

STUDIES IN
ROBOTIC VISION, OPTICAL ILLUSIONS AND
NONLINEAR DIFFUSION FILTERING

HENRIK MALM



LUND INSTITUTE OF TECHNOLOGY
Lund University

Centre for Mathematical Sciences
Mathematics

Mathematics
Centre for Mathematical Sciences
Lund University
Box 118
SE-221 00 Lund
Sweden
<http://www.maths.lth.se/>

Doctoral Theses in Mathematical Sciences 2003:6
ISSN 1404-0034

ISBN 91-628-5778-9
LUTFMA-1016-2003

© Henrik Malm, 2003

Printed in Sweden by KFS AB, Lund 2003

Preface

I have always had a fascination for images, be it the impressionistic masterpieces of late 19th century artists, the beauty of a well composed photograph or the modern art of computer graphics. In view of this, it is not surprising that my interest immediately awoke when I, during the pursuit of my Master's degree in computer science and technology, heard about the course in image analysis given at the Department of Mathematics. This in turn opened my eyes to the subject of mathematics itself and made me realize its value and importance. In the spring of 1998, after focusing my studies in this direction, I presented my Master's thesis on the subject of multi-scale image analysis based on partial differential equations.

With my interest in computer vision and image analysis only escalating during the work on my Master's thesis, it felt like a natural step to apply for a PhD position in the Mathematical Imaging Group. I received a position in the project "Dynamic Vision" financed by the Swedish Research Council (for Engineering Sciences). This project has been aimed at investigating different aspects of computer vision when used together with control theory in robotics. My work in the project has come to focus on the task of hand-eye calibration and intrinsic camera calibration in the setting of robotic vision. This research has led to a number of conference papers, [HM9, HM8, HM6, HM4, HM3, HM1].

However, in the beginning of my doctoral work I also partly continued the work on multi-scale image analysis and diffusion methods in image processing and had a rewarding collaboration with Johan Hult and Clemens Kaminski at the Department of Combustion physics which resulted in [HM7].

In the autumn of 2001, I had the privilege to be a guest of Prof. Yiannis Aloimonos and his research group at the Computer Vision Laboratory at the University of Maryland, MD, USA. There, I was introduced to the world of optical illusions and had a stimulating working companion in Dr. Cornelia Fermüller on this subject. The research that I took part in has resulted in [HM5] and [HM2].

The material presented in this thesis is based on the above mentioned research, conducted during my five years as a PhD student. It has been published or submitted to international journals and presented at a number of international conferences. The papers are listed below in reverse chronological order.

Main Papers

- [HM1] Malm, H., Heyden, A., Simplified Intrinsic Camera Calibration and Hand-Eye Calibration for Robot Vision, to appear in *Proc. International Conference on Intelligent Robots and Systems*, Las Vegas, October 2003.
- [HM2] Fermüller, C., Malm, H., Aloimonos, Y., Uncertainty in Visual Processes Predicts Geometrical Optical Illusions, accepted for publication in *Vision Research*.
- [HM3] Malm, H., Heyden, A., Plane-Based Camera Calibration: The Case of Pure Translation, in *Proc. Workshop on Machine Vision and Pattern Recognition*, Nara, Japan, December 2002.
- [HM4] Malm, H., Heyden, A., Self-Calibration from Image Derivatives for Active Vision Systems, in *Proc. International Conference on Automation, Robotics and Computer Vision*, Singapore, December 2002.
- [HM5] Fermüller, C., Malm, H., Aloimonos, Y., Bias in Visual Motion Processes: A Theory Predicting Illusions, in *Proc. Statistical Methods in Video Processing*, Copenhagen, June 2002.
- [HM6] Malm, H., Heyden, A., Stereo Head Calibration from a Planar Object, in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Kauai, Hawaii, December 2001.
- [HM7] Malm, H., Sparr, G., Hult, J., Kaminski, C.F., Nonlinear Diffusion Filtering of Images Obtained by Planar Laser Induced Fluorescence Spectroscopy, in *Journal of the Optical Society of America A*, 17, pp. 2148-2156, 2001.
- [HM8] Malm, H. and Heyden, A., A New Approach to Hand-Eye Calibration, in *Proc. International Conference on Pattern Recognition*, Barcelona, September 2000.
- [HM9] Malm, H. and Heyden, A., Hand-Eye Calibration from Image Derivatives, in *Proc. European Conference on Computer Vision*, Dublin, June 2000.

Subsidiary Papers

- [HM10] Olsson, T., Bengtsson, J., Johansson, R., Malm, H., Force Control and Visual Servoing Using Planar Surface Identification, in *Proc. International Conference on Robotics and Automation*, Washington D.C., May 2002.
- [HM11] Fermüller, C., Malm, H., Aloimonos, Y., Uncertainty in Visual Processes Predicts Geometric Optical Illusions, *Technical Report CS-TR-4251*, University of Maryland, May 2001.

-
- [HM12] Malm, H. and Heyden, A., A Direct Method for Hand-Eye Calibration and Depth Reconstruction, in *Proc. SSAB Symposium on Image Analysis*, Halmstad, March 2000.
- [HM13] Malm, H., Sparr, G., Hult, J., Kaminski, C.F., An Application of Non-linear Diffusion Filtering on PLIF Images of Flames, in *Proc. SSAB Symposium on Image Analysis*, Halmstad, March 2000.
- [HM14] Malm, H., Review and Practical Comparisons of Multiscale Analysis Methods, in *Proc. SSAB Symposium on Image Analysis*, Uppsala, March 1998.

Acknowledgments

First of all I would like to thank my supervisors Anders Heyden and Gunnar Sparr for their helpful comments and support. I am also grateful to the other members of the Mathematical Imaging Group who has taken their time to discuss theoretical and practical issues related to this thesis. In this respect I would especially like to thank Henrik Stewenius, Fredrik Kahl, Magnus Oskarsson, Kalle Åström, Björn Johansson and Anders P Eriksson.

For their careful reading of this thesis, I would like to thank Gunnar Sparr, Anders Heyden, Magnus Oskarsson and Anders Ericsson.

Another group of people who has always been there when I needed their help and support is the team at the Robotics Lab at the Department of Automatic Control. Special thanks goes out to Tomas Olsson, Johan Bengtsson and Anders Robertsson.

In the autumn of 2000, I had a very enjoyable and rewarding stay at the Computer Vision Lab at the University of Maryland. Warm thanks to Yiannis Aloimonos and Cornelia Fermüller for this opportunity and for the fruitful collaboration.

I am also thankful to Johan Hult and Clemens Kaminiski for the successful collaboration and for answering my questions on combustion physics and to Joachim Weickert for discussions on scale-space theory.

During my time at the Department of Mathematics I have always valued the friendly and enjoyable atmosphere, among the present and former PhD students, very highly. Thank you all for the long wonderful coffee breaks and lunches.

Finally, thanks to my family and friends for always being there.

Contents

1	Introduction	1
I	Calibration Methods for Robotic Vision	5
2	Preliminaries	7
2.1	Basic Concepts	7
2.1.1	Notions from Projective Geometry	7
2.1.2	Modeling of the Camera	9
2.1.3	The Optical Flow Constraint Equation	12
2.2	Calibration Background	13
2.2.1	Intrinsic Camera Calibration	13
2.2.2	Hand-Eye Calibration	14
2.2.3	Stereo Head Calibration	16
3	Plane-based Calibration	19
3.1	Basic Theory	19
3.2	Plane-based Calibration using Translations	20
3.2.1	Calculating the Hand-Eye Orientation	23
3.3	Plane-based Stereo Head Calibration	24
3.3.1	Stereo Head Calibration using Translations	24
3.3.2	Stereo Head Calibration using General Motions	26
3.4	Experiments	28
3.4.1	Experiments using Translational Motions	29
3.4.2	Experiments using a Stereo Head and General Motions	33
3.5	Conclusions	35
4	A Direct Method for Calibration	37
4.1	A Basic Photometric Relation	38
4.2	Calibration from Translational Motions	40
4.3	A Photometric Constraint for General Motions	42
4.4	Calculation of Camera Position	44
4.5	A Comment on a Nonlinear Optimization Stage	45
4.6	Estimating the Derivatives	45
4.7	Experiments	46

CONTENTS

4.7.1	Experiments using Computer Generated Images	46
4.7.2	Experiments using a Real Robot Vision System	51
4.8	Conclusions	53
II	Geometrical Optical Illusions	55
5	Uncertainty in Visual Processes Predicts Geometrical Optical Illusions	57
5.1	Bias in Gray Value Estimation	60
5.1.1	Basic Theory	60
5.1.2	Illustrative Examples	62
5.1.3	A Comment on Smoothing and Line Intersection	69
5.2	Bias in the Estimation of Line Elements	69
5.2.1	Basic Theory	72
5.2.2	Illustrative Examples	74
5.3	Bias in Motion Estimation	76
5.3.1	Illustrative examples	78
5.4	Other Theories of Illusion	82
5.5	Discussion and Conclusions	84
III	Nonlinear Diffusion in Image Processing	87
6	Nonlinear Diffusion Filtering of Images Obtained by Planar Laser-induced Fluorescence Spectroscopy	89
6.1	Nonlinear Diffusion and Scale-Space Theory	89
6.1.1	Numerical Implementation	92
6.2	Planar Laser-Induced Fluorescence Spectroscopy	96
6.3	Properties of Nonlinear Diffusion	99
6.4	Nonlinear Diffusion Filtering of OH PLIF Images	105
6.5	Flame Front Identification	107
6.6	Conclusions	111
A	Expected Value of Least Squares Solutions for the Estimation of Intersections and Motion	113
	Bibliography	117

Chapter 1

Introduction

The work collected in this thesis can at first sight seem to be quite diverse. It is divided into three parts, which all deal with images in one form or the other, but the different settings range from robotic camera sensors over human perception to physical measurement setups. However, entering deeper into each part, common theoretical ideas and computational aspects will become apparent which reveals unexpected connections.

An overview of the thesis and its three subjects will here be presented while deeper introductions are given in the beginning of each part.

Part I: Calibration Methods for Robotic Vision

Making robots see has been a task that has fascinated researchers ever since the beginning of the study of image analysis and computational vision. The interest in this stems not only from the sci-fi dream of constructing humanoids, but also from technical solutions which aim at emulating (and refining) the human movability and visional capacity to perform monotonic and dangerous tasks. The affordability and versatility of digital cameras often makes them attractive for industrial applications in favor of other robotic sensors such as range laser scanners.

When using a camera as a visual sensor for a robot the camera could either be located at a fixed position outside the workspace of the robot or it could be attached to a limb of the robot and follow its movements. The work presented here mainly concerns the latter case.

Before the camera can be used to guide the robot in performing a task, such as finding and grasping objects, there are some quantities in the setup that usually need to be calculated. These include intrinsic parameters in the computational model of the camera such as e.g. the focal length. The procedure of estimating these parameters is called **intrinsic camera calibration**. Further, in the case that the camera is attached to the hand (end-effector) of the robot, the relative orientation and position of the camera in relation to the robot hand need to be determined. Calculating this rigid transformation is referred to as **hand-eye calibration**. Also, in the case of the utilization of two or more cameras, the relative orientation and position among these needs to be obtained.

In this first part of the thesis, two different approaches to the above mentioned calibration tasks are presented and analyzed. The presentation starts with some background material and theoretical preliminaries in Chapter 2. The two approaches are then discussed in Chapters 3 and 4, respectively.

Part II: Geometrical Optical Illusions

In the next part of the thesis we turn to the fascinating subject of visual illusions and specifically to the subgroup known as **geometrical optical illusions**. These are illusions that can be seen in two dimensional geometrical patterns and line drawings. The purpose of this work is to build an explanatory framework encompassing many of the illusions in this class. We propose that bias in the estimation processes of different features in the image is a major contributive factor to the perceived illusory effects. This bias results from errors stemming from uncertainty and noise in the visual process. The following types of errors are considered: Errors in the position and gray values of image points, errors in the position and orientation of small edge elements (edgels) and errors in the direction and lengths of motion vectors perpendicular to edges. For each type of error model, examples of illusions best explained by that model are presented and analyzed.

No assumptions are made on the specific type of visual system that is considered in this part of the thesis, artificial or natural, although the findings clearly apply to the human visual system. The discussion highlights the importance of considering noise and uncertainty in any visual process. This also applies to computer vision algorithms and should be considered for example when constructing calibration methods as those in Part I.

There are also more specific theoretical connections to the other parts of the thesis. The estimation of motion discussed here is related to the calculations in the calibration method presented in Chapter 4 of Part I. Further, the model of errors in gray value is closely related to the scale-space theory from which the nonlinear diffusion filtering techniques discussed in Part III can be derived. Part II consists of Chapter 5 together with Appendix A.

Part III: Nonlinear Diffusion in Image Processing

The last part of the thesis is devoted to the use of **nonlinear diffusion filtering** to process images obtained by **planar laser-induced fluorescence spectroscopy** (PLIF). PLIF spectroscopy is an imaging technique that, due to its high temporal resolution and sensitivity, is very well suited to study combustion processes occurring in turbulent flames. Of special interest is to extract the flame front in the obtained images, since this is where the chemical reactions in the flame take place. However, because of limited detector resolution and inflicting physical phenomena, the flame boundaries in the PLIF images are not very sharp. Also, the imaging process suffers from a high noise level, due to e.g. stray laser light.

To process these images, nonlinear diffusion filtering is ideal. It will smooth out noise and uninteresting detail while sharpening the flame boundaries. The usage of nonlinear diffusion techniques in image processing originally stems from the theory of scale-spaces. To create well-defined multi-scale image representations the linear diffusion equation is commonly used. However, linear diffusion filtering will in the process of smoothing the

image also blur and dislocate edges. To prevent this, Perona and Malik [104] proposed a nonlinear diffusion equation where the amount of smoothing depends on the differential structure in the image, so that interesting edges can be preserved and enhanced.

In Chapter 6, which solely constitutes Part III, introductions to nonlinear diffusion and scale-space theory in image processing and to PLIF imaging of turbulent flames are given. Further, the general properties of nonlinear diffusion is examined and its application and adaption to the current PLIF images is discussed and analyzed.

Contributions

Starting with Part I, the theoretical contributions lies in Chapter 3 and 4. The work presented in Chapter 3 constitutes extensions of the theory on plane-based intrinsic camera calibration originally proposed by Zhang [141] and Sturm and Maybank [119]. The current theory has been developed in [80], [81] and [83].

The calibration method presented in Chapter 4 was inspired by the pioneering work of Negadaripour and Horn [96] and Horn and Weldon [62] on direct methods for scene reconstruction. During the early development of the current theory the work on direct estimation of structure and motion by Stein and Sashua [117] came to our attention. The theory in Chapter 4 is closely related to this work but instead of structure estimation it focuses directly on calibration and is adapted to a robot vision setting. The presented theory also has a lot in common with the work on self-calibration from image derivatives by Brodsky and Fermüller [8].

To the theory presented in Chapter 3 and 4, A. Heyden has contributed with some ideas and comments. The final version of the theory in Chapter 4 is the result of fruitful discussions with H. Stewenius. The theory has gradually been developed in [78], [79] and [82].

The theoretical contributions in Part II lies mainly in the idea of viewing illusory effects as the result of bias in visual estimation processes. This connects a large number of illusions that formerly has been viewed as unrelated. The work on bias in the estimation of gray values and line intersections has been a collaboration with C. Fermüller and Y. Aloimonos. The work on bias in motion estimation, particularly in view of the Ouchi illusion, is work previously presented by C. Fermüller, R. Pless and Y. Aloimonos [101] and is included for the sake of completeness. The current presentation is an adaptation of [36].

Finally, the contributions of Part III lies in the thorough study of the properties of nonlinear diffusion filtering and its successful application to images obtained by planar laser-induced fluorescence spectroscopy. There are no new contributions when it comes to the applied theory, which is originally due to Perona and Malik [104]. The special form of the equation used is due to Weickert [130]. The influence of the parameters in this equation on the present application is carefully studied. The presentation follows [84] with some extensions on scale-space theory.

Part I

**Calibration Methods for Robotic
Vision**

Chapter 2

Preliminaries

Some basic concepts needed in the forthcoming chapters will be introduced in this chapter. The different calibration tasks discussed in this part of the thesis are also presented with some references to previous work. More specific background material is then given in the beginning of each chapter.

2.1 Basic Concepts

We start by briefly introducing some notions from projective geometry that will be used in the theoretical discussions. These notions are of particular relevance to Chapter 3 but will also assist the derivation of the camera model in Section 2.1.2. Finally, the optical flow constraint equation will be derived which is a cornerstone in the calibration method presented in Chapter 4.

2.1.1 Notions from Projective Geometry

Projective geometry is a powerful mathematical tool, very well suited for theoretical reasoning in geometric computational vision. We will here review the notions from projective geometry that are utilized in this part of the thesis. For more thorough presentations cf. [19, 113, 55]. We begin with a basic definition.

Definition 2.1.1. Let V be a vector space of dimension $n+1$. The set of one dimensional linear subspaces of V is called the **projective space of dimension n** , denoted by \mathbb{P}^n . ■

V will be referred to as the **underlying vector space**. A **point** in \mathbb{P}^n is represented by a vector

$$\mathbf{x} = [x_1 \quad x_2 \quad \dots \quad x_{n+1}]^T \in V ,$$

where at least one $x_i \neq 0$. Two vectors \mathbf{x} and \mathbf{y} represent the same point in \mathbb{P}^n if and only if there exists a scalar $\lambda \neq 0$ such that $\mathbf{x} = \lambda\mathbf{y}$. Each such representation is called the **homogeneous coordinates** of the point. In \mathbb{P}^2 and \mathbb{P}^3 , we will often use the notations $\mathbf{x} = [x \quad y \quad z]^T$ and $\mathbf{X} = [X \quad Y \quad Z \quad W]^T$, respectively. For a point in \mathbb{P}^2 where $z \neq 0$ and a point in \mathbb{P}^3 where $W \neq 0$ the representations $\mathbf{x} = [x \quad y \quad 1]^T$ and $\mathbf{X} = [X \quad Y \quad Z \quad 1]^T$ are most commonly used in this text.

Definition 2.1.2. A **projective transformation**, or **homography**, $\mathbb{P}^n \ni \mathbf{x} \mapsto \mathbf{y} \in \mathbb{P}^n$, is a bijective transformation which, using homogeneous coordinates, can be written

$$\lambda\mathbf{y} = A\mathbf{x} ,$$

where A is a non-singular $(n + 1) \times (n + 1)$ matrix. ■

Note that the matrix associated with a given projective transformation is defined up to a nonzero scale factor. The notation \sim will occasionally be used to represent equality up to scale. In this notation, $\mathbf{y} \sim A\mathbf{x}$ in Definition 2.1.2.

Definition 2.1.3. A **projective basis** in \mathbb{P}^n is a set of $n + 2$ points such that for no subset of $n + 1$ points, the corresponding representatives in the underlying space V are linearly dependent. ■

The following canonical basis, called the **standard projective basis**, is often used: $\mathbf{e}_i = [0 \ \dots \ 1 \ \dots \ 0]^T$, $i = 1, \dots, n + 1$, where 1 is in the i th position, and $\mathbf{e}_{n+2} = [1 \ \dots \ 1]^T$. By the next theorem, the so-called **fundamental theorem of projective geometry** cf. [113], it follows that any projective basis can be transferred to the standard basis.

Theorem 2.1.1. *Given a projective basis, $(\mathbf{x}_i)_{i=1}^{n+2}$, in \mathbb{P}^n , then there exists a projective transformation A , that maps the basis to the standard projective basis, i.e.*

$$\lambda_i \mathbf{e}_i = A\mathbf{x}_i, \quad \lambda_i \neq 0, \quad i = 1, \dots, n + 2.$$

Moreover, A is uniquely determined up to a scale factor. ■

We will also need projective transformations between spaces of different dimensions.

