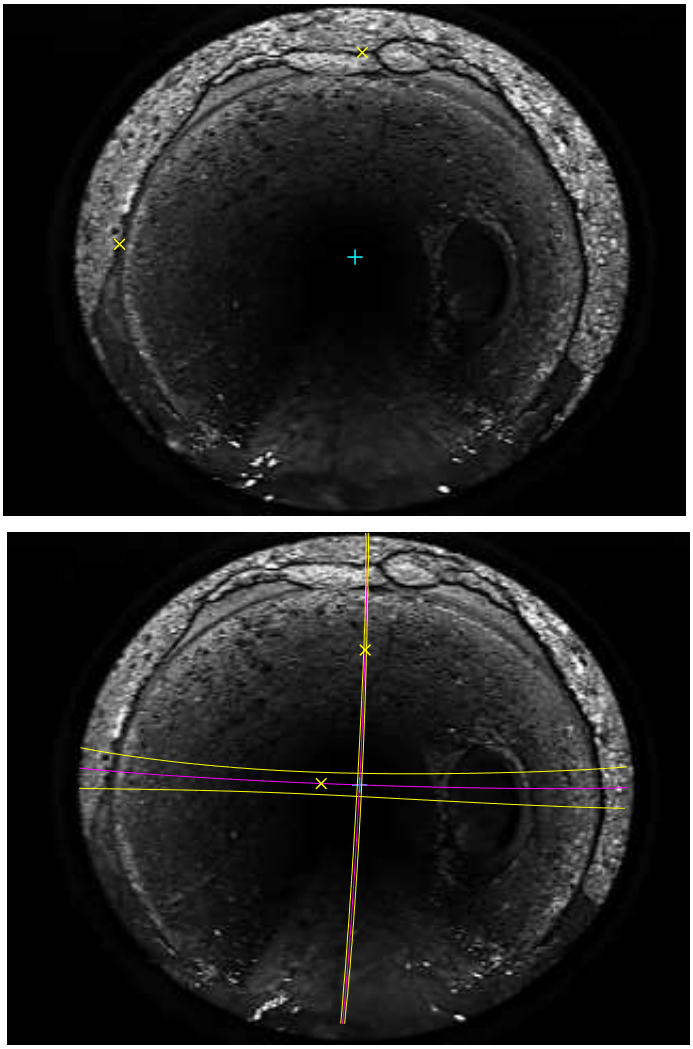


**Figure 7.1:** Estimated epipolar geometry for two fish-eye images. Two points in the first image are chosen (yellow crosses) and their epipolar curves (magenta curves) are plotted to the second image. The yellow curves are the confidence intervals of the epipolar curves. The envelope of the horizontal curve is broad because a very large value of $k^2 = 1000$ was chosen in (6.29) in order to better illustrate the error bounds. The narrow confidence interval of the vertical curve is the 95 % envelope that corresponds to a value $k^2 = 5.99$ and was used to define the search region for the correspondence in our experiments.