Definition 2.1.4. A **projection** from \mathbb{P}^m to \mathbb{P}^n , where $m > n$, is a surjective transformation that can be written

$$\lambda \mathbf{y} = A\mathbf{x} ,$$

where A denotes a $(n + 1) \times (m + 1)$ matrix of rank $n + 1$. ■

Note that, just as for a projective transformation from n to n , a projection is only defined up to a nonzero scale factor.

Two notions from projective geometry that will be used in Chapter 3 are the plane at infinity and the absolute conic. These special geometric primitives and their roles in projective geometry will here briefly be presented.

The **affine space** \mathbb{A}^n is defined as \mathbb{P}^n with the omission of one hyperplane. In the canonical frame, this plane is normally chosen as $x_{n+1} = 0$ and is called the **plane at infinity**. We denote the plane at infinity by π_∞ . The coordinate representation of the points, in the canonical frame, is described by the one-to-one map

$$\mathbb{A}^n \ni (x_1, x_2, x_3) \leftrightarrow (x_1, x_2, x_3, 1) \in \mathbb{P}^n \setminus \pi_\infty .$$

The points on the plane at infinity can be interpreted as directions in \mathbb{A}^n , where the point $[x_1 \ x_2 \ \dots \ x_n \ 0]^T$ corresponds to the direction $[x_1 \ x_2 \ \dots \ x_n]^T \in \mathbb{A}^n$. A

projective transformation preserving the plane at infinity is called an **affine transformation**. It is easy to see that such transformations are characterized by having a representation

$$\begin{bmatrix} A & \mathbf{b} \\ 0 & c \end{bmatrix},$$

where A is a non-singular $n \times n$ matrix, \mathbf{b} denotes an n -vector and c is a non-zero scalar. The affine transformations form a subgroup of the projective transformation group called the **affine transformation group**.

In the same way as the plane at infinity determines the embedding of \mathbb{A}^n in \mathbb{P}^n , the Euclidean space \mathbb{E}^n can be embedded in \mathbb{A}^n by singling out a special **quadric** on the plane at infinity. Generally, a quadric in \mathbb{P}^n is defined as the set of points $\mathbf{x} \in \mathbb{P}^n$ such that $\mathbf{x}^T Q \mathbf{x} = 0$ where Q is a symmetric $(n+1) \times (n+1)$ matrix. In the canonical frame, the planar quadric determining \mathbb{E}^n is normally chosen as the intersection of the quadric $\sum_{i=1}^n x_i^2 = 0$ with the plane at infinity $x_n = 0$. This quadric is called the **absolute quadric** and is denoted by Ω_∞ . An affine transformation that keeps the absolute quadric invariant is called a **similarity transformation**, and can be written

$$A = \begin{bmatrix} \sigma R & \mathbf{b} \\ \mathbf{0} & c \end{bmatrix},$$

where R is an orthogonal $n \times n$ matrix, \mathbf{b} denotes an n -vector and c denotes a non-zero scalar. In particular, if $\det(A) = 1$, A is called a **Euclidean transformation**. The Euclidean transformations form a subgroup of the affine transformation group. In \mathbb{P}^3 , Ω_∞ is the intersection of $\sum_{i=1}^3 x_i^2 = 0$ and $x_4 = 0$ and is called the **absolute conic**. In Chapter 3 the projection (or image) of the absolute conic in the image plane plays a central role.

2.1.2 Modeling of the Camera

To be able to draw any conclusions about an observed scene from images taken by a camera, we need to have a model for how the images are created. Although most cameras use an intricate set of lenses to modify the path of light, a very simple projection model called the **pinhole camera model** has shown to be sufficient for most applications. In this model a point \mathbf{X} in \mathbb{P}^3 is perspective projected onto an **image plane**, forming a model for \mathbb{P}^2 . That is, if we draw a line between the object point \mathbf{X} and the **focal point**, also called the **camera center**, the corresponding image point \mathbf{x} will be the point of intersection of this line and the image plane, cf. Figure 2.1.

Suppose the camera and the scene is embedded in a Euclidean space, which has a metric. The point on the image plane which is closest to the focal point is then called the **principal point** and the distance between these two points is called the **focal length**, denoted by f . The **optical axis** is the axis through the focal point and the principal point.

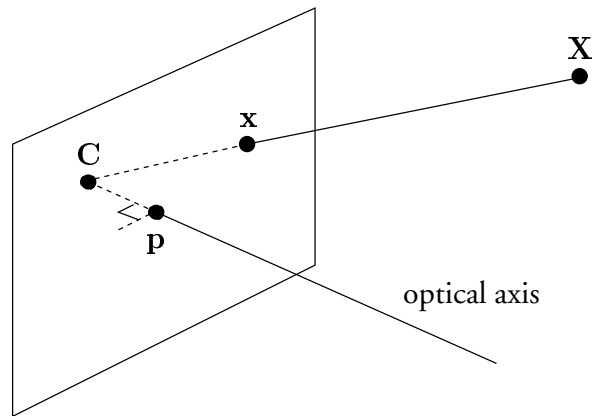


Figure 2.1: The pinhole camera model. \mathbf{X} is an object point whose projection in the image plane is the image point \mathbf{x} . \mathbf{C} is the focal point (or camera center) and \mathbf{p} is the principal point.

If we introduce a coordinate system in the underlying Euclidean space such that the focal point is at the origin $(0, 0, 0)$, the optical axis coincides with the Z -axis and such that the image plane is $Z = f$, then the projection of an object point (X, Y, Z) to an image point (x, y) simply becomes

$$\begin{cases} x = f \frac{X}{Z} \\ y = f \frac{Y}{Z} \end{cases} . \quad (2.1)$$

We often also scale the coordinate system so that $f = 1$. The projection equations (2.1) can be reformulated as the matrix equation

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} [I_{3 \times 3} \quad \mathbf{0}_{3 \times 1}] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \quad (2.2)$$

where $\lambda = Z$.

To make the model more general we allow for an independent coordinate system transformation in the image plane. For example, if the principal point has coordinates

(x_0, y_0) in the image plane, the projection (2.2) instead becomes

$$\lambda \mathbf{x} = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} [I_{3 \times 3} \quad \mathbf{0}_{3 \times 1}] \mathbf{X}. \quad (2.3)$$

Image coordinates are usually measured in pixels and the number of pixels per unit distance can differ between the x - and y -axes. The ratio between these two scale factors is called the **aspect ratio** and is usually denoted by γ . Also, to allow for non-perpendicular image axes the **skew** s is introduced. If we take these two additional quantities into consideration the projection equation becomes

$$\lambda \mathbf{x} = K [I_{3 \times 3} \quad \mathbf{0}_{3 \times 1}] \mathbf{X}, \quad (2.4)$$

where

$$K = \begin{bmatrix} f & sf & x_0 \\ 0 & \gamma f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

In practice, however, the skew can often be neglected, i.e. $s = 0$. The parameters included in K are usually called the **intrinsic parameters** of the camera.

It is often convenient to fix a world coordinate system in the scene that differs from the one attached to the camera. The projection equation will then include a Euclidean transformation that first transforms the object point from the world to the camera coordinate system before projecting it onto the image plane. Let this Euclidean transformation be given by

$$\mathbf{X} \mapsto \begin{bmatrix} R & -R\mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \mathbf{X}, \quad (2.6)$$

where R denotes a 3×3 rotation matrix and \mathbf{t} is a non-homogeneous translation vector. The null space of the matrix representing the projection model determines the focal point of the camera. When choosing the fourth column in the transformation matrix above as $-R\mathbf{t}$, the focal point has the coordinates $\mathbf{C} = [t^T \quad 1]^T$. Using this transformation together with (2.4) yields

$$\lambda \mathbf{x} = K \underbrace{[R \quad -R\mathbf{t}]}_P \mathbf{X} = P\mathbf{X}. \quad (2.7)$$

The parameters R and \mathbf{t} are often called the **extrinsic parameters** and the 3×4 matrix P is called the **camera matrix**. If the calibration matrix K is unknown the camera is said to be **uncalibrated**.

Observe that there are 11 independent parameters in this camera model, 3 for the orientation R of the camera, 3 for the position t of the camera and 5 parameters in the calibration matrix K . Note that this is the same number of parameters as in a projection from \mathbb{P}^3 to \mathbb{P}^2 . In fact, there is a one-to-one mapping from a general 3×4 projection matrix P to a camera with 11 independent intrinsic and extrinsic parameters.

2.1.3 The Optical Flow Constraint Equation

A constraint equation on the motion of the image points in an image sequence will here be derived. Think of an image sequence as a continuous stream of images. Let $I(x, y, t)$ be the gray level intensity at the image point (x, y) at time t . Further, let $u(x, y)$ and $v(x, y)$ denote the components of the motion field in the image plane in the x and y directions, respectively. Assuming that the irradiance at a point in the scene is (locally) invariant to the viewing angle and distance we expect the following equation to be fulfilled,

$$I(x + u\delta t, y + v\delta t, t + \delta t) = I(x, y, t) . \quad (2.8)$$

That is, the intensity at point $(x + u\delta x, y + v\delta y)$ at time $t + \delta t$ will be the same as the intensity at (x, y) at time t . If it is assumed that the intensity varies smoothly with x , y and t , the left hand side in (2.8) can be expanded in a Taylor series giving

$$I(x, y, t) + \delta x \frac{\partial I}{\partial x} + \delta y \frac{\partial I}{\partial y} + \delta t \frac{\partial I}{\partial t} + e = I(x, y, t) . \quad (2.9)$$

Here, e is the error term of order $\mathcal{O}((\delta t)^2)$. By cancelling out $I(x, y, t)$, dividing by δt and taking the limit as $\delta t \rightarrow 0$, we obtain

$$I_x u + I_y v + I_t = 0 , \quad (2.10)$$

where

$$\begin{cases} u = \frac{\delta x}{\delta t} = \dot{x} , \\ v = \frac{\delta y}{\delta t} = \dot{y} , \end{cases} \quad (2.11)$$

and $I_x = \frac{\partial I}{\partial x}$, $I_y = \frac{\partial I}{\partial y}$ and $I_t = \frac{\partial I}{\partial t}$.

Equation (2.10) is called the **optical flow constraint equation** since it is often used to calculate the estimation of the motion field (u, v) called optical flow. Rewrite (2.10) in the form

$$(I_x, I_y) \cdot (u, v) = -I_t . \quad (2.12)$$

It is clear that the optical flow constraint equation determines the motion field in the direction of the intensity gradient (I_x, I_y) . However, the component perpendicular to

the gradient is still undetermined. This ambiguity is referred to as the **aperture problem**. There has appeared numerous techniques for calculating optical flow, cf. [6] for an overview. Every such technique uses some kind of smoothness assumption on the motion field to deal with the aperture problem. One of the simplest, most accurate and most popular optical flow methods is originating from a paper by Lucas and Kanade [75]. They apply a locally constant model for the motion field (u, v) and solves a weighted system of equations of the form (2.10) by least squares estimation in a small neighborhood around each point. This model for motion estimation will be applied in Part II of this thesis when discussing errors in the perception of motion.

The calibration model presented in Chapter 4 makes use of (2.10) but does not rely on the estimation of optical flow. That is, it only uses the information of the motion field perpendicular to edges, which is obtained directly from the spatial (I_x, I_y) and temporal I_t image derivatives. The component of the motion field in the direction of the gradients is usually called the **normal flow**. Methods which use the normal flow directly for visual computations have been coined **direct methods**, cf. [96, 62, 117]. See Chapter 4 for further details.

2.2 Calibration Background

2.2.1 Intrinsic Camera Calibration

Intrinsic camera calibration is the process of estimating the parameters in the matrix K (2.5) in the camera model. After these intrinsic parameters have been determined, metric information of the scene in view can be extracted from the images.

A standard method for camera calibration is the so-called **DLT algorithm** (Direct Linear Transform). Here, the geometry of an observed object in 3D space is assumed to be known with very good precision. Usually this object consists of two or three perpendicular planes with some sort of grid pattern where the relative position of the grid corners are known. The projection of these corners are then extracted from the images of the object using e.g. the Harris corner detector [54]. In this way both the object points \mathbf{X}_i and the image points \mathbf{x}_i are known and the linear relation (2.7) is used to calculate the camera matrix P . A minimum number of 6 points are needed for the calculation.

After the linear calculation of P the estimation is most often improved by minimizing the the following reprojection error over P ,

$$\sum_i \|\mathbf{x}_i - P\mathbf{X}_i\|^2 \quad (2.13)$$

This can be done e.g. using Levenberg-Marquardt minimization [85].

When a good estimate of P has been obtained the first three columns of the 3×4 matrix can be factored into an upper triangular matrix with positive diagonal elements

and a rotation matrix, by a modified **QR-decomposition** [46]. This yields the calibration matrix K and the pose (extrinsic parameters) of the camera, R and t , according to (2.7).

A practical problem with the method outlined above is that the calibration object is required to be three-dimensional for the calculation of P . The construction of an accurate calibration object is considerably simplified if the object is allowed to be planar. A method for plane-based camera calibration was introduced independently by Zhang [141] and Sturm and Maybank [119]. The calibration theory presented in Chapter 3 constitutes extensions of the theory presented in these papers. See Chapter 3 for further details on calibration from a planar object.

Another category of calibration techniques are the so-called **self-calibration** (or auto-calibration) algorithms, cf. [88, 31, 58, 74, 122]. See [55] for an overview. These methods do not use any calibration object and rely only on the rigidity of the scene and different assumptions on the intrinsic parameters. Often it is assumed that the calibration parameters stay fixed throughout the image sequence, but self-calibration from much weaker assumptions is possible [59, 107]. All the methods cited above rely on the extraction and matching of features such as corner points. However, an algorithm for self-calibration based directly on the spatial and temporal image derivatives has also been proposed, cf. [8]. This algorithm is related to the calibration method presented in Chapter 4.

2.2.2 Hand-Eye Calibration

In this thesis a robot vision system where the camera (or multiple cameras) is attached to the end-effector of a robot arm is considered, cf. Figure 2.2. The end-effector will be referred to as the **robot hand**.

The process of estimating the Euclidean transformation between the coordinate frame of the camera and the coordinate frame of the robot hand is called **hand-eye calibration**. In the literature this transformation is usually denoted by X and we will use this notation in this section also. It should not be mixed up with the coordinates of a 3D point. The rotational part of X will occasionally be called the hand-eye orientation and the translational part will be called the hand-eye translation. The problem of hand-eye calibration is traditionally formulated as solving the following matrix equation with respect to X ,

$$AX = XB, \quad (2.14)$$

where $A = A_2^{-1}A_1$ and $B = B_2B_1^{-1}$. The transformations A_1 , A_2 , B_1 and B_2 are illustrated in Figure 2.3. If the robot has been calibrated, i.e. if we can move the robot hand to a specified orientation and position with high accuracy, the transformations A_1 and A_2 are obtained directly. The transformations B_1 and B_2 from a calibration object to the camera coordinate frames are obtained as the extrinsic parameters from the camera calibration methods discussed in the last section.

A number of different approaches has been proposed for solving (2.14). In the earliest works [115, 124, 17] linear solution methods were developed which separated the calculation of the rotational part and the translational part of the hand-eye transformation



Figure 2.2: A robot with a hand-eye configuration.

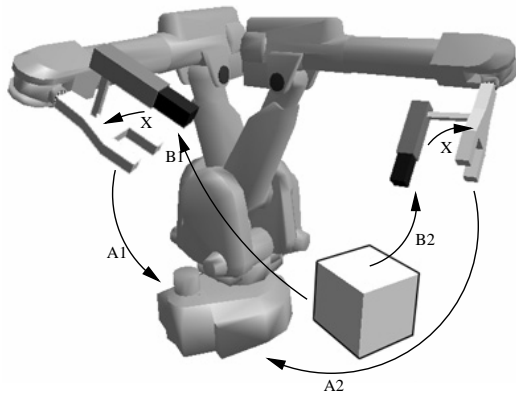


Figure 2.3: The transformations in the hand-eye calibration problem [98].

X . For each motion (image pair) an instance of (2.14) is established and in [124] it was shown that a minimum of two motions with non-parallel rotation axes are required to solve for X . In [144, 24, 109] the equation (2.14) is slightly reformulated to also include the unknown transformation from the robot base coordinate frame to the coordinate frame of the calibration object. Methods for solving for the rotational and translational part of X simultaneously have been presented in [15] and [20], where dual quaternions are used in [20] for the algebraic formulation. Another method with simultaneous consideration of the rotational and translational parts is proposed in [61], where a nonlinear minimization procedure is presented. Also, a variant of the standard formulation (2.14) is considered in [61], where the intrinsic and extrinsic parameters in the camera matrix need not to be made explicit.

A method that do not use a calibration grid is introduced in [5]. This method uses so-called **structure-from-motion** to formulate a variant of (2.14), now including an extra unknown scale factor for each image pair. The structure-from-motion problem in computer vision is the task of determining the structure of the scene and the 3D motion of the camera using only the correspondences of image points. The algorithm for structure-from-motion used in [5] uses a calibrated camera to reconstruct the camera motion up to a similarity transformation, cf. [18]. Different restricted motions, i.e. pure translations and pure rotations, are considered as well as general motions. However, the method has the drawback that image points have to be matched and tracked from the beginning to the end of the robot trajectory.

Finally, a technique for both intrinsic camera calibration and hand-eye calibration using specially designed motions has been proposed in [76]. This technique also does not need a calibration object and has common grounds with the calibration method presented in Chapter 4. However, the technique still assumes that image point correspondences are established. In Chapter 4 a method for intrinsic camera calibration and hand-eye calibration is presented which does not need extraction and tracking of image points.

2.2.3 Stereo Head Calibration

By a **stereo head** (or stereo rig) we refer to a configuration with two cameras rigidly mounted as in Figure 2.4. Stereo heads are very common in robotic vision since depth information can be obtained directly from feature correspondences once the configuration has been calibrated. The cameras are generally assumed to have different intrinsic and extrinsic parameters.

Stereo head calibration is the process of determining the intrinsic parameters of the two cameras and the relative rigid orientation and translation between them (the relative pose). In Chapter 3, the relations that arise when a stereo head is viewing a plane with known metric structure under translational and general motions are studied. This constitutes extensions of the plane-based calibration theory presented by Zhang [142] and Sturm and Maybank [119].

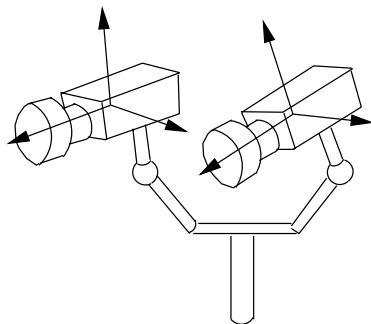


Figure 2.4: A stereo head configuration.

A traditional way of finding the relative rigid transformation between the two cameras in a stereo head is to calculate the so-called **fundamental matrix** F from image correspondences. After calibrating the cameras separately with respect to the intrinsic parameters, the **essential matrix** E can be obtained from F . From the essential matrix we can then obtain the rigid transformation using a singular value decomposition, cf. [55, 29] for further details. The recent research on stereo head calibration has focused on self-calibration algorithms. Several authors have proposed new methods and examined the geometrical relations, cf. [145, 143, 22, 1, 60]. In [21], Demirdjian et al. present a method for self-calibration of a stereo head using a planar scene.

Chapter 3

Plane-based Calibration

In this chapter some extensions of the theory on plane-based calibration introduced by Zhang [141, 142] and Sturm and Maybank [119] will be presented. The use of a planar calibration object greatly simplifies its construction and high precision of the object points is easy to achieve. Calibration from a plane using translational motions has earlier been addressed by Tsai [123] who presented a rigorous calibration theory including accuracy evaluations. However, the motivation for the current work has been to extend the theory of [142, 119] and find simple relationships that fit into that framework.

The basic theory from [141, 119] is presented in Section 3.1. The additional relations that arise when it is assumed that the motions used are pure translations are then studied in Section 3.2. It is also shown what additional calculations and motions of the robot hand that are needed to obtain the hand-eye orientation in a robot vision setting. In Section 3.3 we turn to stereo head calibration and explore the relations that arise in this setting. Then, in Section 3.4, results of experiments on different realizations of the current theory are presented. The chapter is ended in Section 3.5 with some conclusions.

3.1 Basic Theory

We will throughout this chapter use the perspective pinhole camera model, cf. Section 2.1.2. That is, the projection is governed by (2.7), where K is defined in (2.5). However, the translation vector \mathbf{t} will in this chapter denote the translation from the focal point of the camera to the origin of the world coordinate system in the coordinate frame of the *camera*. Thus, the camera matrix P will have the form $P = K [R \ \mathbf{t}]$.

The orientation and origin of the world coordinate system can be chosen so that the plane of the calibration object is $Z = 0$. We then get

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K [\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{r}_3 \ \mathbf{t}] \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = K [\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{t}] \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}, \quad (3.1)$$

where \mathbf{r}_i is the i :th column of R . In this way an object point is related to the corresponding image point by a homography H , cf. Definition 2.1.2,

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}, \quad H = \frac{1}{\lambda} K [\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{t}]. \quad (3.2)$$

This homography between the object plane and the image plane can be estimated for each image, in the same manner as estimating the camera matrix P using the DLT algorithm, cf. [140, 55].

Let \mathbf{h}_i be the i :th column in H . Then,

$$[\mathbf{h}_1 \quad \mathbf{h}_2 \quad \mathbf{h}_3] = \frac{1}{\lambda} K [\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{t}] \quad (3.3)$$

and

$$\mathbf{r}_1 = \lambda K^{-1} \mathbf{h}_1, \quad (3.4)$$

$$\mathbf{r}_2 = \lambda K^{-1} \mathbf{h}_2. \quad (3.5)$$

Introduce $\omega = K^{-T} K^{-1}$. Since \mathbf{r}_1 and \mathbf{r}_2 are orthonormal the following constraints, involving \mathbf{h}_1 , \mathbf{h}_2 and ω , can be derived from (3.4) and (3.5):

$$\mathbf{r}_1^T \mathbf{r}_1 = \lambda^2 \mathbf{h}_1^T K^{-T} K^{-1} \mathbf{h}_1 = \lambda^2 \mathbf{h}_1^T \omega \mathbf{h}_1 = 1, \quad (3.6)$$

$$\mathbf{r}_2^T \mathbf{r}_2 = \lambda^2 \mathbf{h}_2^T K^{-T} K^{-1} \mathbf{h}_2 = \lambda^2 \mathbf{h}_2^T \omega \mathbf{h}_2 = 1, \quad (3.7)$$

$$\mathbf{r}_1^T \mathbf{r}_2 = \lambda^2 \mathbf{h}_1^T K^{-T} K^{-1} \mathbf{h}_2 = \lambda^2 \mathbf{h}_1^T \omega \mathbf{h}_2 = 0. \quad (3.8)$$

From these equations the unknown scale factor λ can be eliminated:

$$\mathbf{h}_1^T \omega \mathbf{h}_2 = 0, \quad (3.9)$$

$$\mathbf{h}_1^T \omega \mathbf{h}_1 = \mathbf{h}_2^T \omega \mathbf{h}_2. \quad (3.10)$$

The matrix ω actually represents the image of the absolute conic, cf. Section 2.1.1, and we now have two linear constraints on this symmetric matrix from each image of the plane. By using three different views of the plane there are enough constraints to solve for ω . The intrinsic camera matrix K can then be obtained by Cholesky factorization, cf. [46], and matrix inversion.

3.2 Plane-based Calibration using Translations

In this section we will examine the problem of calibration from images of a planar object when the relative orientation between the object and the camera does not change, i.e. the motion of the camera between successive images is a pure translation. This case is degenerate in the standard formulation from the previous section. This is evident from the fact that the column vectors \mathbf{h}_1 and \mathbf{h}_2 only depend on K and on \mathbf{r}_1 and \mathbf{r}_2 , respectively. Consequently, the constraints (3.9) and (3.10) remain unchanged if the orientation R does not change. However, by using some knowledge about the translational motion new constraints can be formulated. Zhang briefly addresses this possibility in Appendix D of [140]. The theory presented in this section was developed in [81] and [83]. This special case of plane-based calibration is interesting for robotic systems with few degrees of freedom.

We start by examining the number of degrees of freedom of the translational calibration problem to see what can be calculated assuming different amounts of knowledge. Assume that two images are obtained by the camera and that it has performed a pure translational motion between the imaging instants. From each of the two images a homography from the calibration plane to the image plane is estimated. Denote these homographies by H and H' , respectively. Since the first two columns of H are parallel to the first two columns of H' and since the homographies are known only up to scale, we get 11 distinct elements from the two homographies. There are 6 degrees of freedom for the pose of the camera in the first image. Thus, there are $11 - 6 = 5$ degrees of freedom left for the intrinsic parameters and the translational motion. That is, to calculate all the 5 intrinsic parameters we need to know the translation completely. If e.g. the skew is assumed to be zero, $s = 0$, the camera can be calibrated even if the length of the translation is unknown. If it, additionally, is assumed that the aspect ratio $\gamma = 1$, the camera can be calibrated when the direction of the translation is unknown but the length is known.

The constraints for the translational case will now be formulated. If H is expressed as in (3.2), the estimated homography H' after a translation \mathbf{t}' can be written as

$$H' = \frac{1}{\lambda'} K [\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{t} + R\mathbf{t}'] = \quad (3.11)$$

$$= \frac{1}{\lambda'} K [\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{t} + t'_1 \mathbf{r}_1 + t'_2 \mathbf{r}_2 + t'_3 \mathbf{r}_3] , \quad (3.12)$$

where \mathbf{t}' is the translation vector expressed in the coordinate system of the calibration object, i.e. in the world coordinate system.

When estimating the homographies the knowledge of the special structure of these homographies should be used in order to get more robust estimations. When using translational motions the images are obtained with the same orientation of the camera and the first two columns in the associated homographies should be parallel. Since the scalings of the homographies are arbitrary, the homographies for all images with the same orientation can be estimated simultaneously, so that the first two columns are the same for every estimated homography. That is, the scale factors are chosen equal, $\lambda' = \lambda$. This drastically reduces the number of parameters to be estimated and makes the preceding calculations more accurate. In the continuation there will thus only be one scale factor, λ , present.

Consider the third columns in the matrices H and H' ,

$$\mathbf{h}_3 = \frac{1}{\lambda} K \mathbf{t} , \quad (3.13)$$

$$\mathbf{h}'_3 = \frac{1}{\lambda} K (\mathbf{t} + t'_1 \mathbf{r}_1 + t'_2 \mathbf{r}_2 + t'_3 \mathbf{r}_3) . \quad (3.14)$$

Let

$$\hat{\mathbf{h}}_3 = \mathbf{h}'_3 - \mathbf{h}_3 . \quad (3.15)$$

Then, using (3.3),

$$\hat{\mathbf{h}}_3 = \frac{1}{\lambda} K(t'_1 \mathbf{r}_1 + t'_2 \mathbf{r}_2 + t'_3 \mathbf{r}_3) = t'_1 \mathbf{h}_1 + t'_2 \mathbf{h}_2 + \frac{1}{\lambda} K t'_3 \mathbf{r}_3 \quad , \quad (3.16)$$

and, consequently,

$$\mathbf{r}_3 = \frac{\lambda}{t'_3} K^{-1}(\hat{\mathbf{h}}_3 - t'_1 \mathbf{h}_1 - t'_2 \mathbf{h}_2) \quad . \quad (3.17)$$

In search for new calibration constraints, containing $\hat{\mathbf{h}}_3$ and \mathbf{t}' , scalar products including \mathbf{r}_3 are written down. Taking the scalar product of the orthonormal vectors \mathbf{r}_1 and \mathbf{r}_3 and using (3.6) and (3.8) gives

$$\begin{aligned} \mathbf{r}_1^T \mathbf{r}_3 &= \frac{\lambda^2}{t'_3} \mathbf{h}_1^T K^{-T} K^{-1} (\hat{\mathbf{h}}_3 - t'_1 \mathbf{h}_1 - t'_2 \mathbf{h}_2) \\ &= \frac{\lambda^2}{t'_3} (\mathbf{h}_1^T \omega \hat{\mathbf{h}}_3 - t'_1 \mathbf{h}_1^T \omega \mathbf{h}_1 - t'_2 \mathbf{h}_1^T \omega \mathbf{h}_2) \\ &= \frac{\lambda^2}{t'_3} (\mathbf{h}_1^T \omega \hat{\mathbf{h}}_3 - \frac{t'_1}{\lambda^2}) = 0 \quad . \end{aligned}$$

This implies that

$$\mathbf{h}_1^T \omega \hat{\mathbf{h}}_3 = \frac{t'_1}{\lambda^2} \quad . \quad (3.18)$$

Similarly, the scalar product of \mathbf{r}_2 and \mathbf{r}_3 yields

$$\mathbf{h}_2^T \omega \hat{\mathbf{h}}_3 = \frac{t'_2}{\lambda^2} \quad . \quad (3.19)$$

It remains to consider the scalar product of \mathbf{r}_3 with itself, i.e. the norm condition $\|\mathbf{r}_3\| = 1$,

$$\begin{aligned} \mathbf{r}_3^T \mathbf{r}_3 &= \frac{\lambda^2}{t'^2_3} (\hat{\mathbf{h}}_3^T - t'_1 \mathbf{h}_1^T - t'_2 \mathbf{h}_2^T) \omega (\hat{\mathbf{h}}_3 - t'_1 \mathbf{h}_1 - t'_2 \mathbf{h}_2) \\ &= \frac{\lambda^2}{t'^2_3} (\hat{\mathbf{h}}_3^T \omega \hat{\mathbf{h}}_3 - \frac{t'^2_1}{\lambda^2} - \frac{t'^2_2}{\lambda^2}) = 1 \quad , \end{aligned}$$

implying that

$$\hat{\mathbf{h}}_3^T \omega \hat{\mathbf{h}}_3 = \frac{\|\mathbf{t}'\|^2}{\lambda^2} \quad . \quad (3.20)$$

By letting $\bar{\omega} = \lambda^2 \omega$, the complete set of constraints arising from two images of a plane when the camera undergoes a translational motion $\mathbf{t}' = (t'_1, t'_2, t'_3)$ are

$$\mathbf{h}_1^T \bar{\omega} \mathbf{h}_2 = 0, \quad (3.21)$$

$$\mathbf{h}_1^T \bar{\omega} \mathbf{h}_1 = 1, \quad (3.22)$$

$$\mathbf{h}_2^T \bar{\omega} \mathbf{h}_2 = 1, \quad (3.23)$$

$$\mathbf{h}_1^T \bar{\omega} \hat{\mathbf{h}}_3 = t'_1, \quad (3.24)$$

$$\mathbf{h}_2^T \bar{\omega} \hat{\mathbf{h}}_3 = t'_2, \quad (3.25)$$

$$\hat{\mathbf{h}}_3^T \bar{\omega} \hat{\mathbf{h}}_3 = \|\mathbf{t}'\|^2. \quad (3.26)$$

After solving the system (3.21)-(3.26) in the unknowns of $\bar{\omega}$, t'_1 , t'_2 and $\|\mathbf{t}'\|$, a Cholesky factorization of $\bar{\omega}$ is performed. Inverting and scaling the resulting matrix gives us the intrinsic calibration matrix K . The scale factor λ is easily found since K has an element equal to 1 in the bottom right position.

3.2.1 Calculating the Hand-Eye Orientation

Considering a robot vision system, it will now be discussed what is needed to extend the calculations to also include computation of the rotational part of the hand-eye transformation, cf. Section 2.2.2. For calculation of the translational part, rotational motions need to be used, cf. [124]. Since this section concerns calibration from translational motions we will focus on estimating the hand-eye orientation, R_{he} .

The calculation of R_{he} is very direct and can be based on a minimum number of two translations. Consider two translations \mathbf{t}' and \mathbf{t}'' in the coordinate system of the calibration object. From the second translation \mathbf{t}'' a new set of equations of the form (3.24)-(3.26) is obtained. If $\hat{\mathbf{h}}_3$ is the analogue of $\hat{\mathbf{h}}_3$ for the second translation, one of the three new constraints is

$$\hat{\mathbf{h}}_3^T \bar{\omega} \hat{\mathbf{h}}_3 = \|\mathbf{t}''\|^2. \quad (3.27)$$

To calculate the hand-eye orientation, it is assumed that the translation vectors are known in the robot hand coordinate system. Since the lengths of these vectors are equal to the lengths of the corresponding translation vectors in the coordinate system of the calibration object, $\|\mathbf{t}'\|$ and $\|\mathbf{t}''\|$ are known quantities. In this way we have five constraints on $\bar{\omega}$ and the camera can be calibrated if zero skew, $s = 0$, is assumed.

For the estimation of R_{he} we want to find the representation \mathbf{t}'_c , in the camera coordinate system, of the corresponding translation \mathbf{t}'_r , in the robot hand coordinate system, which in turn corresponds to the translation \mathbf{t}' in the coordinate system of the calibration plane. The homography H' after the translation \mathbf{t}' can be represented as

$$H' = \frac{1}{\lambda} K [\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{t} + \mathbf{t}'_c]. \quad (3.28)$$

Using the vector $\hat{\mathbf{h}}_3$ introduced earlier, cf. (3.15), we get

$$\hat{\mathbf{h}}_3 = \frac{1}{\lambda} K \mathbf{t}'_c . \quad (3.29)$$

For the calculation of R_{he} the length of \mathbf{t}'_c is not of importance and

$$\mathbf{t}_c \sim K^{-1} \hat{\mathbf{h}}_3 . \quad (3.30)$$

The hand-eye orientation R_{he} is calculated by relating the translation vectors in the camera coordinate system to the corresponding translation vectors in the robot hand coordinate system. After calculating \mathbf{t}'_c and \mathbf{t}''_c corresponding to \mathbf{t}'_r and \mathbf{t}''_r , respectively, a third corresponding pair of translations can be constructed using $\mathbf{t}'''_c = \mathbf{t}'_c \times \mathbf{t}''_c$ and $\mathbf{t}'''_r = \mathbf{t}'_r \times \mathbf{t}''_r$. After normalizing the translation vectors, R_{he} can be calculated using

$$R_{he} = [\mathbf{t}'_c \quad \mathbf{t}''_c \quad \mathbf{t}'''_c] [\mathbf{t}'_r \quad \mathbf{t}''_r \quad \mathbf{t}'''_r]^{-1} . \quad (3.31)$$

When using (3.31) numerically, the matrix R_{he} will not be exactly orthogonal. To find the nearest orthogonal matrix in the Frobenius norm a singular value decomposition can be performed on R_{he} , $R_{he} = USV^T$. The orthogonal matrix $\bar{R}_{he} = UV^T$ is then chosen as the rotation matrix.

3.3 Plane-based Stereo Head Calibration

In this section, the relations that arise when calibrating of a stereo head configuration using a planar calibration object will be studied. The discussion will be divided into two subsections. Firstly, in Section 3.3.1, the theory on using pure translational motions from Section 3.2 will be applied to the stereo case. Secondly, in Section 3.3.2, general motions of the stereo head will be considered and relations that can be derived in this case will be discussed.

Throughout this section entities corresponding to the right camera in the stereo head will be denoted by an r and entities corresponding to the left camera will be denoted by an l .

3.3.1 Stereo Head Calibration using Translations

One interesting application of the theory presented in Section 3.2 is the calibration of a stereo head using pure translational motions. If the stereo head is translated, the translation vector \mathbf{t}' , cf. (3.11), will be the same for the two cameras in the coordinate system of the calibration object. This can be used to make a simultaneous calibration of the two cameras.

Assume that the length of the translational motion of the stereo head is known. If, additionally, zero skew is assumed, $s = 0$, the following equations can be used to calibrate the cameras,

$$\mathbf{h}_1^r T \bar{\omega}^r \mathbf{h}_2^r = 0 , \quad (3.32)$$

$$\mathbf{h}_1^r T \bar{\omega}^r \mathbf{h}_1^r = 1 , \quad (3.33)$$

$$\mathbf{h}_2^r T \bar{\omega}^r \mathbf{h}_2^r = 1 , \quad (3.34)$$

$$\mathbf{h}_1^l T \bar{\omega}^l \mathbf{h}_2^l = 0 , \quad (3.35)$$

$$\mathbf{h}_1^l T \bar{\omega}^l \mathbf{h}_1^l = 1 , \quad (3.36)$$

$$\mathbf{h}_2^l T \bar{\omega}^l \mathbf{h}_2^l = 1 , \quad (3.37)$$

$$\mathbf{h}_1^r T \bar{\omega}^r \hat{\mathbf{h}}_3^r - \mathbf{h}_1^l T \bar{\omega}^l \hat{\mathbf{h}}_3^l = 0 , \quad (3.38)$$

$$\mathbf{h}_2^r T \bar{\omega}^r \hat{\mathbf{h}}_3^r - \mathbf{h}_2^l T \bar{\omega}^l \hat{\mathbf{h}}_3^l = 0 , \quad (3.39)$$

$$\hat{\mathbf{h}}_3^r T \bar{\omega}^r \hat{\mathbf{h}}_3^r = \|\mathbf{t}'\|^2 , \quad (3.40)$$

$$\hat{\mathbf{h}}_3^l T \bar{\omega}^l \hat{\mathbf{h}}_3^l = \|\mathbf{t}'\|^2 . \quad (3.41)$$

In the case of unknown length, but known direction, of the translation the last four equations can be written

$$\mathbf{h}_1^r T \bar{\omega}^r \hat{\mathbf{h}}_3^r = t'_1 , \quad (3.42)$$

$$\mathbf{h}_1^l T \bar{\omega}^l \hat{\mathbf{h}}_3^l = t'_1 , \quad (3.43)$$

$$\mathbf{h}_2^r T \bar{\omega}^r \hat{\mathbf{h}}_3^r = t'_2 , \quad (3.44)$$

$$\mathbf{h}_2^l T \bar{\omega}^l \hat{\mathbf{h}}_3^l = t'_2 , \quad (3.45)$$

$$\hat{\mathbf{h}}_3^r T \bar{\omega}^r \hat{\mathbf{h}}_3^r - \hat{\mathbf{h}}_3^l T \bar{\omega}^l \hat{\mathbf{h}}_3^l = 0 . \quad (3.46)$$

After calculation of the camera matrices K^r and K^l , the pose of the cameras, i.e. the orientation and position of the cameras in relation to the calibration object, can easily be obtained from relation (3.3). The following formulas are obtained for the columns in R and for \mathbf{t} ,

$$\mathbf{r}_1 = \lambda K^{-1} \mathbf{h}_1 , \quad (3.47)$$

$$\mathbf{r}_2 = \lambda K^{-1} \mathbf{h}_2 , \quad (3.48)$$

$$\mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2 , \quad (3.49)$$

$$\mathbf{t} = \lambda K^{-1} \mathbf{h}_3 , \quad (3.50)$$

where $\lambda = 1/\|K^{-1} \mathbf{h}_1\| = 1/\|K^{-1} \mathbf{h}_2\|$, cf. [142]. From this, the relative rigid transformation \hat{T} between the two cameras in the stereo head can be obtained and the stereo head is then completely calibrated. Thus, this can be done from one single translation, if zero skew is assumed and either the direction or length of the translation is known.

3.3.2 Stereo Head Calibration using General Motions

The case of general motions of the stereo head will now be considered. It will be studied how the rigid transformation $\hat{T} = [\hat{R} \mid \hat{\mathbf{t}}]$ between the two cameras can be included into the representation of the present homographies. For readability, the bar separates the leftmost 3×3 matrix from the rightmost 3-vector, in the 3×4 transformation matrix. Following the notation in the last section, an r and an l will be used to denote the right and left cameras, respectively.