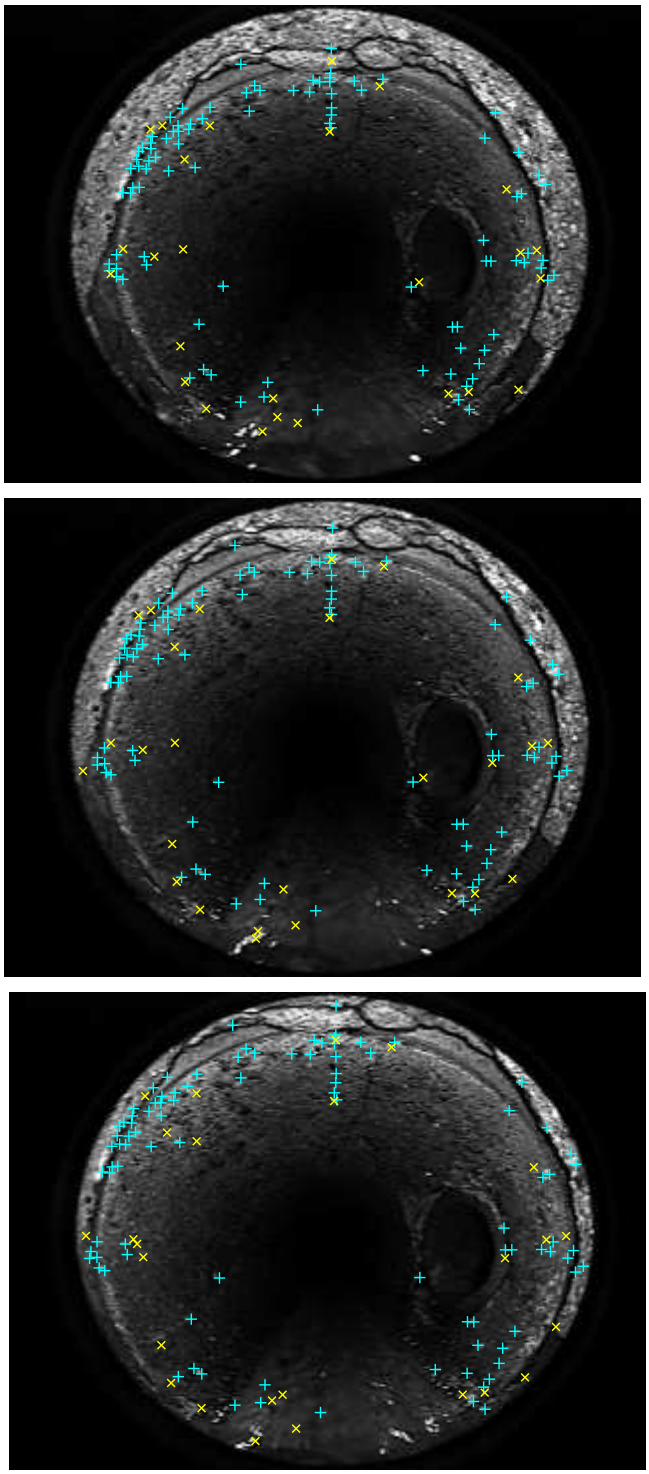


**Figure 7.2:** The initial matches over three frames. The matches denoted by yellow crosses were classified outliers on the basis of the robustly estimated three-view geometry.

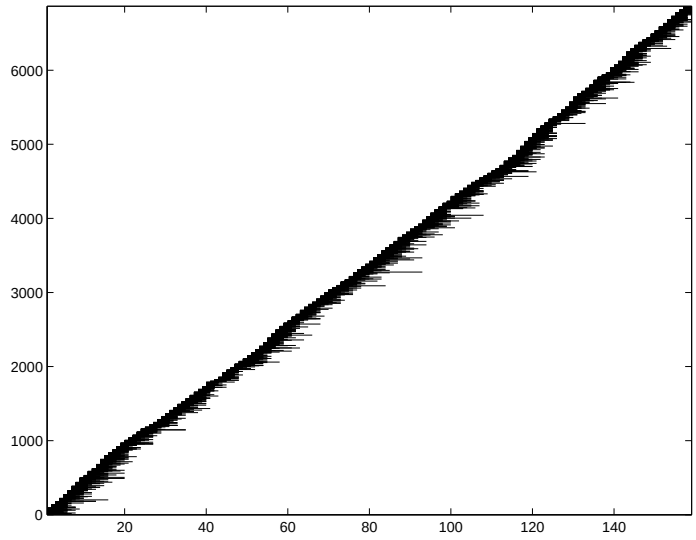



**Figure 7.3:** The tracked interest points through a sequence of 159 sewer images. There are 6864 interest points in total. Each horizontal line segment represents a correspondence chain. The length of the pipe section covered by the images is about two meters.

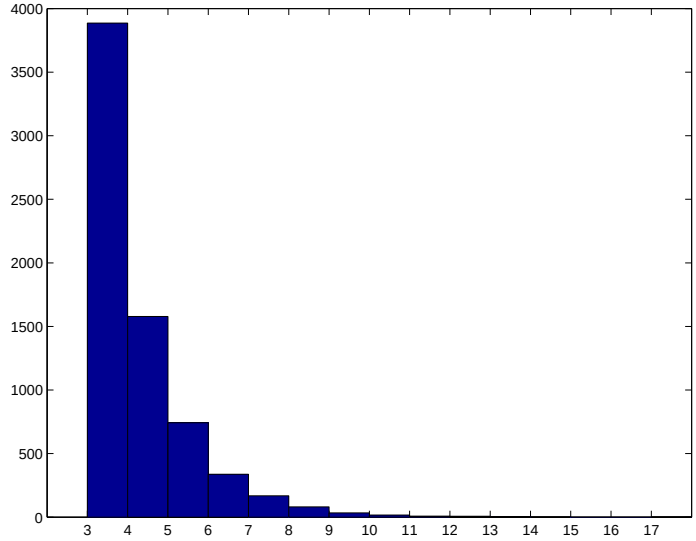



**Figure 7.4:** Histogram of the lengths of the correspondence chains shown in Fig. 7.3. Most of the chains have length of three, i.e., they appear only in one triple of images.

## 7.3 Reconstruction

After the point correspondences are established over all the images, the task is to recover their 3D coordinates. Assuming that the cameras are calibrated and the image measurement errors are Gaussian, the maximum likelihood solution for the structure and motion is obtained by minimising

$$\sum_{i=1}^{M}\sum_{j=1}^{N} \delta_{ij}\, d(\mathbf{m}_j^i, \mathcal{P}_j(\hat{\mathbf{X}}^i))^2 \tag{7.3}$$

over the 3D points $\hat{\mathbf{X}}^i$ and the external camera parameters in $\mathcal{P}_j$. Here $\mathbf{m}_j^i$ are the measured coordinates of point $i$ in view $j$ and $\delta_{ij}$ is either 1 or 0 indicating whether the point $i$ is found in view $j$. The explicit form of the imaging functions $\mathcal{P}_j$ depends on the camera model. For our fish-eye lens camera, we used the extended camera model of Section (3.2.3) with the 23 internal parameters and minimised (7.3) by the Levenberg-Marquardt algorithm.

The estimation of structure and motion by minimising (7.3) directly is called *bundle adjustment* because it involves adjusting the bundle of rays between each camera and the set of 3D points. Because reconstruction is possible only up to a similarity transformation, which has 7 degrees of freedom, the number of free parameters in the minimisation problem is $(3M + 6N - 7)$. When $N$ and $M$ increase the minimisation becomes costly and eventually impossible. Hence, one is forced to partition long image sequences into shorter sections, bundle adjust them individually and then merge the partial reconstructions.