In essence, our discussion will result in a way of calculating the infinite homography H_∞ , cf. Section 2.1.1, between the image planes of the two cameras. This will be done, firstly, by a similar methodology and notation as in the previous sections of this chapter, by writing down equations for the homographies H^l and H^r from the calibration plane to the left and right image planes, respectively. Secondly, a slightly more direct derivation will be presented. The calculation of H_∞ for a stereo head has previously been addressed by Zisserman et al. [145] in the case of self-calibration.

By using the transformation \hat{T} , the orientations and positions of the left and right camera in relation to the calibration plane can be related in the following way,

$$[R^l \mid \mathbf{t}^l] = [\hat{R} \mid \hat{\mathbf{t}}] \begin{bmatrix} R^r & \mathbf{t}^r \\ 0 & 1 \end{bmatrix}. \quad (3.51)$$

It then follows that

$$\lambda^l \begin{bmatrix} x^l \\ y^l \\ 1 \end{bmatrix} = K^l [\hat{R} \mid \hat{\mathbf{t}}] \begin{bmatrix} R^r & \mathbf{t}^r \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix}, \quad (3.52)$$

$$\lambda^l \begin{bmatrix} x^l \\ y^l \\ 1 \end{bmatrix} = K^l [\hat{R}R^r \mid \hat{R}\mathbf{t}^r + \hat{\mathbf{t}}] \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix}, \quad (3.53)$$

$$\lambda^l \begin{bmatrix} x^l \\ y^l \\ 1 \end{bmatrix} = K^l \hat{R} [\mathbf{r}_1^r \ \mathbf{r}_2^r \ \mathbf{r}_3^r (\mathbf{t}^r + \hat{R}^T \hat{\mathbf{t}})] \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix}, \quad (3.54)$$

and accordingly

$$H^l = \frac{1}{\lambda^l} K^l \hat{R} [\mathbf{r}_1^r \ \mathbf{r}_2^r (\mathbf{t}^r + \hat{R}^T \hat{\mathbf{t}})]. \quad (3.55)$$

For the right camera we have, cf. (3.2),

$$H^r = \frac{1}{\lambda^r} K^r [\mathbf{r}_1^r \ \mathbf{r}_2^r \ \mathbf{t}^r]. \quad (3.56)$$

By eliminating \mathbf{r}_1^r , \mathbf{r}_2^r and \mathbf{t}^r between (3.55) and (3.56) the following relations between the columns in H^l and H^r are obtained,

$$\mathbf{h}_1^r = \frac{\lambda^l}{\lambda^r} K^r \hat{R}^T K^{l-1} \mathbf{h}_1^l, \quad (3.57)$$

$$\mathbf{h}_2^r = \frac{\lambda^l}{\lambda^r} K^r \hat{R}^T K^{l-1} \mathbf{h}_2^l, \quad (3.58)$$

$$\mathbf{h}_3^r = \frac{\lambda^l}{\lambda^r} K^r \hat{R}^T K^{l-1} \mathbf{h}_3^l - \frac{1}{\lambda^r} K^r \hat{R}^T \hat{\mathbf{t}}. \quad (3.59)$$

Using images from at least two stereo views (two positions of the stereo head) the matrix $K^r \hat{R}^T K^{l-1}$ can be obtained from equations (3.57) and (3.58), up to a scale factor. This follows from the fact that $K^r \hat{R}^T K^{l-1}$ is independent of the pose of the stereo head, while the homographies from the calibration plane to the image planes change for each position. The calculated matrix is denoted by

$$H^\infty = K^r \hat{R}^T K^{l-1}, \quad (3.60)$$

since it actually is an expression for the infinite homography between the images planes of the cameras in the stereo head, i.e. the homography between the image planes induced by the plane at infinity, cf. [55].

The infinite homography can be used for relating the matrices $\omega^l = K^{l-T} K^{l-1}$ and $\omega^r = K^{r-T} K^{r-1}$ in the following way,

$$\begin{aligned} \omega^l = K^{l-T} K^{l-1} &= H^{\infty T} K^{r-T} \hat{R}^T \hat{R} K^{r-1} H^\infty = \\ &= H^{\infty T} K^{r-T} K^{r-1} H^\infty = H^{\infty T} \omega^r H^\infty. \end{aligned} \quad (3.61)$$

Using (3.61) together with the single camera constraints for each camera, (3.9) and (3.10), the two cameras in the stereo head can be calibrated simultaneously. In [145], in the case of self-calibration, it is argued that (3.61) does not provide any additional constraints on the calibration problem. Further, we have observed that two motions, the minimum number for single camera calibration, still are needed for calibrating the stereo head after adding (3.61) in the calculations. However, in the experiments presented in Section 3.4.2 it is shown how the addition of (3.61) influences the calculations under noisy conditions.

A Different Formulation

The discussion above reveals an efficient way to calculate the infinite homography between the two image planes in a stereo head, from two stereo views of a plane. The relation between the infinite homography H^∞ and the estimated homographies, H^r and H^l , can also be derived in a slightly more direct way. Let $H^{r'}$ and $H^{l'}$ be the homographies from the plane at infinity to the right and the left image plane, respectively. These

homographies can be expressed in a similar way as the homographies from the calibration plane to the image plane, cf. (3.1), but instead of setting $Z = 0$ we use the fact that the last element in the homogeneous coordinates of the points at infinity is equal to zero,

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \sim K[\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{r}_3 \ t] \begin{bmatrix} X \\ Y \\ Z \\ 0 \end{bmatrix} = K[\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{r}_3] \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} . \quad (3.62)$$

Comparing (3.62) to (3.1), it is observed that the first two columns of H^r and H^{lr} (H^l and H^{ll}) actually are equal up to scale, i.e.

$$\mathbf{h}_i^r \sim \mathbf{h}_i^{lr}, \quad i = 1, 2 , \quad (3.63)$$

$$\mathbf{h}_i^l \sim \mathbf{h}_i^{ll}, \quad i = 1, 2 . \quad (3.64)$$

Since, by definition,

$$H^\infty = H^{lr} H^{ll^{-1}} , \quad (3.65)$$

the following holds for the first two columns of H^{ll} and H^{lr} ,

$$H^\infty \mathbf{h}_i^{ll} = \mathbf{h}_i^{lr}, \quad i = 1, 2 . \quad (3.66)$$

Because of (3.63) and (3.64), we once again arrive at (3.57) and (3.58),

$$\mathbf{h}_1^r \sim H^\infty \mathbf{h}_1^{ll} , \quad (3.67)$$

$$\mathbf{h}_2^r \sim H^\infty \mathbf{h}_2^{ll} . \quad (3.68)$$

A Note on Determining \hat{R}

At the end of Section 3.3.1 it was commented on how to calculate the rigid transformation $\hat{T} = [\hat{R} \mid \hat{\mathbf{t}}]$ between the left and the right cameras of the stereo pair. However, \hat{R} can be obtained without first calculating the transformations from the calibration plane to the cameras. If H^∞ has been calculated, \hat{R} can be obtained from (3.60),

$$\hat{R} = K^{lT} H^{\infty T} K^{r-T} . \quad (3.69)$$

3.4 Experiments

Results of experiments, using different implementations of the theory derived in this chapter, will here be presented. We will start by studying the practical performance of plane-based calibration from translations in Section 3.4.1. Then, in Section 3.4.2, the practical effect of relation (3.61) on the calibration of a stereo head from a planar object is analyzed.

3.4.1 Experiments using Translational Motions

For the experiments on calibration using translational motions both synthetic image data and images from a real robot vision system have been used.

Synthetic Data

Calibration has been performed on computer generated data to get a measurement on how sensitive the calculations are to errors and noise in the input data. Projections of a grid with 9×6 points were calculated. The simulated camera was rotated first 6° degrees around the x -axis, then 30° degrees around the y -axis and then finally -12° degrees around the z -axis in relation to the calibration grid. The distance between the grid points, both horizontally and vertically, was 5 length units (l.u.) which is to be compared to the initial position of the camera which was 100 l.u. from the plane along the z -axis and 10 l.u. along the y -axis. We have looked at three different cases:

Case 1 The length of the translational motion \mathbf{t}' is known, but the direction is unknown. The aspect ratio α and the skew s is assumed to be 1 and 0, respectively.

Case 2 The direction of \mathbf{t}' is known, but the length is unknown. The skew s is assumed to be 0.

Case 3 Both the length and the direction of \mathbf{t}' is known. All intrinsic parameters are calculated.

Noise with a standard deviation of 0.5 pixels was added to the projected points. This is to be compared to the artificial image size which was approximately 320×240 pixels. The calibrations were performed 100 times and the means and the standard deviations of the different parameters were calculated. Two camera positions, i.e. two images, were used with the length of the translation being $\|\mathbf{t}'\| = 15$ and the direction being parallel to the vector $(5, 3, 10)$. The correct calibration matrix was

$$K = \begin{bmatrix} 650 & 0 & 160 \\ 0 & 650 & 120 \\ 0 & 0 & 1 \end{bmatrix}. \quad (3.70)$$

The resulting means and standard deviations of the estimated parameters were in the three different cases,

$$\begin{aligned}
 K_1 &= \begin{bmatrix} 650.7 \pm 14.6 & 0 & 158.9 \pm 8.4 \\ 0 & 650.7 \pm 14.6 & 121.1 \pm 11.1 \\ 0 & 0 & 1 \end{bmatrix}, \\
 K_2 &= \begin{bmatrix} 650.5 \pm 24.4 & 0 & 159.4 \pm 10.5 \\ 0 & 650.3 \pm 23.5 & 121.3 \pm 11.1 \\ 0 & 0 & 1 \end{bmatrix}, \\
 K_3 &= \begin{bmatrix} 651.1 \pm 15.7 & -0.03 \pm 2.2 & 159.3 \pm 6.4 \\ 0 & 650.8 \pm 14.6 & 121.1 \pm 9.1 \\ 0 & 0 & 1 \end{bmatrix}.
 \end{aligned}$$

Considering that the minimal case of two images were used for the calculations, the standard deviations are rather reasonable. The use of more images would probably make the estimations more precise. It can be seen from these simulations that case 2 is the most unstable, while the other two cases seem to be rather equivalent in this respect.

Real Data

The robot used for the experiments on a real robot vision system is a modified ABB IRB2000, cf. Figure 3.1, which is capable of moving in all six degrees of freedom. The camera is mounted by a ball head camera holder on the hand of the robot so that the orientation of the camera can be changed with respect to the orientation of the robot hand coordinate system.

Figure 3.2 shows an image from one of the used image sequences, picturing the calibration plane. In the following experiments approximately ten images were used in each sequence.

Experiments have been performed for both a single camera and a stereo head. In the stereo head case, the head was simulated by making two identical translational motions, starting from two different locations using different orientations and settings of the camera. The two different camera settings will be referred to as the left and the right camera, respectively.

As a reference, the cameras were calibrated using general motions and the basic constraints for plane-based calibration, (3.9) and (3.10). This resulted in the following calibration matrices,

$$K_{general}^l = \begin{bmatrix} 1063.1 & -6.2 & 328.9 \\ 0 & 1094.0 & 247.9 \\ 0 & 0 & 1 \end{bmatrix},$$

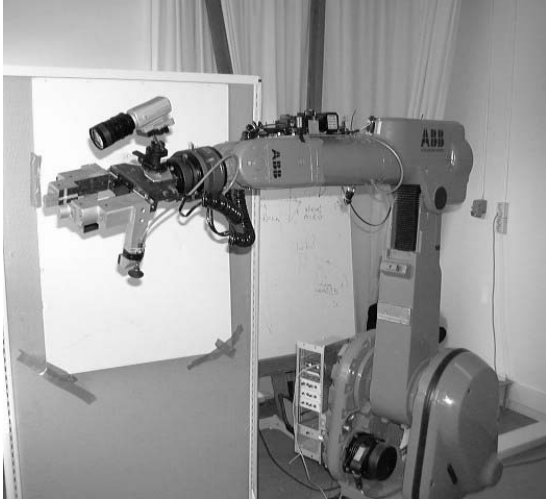


Figure 3.1: The robot used in the experiments.

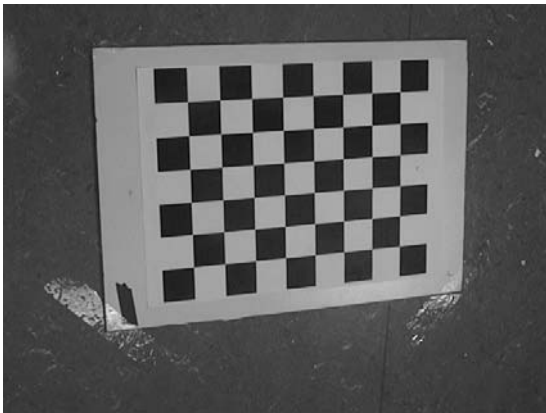


Figure 3.2: An image from one of the sequences used.

and

$$K_{general}^r = \begin{bmatrix} 1089.6 & -9.9 & 335.1 \\ 0 & 1126.4 & 231.9 \\ 0 & 0 & 1 \end{bmatrix},$$

To test the practical performance of calibration from pure translations, the left camera was first translated in the direction orthogonal to the floor, i.e. $\mathbf{t}' = (0, 0, 1)$. By using equations (3.21)-(3.25) the camera was calibrated assuming zero skew, $s = 0$. The following intrinsic camera matrix was obtained,

$$K_{(0,0,1)}^l = \begin{bmatrix} 1019.4 & 0 & 339.3 \\ 0 & 1047.8 & 235.5 \\ 0 & 0 & 1 \end{bmatrix}.$$

The value on the focal length is clearly underestimated compared to $K_{general}^l$.

The left camera was also calibrated using a motion sequence where the translation direction was changed after each image. Here, the directions of the translations were unknown and we instead used a known fixed length of the translations. Hence, equations (3.21)-(3.23) and multiple instances of (3.26) were used for the calculations. The following matrix was obtained,

$$K_{change}^l = \begin{bmatrix} 1030.7 & 0 & 337.4 \\ 0 & 1076.5 & 236.9 \\ 0 & 0 & 1 \end{bmatrix}.$$

With respect to $K_{general}^l$, K_{change}^l seems to be a better estimate than $K_{(0,0,1)}^l$. This is in accordance with the observations made using the synthetic input data.

The stereo head configuration of the left and right camera has also been calibrated using the theory presented in Section 3.3.1. The camera pair was translated in the direction orthogonally to the floor, but this information was actually not used. Instead a known length of the translations was used and equations (3.32)-(3.41) were applied. The following two matrices were calculated:

$$K_{stereo}^l = \begin{bmatrix} 1058.7 & 0 & 353.2 \\ 0 & 1092.8 & 231.4 \\ 0 & 0 & 1 \end{bmatrix},$$

and

$$K_{stereo}^r = \begin{bmatrix} 1091.8 & 0 & 310.1 \\ 0 & 1125.1 & 252.9 \\ 0 & 0 & 1 \end{bmatrix}.$$

The result obtained here is very close to the reference method, where general motions of the cameras were used, at least with respect to the focal length and the aspect ratio.

When, as in the last experiment, the lengths of the translations are known, and it is known that the translation direction is unchanged through the whole image sequence, additional constraints can be used in the homography estimation. Indeed, by using the known translation lengths in the estimation only 11 parameters in total need to be estimated for all the homographies together. By using as much apriori information of the structure of the homographies as possible the calibration process becomes more stable and accurate.

3.4.2 Experiments using a Stereo Head and General Motions

To test the influence of relation (3.61) on the calibration of a stereo head, experiments on synthetic data have been performed. This allows for a study of the calibration procedure in the presence of different amounts of added noise.

Our input data consist of a planar grid with 10×14 points. Six different stereo views of the plane were used. Following [140], we describe the rotations by a vector \mathbf{r} which is parallel to the rotation axis, having a length that is equal to the angle of rotation measured in degrees. The translations are described by a vector \mathbf{t} . The six orientations and positions of the right hand camera in the stereo pair were chosen as follows:

$$\begin{aligned} \mathbf{r}_1 &= (20, 0, 0), \mathbf{t}_1 = (-9, -12.5, 100); \\ \mathbf{r}_2 &= (0, 20, 0), \mathbf{t}_2 = (-9, -12.5, 110); \\ \mathbf{r}_3 &= \frac{1}{\sqrt{5}}(-30, -30, -15), \mathbf{t}_3 = (-10.5, -12.5, 125); \\ \mathbf{r}_4 &= (0, 20, 30), \mathbf{t}_4 = (-5, -10, 95); \\ \mathbf{r}_5 &= (15, 0, 25), \mathbf{t}_5 = (-13, -15, 105); \\ \mathbf{r}_6 &= (-20, 10, 0), \mathbf{t}_6 = (-10, -10, 115) . \end{aligned}$$

The left camera was then rotated and translated using $\mathbf{r} = (0, 30, 0)$ and $\mathbf{t} = (-60, 0, 0)$ with respect to the right camera.

The two simulated cameras were calibrated both separately, using only the standard constraints (3.9) and (3.10), and simultaneously, using the addition of (3.61). Noise was added to the projected image points, from a variance of 0.1 pixels up to 0.8 pixels in steps of 0.1. In Figures 3.3 and 3.4, the mean of the relative error from 100 simulations of estimating α is plotted for the left and the right camera, respectively. The solid line corresponds to the use of (3.61) and the dashed line to estimation without using this relation. It can be seen that (3.61) gives an improvement of the calibration for the left camera, while it hardly affects the right camera. The result is similar for the other intrinsic parameters. Probably the views of the calibration grid by the left camera are more sensitive to the added noise, and the right camera helps the left camera to achieve better estimations through the simultaneous calibration.

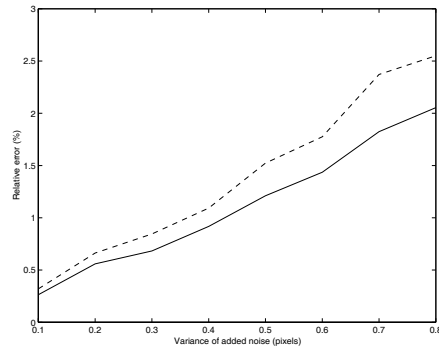


Figure 3.3: Mean of the relative errors in the estimation of α for the left camera. The solid line corresponds to the use of (3.61). The dashed line corresponds to calculation without using this relation.

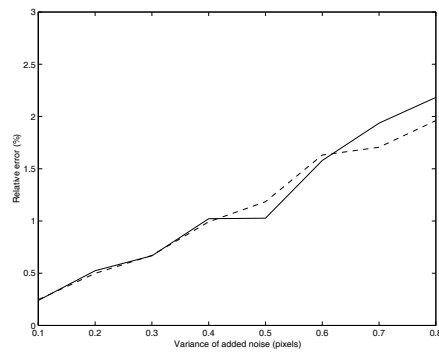


Figure 3.4: The analogue to Figure 3.3 for the right camera.

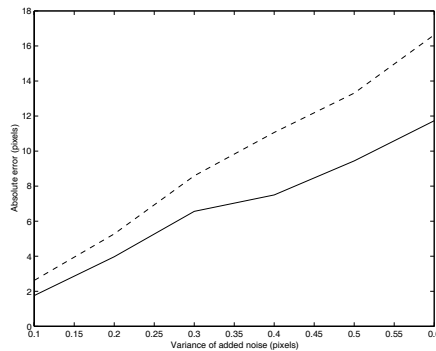


Figure 3.5: The mean of the relative error when estimating u_0 for left camera, when this camera is disturbed by twice the amount of noise as the right camera.

One application of the findings presented here would be the case of calibrating two cameras, when it is known that one of the cameras is more difficult to calibrate than the other one. In this case it could be a good idea to first calibrate the superior camera on its own and then calibrate the two cameras simultaneously. This would probably lead to more accurate estimates of the intrinsic parameters for the more error prone camera.