Solving a large nonlinear minimisation problem requires a good initialisation. We compute the initialisation by a hierarchical approach building from image triplets. The method is similar to that in [Fitzgibbon98b] but in the calibrated case it is somewhat simpler and is described in the following.

### 7.3.1 Hierarchical Merging of Sub-Sequences

As described in Section 7.1 the final estimate of the three-view geometry for each image triplet is computed by minimising (6.31). Thus, each triplet is bundle adjusted separately and a metric reconstruction is obtained for each of them. The aim is to merge the reconstructions of overlapping triplets into longer sub-sequences, bundle adjust them and then merge again. Iterating the two steps, bundle adjusting and merging, leads to a hierarchical algorithm which eventually gives an initial reconstruction for the whole sequence. Because the sub-sequences are bundle adjusted at each hierarchical level the initial solution should be close to the true minimum. The advantage of the hierarchical approach is that the error is optimally distributed over the whole sequence.

Consider two 3D point sets that are the reconstructed correspondences from two overlapping sub-sequences. Due to overlap some points are common to both sets. Using these 3D point correspondences one may merge the reconstructions, i.e., transform the point sets into a common coordinate frame. In the case of uncalibrated perspective cameras the two point sets are related via a projective transformation of the 3-space but in the calibrated case the transformation is a similarity transformation. This is an advantage because there is a non-iterative algorithm for computing the least-squares solution of the similarity transformation from 3D point correspondences [Arun87, Umeyama91]. Hence, the iterative

estimation of the 3-space homographies is avoided in our implementation compared to [Fitzgibbon98b].

### 7.3.2 Results

In this section we show some examples of computed reconstructions. The experimented sewer video sequence is the same as in Fig. 7.3, where the point correspondence chains were illustrated. With our current implementation we did not bundle adjust the entire sequence of 159 views as a whole since the number of parameters is too large for a medium-scale Levenberg-Marquardt implementation. Since a single point is typically found only in very few views the structure of the optimisation problem is sparse (see Fig. 7.3), hence, sparse optimisation methods would give significant advantage [Triggs00, Hartley00]. Nevertheless, in this work we confined ourselves to compute the reconstruction by simply concatenating partial reconstructions that were bundle adjusted separately.

In Fig. 7.5, there is a three-dimensional reconstruction of points computed from correspondences over 35 images, i.e., frames 106-140 of the sequence in Fig. 7.3. The reconstruction was computed by using the hierarchical approach described above. There were 1512 points in total and the RMS (root-mean-squared) projection error after the final bundle adjustment was 0.26 pixels. There are few reconstructed points in the bottom part of the pipe since it is difficult to find correspondences from the water region. The points inside the pipe near the roof form an interesting detail in the reconstruction. They correspond to a root hanging from the roof of the pipe. Top and side views of the reconstructed points are shown in Fig. 7.6.

In order to obtain a reconstruction of the pipe section covered by the whole sequence of 159 images, we concatenated six partial reconstructions. These were computed from the following sub-sequences: 1-35, 33-55, 53-87, 85-108, 106-140, and 138-159. One of the partial reconstructions was already illustrated in Figs. 7.5 and 7.6. The others were computed in a similar way. As there are three view overlaps between successive sub-sequences, the partial reconstructions have common points and the reconstructions can be transformed into a common coordinate frame [Umeyama91].

In Figs. 7.7 and 7.8, top and side views of the concatenated reconstruction are shown. The thick part near the beginning of the pipe is a pipe socket that is visible because the pipe joint between two concrete sections is displaced. The side view shows that the pipe is bent downwards. Visual inspection of the original video showed that the pipe actually seems to be slightly bent, but the bending in Fig. 7.8 is probably exaggerated due to the accumulation of error in the concatenation. This is possible because the concatenated reconstructions have an overlap of only three views. Thus, they are merged on the basis of quite a few points on a short interval in the longitudinal direction. In order to avoid this kind of accumulation of error one should bundle adjust as long sequences as possible and when concatenating the partial reconstructions should have more significant degree of overlap.

Computing a reconstruction like that shown in Figs. 7.7 and 7.8 is computationally demanding. By our current Matlab implementation it took several days on a 2.2 GHz Pentium 4 workstation. To improve the computational efficiency,

one of the most important fields of improvement is the bundle adjustment optimisation.

As the reconstructions shown in Figs. 7.5-7.8 are only sets of 3D points, one must fit some parametric model to the reconstructed points to acquire a real 3D model of the pipe. A flexible cylindrical tube would be suitable model. The model fitting should be quite straightforward as long as the point set is not too sparse. However, the experiments were left for the future work.

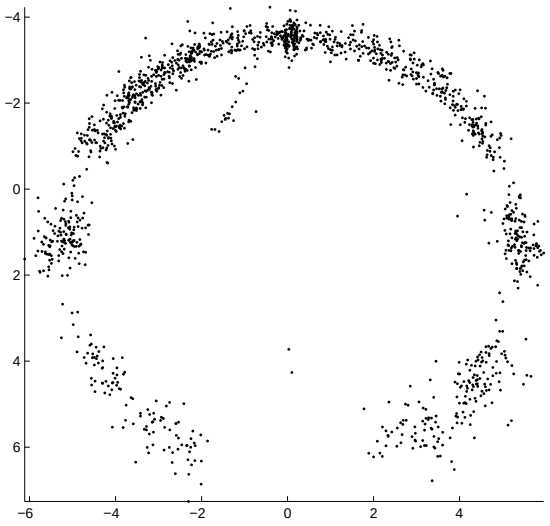
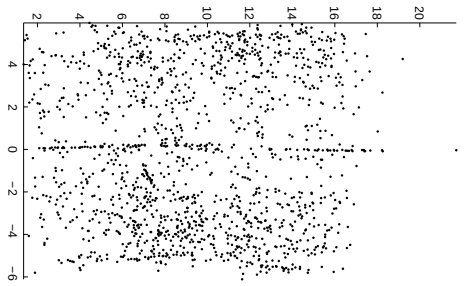
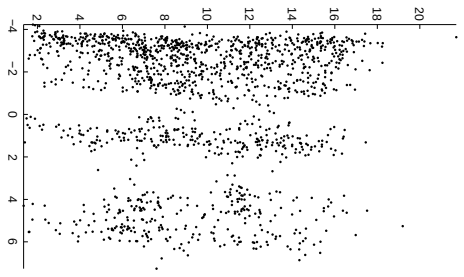



**Figure 7.5:** Front view of the reconstructed 3D points. The reconstruction is computed from point correspondences over a sequence of 35 images. There are 1512 points and the RMS projection error, i.e., the average distance between projected and measured points, is 0.26 pixels. Notice also the root hanging from the roof of the pipe.




(a)




(b)

**Figure 7.6:** Top (a) and side (b) views of the reconstructed points in Fig. 7.5

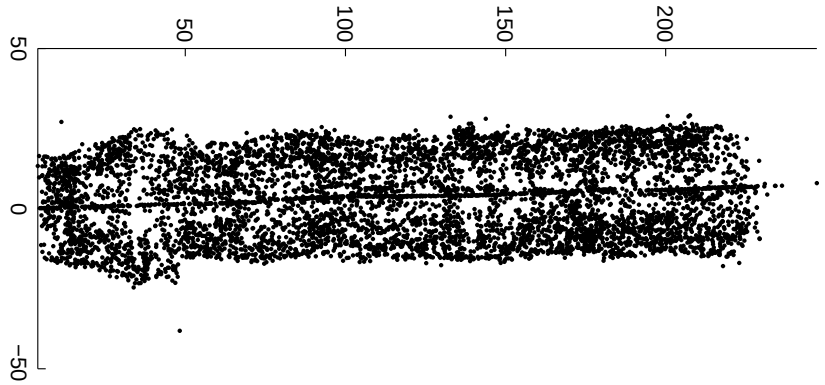



**Figure 7.7:** Top view of the reconstructed points for the sewer image sequence in Fig. 7.3. The reconstruction was computed by concatenating six shorter reconstructions, such as that in Fig. 7.6. The thick part near the beginning of the pipe is a pipe socket in a displaced pipe joint.

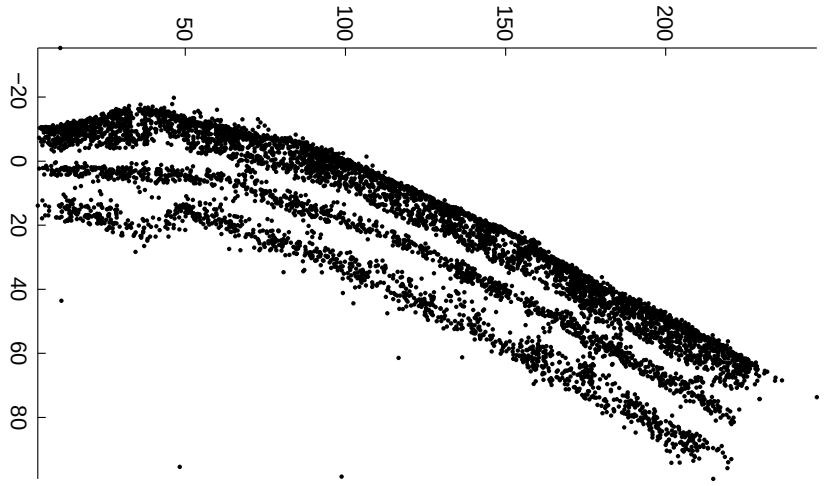



**Figure 7.8:** Side view. The pipe seems to have bent downwards. For the most part the bending is caused by the accumulation of error in the concatenation, hence, when concatenating partial reconstructions they should have greater overlap than used here. The points clearly outside the pipe are such erroneous correspondences that are consistent with the three-view geometry but do not correspond to a real object point.

# Chapter 8

# Conclusions

In this thesis, we have described how the interior shape of a sewer pipe can be automatically recovered from a video sequence taken by a moving camera. An important part of the work is the calibration of a fish-eye lens camera. A generic camera calibration method was developed and implemented as a calibration toolbox on Matlab. Second part of the work is the automatic structure recovery from calibrated image sequences. Methods of modern geometric computer vision were successfully applied to the fish-eye case and the experiments with real sewer videos showed that the shape measurements are possible in practice.

From the scientific point of view, perhaps the most important result of this thesis is the proposed general camera model and the camera calibration method, which is based on viewing a planar calibration object. The calibration experiments verified that the proposed camera model is suitable for both conventional cameras and fish-eye lens cameras. By using circular control points a relatively high level of accuracy was achieved in calibration. This is promising considering the aim of using fish-eye lenses, or other types of lenses suffering from severe distortion, in measurement purposes.