Figure 3.5 shows an example of calibrating two cameras when the left camera is disturbed by twice the amount of noise as the right camera. The plot shows the mean error in pixels of 50 simulations, when estimating u_0 for the left camera. The noise level in this camera is twice as high as indicated on the horizontal axis. The dashed line represents single camera calibration and the solid lines represents simultaneous calibration of the two cameras. The same effect is seen for the other intrinsic camera parameters.

3.5 Conclusions

Extensions of the plane-based calibration theory due to Zhang [142] and Sturm and Maybank [119] have been presented in this chapter. It has been shown what relations that can be derived if the motion is assumed to be pure translational. This theory is then considered in the setting of hand-eye calibration and stereo head calibration. Experimental studies using the derived equations shows that the minimal realizations of the theory do not give very accurate results. However, using many different motions or knowledge of both the direction and length of the translations improve the estimations. One application that seems to work very well is the simultaneous calibration of the two cameras in a stereo head using translational motions. The results obtained in that setting shows strong agreement with the results obtained using the basic constraints for plane-based calibration and general motions.

CHAPTER 3.

A method for calculating the infinite homography between the two image planes in a stereo head, using the homographies estimated between the calibration plane and the image planes, has also been derived. Its possible usage for simultaneous calibration of the two cameras in the stereo head has been discussed and illustrated using experiments.

Chapter 4

A Direct Method for Calibration

A method for intrinsic camera calibration and hand-eye calibration, based directly on the spatial and temporal image derivatives in an image sequence, will be derived in this chapter. The target system for this algorithm is, just as in the previous chapter, a robot vision system with a camera mounted on the hand of the robot. It is assumed that the robot has been calibrated so that exact knowledge of the motion of the robot hand is available.

In [62] and [96], Horn and co-workers coined the term **direct methods** for algorithms using only the component of the motion field in the direction of the intensity gradient, directly available from the optical flow constraint equation (2.10), as input data. This pioneering work on direct methods for shape recovery and 3D motion estimation has been continued by several authors [52, 33, 34, 118, 117, 97]. The work presented in this chapter has most in common with the work of Stein and Shashua [118, 117]. The constraint equations derived by these authors are closely related to the relations presented here. However, in [118, 117] calibrated cameras were considered and camera motion and scene structure were estimated from general motions. In our presentation, a robot vision setting is considered and the main objective is intrinsic camera calibration and hand-eye calibration. The calibration is made possible by assuming that the robot hand can be moved in a predetermined way.

The equations derived in the work presented here are constraints directly on the unknown calibration parameters, i.e. the intrinsic camera parameters and the hand-eye transformation. The constraint equations are constructed solely from the spatial and temporal image derivatives and the motion parameters of the robot hand. The motion field for uncalibrated instantaneous rigid motion has previously been studied by Viéville et al. [127] and Fermüller et al. [8]. In [8], a method for self-calibration from image derivatives is presented. The algorithm is very general and calculates the intrinsic camera parameters using general unknown motions of the camera. To this end, a smoothness constraint on the surfaces in view is formulated. Using this constraint, an iterative process is constructed to separate the translational and rotational components of the motion field. The calibration parameters are then estimated from the rotational components of several flow fields. By assuming that we can control the motion of the robot hand, our calculations become considerably simpler than those in [8]. No smoothness assumptions need to be made and by choosing special motions, such as pure translations, the calculations are linear in the unknowns.

Our first results on using image derivatives directly for robotic camera calibration was presented in [78] and [79]. In these papers only the hand-eye calibration problem was addressed. In [82] a technique for intrinsic camera calibration was added to the theory.

The calibration method and its theoretical presentation has been considerably simplified during its development.

We start the presentation of the calibration theory in Section 4.1 by deriving a basic relation that will be a cornerstone in the forthcoming chapters. Then, in Section 4.2, it is shown how the intrinsic camera parameters and the orientation of the camera in relation to the robot hand can be calculated simultaneously, using translational motions of the robot hand. A calibration constraint for general motions is then presented in Section 4.3. Using this constraint, motions including a rotational component are then utilized in Section 4.4 to calculate the position of the camera in relation to the robot hand. By using the result of the computations presented in Section 4.2 as additional input data, the calculation becomes linear in the unknowns. Next, in Section 4.5, the possibility of a preceding nonlinear optimization, using the results from the previous calculations as starting point, is discussed. After the full calibration algorithm has been presented, details on the practical estimation of the derivatives are studied in Section 4.6. Results of experiments on both computer generated and real data are then presented in Section 4.7, and different aspects of the method are highlighted. The chapter is ended with a concluding discussion in Section 4.8.

4.1 A Basic Photometric Relation

We will start by deriving a photometric constraint on the motion of a 3D point that is seen by an uncalibrated camera undergoing a general motion. Further, the camera is mounted on a robot hand, implying a rigid transformation between the hand and the camera. Let $\mathbf{X} = (X, Y, Z)$ be the (non-homogeneous) coordinates of a 3D point in the coordinate system of the robot hand and let the time derivative of this point, i.e. the motion vector, be denoted by $\dot{\mathbf{X}} = (\dot{X}, \dot{Y}, \dot{Z})$. Let $\mathbf{x} = (x, y)$ the (non-homogeneous) coordinates of the projection of \mathbf{X} in the image plane.

Let the rigid transformation between the robot hand coordinate system and the camera coordinate system, i.e. the hand-eye transformation, be given by $T_h = [R_h \quad -R_h \mathbf{t}_h]$. Here, the rotation matrix R_h represents the relative orientation between the robot hand coordinate system and the camera coordinate system. The non-homogeneous vector \mathbf{t}_h represents the translation between the origin of the robot hand coordinate system and the focal point of the camera. The projection of a point in the robot hand coordinate system to the image plane of the camera is then described by

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = KR_h [I \quad -\mathbf{t}_h] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \quad (4.1)$$

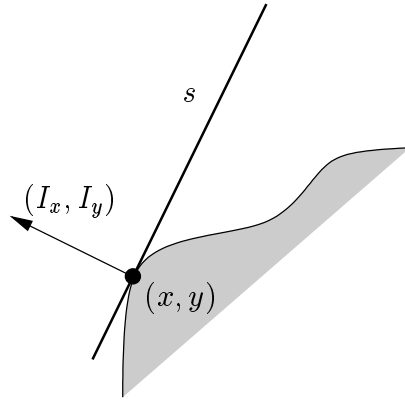


Figure 4.1: The tangent line s of the grey level contour at image point (x, y) .

Let $A = KR_h$, so that the projection (4.1) can be written as

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = A(\mathbf{X} - \mathbf{t}_h) . \quad (4.2)$$

Denote by I_x and I_y the spatial derivatives in the x - and y -directions, respectively. The temporal derivative is denoted by I_t . Following [117] we introduce the notation

$$\mathbf{s} = \begin{bmatrix} I_x \\ I_y \\ -(I_x x + I_y y) \end{bmatrix} . \quad (4.3)$$

Note that \mathbf{s} represents the tangent line of the grey level contour at image point (x, y) , cf. Figure 4.1.

If we denote the homogeneous coordinates of (x, y) by $\mathbf{p} = [x \ y \ 1]^T$, the fact that \mathbf{p} lies on the line s can be expressed as $\mathbf{s}^T \mathbf{p} = 0$, or by using (4.2) as

$$\mathbf{s}^T A(\mathbf{X} - \mathbf{t}_h) = 0 . \quad (4.4)$$

Differentiating this expression with respect to time gives

$$\lambda \dot{\mathbf{s}}^T \mathbf{p} + \mathbf{s}^T A \dot{\mathbf{X}} = 0 . \quad (4.5)$$

When discussing the estimation of the derivatives in Section 4.6, we will mention that it can be advantageous to let the spatial derivatives be smoothed in time through the image

sequence. This does, however, as the following shows, not influence the result of the current derivation. The temporal derivative of s is

$$\dot{\mathbf{s}} = \begin{bmatrix} \dot{I}_x \\ \dot{I}_y \\ -(\dot{I}_x x + \dot{I}_y y) - (I_x u + I_y v) \end{bmatrix}, \quad (4.6)$$

where $(u, v) = (\dot{x}, \dot{y})$. Since, according to the optical flow constraint equation (2.10), $I_x u + I_y v = -I_t$, we get

$$\dot{\mathbf{s}}^T \mathbf{p} = \dot{I}_x x + \dot{I}_y y - (\dot{I}_x x + \dot{I}_y y) - (I_x u + I_y v) = I_t, \quad (4.7)$$

and finally, by inserting this in (4.5), the basic photometric relation is reached,

$$\mathbf{s}^T A \dot{\mathbf{X}} + \lambda I_t = 0. \quad (4.8)$$

4.2 Calibration from Translational Motions

If the motion of the robot hand is a pure translation, the motion of the world points relative to the robot hand coordinate system is a translation with reverse direction. Let \mathbf{D}_h be the translation vector of the robot hand. Since the motion of the robot hand is assumed known, \mathbf{D}_h is known. Now, for the motion of a world point in the robot hand coordinate system, $\dot{\mathbf{X}} = -\mathbf{D}_h$. Using this in (4.8) gives

$$\mathbf{s}^T A \mathbf{D}_h - \lambda I_t = 0. \quad (4.9)$$

Besides the unknown matrix A , this equation also contains the unknown scalar λ . By the modeling of the camera in Section 2.1.2, this quantity corresponds to the distance from the image plane to the point \mathbf{X} along the Z -axis in the camera coordinate system and it is consequently generally different for every image point \mathbf{x} . To get a linear constraint for A only, we make a second translation $\bar{\mathbf{D}}_h$. If the two image sequences resulting from the translations have one image in common and if the temporal derivatives I_t and \bar{I}_t are calculated with respect to this image, λ is the same in the constraint resulting from the second translation,

$$\bar{\mathbf{s}}^T A \bar{\mathbf{D}}_h - \lambda \bar{I}_t = 0. \quad (4.10)$$

If, in the image sequence, the direction of translation changes after each obtained image, \mathbf{D}_h and $\bar{\mathbf{D}}_h$ are simply two successive translations which have an image in common at the end of \mathbf{D}_h and at the starting point of $\bar{\mathbf{D}}_h$, cf. Figure 4.2A. However, a better estimation of I_t (and \bar{I}_t) is given if several images, obtained using the same direction of translation, are used for its computation. Then, the image common for the two directions of translations are obtained at the location marked in Figure 4.2B. Note that the vector s also is

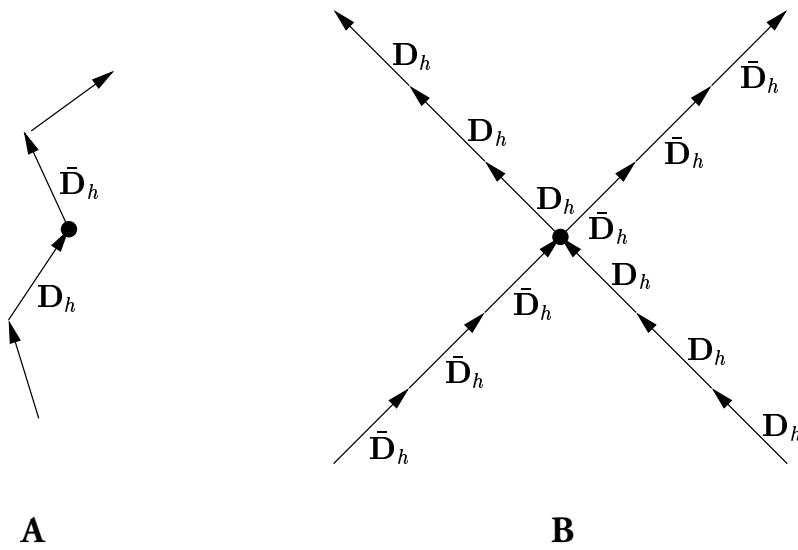


Figure 4.2: The image common for the two translation directions \mathbf{D}_h and $\bar{\mathbf{D}}_h$ are obtained at location marked with a black dot. (A) illustrates the situation where I_t (and \bar{I}_t) are calculated using only two images and (B) illustrates the situation where several images, obtained using the same translation direction, are used for the calculation of I_t (and \bar{I}_t).

marked with a bar in 4.10. This is because the spatial derivatives, in the implementations, can be made to have a time dependence, cf. Equation (4.6) and Section 4.6. Thus, s also depends on the direction of translation.

By eliminating λ between (4.9) and (4.10), the following linear constraint in the elements of A is obtained,

$$\bar{I}_t \mathbf{s}^T A \mathbf{D}_h - I_t \bar{\mathbf{s}}^T A \bar{\mathbf{D}}_h = 0 \quad (4.11)$$

We have one equation of this form for every pixel in the image and for each pair of translations, but only 9 unknowns. Let a_{ij} be the element in row i and column j in A . Equation (4.11) can then be rewritten as,

$$\begin{bmatrix} I_x (\bar{I}_t \mathbf{D}_h - I_t \bar{\mathbf{D}}_h) \\ I_y (\bar{I}_t \mathbf{D}_h - I_t \bar{\mathbf{D}}_h) \\ -(I_x x + I_y y) (\bar{I}_t \mathbf{D}_h - I_t \bar{\mathbf{D}}_h) \end{bmatrix}^T = 0 \cdot \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \\ a_{31} \\ a_{32} \\ a_{33} \end{bmatrix} \quad (4.12)$$

It can be observed that the system matrix, constructed from one instance of (4.12) for each pixel, only has rank 6. Thus, at least one more translation, $\bar{\bar{\mathbf{D}}}_h$, is needed to solve for A . Adding the analogue of (4.12), corresponding to e.g. the translation pair $\bar{\mathbf{D}}_h$ and $\bar{\bar{\mathbf{D}}}_h$, to the linear system gives a system matrix of rank 8. That is, 3 different translations are sufficient. After solving for the elements in A , the resulting matrix is factorized using a modified QR-decomposition [46], yielding a right diagonal matrix K and an orthogonal matrix R_h . The intrinsic camera parameters and the orientation of the camera are consequently obtained simultaneously.

4.3 A Photometric Constraint for General Motions

As a preparation for Section 4.4, on calculating the position of the camera in relation to the robot hand, we will here derive analogue expressions for (4.9) and (4.11) for general motions, i.e. motions including both rotation and translation.

The infinitesimal motion model introduced by Longuet-Higgins and Prazdny [73] will be applied. According to this model the general motion vector of a point \mathbf{X} , in the robot hand coordinate system, can be expressed as

$$\dot{\mathbf{X}} = -\mathbf{D}_h - \boldsymbol{\Omega}_h \times \mathbf{X} \quad , \quad (4.13)$$

where \mathbf{D}_h is the translation vector and $\boldsymbol{\Omega}_h$ is the axis of rotation in robot hand coordinate system. Since the motion of the robot hand is assumed to be known, \mathbf{D}_h and $\boldsymbol{\Omega}_h$ are known. For this motion model to be valid, we have to assume small-angle rotations and that the translation is small compared to the distance to the objects in the scene. However, these restrictions fits us well since small motions still have to be assumed for the optical flow constraint equation (2.10) to be valid, cf. Section 2.1.3.

To be able to work directly with quantities in the images, the constraints should be expressed in the image coordinates $\mathbf{x} = (x, y)$ instead of the 3D coordinates \mathbf{X} . To this end, (4.2) is used to exchange \mathbf{X} for $\mathbf{p} = [x \ y \ 1]^T$ in (4.13). Singling out \mathbf{X} on the right hand side of (4.2) results in

$$\lambda A^{-1} \mathbf{p} + \mathbf{t}_h = \mathbf{X} . \quad (4.14)$$

A vector product can always be expressed using a skew-symmetric matrix.

If $\mathbf{a} = [a_1 \ a_2 \ a_3]^T$ and $\mathbf{b} = [b_1 \ b_2 \ b_3]^T$, $\mathbf{a} \times \mathbf{b}$ can be written as

$$[\mathbf{a}]_{\times} \mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} .$$

Using this notation, inserting (4.14) in (4.13) results in

$$\dot{\mathbf{X}} = -\mathbf{D}_h - [\boldsymbol{\Omega}_h]_{\times} \mathbf{t}_h - \lambda [\boldsymbol{\Omega}_h]_{\times} A^{-1} \mathbf{p} . \quad (4.15)$$

This expression for $\dot{\mathbf{X}}$ is now inserted in the resulting relation (4.8) from Section 4.1,

$$\mathbf{s}^T A(\mathbf{D}_h + [\boldsymbol{\Omega}_h]_{\times} \mathbf{t}_h) + \lambda \mathbf{s}^T A [\boldsymbol{\Omega}_h]_{\times} A^{-1} \mathbf{p} - \lambda I_t = 0 . \quad (4.16)$$

Equation (4.16) is the analogue of (4.9) for a general motion. Just as in Section 4.2, a second motion is performed in order to eliminate the unknown quantity λ . The requirement is, as in the translational case, that the image sequences have one image in common and that the two temporal derivatives are calculated with respect to this image. The constraint arising from the second motion, $(\bar{\mathbf{D}}_h, \bar{\boldsymbol{\Omega}}_h)$, will look like

$$\bar{\mathbf{s}}^T A(\bar{\mathbf{D}}_h + [\bar{\boldsymbol{\Omega}}_h]_{\times} \mathbf{t}_h) + \lambda \bar{\mathbf{s}}^T A [\bar{\boldsymbol{\Omega}}_h]_{\times} A^{-1} \mathbf{p} - \lambda \bar{I}_t = 0 . \quad (4.17)$$

Eliminating λ from (4.16) and (4.17) gives

$$\mathbf{s}^T \mathbf{D}_c (\bar{I}_t - \bar{\mathbf{s}}^T A [\bar{\boldsymbol{\Omega}}_h]_{\times} A^{-1} \mathbf{p}) - \bar{\mathbf{s}}^T \bar{\mathbf{D}}_c (I_t - \mathbf{s}^T A [\boldsymbol{\Omega}_h]_{\times} A^{-1} \mathbf{p}) = 0 , \quad (4.18)$$

where

$$\begin{aligned} \mathbf{D}_c &= A(\mathbf{D}_h - \mathbf{t}_h \times \boldsymbol{\Omega}_h) , \\ \bar{\mathbf{D}}_c &= A(\bar{\mathbf{D}}_h - \mathbf{t}_h \times \bar{\boldsymbol{\Omega}}_h) . \end{aligned}$$

This is a photometric constraint for an uncalibrated camera undergoing two general motions and the analogue of (4.11) in Section 4.2.

4.4 Calculation of Camera Position

In Section 4.2, translational motions were used to calculate the matrix $A = KR_h$, i.e. the intrinsic camera parameters and the orientation of the camera in relation to the robot hand. To calculate the full hand-eye transformation we also need to estimate the position of the camera in relation to the robot hand, i.e. the vector \mathbf{t}_h . Pure translational motions will not give any information about this vector, i.e. its calculation requires motions also including a rotational part, cf. [124].

Unfortunately, the photometric constraint for general motions (4.18) is highly non-linear in the unknowns A and \mathbf{t}_h . No simple linear way of using this constraint to solve for A and \mathbf{t}_h simultaneously has been found. By applying a sequential methodology we can however first make translational motions to calculate A and then use this estimate to simplify the calculation of \mathbf{t}_h . Below, it is therefore assumed that the image coordinates have been adjusted for A , in the sense that the point \mathbf{p} is transformed using $\mathbf{p} \mapsto A^{-1}\mathbf{p}$ and that the line \mathbf{s} is transformed using $\mathbf{s} \mapsto A^T\mathbf{s}$. These transformations will hereafter be assumed and below A is replaced by the identity matrix.

Two different alternatives on the calculation of \mathbf{t}_h will be presented. In the first one it is assumed that pairs of motions of one rotational motion and one pure translational motion are performed, where one image is common for the two motions, a similar situation as in Figure 4.2. Furthermore, it is assumed that the rotation is centered around the origin of the robot hand coordinate system. The second method uses general motions sequences and applies (4.18) where A is assumed to be known.