In this work, tracking and reconstruction of points from fish-eye image sequences were described in a general framework that also extends to other kinds of calibrated cameras. For example, the approach may be useful when making measurements with wide-angle lenses which deviate from the usual pinhole camera model. Hence, although our experiments were done with the sewer videos, the implemented methods have a more general applicability. They can be used also in other applications to recover scene structure from video sequences taken by a calibrated camera.

From the sewer measurement application point of view, the findings of this work are interesting. It is an important result that the shape of a sewer pipe may be recovered solely from the video. Nevertheless, a lot of work is still needed before the methods of this thesis can be used in real sewer pipe inspections. For instance, although the interest point extraction and tracking succeeded well for the test video sequence, it is possible that the approach does not work as well in all cases, especially plastic pipes may be difficult. Another drawback of the structure from motion approach is its computational complexity.

However, despite the above difficulties this thesis has been fruitful also from the application viewpoint. The camera calibration is a prerequisite for any optical measurement of a sewer pipe and is needed even if structured light were used

to measure the shape of a pipe. Measuring the cross-sectional shape of sewer pipes by using a camera and a light pattern is perhaps the direction for future work which would lead to practical improvements in sewer pipe inspections in a short run.

# Appendix A

# Projective Geometry

In this appendix, we introduce the basic concepts and notations in projective geometry. We mainly concentrate on planar geometry but the geometry of 3-space is just a straightforward generalization of the planar case for the most part. First we state two general definitions and then go on to the 2D and 3D cases. The treatment is based on the book [Hartley00].

**Definition A.0.1** *The set of one-dimensional subspaces of* $\mathbb{R}^{n+1}$ *is called the projective space of dimension* $n$ *and denoted by* $\mathbb{P}^n$.

**Definition A.0.2** *Any representation of* $\mathbf{x} \in \mathbb{P}^n$ *of the form* $\mathbf{x} = (x_1, \ldots, x_{n+1})^\top$ *is called* homogeneous coordinates *for* $\mathbf{x}$.

## A.1 Projective Geometry of 2D

### A.1.1 Points and Lines

**Homogeneous representation** In planar projective geometry points are considered as elements of space $\mathbb{P}^2$ and represented by their homogeneous coordinates. Homogeneous representation of a point $(x, y)^\top$ is obtained by adding a final coordinate of 1, i.e. $\mathbf{x} = (x, y, 1)^\top$. A homogeneous vector $\mathbf{x} = (x_1, x_2, x_3)^\top$, assuming $x_3 \neq 0$, represents the point $(x_1/x_3, x_2/x_3)^\top$ in the usual inhomogeneous coordinates.

The advantage of projective geometry is that besides points also lines are elements of the *same* space $\mathbb{P}^2$. This may be motivated as follows. A line in the plane is usually defined by an equation such as $ax + by + c = 0$. This suggests that a line may be represented as a 3-vector, $(a, b, c)^\top$. However, the above equation multiplied by an arbitrary non-zero scalar defines still the same line. Therefore it is reasonable to consider all 3-vectors related by an overall scaling as equivalent representations of a line, i.e., $\mathbf{l} = (a, b, c)^\top \hat{=} \lambda(a, b, c)^\top,\ \lambda \neq 0$.

**Incidence** A point and a line are *incident* if the former lies on the latter. In homogeneous representation this relationship has a simple expression which follows directly from the general form of the line equation:

**Result A.1.1** *The point* $\mathbf{x}$ *lies on the line* $\mathbf{l}$ *if and only if* $\mathbf{x}^\top \mathbf{l} = 0$.

**Intersection of lines** Two (non-parallel) lines intersect at a single point which lies on both lines. The intersection point has a simple algebraic expression which

is easily derived from the above incidence relation:

**Result A.1.2** *The intersection of two lines* $\mathrm{l}$ *and* $\mathrm{l}'$ *is the point* $\mathbf{x} = \mathrm{l} \times \mathrm{l}'$.

**Line joining points** A line is defined by two points lying on it. Analogously to the previous result it holds:

**Result A.1.3** *The line through two points* $\mathbf{x}$ *and* $\mathbf{x}'$ *is* $\mathrm{l} = \mathbf{x} \times \mathbf{x}'$.

**Ideal points** The homogeneous vectors of the form $(x_1, x_2, 0)^\top$ do not have inhomogeneous representation. They are interpreted as points that lie at infinity and are called *ideal points*. When result A.1.2 is applied to parallel lines (homogeneous 3-vectors whose first two coordinates have the same ratio), the intersection point is an ideal point. Therefore it is said that two parallel lines intersect at infinity.

**The line at infinity** According to result A.1.3 the line joining any two ideal points is $\mathrm{l}_\infty = (0, 0, 1)^\top$. Again, there does not exist inhomogeneous interpretation for this line and it is called *the line at infinity*. One may easily confirm that all ideal points lie on the line at infinity.

**Duality** In projective geometry points and lines have a symmetric role as can be noticed from the above results. Every element of space $\mathbb{P}^2$ has an interpretation both as a line and a point. Since it is just a question of interpretation the roles of points and lines may be swapped. This is the idea behind the *duality principle* which is stated as follows [Hartley00]: *To any theorem of two-dimensional projective geometry there corresponds a dual theorem which may be derived by interchanging the roles of points and lines in the original theorem.*

### A.1.2 Conics

A conic is a curve described by a second-degree equation in the plane. In Euclidean geometry conics are of three main types: hyperbola, ellipse and parabola. In projective geometry all of them can be represented by a $3 \times 3$ symmetric matrix. This is seen as follows.