Alternative 1

Rewrite equation (4.16) for the case of pure rotation of the robot hand and known A ,

$$\mathbf{s}^T [\boldsymbol{\Omega}_h]_{\times} \mathbf{t}_h + \lambda (\mathbf{s}^T [\boldsymbol{\Omega}_h]_{\times} \mathbf{p} - I_t) = 0 \quad . \quad (4.19)$$

Assume that a translation $\bar{\mathbf{D}}_h$ is also performed, resulting in the constraint (4.10), and that the image sequences for the rotational and translational motions have one image in common. If the temporal derivatives are calculated with respect to this image, λ is the same in (4.19) and (4.10). Eliminating λ between (4.19) and (4.10) results in

$$(\bar{I}_t \mathbf{s}^T [\boldsymbol{\Omega}_h]_{\times}) \mathbf{t}_h = -\bar{\mathbf{s}}^T \bar{\mathbf{D}}_h (\mathbf{s}^T [\boldsymbol{\Omega}_h]_{\times} \mathbf{p} - I_t) \quad . \quad (4.20)$$

We get one constraint of this kind for every pixel in the image. The system matrix, constructed from one instance of (4.20) for each pixel, will only have rank 2 because of the skew-symmetric matrix $[\boldsymbol{\Omega}_h]_{\times}$ in the coefficient for \mathbf{t}_h . On account of this, a minimum of two pairs of translational and rotational motions are needed to solve for \mathbf{t}_h . However, the translational motion can be the same for both pairs.

Alternative 2

In this case general motions of the robot hand are used. The calculations are based on (4.18) assuming known A . Rewriting this equation, the following constraint on \mathbf{t}_h is obtained,

$$(B\bar{\mathbf{s}}^T[\bar{\boldsymbol{\Omega}}_h]_{\times} - \bar{B}\mathbf{s}^T[\boldsymbol{\Omega}_h]_{\times})\mathbf{t}_h = -B\bar{\mathbf{s}}^T\bar{\mathbf{D}}_h + \bar{B}\mathbf{s}^T\mathbf{D}_h, \quad (4.21)$$

where

$$\begin{aligned} B &= I_t - \mathbf{s}^T[\boldsymbol{\Omega}_h]_{\times}\mathbf{p}, \\ \bar{B} &= \bar{I}_t - \bar{\mathbf{s}}^T[\bar{\boldsymbol{\Omega}}_h]_{\times}\mathbf{p}. \end{aligned}$$

If both motions, $(\mathbf{D}_h, \boldsymbol{\Omega}_h)$ and $(\bar{\mathbf{D}}_h, \bar{\boldsymbol{\Omega}}_h)$, contains significant rotational parts, the system matrix constructed from one instance of (4.21) for each pixel has full rank. For a right hand side to exist, one of the motions also needs to include a significant translational part. If these conditions are fulfilled \mathbf{t}_h can be calculated from two general motions.

4.5 A Comment on a Nonlinear Optimization Stage

After A and \mathbf{t}_h has been calculated linearly, using the procedures in Sections 4.2 and 4.4, a nonlinear optimization scheme could be applied with the linear solutions as starting point. To this end, general motions are used and the constraint (4.18) is applied. The nonlinear least squares error can be minimized with respect to A and \mathbf{t}_h using, for example, the Levenberg-Marquardt method [85]. An analytical expression for the Jacobian is rather easy to obtain.

However, in our experiments, we did not find that the nonlinear optimization stage improved the results. A deeper analysis has to be performed to fully understand the structure of the cost function. For 3D motion estimation, it has been argued in [35] and [117] that the minima of the adopted cost functions lie in a narrow “valley”, which indicates an ambiguity between the translational and rotational parameters. A similar situation could be present in our setting. Intuitively, an ambiguity between the location of the principal point and the translational part of the hand-eye transformation seems probable.

4.6 Estimating the Derivatives

Since direct calibration is based solely on the spatial and temporal image derivatives, success of the algorithm depends heavily on good estimates of the derivatives. The most successful way of estimating the derivatives that we have found is by convolving the images

with differentiated Gaussian kernels. For example, the kernel used for the x -derivative looks like

$$G_x(x, y) = \frac{-x}{\sqrt{2\pi\sigma_x^3\sigma_y}} e^{-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2}} \quad (4.22)$$

This means that the derivative estimations include smoothing. Actually, it seems like the calculations benefit from a rather large degree of smoothing. Since we are not interested in the reconstruction of any scene structure, we can allow ourselves to use quite large smoothing kernels. In the experiments, kernels as large as $\sigma = 7$ have been used, where the width is expressed in pixels.

Concerning the temporal derivative, we also got the best results by weighting the intensities over time using a one-dimensional differentiated Gaussian kernel. This means that a number of images must be obtained while performing the same motion, cf. Figure 4.2. In the case where the motion change after each image, the temporal derivative is estimated simply by the difference in intensities of two subsequent images.

It has been noticed that the calculations benefit from smoothing the images in the spatial direction before calculating the temporal derivatives. Also, smoothing the spatial derivatives through time in the same degree as in the estimation of the temporal derivative improved the results. In the case where the motion change after each image, this smoothing consists of a simple averaging of the spatial derivatives in two subsequent images.

4.7 Experiments

Experiments on both computer generated image sequences and on image sequences from a real robot vision system have been carried out. The advantage of the computer generated experiments is that we have exact knowledge of the ground truth, i.e. the correct intrinsic parameters and the correct hand-eye transformation. Further, the images are initially noise free and the lighting conditions are controlled.

4.7.1 Experiments using Computer Generated Images

The images in these experiments were generated using the raytracing software POV-ray 3.1. The scene in view consisted of two textured perpendicular planes (a floor and a wall) with a small cube placed on the “floor”, cf. Figure 4.3.

Motion of the robot hand and the induced motion of the camera was simulated. The rotation matrix R_{he} describing the orientation of the camera in relation to the robot hand was chosen as

$$R_{he} = \begin{bmatrix} 0.8365 & -0.4084 & -0.3653 \\ 0.2241 & 0.8634 & -0.4520 \\ 0.5000 & 0.2962 & 0.8138 \end{bmatrix},$$

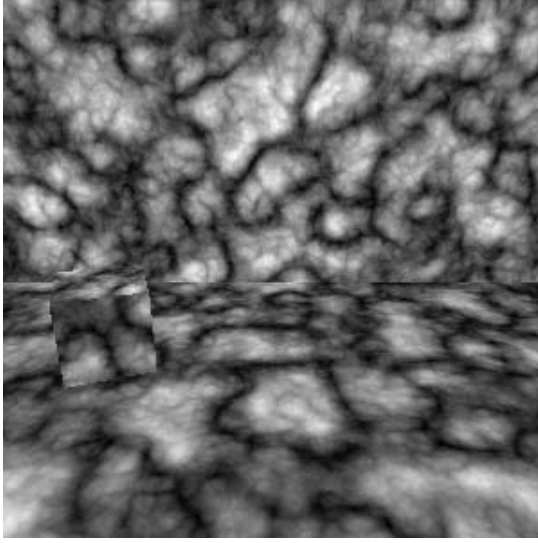


Figure 4.3: An image from the computer generated sequence.

i.e. a rotation 20° around the x-axis followed by a rotation 30° around the y-axis and a rotation 15° around the z-axis, using a fixed coordinate system.

The position of the camera in relation to the robot hand was described by the translation vector $(T_X, T_Y, T_Z) = (4, 2, 3)$. The unit of this vector is the length unit of the computer graphics scene that the artificial camera is seeing. In our experiments the wall in the background of the images was located at a distance of 15 length units (l.u.) from the initial position of the camera. The calibration matrix K was chosen as

$$K = \begin{bmatrix} 260.67 & 0 & 20 \\ 0 & 347.56 & 10 \\ 0 & 0 & 1 \end{bmatrix},$$

i.e. the focal length f was 347.56, the aspect ratio $\alpha = 0.75$, the skew $s = 0$ and the principal point (x_0, y_0) was $(20, 10)$, where the center of the image is located at $(0, 0)$. The origin of the coordinate system of the image plane was located in the middle of the image. We have tested the influence on the calibration result of mainly four aspects of the calibration algorithm; the choice of translation directions, the choice of step size in the translation δ , the choice of width of the smoothing kernels σ_s and the sensitivity to added noise with standard deviation σ_n .

The intrinsic parameters and hand-eye orientation were calculated using the method

described in Section 4.2. Following this, the hand-eye translation was calculated using alternative 1 in Section 4.4. For each motion performed, 9 images were obtained.

First we tested the influence of the choice of translation directions. The three translation directions were chosen as $\mathbf{D} = (0, -\sin \theta, \cos \theta)$, $\bar{\mathbf{D}} = (-\sin \theta, 0, \cos \theta)$ and $\bar{\bar{\mathbf{D}}} = (\frac{1}{\sqrt{2}} \sin \theta, \frac{1}{\sqrt{2}} \sin \theta, \cos \theta)$ in the robot hand coordinate system. Results on the intrinsic calibration when varying θ are shown in Table 4.1. We see that f , α and s are

θ	f	α	s	x_0	y_0
	347.56	0.75	0	20	10
10°	347.86	0.7524	0.0000	18.41	12.61
20°	348.26	0.7516	-0.0003	20.64	8.96
30°	347.24	0.7511	-0.0011	18.79	9.52
40°	347.07	0.7517	-0.0012	19.04	9.94
50°	347.58	0.7512	-0.0012	19.62	9.90
60°	347.34	0.7510	-0.0004	19.98	11.59
70°	347.36	0.7479	0.0003	20.64	11.33
80°	346.15	0.7482	0.0018	21.78	12.58

Table 4.1: Results on the intrinsic calibration when changing the translation directions.

well estimated for most of the angles in the interval, but the principal point (x_0, y_0) is more unstable. This instability of the principal point will also be noticeable in the other experiments presented here. The results seem most accurate in the middle section of the interval, i.e. when the translation directions are not too close together and not too near a planar configuration. This tendency can also be noticed when studying the influence of the choice of translation directions on the orientation and position calculations, but for these parameters the degradation of the result is more apparent for the largest values on the angle θ than for the smallest. In the following, the translation directions corresponding to $\theta = 50^\circ$ has been used.

The spatial and temporal derivatives in the calculations were approximated by convolving the images with differentiated Gaussian kernels with standard deviation σ_s , cf. Section 4.6. The next thing we tested was to vary σ_s , i.e. varying the width of the smoothing kernels. In these experiments the same width was used for both the spatial and temporal derivatives. The results are shown in Table 4.2 for the extrinsic calibration and in Table 4.3 for the intrinsic calibration. For the orientation calculation the result is shown as the deviation in degrees of the calculated rotation axis from the real rotation axis. The width of the smoothing kernels σ_s is expressed in pixels. In this noise free experiment it seems like the quality is rather unaffected by the choice of smoothing kernels, as long as it is not too small, i.e. less than 1.

The most critical parameter in the choice of motion seems to be the step size between

σ_s	$d\theta$	T_X	T_Y	T_Z
	0	4	2	3
0.5	2.8829	4.4312	1.5220	1.2453
1.0	0.3017	4.0140	1.9856	2.9961
1.5	0.0817	3.9955	2.0018	3.0092
2.0	0.0586	3.9982	2.0000	3.0079
2.5	0.0287	4.0017	2.0008	3.0060
3.0	0.0440	4.0061	2.0030	3.0035

Table 4.2: Results on the extrinsic calibration when changing the width σ_s of the smoothing kernels.

σ_s	f	α	s	x_0	y_0
	347.56	0.75	0	20	10
0.5	368.01	0.6546	-0.0514	13.90	84.54
1.0	347.58	0.7517	-0.0012	19.62	9.90
1.5	347.34	0.7515	-0.0008	20.55	9.89
2.0	347.26	0.7513	-0.0005	20.81	9.73
2.5	347.25	0.7512	-0.0002	20.99	9.53
3.0	347.30	0.7510	-0.0002	21.15	9.51

Table 4.3: Results on the intrinsic calibration when changing the width of the smoothing kernels σ_s .

the images in the image sequences. This is of course related to how far away the main objects in the current view are located. As mentioned earlier, in our experiments the wall in the background of the images is located at a distance of 15 l.u. . This is to be compared to 0.1 l.u. , which is the step size used in most of the experiments presented here. In Table 4.4 we see that just doubling the step size to 0.2 l.u. reduces the quality of the extrinsic calibration considerably. The same is true for the intrinsic calibration, especially for the calculation of the principal point. However, as seen in the bottom part of Table 4.4, by increasing the width of the smoothing kernels, σ_s , much more accurate estimations can be obtained.

The last thing we tested was the sensitivity to noise. Gaussian distributed noise was added with increasing standard deviation σ_n . The standard deviation is here measured in gray levels. The number of gray levels in the images was 256. The method is quite insensitive to rather large amounts of noise, especially when increasing the width of the smoothing kernel σ_s . In Figure 4.4 and 4.5 we show the influence of the noise level on the focal length and on the position and orientation calculations using two different widths

$\delta; \sigma_s$	$d\theta$	T_X	T_Y	T_Z
	0	4	2	3
0.05;1	0.0560	4.0426	1.9765	2.9720
0.10;1	0.3017	4.0140	1.9856	2.9961
0.15;1	0.8337	4.1129	1.9972	2.9597
0.20;1	1.6260	4.2541	2.0213	2.9367
0.25;1	3.8763	4.3972	2.1212	2.8765
0.05;2	0.0222	4.0004	2.0038	3.0051
0.10;2	0.0586	3.9982	2.0000	3.0079
0.15;2	0.0549	4.0033	2.0004	3.0051
0.20;2	0.4536	4.0157	2.0087	3.0021
0.25;2	2.1206	4.0332	2.0821	2.9666

Table 4.4: Results on the extrinsic calibration when changing the translation step size δ and the width of the smoothing kernels σ_s .

σ_s of the smoothing kernels. The dotted lines in the figures corresponds to the results using $\sigma_s = 1$ and the solid line corresponds to $\sigma_s = 2$. The unit on the horizontal axes is the standard deviation σ_n , measured in pixels. The unit on the vertical axis in the right diagram in Figure 4.5 is the deviation in degrees from the correct rotation axis. As can be seen, the improvement of the results is rather dramatic when doubling σ_s . The good handling of noise is probably due to the enormous amount of data that the calculations are based on. In the images used here, the actual number of pixels that gives rise to one constraint each was more than 85000 (The images were originally 301×301 pixels but shrunk a bit during the smoothing).

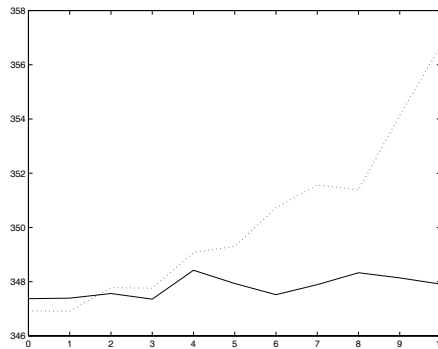


Figure 4.4: Sensitivity to noise of the focal length.

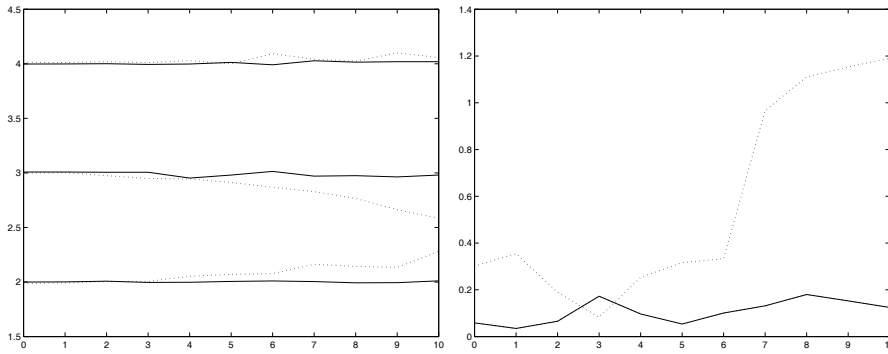


Figure 4.5: Sensitivity to noise of the position calculation (left) and orientation calculation (right).

4.7.2 Experiments using a Real Robot Vision System

To try the method on a real robot vision system, we used the same robot vision system as in the experiments of Chapter 3, cf. Section 3.4.1.

The method described in Section 4.2 was used to calculate the intrinsic camera parameters and the hand-eye orientation. Unfortunately, no satisfactory results have been obtained for the calculation of the camera position using real image data. Possible reasons for this is discussed in the conclusions of this chapter, Section 4.8.

Figure 4.6 shows an image from the image sequence used in these experiments. The scene consists of a carpet, layed out on the floor approximately 1.5 m from the camera and the robot hand. Objects were put under the carpet so that the scene would not be completely planar. The size of the images are 240×320 pixels. The robot hand was moved along the x , y and z axes of the robot hand coordinate system and 8 images on each side of the origin along each axis were obtained. For the estimation of the temporal derivative, a smoothing kernel of width $\sigma_s = 1$ was used.

As a reference, the intrinsic camera calibration was also performed using the Camera Calibration Toolbox for Matlab by J.-Y. Bouguet. This is an implementation of the plane-based calibration techniques proposed by Zhang [142] and Sturm and Maybank [119], cf. Section 3.1. We used 15 images of a calibration plane from different viewpoints for this calibration. For the orientation calculation we used as reference the method for hand-eye calibration proposed by Tsai and Lenz [124].

The result of the intrinsic calibration using the Calibration Toolbox was

$$K_{toolbox} = \begin{bmatrix} 573.98 \pm 6.14 & 0.00 \pm 0.00 & 146.11 \pm 5.49 \\ 0 & 596.42 \pm 6.48 & 130.12 \pm 4.73 \\ 0 & 0 & 1 \end{bmatrix}.$$

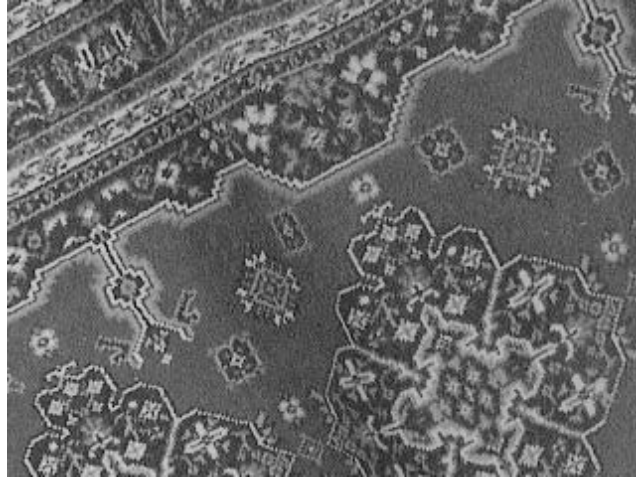


Figure 4.6: An image from the real image sequence.

Here, a nonlinear optimization stage was applied, which gave estimations of the uncertainties for the different parameters.

Two image sequences with different step sizes δ , i.e. the distance between each obtained image, were used in the experiments. Starting with the step size $\delta = 0.5$ cm, the following calibration matrices were obtained using two different widths of the spatial smoothing kernels, $\sigma_s = 1$ and $\sigma_s = 3$,

$$K_{\sigma_s=1} = \begin{bmatrix} 572.05 & 8.66 & 162.10 \\ 0 & 595.12 & 132.17 \\ 0 & 0 & 1 \end{bmatrix} \quad K_{\sigma_s=3} = \begin{bmatrix} 580.66 & 3.38 & 173.11 \\ 0 & 595.59 & 146.58 \\ 0 & 0 & 1 \end{bmatrix} .$$

Using the step size $\delta = 1.0$ cm the following calibration matrices were obtained using $\sigma_s = 3$ and $\sigma_s = 5$,

$$K_{\sigma_s=3} = \begin{bmatrix} 573.09 & 4.01 & 130.01 \\ 0 & 595.12 & 127.45 \\ 0 & 0 & 1 \end{bmatrix} \quad K_{\sigma_s=5} = \begin{bmatrix} 570.78 & 7.03 & 140.20 \\ 0 & 589.91 & 125.82 \\ 0 & 0 & 1 \end{bmatrix} .$$

The agreement of the method proposed in this chapter using translational motions with the plane-based calibration technique for general motions is striking. The instability of the principal point that could be noted in the experiments on the computer generated data is also apparent here. The uncertainty in the estimation of the principal point is probably due to the small movements of the camera, compared to the distance to the object in the scene.

The hand-eye calibration method of Tsai and Lenz, using the extrinsic parameters from the plane-based camera calibration, gave the following rotation matrix for the current setup,

$$R_{Tsai} = \begin{bmatrix} 0.8666 & 0.4858 & -0.1139 \\ -0.4987 & 0.8510 & 0.1644 \\ 0.0171 & -0.1993 & 0.9798 \end{bmatrix},$$

This amounts to a rotation of -11.50° around the x -axis, -0.98° around the y -axis and -29.91° around the z -axis, using a fixed reference frame. This can be compared to the hand-eye orientation obtained by our method, using step size $\delta = 1$ cm and spatial smoothing kernel $\sigma_s = 3$,

$$R_{Malm} = \begin{bmatrix} 0.8707 & 0.4855 & 0.0780 \\ -0.4915 & 0.8547 & 0.1668 \\ 0.0143 & -0.1836 & 0.9829 \end{bmatrix},$$

which amounts to a rotation -10.58° around the x -axis, -0.81° around the y -axis and -29.44° around the z -axis.

4.8 Conclusions

In this chapter, a method for intrinsic camera calibration and hand-eye calibration, based directly on the spatial and temporal image derivatives in an image sequence, has been presented. The method is divided into mainly two stages, where in the first stage the intrinsic camera parameters and the hand-eye orientation are estimated from an image sequence obtained during translational motions. In the second stage, the position of the camera in relation to the robot hand is obtained using motions also including a rotational part.

The method has been tested on both computer generated images and images obtained using a real robot vision system. The performance of the algorithm is very good using the computer generated images. The calculations are rather unaffected by, for example, the choice of translation directions and the width of the smoothing kernels in this setting. The method here also shows a remarkable insensitivity to added noise.

Using real images, the algorithm is however much less robust. We have, for example, not been able to obtain a reasonable value on the position of the camera in relation to the robot hand, even though this parameter is very well estimated using the computer generated images. The main reason for this large difference in performance is probably that the field of view of the camera used in the real experiments is much more narrow than for the simulated camera in the computer generated case. Studies on scene reconstruction and motion estimation from image flows [117, 35] have shown that a rotational flow field is very hard to discriminate from a translational flow field if the field of view is small.

Accurate results can, as shown by the experiments, be obtained for the focal lengths and the hand-eye orientation and for the estimation of these parameters the approach is probably useful. However, difficult or changing lighting conditions, and textures where a good estimate of the derivatives is hard to achieve, can still have a considerable negative effect on the result. For example, a fluorescent light in our robot laboratory was noted to have negative influence on the estimations.

A deeper analysis of the method and more experimental studies are needed to fully evaluate its usefulness. One idea for further development of the method is to apply a coarse-to-fine iterative scheme, cf. [117], to be able to use larger motions.

Part II

Geometrical Optical Illusions

Chapter 5

Uncertainty in Visual Processes Predicts Geometrical Optical Illusions

By their way of deceiving our eyes and mind, optical illusions have always been a source of fascination. Most people are familiar with some illusions, e.g. the Müller-Lyer illusion, Figure 5.1A, or the illusory drawings by Escher or Reutersvärd, Figure 5.1B. Through history architects have used optical illusions to alternate the perception of a space or structure. For example, the architects of ancient Greece made their columns slightly swollen to compensate for the fact that tall thin structures appear to be slightly shrunken in the middle. Another effect, observed by almost everyone and documented on very early in history, is the moon illusion, i.e. the phenomenon that the moon appears to be smaller at zenith than at the horizon. Ptolemy 150AD gave the explanation for this that is still considered today.

In the nineteenth century, as scientists started to get interested in perception, optical illusions became the subject of systematic study. In different mind-related fields such as psychology, neurology, physiology, and philosophy, studies of illusions have been performed. A strategy for finding out how correct perception operates is to observe situations where it fails. By carefully altering the stimuli and testing the changes in visual performance, researchers have tried to gain insight into the principles of perception. From a large number of experimental studies a lot has been learned about the parametric influences on illusory patterns, i.e. which orientation, thickness, number of lines etc. that causes the maximum or minimum illusory effect. Using this knowledge it is possible to invent variations on illusory figures. To do this there is no need to understand in any depth why we perceive the illusion.

At this time, hardly anyone involved in the study of perception would acknowledge that there exists any general theory explaining optical illusions. It is therefore an intriguing task to try to reduce all observations to a few major principles underlying large classes of illusions, that later might be expressed in one all-embracing theory. In this chapter one such major principle is presented that builds an explanatory framework for the best known and most studied illusions of all, the **geometrical optical illusions**.

The Proposed Theory

The term geometrical optical illusions, or *geometrisch-optische Täuschungen*, was coined by Opper [99] in a paper about the overestimation of an interrupted as compared to an uninterrupted extent, cf. Figure 5.2, later called the Opper-Kundt illusion [71]. The term is now used for any illusion that can be seen in line drawings. Other famous illusions in

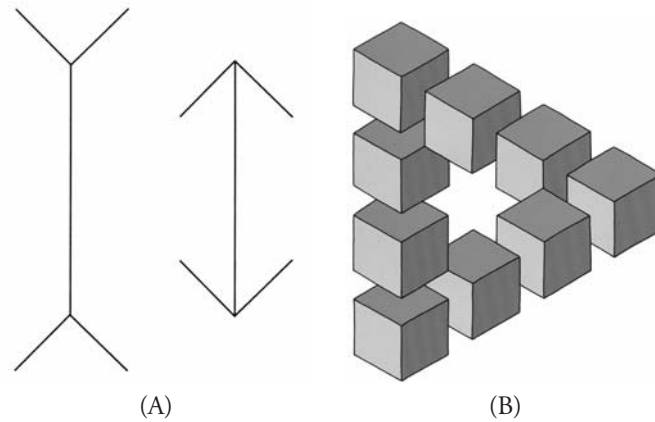


Figure 5.1: (A) The Müller-Lyer illusion, (B) Opus 1 (1934) by Oscar Reutersvärd [108]. See Sparr [116] and Heyden [57] for computational vision analyses of the “impossible” drawings by e.g. Escher and Reutersvärd.

this class include the before mentioned Müller-Lyer illusion [90], the Fraser spiral [39], and the Poggendorff and Zöllner illusions [146]. The contrast effect, i.e. the underestimation of a small extent near a large one and the overestimation in brightness of a bright entity next to a dark one etc. [102], is usually also included here. The number of illusory patterns that fall into this class is very large, and the underlying phenomena can seem to be quite diverse.

The claim of this work is that the statistics of visual computations is the cause, or one of the major causes, underlying geometrical optical illusions and illusory effects due to motions signals in line drawings. When interpreting a pattern, features in the image such as points and lines, intersection of lines or local image motion must be derived, i.e. they must be estimated from the input data. Noise in the data, whose origin can be of different kinds, causes the estimation of these features to be biased. As a result, the locations and orientations of the features are perceived erroneously and the appearance of the pattern is altered. The bias occurs in any visual processing, but under average conditions it is not large enough to be noticeable. Illusory patterns are constructed such that the bias is strongly pronounced.

Turning to the human visual system, our eyes receive as input a sequence of images. Even if we fixate on a static pattern eye movements are performed and a series of images is gathered. According to experts on eye movements, cf. [12], the data from a number of retinal images are combined into one image representation. Helmholtz [128] refers to the representation of the retinal image as the field of fixation, *Blickfeld*, and to the representation that moves with the eye as the visual globe, *Sehfeld*. More recently the

terms retinotopic frame and stable feature frame have been used [32].



Figure 5.2: The Oppel-Kundt illusion [110].

The early visual processing apparatus extracts local image measurements from the image sequence which serve as input to further estimation processes. For the modeling conducted here we consider the following kinds of measurements: the positions and gray value magnitudes of image points, the orientations and positions of small edge elements (edgels) and the direction and lengths of image motion vectors perpendicular to local edges (cf. normal flow, Section 2.1.3 and Chapter 4). These image measurements can only be derived within a range of accuracy, in other words, noise influence the measured quantities. The first interpretation processes consist of estimating local edges using as input all the available image points, estimating the intersections of lines using all the edgels and computing local image motion using all the normal flow measurements in a small region. These estimation processes are biased. Thus the perceived positions of image points and edges are shifted, their directions tilted and their motion wrongly estimated. These perceived features then serve as input to the next higher level in the interpretation process. Long straight lines or general curves are fitted to the edgels and this gives rise to the tilted and displaced straight lines and distorted curves that are perceived in many illusionary patterns. In the case of motion, the local image measurements are combined in segmentation and 3D motion estimation processes to interpret the spatiotemporal content of the scene. It should be emphasized that the estimation processes discussed are not specific for the human vision system but are applied by most visual systems, natural or artificial.

In the case of gray value estimation, theories related to our approach have previously been presented. For example, optical and neural blur has been argued to be the cause of some geometrical optical illusions [49, 50, 43, 44, 45]. Also, models of band-pass filtering have been proposed to account for a number of illusions [95, 93, 91, 9, 26]. These studies invoked in intuitive terms the concept of noise and uncertainty, since band-pass filtering constitutes a model of edge detection in noisy gray level images and since optical and neural blurring are related to the smoothing that we will use to model uncertainty in grey value. The theme of our study is that these previous findings constitute a special case of a more general principle, namely that uncertainty and noise cause bias in the estimation of image features. It will be shown that this principle accounts for a large number of geometrical optical illusions that previously have been considered unrelated.

In the following three sections, bias in grey value estimation, in the estimation of edge elements and bias in motion estimation will be analyzed in succession. For each estimation model, we discuss a number of illusions that are best explained by that model. However, it should be emphasized that all three kinds of data are used in almost any

interpretation process and that they are certainly related. Noisy edgel data and noisy spatiotemporal data can for example be derived from noisy gray values. The relationships between the estimation processes are not investigated in this work. After these three main sections an overview of previous related work on illusions is given in Section 5.4. This is then followed by some further discussion and conclusions in Section 5.5. The calculation of the expected value of the utilized least squares estimations is given in Appendix A.

5.1 Bias in Gray Value Estimation

5.1.1 Basic Theory

Let the intensity in an ideal noise free image sequence, parameterized by image position (x, y) and time t , be $I(x, y, t)$. The images retrieved at the retina can be thought of as a noisy version of the ideal signal. In general, two kinds of noise sources must be considered, noise in the gray level value and noise in the spatiotemporal location.

At every point (x, y, t) an additive, independently and identically distributed noise term $\delta I(x, y, t)$ with zero mean and covariance matrix Λ_g is considered. This represents the error in gray level value and is modeled as a Gaussian distribution, i.e. $\delta I(x, y, t) \approx N(0, \Lambda_g)$. Error in position is also considered. Suppose that the ideal signal from the scene should be received at location (x, y) in the image. The positional uncertainty is modeled using a Gaussian distribution centered at (x, y) with covariance matrix Λ_p . The expected value of the intensity thus amounts to the convolution of the signal with a spatiotemporal Gaussian kernel, i.e.

$$E((I(\mathbf{x}) + N(0, \Lambda_g)) * G(\mathbf{x}, \Lambda_p)) , \quad (5.1)$$

where $\mathbf{x} = (x, y, t)$ and

$$G(\mathbf{x}, \Lambda_p) = \frac{1}{(2\pi)^{3/2} |\Lambda_p|^{1/2}} e^{-\frac{1}{2} \mathbf{x}^t \Lambda_p^{-1} \mathbf{x}} . \quad (5.2)$$

In the modeling of static signals only the spatial variables need to be considered. Furthermore, for simplicity, the spatial noise components in the x - and y -directions are assumed to be identical and independent throughout this chapter, where σ_g is the standard deviation in the gray level noise and σ_p is the standard deviation in the positional noise. The expected value of the intensity is then

$$\begin{aligned} E((I(x, y) + N(0, \sigma_g)) * G(x, y, \sigma_p)) = \\ E(I(x, y) * G(x, y, \sigma_p)) + E(N(0, \sigma_g) * G(x, y, \sigma_p)) . \end{aligned} \quad (5.3)$$

Edges will be modeled as the extrema of the first-order derivatives [11], or as the zero crossings of the Laplacian operator $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$, cf. [87, 86]. As seen in (5.3), the

expected value can be written as the sum of two terms. The second term, the expected value of $N(0, \sigma_g) * G(x, y, \sigma_p)$, and its derivatives are independent of the position. This means that noise in the gray value does not affect the expected locations of edges and for the purpose of this analysis it can be ignored.

Note that

$$G(\mathbf{x}, \sigma_p) * \frac{\partial I(\mathbf{x})}{\partial \mathbf{x}} = \frac{\partial G(\mathbf{x}, \sigma_p)}{\partial \mathbf{x}} * I(\mathbf{x}) \quad , \quad (5.4)$$

and, accordingly, that

$$G(x, y, \sigma_p) * \nabla^2 I(x, y) = \nabla^2 G(x, y, \sigma_p) * I(x, y) \quad . \quad (5.5)$$

Hence, differentiating $I(x, y)$ and then smoothing it with a Gaussian is equivalent to calculating the differentiated Gaussian distribution function and use this for the convolution with $I(x, y)$, cf. Section 4.6 and Section 6.1.1. In the case of second-order derivatives, convolution with the **Laplacian of Gaussian** (LoG) filter, $\nabla^2 G(x, y, \sigma)$, and then locating the zero-crossings is a common method for edge detection in image analysis [87]. The LoG filter can be closely approximated with the difference of two Gaussian kernels, a Difference of Gaussians filter (DoG). It has been argued that the edges in the human early visual system is computed using physiological realizations of DoG filters [111, 28, 136, 135, 86]. Thus we are interested in the locations (x, y) where

$$\left[\left(\frac{\partial G(x, y, \sigma_p)}{\partial x} * I(x, y) \right)^2 + \left(\frac{\partial G(x, y, \sigma_p)}{\partial y} * I(x, y) \right)^2 \right]^{1/2} \quad , \quad (5.6)$$

has a local maximum, or where

$$\nabla^2 G(x, y, \sigma_p) * I(x, y) = 0 \quad . \quad (5.7)$$

So the implementation of edge detection in the visual system could be one source of positional uncertainty. Another interpretation is that there is uncertainty in the position since a sequence of images is combined into one image representation and errors occur in the geometric compensation for location. A third interpretation is that the eye can be viewed as a defocused imaging system, which even for only one image taken at one instant in time can be modeled by Gaussian smoothing. Since repeated convolution with different Gaussian kernels is equivalent to convolution with a wider Gaussian kernel (with a standard deviation that is the square root of the sum of the variances of the kernels), any combination of the above mentioned noise sources is also well approximated by our model.

Gaussian smoothing of static images has been intensively studied in the literature on linear scale space theory, cf. Section 6.1. A large number of theoretical results has been derived which can be employed straightforwardly in our study. For example, Lindeberg [72] derives formulae for the instantaneous velocity of edge points in the direction normal

to edges when the scale parameter t is varied, the so-called **drift velocity**. This can be used when studying how the estimation of the position of an edge changes under different amounts of positional noise, i.e. under different amounts of Gaussian smoothing.

Consider at every edge point p_0 a local orthonormal coordinate system (η, ξ) with the η -axis parallel to the spatial gradient direction at p_0 and the ξ -axis perpendicular to it, cf. gauge coordinates in Section 6.1. If the edges are given as the zero-crossings of the Laplacian, the drift velocity $(\partial_t \eta, \partial_t \xi)$ amounts to

$$(\partial_t \eta, \partial_t \xi) = -\frac{\nabla^2 (\nabla^2 I)}{2 \left((\nabla^2 I_\eta)^2 + (\nabla^2 I_\xi)^2 \right)} (\nabla^2 I_\eta, \nabla^2 I_\xi) . \quad (5.8)$$

For a straight edge, where all the directional derivatives in the η -direction are zero, this simplifies to

$$(\partial_t \eta, \partial_t \xi) = -\frac{1}{2} \frac{I_{\xi\xi\xi\xi}}{I_{\xi\xi\xi}} (0, 1) . \quad (5.9)$$

The scale space behavior of straight edges is illustrated in Figure 5.3. Three cases are considered. Firstly, edges between a dark and a bright region do not change location under Gaussian smoothing, cf. Figure 5.3A. Secondly, the two edges at the boundaries of a bright line, or bar, in a dark region drift apart, assuming that the width of the smoothing kernel is large enough so that the whole bar affects the edges, cf. Figure 5.3B. Finally, the effect of the smoothing on a line of medium brightness next to a bright and dark region is to move the two edges towards each other, Figure 5.3C. These observations suffice to explain a number of illusions.

5.1.2 Illustrative Examples

Two geometrical optical illusions provided by the Department of Psychology at Akita University [2] will be studied. These illusionary patterns consist of regular grids and checkerboard patterns with small squares superimposed on them. The effect of the squares is to change the appearance of straight lines to bended curves.

Figure 5.4 shows one of the figures, a black square grid on a white background with small black squares superimposed. Figure 5.4B shows the central part of the figure magnified. The black squares at the center of the squares have been removed for sake of clarity since they do not notably affect the perception of the illusion. In Figure 5.4C the result of edge detection on the raw image using the Laplacian of a Gaussian (LoG) is shown. Figure 5.4D shows the smoothed image which results from filtering with a Gaussian kernel with standard deviation $5/4$ times the width of the bars. In Figure 5.4E the results of edge detection on the smoothed image using a LoG is shown. (This is clearly the same as performing edge detection on the raw image using a LoG with larger standard deviation).