A quadratic curve has the general form

$$ax^2 + 2bxy + cy^2 + 2dx + 2ey + f = 0, \tag{A.1}$$

which may be written in matrix form

$$(x, y, 1) \begin{bmatrix} a & b & d \\ b & c & e \\ d & e & f \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = 0. \tag{A.2}$$

Denoting the above symmetric matrix by $\mathbf{C}$ and changing to homogeneous notation gives

$$\mathbf{x}^\top \mathbf{C}\, \mathbf{x} = 0. \tag{A.3}$$

The matrix $\mathbf{C}$ may be multiplied by a non-zero scalar without altering the conic defined by (A.3). Thus, $\mathbf{C}$ is a homogeneous representation of a conic, i.e., a *homogeneous matrix* that is defined only up to scale.

### A.1.3 Projective Transformations

A planar projective transformation is an invertible mapping from points in $\mathbb{P}^2$ to points in $\mathbb{P}^2$ that maps lines to lines. It is also called a *projectivity, collineation* or a *homography*. From the computational viewpoint the following definition is convenient.

**Definition A.1.4** *A planar projective transformation is a linear transformation on homogeneous 3-vectors represented by a non-singular* $3 \times 3$ *matrix:*

$$\mathbf{x}' = \mathbf{H}\mathbf{x} \tag{A.4}$$

Like the homogeneous representation of a point is scale invariant so is the matrix representation of a projectivity. The non-singular matrix $\mathbf{H}$ is a homogeneous matrix that may be multiplied by a non-zero scale factor without altering the projective transformation. An example of a projective transformation is the *central projection* which is a linear transformation on homogeneous coordinates but non-linear on inhomogeneous coordinates.

When the points on a line $\mathbf{l}$ are mapped onto another line by a known projectivity the homogenous coordinates of the transformed line are obtained as follows.

**Result A.1.5** *Under a point transformation* $\mathbf{x}' = \mathbf{H}\mathbf{x}$*, a line* $\mathbf{l}$ *transforms to* $\mathbf{l}' = \mathbf{H}^{-\top}\mathbf{l}$.

*Proof.* $0 = \mathbf{x}^\top \mathbf{l} = \mathbf{x}^\top \mathbf{H}^\top \mathbf{H}^{-\top}\mathbf{l} = \mathbf{x}'^\top \mathbf{H}^{-\top}\mathbf{l}$ □

The transformation rule for a conic is derived in a similar manner.

**Result A.1.6** *Under a point transformation* $\mathbf{x}' = \mathbf{H}\mathbf{x}$*, a conic* $\mathbf{C}$ *transforms to* $\mathbf{C}' = \mathbf{H}^{-\top}\mathbf{C}\mathbf{H}^{-1}$.

*Proof.* $0 = \mathbf{x}^\top \mathbf{C}\mathbf{x} = \mathbf{x}^\top \mathbf{H}^\top \mathbf{H}^{-\top}\mathbf{C}\mathbf{H}^{-1}\mathbf{H}\mathbf{x} = \mathbf{x}'^\top \mathbf{H}^{-\top}\mathbf{C}\mathbf{H}^{-1}\mathbf{x}'$ □

## A.2 Projective Geometry of 3D

In projective 3-space $\mathbb{P}^3$ points and *planes* are represented by homogeneous 4-vectors. Hence, in $\mathbb{P}^3$ points and planes are dual analogous to the point-line duality in $\mathbb{P}^2$. Now the incidence relation $\boldsymbol{\pi}^\top \mathbf{X} = 0$ (zero inner product of two 4-vectors) expresses that the point $\mathbf{X}$ is on the plane $\boldsymbol{\pi}$. The ideal points lie on the plane at infinity, $\boldsymbol{\pi}_\infty = (0, 0, 0, 1)^\top$. Three planes, in a general position, intersect in a unique point (which lies on the plane at infinity in the case of coplanar planes) and three non-collinear points define a plane. The counterpart of a conic in $\mathbb{P}^3$ is a quadric, which is represented by a homogeneous $4 \times 4$ matrix. For example, ellipsoids are quadrics. The definition of projectivities in $\mathbb{P}^3$ is similar to the planar case, they are represented by non-singular homogeneous $4 \times 4$ matrices. The transformation rules of planes, quadrics and dual quadrics are similar to the corresponding transformation rules of lines, conics and dual conics in the planar case.

# Bibliography


[2d3] 2d3. *boujou* camera tracker. `http://www.2d3.com`.

[Arun87] Arun, K. S., Huang, T. S. and Blostein, S. D. Least-squares fitting of two 3-D point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698–700, 1987.

[Bakstein02] Bakstein, H. and Pajdla, T. Panoramic mosaicing with a 180° field of view lens. In *Proc. IEEE Workshop on Omnidirectional Vision*, pages 60–67. 2002.

[Basu95] Basu, A. and Licardie, S. Alternative models for fish-eye lenses. *Pattern Recognition Letters*, 16:433–441, 1995.

[Beardsley96] Beardsley, P. A., Torr, P. H. S. and Zisserman, A. 3D model acquisition from extended image sequences. In *Proc. 4th European Conference on Computer Vision*, pages 683–695. 1996.

[Bouguet04] Bouguet, J. Camera calibration toolbox for Matlab. `http://www.vision.caltech.edu/bouguetj/calib_doc/`, 2004.

[Brandt01] Brandt, S. Multi-resolution matching of uncalibrated images utilizing epipolar geometry and its uncertainty. In *Proc. IEEE International Conference on Image Processing*, pages 213–216. 2001.

[Brandt02] Brandt, S. *Theorems and algorithms for multiple view geometry with applications to electron tomography.* Ph.D. thesis, Helsinki University of Technology, 2002.

[Bräuer-Burchardt01] Bräuer-Burchardt, C. and Voss, K. A new algorithm to correct fish-eye- and strong wide-angle-lens-distortion from single images. In *Proc. IEEE International Conference on Image Processing*, pages 225–228. 2001.

[Brown71] Brown, D. C. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[Chae01] Chae, M. J. and Abraham, D. M. Neuro-fuzzy approaches for sanitary sewer pipeline condition assessment. *Journal of Computing in Civil Engineering*, 15(1):4–14, January 2001.

[Cooper98] Cooper, D., Pridmore, T. P. and Taylor, N. Towards the recovery of extrinsic camera parameters from video records of sewer surveys. *Machine Vision and Applications*, 11:53–63, 1998.

[Cooper01] Cooper, D., Pridmore, T. P. and Taylor, N. Assessment of a camera pose algorithm using images of brick sewers. *Automation in Construction*, 10:527–540, 2001.

[Cox96] Cox, I. J., Hingorani, S. L. and Rao, S. B. A maximum likelihood stereo algorithm. *Computer Vision and Image Understanding*, 63(3):542–567, 1996.

[Csurka97] Csurka, G., Zeller, C., Zhang, Z. and Faugeras, O. Characterizing the uncertainty of the fundamental matrix. *Computer Vision and Image Understanding*, 68(1):18–36, 1997.

[Devernay01] Devernay, F. and Faugeras, O. Straight lines have to be straight. *Machine Vision and Applications*, 13(1):14–24, 2001.

[Faugeras92a] Faugeras, O. Camera self-calibration: theory and experiments. In *Proc. European Conference on Computer Vision*, pages 321–334. Springer, 1992.

[Faugeras92b] Faugeras, O. What can be seen in three dimensions with an uncalibrated stereo rig? In *Proc. European Conference on Computer Vision*, pages 563–578. Springer, 1992.

[Faugeras01] Faugeras, O. and Luong, Q.-T. *The Geometry of Multiple Images*. The MIT Press, 2001.

[Fischler81] Fischler, M. A. and Bolles, R. C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. Assoc. Comp. Mach.*, 24(6):381–395, 1981.

[Fitzgibbon98a] Fitzgibbon, A. W. and Zisserman, A. Automatic 3D model acquisition and generation of new images from video sequences. In *Proc. European Signal Processing Conference*, pages 1261–1269. 1998.

[Fitzgibbon98b] Fitzgibbon, A. W. and Zisserman, A. Automatic camera recovery for closed or open image sequences. In *Proc. European Conference on Computer Vision*, pages 311–326. Springer-Verlag, 1998.

[Fitzgibbon01] Fitzgibbon, A. W. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*. 2001.

[Fraunhofer AIS] Fraunhofer AIS. Sewer robots at Fraunhofer AIS. `http://www.sewerobots.de`.

[Golub96] Golub, G. and Van Loan, C. *Matrix Computations*. The Johns Hopkins University Press, 1996.

[Gooch96] Gooch, R. M., Clarke, T. A. and Ellis, T. J. A semi-autonomous sewer surveillance and inspection vehicle. In *Proc. IEEE Intelligent Vehicles*, pages 64–69. 1996.

[Harris88] Harris, C. and Stephens, M. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conference*. 1988.

[Hartley92] Hartley, R. Estimation of relative camera positions for uncalibrated cameras. In *Proc. European Conference on Computer Vision*, pages 579–587. Springer, 1992.

[Hartley00] Hartley, R. and Zisserman, A. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[Heikkilä96] Heikkilä, J. and Silvén, O. Calibration procedure for short focal length off-the-shelf CCD cameras. In *Proc. International Conference on Pattern Recognition*, pages 166–170. 1996.

[Heikkilä00a] Heikkilä, J. Camera calibration toolbox for Matlab. `http://www.ee.oulu.fi/~jth/calibr/`, 2000.

[Heikkilä00b] Heikkilä, J. Geometric camera calibration using circular control points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1066–1077, October 2000.

[Hertzberg96] Hertzberg, J. and Kirchner, F. Landmark-based autonomous navigation in sewerage pipes. In *Proc. First Euromicro Workshop on Advanced Mobile Robots*, pages 68–73. 1996.

[Heyden96] Heyden, A. and Åström, K. Euclidean reconstruction from constant intrinsic parameters. In *Proc. International Conference on Pattern Recognition*. 1996.

[Heyden97a] Heyden, A. Projective structure and motion from image sequences using subspace methods. In *Proc. Scandinavian Conference on Image Analysis*, pages 963–968. 1997.

[Heyden97b] Heyden, A. and Åström, K. Euclidean reconstruction from image sequences with varying and unknown focal length and principal point. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*. 1997.

[Heyden00] Heyden, A. Tutorial: Multiple view geometry. Tutorial in conjunction with ICPR2000, 2000.

[Johansson01] Johansson, M., Kahl, F. and Heyden, A. VISIRE: From video to VRML. In *Proc. Scandinavian Conference on Image Analysis*. 2001.

[Kannala04] Kannala, J. and Brandt, S. A generic camera calibration method for fish-eye lenses. In *Proc. International Conference on Pattern Recognition*. 2004.

[Kolesnik02] Kolesnik, M. and Streich, H. Visual orientation and motion control of MAKRO – adaptation to the sewer environment. In *Proc. SAB'2002 - Simulation of Adaptive Behavior*. 2002.

[Kuntze98] Kuntze, H.-B. and Haffner, H. Experiences with the development of a robot for smart multisensoric pipe inspection. In *Proc. IEEE International Conference on Robotics and Automation*, pages 1773–1778. 1998.

[Leica] Leica. Leica Geosystems HDS, Inc. `http://www.cyra.com`.

[Longuet-Higgins81] Longuet-Higgins, H. C. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(10), 1981.

[Lucas81] Lucas, B. and Kanade, T. An iterative image registration technique with an application to stereo vision. In *Proc. 7th International Joint Conference on Artificial Intelligence*. 1981.

[Martinec02] Martinec, D. and Pajdla, T. Structure from many perspective images with occlusions. In *Proc. European Conference on Computer Vision*, pages 355–369. 2002.

[Mičušík03] Mičušík, B. and Pajdla, T. Estimation of omnidirectional camera model from epipolar geometry. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 485–490. 2003.

[Miyamoto64] Miyamoto, K. Fish eye lens. *Journal of the Optical Society of America*, 54(8):1060–1061, 1964.

[Morris00] Morris, D. D. and Kanade, T. Image-consistent surface triangulation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*. 2000.

[Pantsar00] Pantsar, T. *Detection of surface cracks and pipe joints in digital sewer images*. Master's thesis, Helsinki University of Technology, 2000.

[PhotoModeler] PhotoModeler. Photogrammetry software. `http://www.photomodeler.com`.

[Pollefeys98] Pollefeys, M., Koch, R. and Van Gool, L. Self calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *Proc. 6th International Conference on Computer Vision*, pages 90–96. 1998.

[Pollefeys99] Pollefeys, M. *Self-calibration and metric 3D reconstruction from uncalibrated image sequences*. Ph.D. thesis, K.U.Leuven, 1999.

[Quan96] Quan, L. Self-calibration of an affine camera from multiple views. *International Journal of Computer Vision*, 19(1):93–105, 1996.

[Rousseeuw87] Rousseeuw, P. and Leroy, A. M. *Robust Regression and Outlier Detection.* Wiley, New York, 1987.

[Ruiz-del-Solar96] Ruiz-del-Solar, J. and Köppen, M. Sewage pipe image segmentation using a neural based architecture. *Pattern Recognition Letters*, 17:363–368, 1996.

[Schenk99] Schenk, T. *Digital Photogrammetry.* TerraScience, 1999.

[Schmid00] Schmid, C., Mohr, R. and Bauckhage, C. Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000.

[Shah96] Shah, S. and Aggarwal, J. Intrinsic parameter calibration procedure for a (high-distortion) fish-eye lens camera with distortion model and accuracy estimation. *Pattern Recognition*, 29(11):1775–1788, 1996.

[ShapeCam] ShapeCam. 3D scanning solution. `http://www.eyetronics.com`.

[Shi94] Shi, J. and Tomasi, C. Good features to track. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition.* 1994.

[Slama80] Slama, C. C., editor. *Manual of Photogrammetry.* American Society of Photogrammetry, fourth edition, 1980.

[Sturm96] Sturm, P. and Triggs, B. A factorization based algorithm for multi-image projective structure and motion. In *Proc. 4th European Conference on Computer Vision*, pages 709–720. 1996.

[Sturm99] Sturm, P. and Maybank, J. On plane based camera calibation: A general algorithm, singularities, applications. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 432–437. 1999.

[Tissainayagam04] Tissainayagam, P. and Suter, D. Assessing the performance of corner detectors for point feature tracking applications. *Image and Vision Computing*, 22:663–679, 2004.

[Tomasi92] Tomasi, C. and Kanade, T. Shape and motion from image streams under orthography: A factorization approach. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[Torr00] Torr, P. H. S. and Zisserman, A. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78:138–156, 2000.

[Triggs00] Triggs, B., McLauchlan, P., Hartley, R. and Fitzgibbon, A. Bundle adjustment — a modern synthesis. In *Vision Algorithms: Theory and Practice.* Springer-Verlag, 2000.

[Umeyama91] Umeyama, S. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):376–380, 1991.

[VGG] VGG. Visual Geometry Group at the University of Oxford. `http://www.robots.ox.ac.uk/~vgg`.

[Xu96] Xu, G. and Zhang, Z. *Epipolar Geometry in Stereo, Motion and Object Recognition.* Kluwer, 1996.

[Xu98] Xu, K., Luxmoore, A. R. and Davies, T. Sewer pipe deformation assessment by image analysis of video surveys. *Pattern Recognition*, 31(2):169–180, 1998.

[Zhang95] Zhang, Z., Deriche, R., Faugeras, O. and Luong, Q.-T. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Journal*, 78:87–119, 1995.

[Zhang98] Zhang, Z. A flexible new technique for camera calibration. *Technical Report MSR-TR-98-71, Microsoft Research*, December 1998.

[Zhang00] Zhang, Z. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, November 2000.