The grid consists of lines (or bars), and the effect of Gaussian smoothing on the bars is to drift the two edges apart. However, at the locations where a square is aligned with

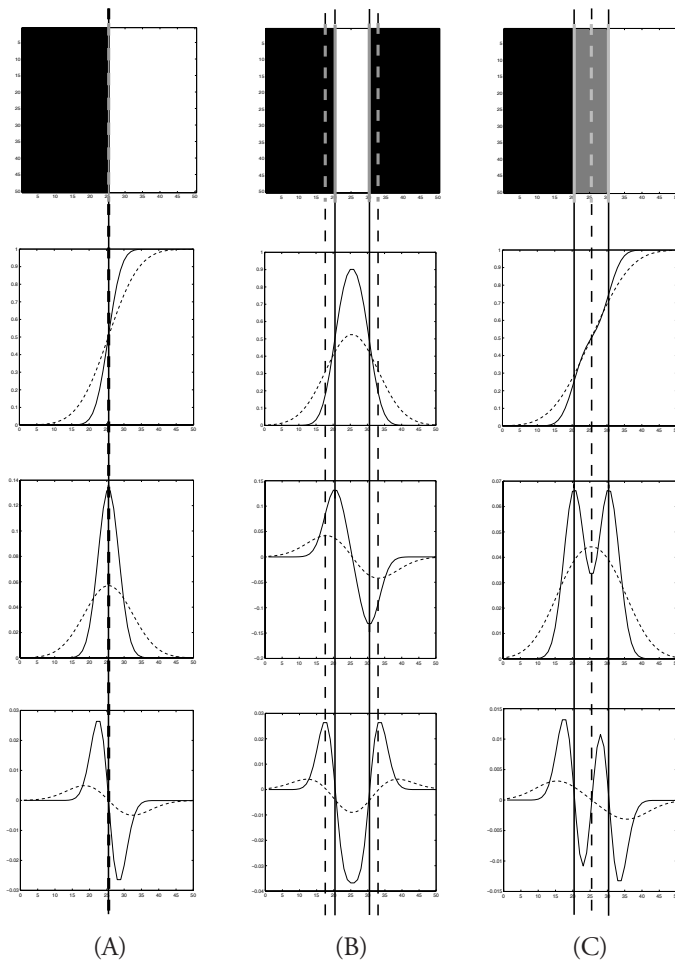
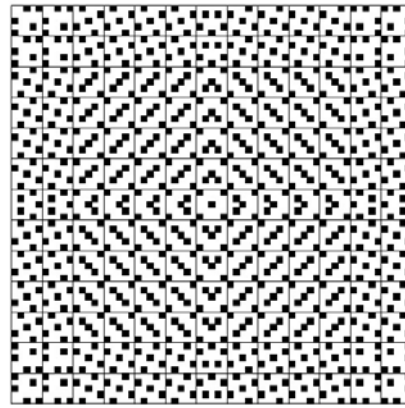
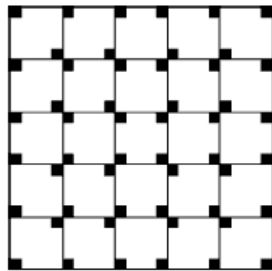


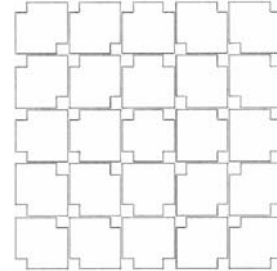
Figure 5.3: The first row shows gray level figures of, (A) a dark region next to a bright region, (B) a bright line in a dark region, and (C) a gray line next to a dark and a bright region. Superimposed on the gray level figures are the edges which have been computed as the zero crossings of the Laplacian of a Gaussian (LoG) for two different widths of the Gaussian. The solid lines correspond to a standard deviation, σ , of 3 pixels and the dashed lines to a σ of 7 pixels. The second row shows cross sections of the two differently smoothed versions of the gray level figures above. The third row shows the first order derivatives, and the fourth row the second order derivatives of these functions. The maxima of the first order derivatives and the zero crossings of the second order derivatives are marked by vertical lines. From their change in position the observations about the movement of edges in scale space can be deduced: In (A) the edge stays in place, in (B) the edges drift apart, and in (C) the edges are getting closer (collapse into one for $\sigma = 7$).



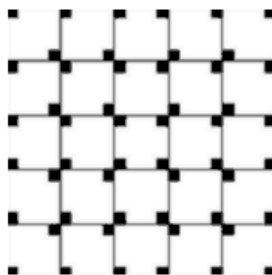
(A)



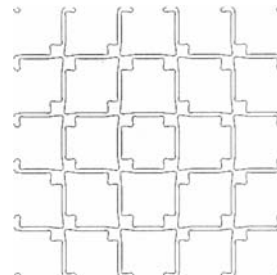
(B)



(C)



(D)



(E)

Figure 5.4: (A) Illusory pattern: “spring” [2]. (B) A small (simplified) part of the illusion to which (C) edge detection, (D) Gaussian smoothing, and (E) smoothing and edge detection have been applied.

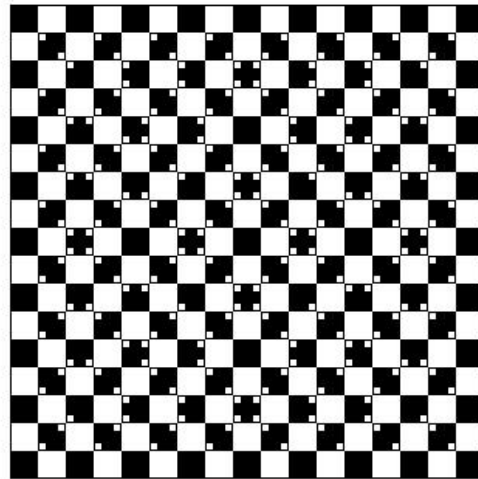
the grid there is only one edge and this edge does not change location under Gaussian smoothing. The net effect of the smoothing is that the edges of the grid lines are no longer straight. The curvature of the edges obtained from the smoothing is qualitatively similar to the curvature perceived in the figure, cf. Figure 5.4A and Figure 5.4E.

Figure 5.5A shows the second pattern from [2]. This pattern consists of a black and white checkerboard with small white squares superimposed in the corners of the black tiles close to the edges. Next to the small white squares short black bars are created – a white area (from a small white square) next to a black bar (from a black checkerboard tile) next to a white area (from a white checkerboard tile). This explains the effect: The edges which are the boundaries of the created bars drift apart under scale-space smoothing and the other edges, between the black and white tiles of the checkerboard, stay in place. The result is that the edges near the locations of the small white squares are pushed outwards toward the white checkerboard tiles. This is illustrated in Figure 5.5B which shows the combined effect of smoothing and edge detection for a part of the pattern. Figures 5.5C and 5.5D zoom in on the drift velocity in the smoothed image as derived from (5.8). Again the effect obtained from the smoothing is the same as the one perceived in the original illusionary pattern.

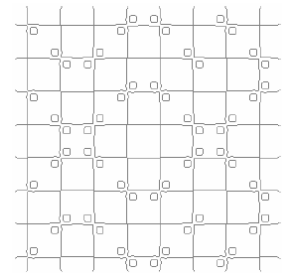
Another geometrical optical illusion in this category is the famous "café wall" illusion, shown in Figure 5.6A. It consists of a black and white checkerboard pattern with alternate rows shifted one half-cycle and with thin "mortar" lines separating the rows. The mortar lines should be midway in luminance between the black and white rectangles to give the strongest illusory effect. The effect is a tilt in the mortar lines, with alternate mortar lines tilted in the opposite directions.

In the café wall figure, at the locations where a mortar line borders both a dark tile and a bright tile, the two edges move towards each other under Gaussian smoothing and for thin lines it takes a relatively small amount of smoothing for the two edges to merge into one. In positions where the mortar line is between two bright regions or where it is between two dark regions, the edges move away from each other. The results of smoothing and edge detection are shown in Figure 5.6C for the magnified part of the pattern shown in Figure 5.6B. Short line segments are apparent here which are tilted with the same sign of the slope as perceived in the original illusionary pattern. Figures 5.6D and 5.6E show the drift velocity in large magnification which illustrates the tendency of the movement of the edges.

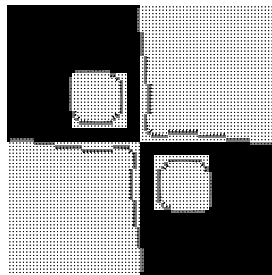
If bias is indeed the main cause of the illusion we should be able to counteract the effect by introducing additional elements. This has been pursued in Figure 5.7A. The additional white and black squares put in the corners of the tiles remove the illusory effect. Figure 5.7B shows a magnified part of the pattern and Figure 5.7C shows the edges detected. As the latter figure shows, the inserted squares partly compensate for the drifting of the edges along the mortar line separating tiles of the same gray value. As a result slightly wavy edges are obtained. The "waviness" is, however, too weak to be perceived (low amplitude, high frequency) and as a result a straight line without tilt is



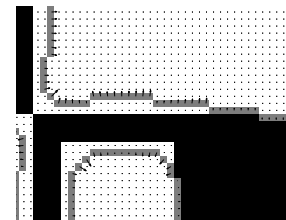
(A)



(B)

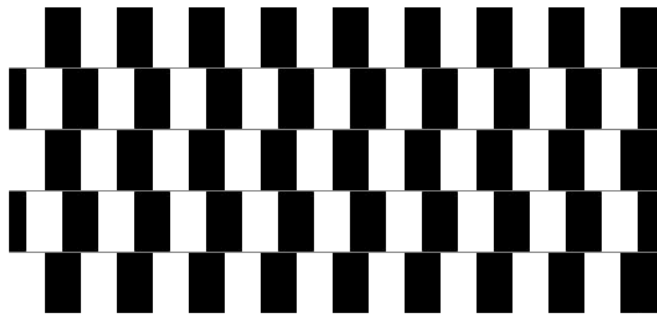


(C)

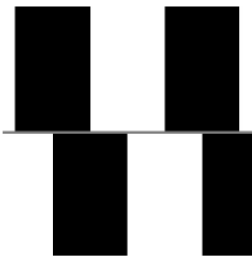


(D)

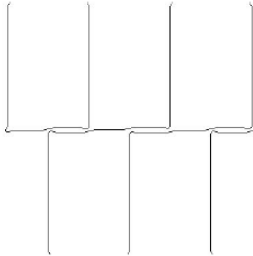
Figure 5.5: (A) Illusory pattern: “waves” [2]. (B) The result of smoothing and edge detection on a part of the pattern. (C) and (D) The drift velocity at edges in the smoothed image logarithmically scaled for parts of the pattern.



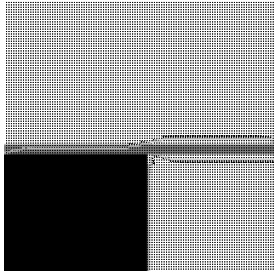
(A)



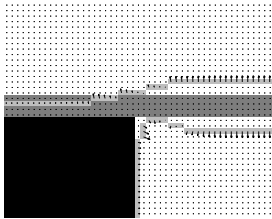
(B)



(C)



(D)



(E)

Figure 5.6: (A) Café wall illusion. (B) Small part of the figure. (C) Result of smoothing and edge detection. (D) and (E) Zoom-ins on the drift velocity.

seen.

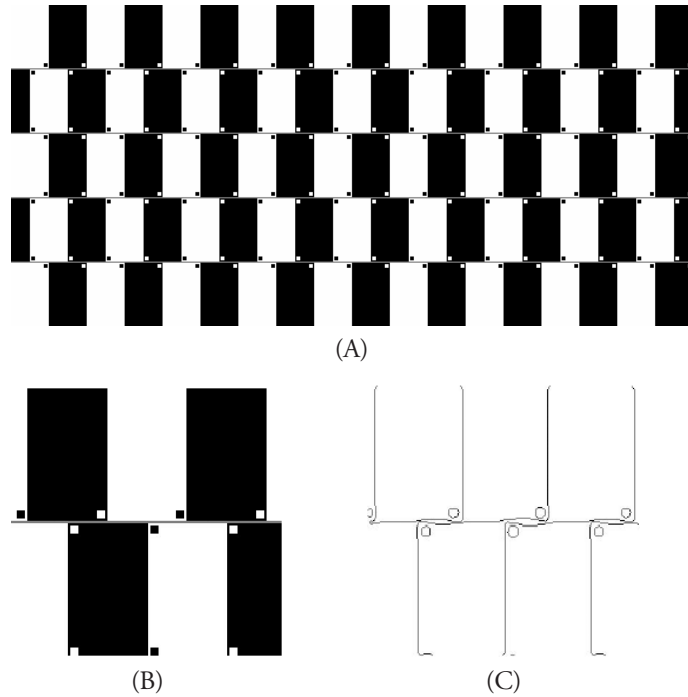


Figure 5.7: Modified café wall pattern. The additional black and white squares change the edges in the filtered image, which counteracts the illusory effect.

A full account of the perception of lines in the above illusions requires additional explanation. The lines are derived in two (or more) processing stages. In the first stage, local edge elements are computed and bias explains the tilting of these elements. The second stage consists of the integration of these local edge elements into longer lines. Our hypothesis is that this integration uses as input the positions and orientations of the short edge elements to compute an approximation of the longer curves. If the linking of edge elements is carried out in this way tilted lines will be computed in the café wall illusion and wavy curves will be derived in the pattern "waves" in Figure 5.4. It is not important to our explanation how the approximation is carried out exactly, be it linear estimation or through some more elaborate nonlinear process. Most probably it is carried out in conjunction with other perceptual grouping processes [49, 50]. The essential thing is that information about the orientation of edgels is used. Even if there are gaps between consecutive edgels the final perception of the pattern will be that of a continuous line or curve. This can be explained by the fact that the human vision system is very good at

perceiving continuity and detecting discontinuity. It can, for example, easily discriminate between smooth curves and jagged discontinuous curves.

Our account of the integration process also explains one of the most forceful of all illusions, Frasers twisted cord illusion [39], also known as the Fraser spiral, and related Fraser figures. The Fraser spiral pattern, cf. Figure 5.8, consists of circles made of black and white elements which together form something similar to a twisted cord, on a checked background. The twisted cords give the perception of being spiral-shaped, rather than a set of circles. The principle of the pattern is easily explained using another Fraser illusion shown in Figure 5.9. Figure 5.9C shows the basic units of all Frasers figures, a short line with a triangle at either end. This appears in both black and white. In the Fraser illusion in Figure 5.9, the short lines are all tilted with the same slope. The edges perceived from these short lines are consequently tilted in the same way and the “weights” in form of the triangles at the ends increases the perception of the slope. From these edge elements an extended tilted line is perceived due to the integration process. In the Fraser spiral the line elements are sections of spirals and an extended spiral is perceived. The illusory effect is also obtained for a twisted cord on a gray background, cf. Figure 5.9B, but it is slightly weaker than with the pattern of the background included. In the perception of each basic element in 5.9C, the visual system probably tries to fit a line to the perceived input data. The triangles then work as weights in this fitting process to increase the slope of the short lines.

5.1.3 A Comment on Smoothing and Line Intersection

In [128], the perceptual effect of a line intersecting a slightly broader line is illustrated as in Figure 5.10. The fine line, as shown in A, appears to be bent in the vicinity of the broader black line, as indicated in exaggeration in B. The effect of smoothing or, equivalently, uncertainty in the positional data, is shown in Figure 5.11. Here, the image in 5.11A is smoothed and the gray value maxima on the intersecting line is indicated in 5.11B and magnified in Figure 5.11C. The resemblance to Figure 5.10B is striking. A more detailed analysis of the behaviour of intersecting lines is the topic of the next section.

5.2 Bias in the Estimation of Line Elements

There is a large group of illusions in which lines intersecting at acute angles are a decisive factor in the illusion. To analyze these line drawings, a slightly different noise model is adopted in this section, with the noise being defined directly on the edge elements. Noise in gray level values results in noise in the estimated edge elements. Another physiologically plausible explanation is that noise in the edge elements is due to quantization errors of the cells analyzing orientation [63].

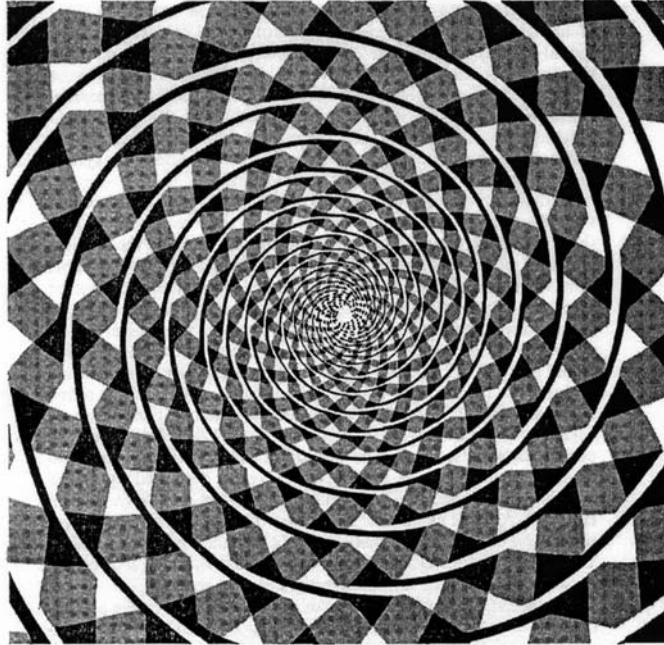


Figure 5.8: Fraser's spiral [39].

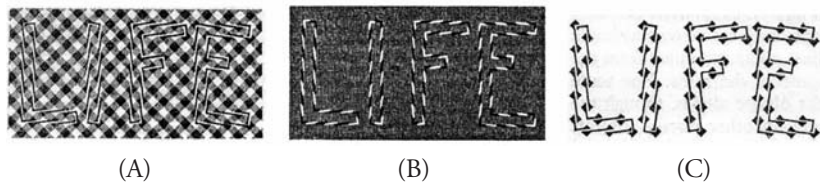


Figure 5.9: (A) Fraser figure with straight twisted cords. (B) The illusion is weaker with a gray background. (C) The basic elements of the figure are tilted elements with a triangle at either end [110].

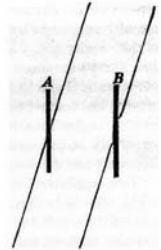


Figure 5.10: The fine line as shown in (A) appears to be bent in the vicinity of the broader black line, as indicated in exaggeration in (B), [128].

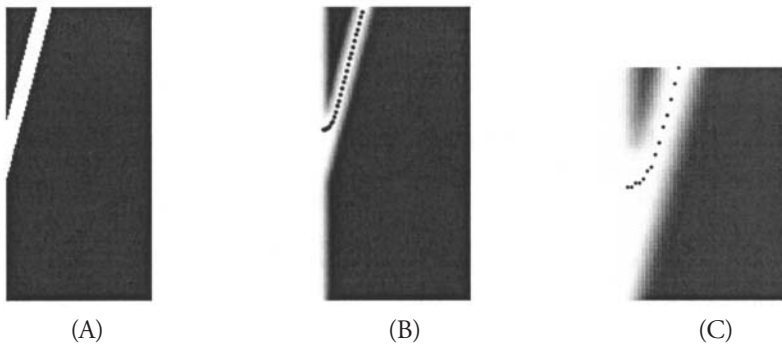


Figure 5.11: (A) A line intersecting a bar at an angle of fifteen degrees. (B) The image has been smoothed and the maxima of the gray level function have been detected and marked with stars. (C) Magnification of intersection area.

5.2.1 Basic Theory

Next the estimated position of an intersection of straight lines is analyzed. The inputs are edge elements, parameterized by the image gradient (I_x, I_y) , which coincides with the normal vector to the edge, and the position of the center of the edge element (x_0, y_0) . The edge elements are influenced by noise, cf. Figure 5.12A. To obtain the intersection of straight lines, imagine a line through every edge element and compute the point closest to all the lines, Figure 5.12B.

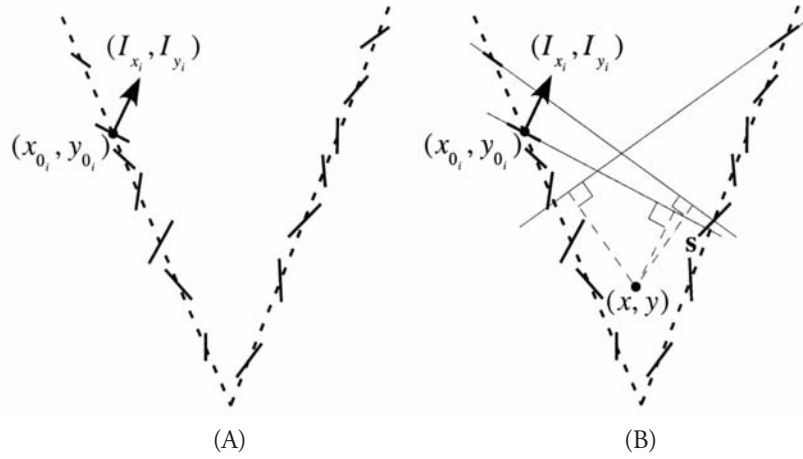


Figure 5.12: (A) The inputs are edge elements parameterized by the position of their centers (x_0, y_0) and the image gradient (I_x, I_y) . (B) The intersection of straight lines is estimated as the point closest to all the “imaginary” lines passing through the edge elements.

In algebraic terms, consider additive, independently identically distributed (i.i.d.) zero mean noise in the parameters of the edgels. In the following unprimed letters are used to denote estimates, primed letters to denote actual values and δ 's to denote errors. Hence, $I_x = I'_x + \delta I_x$, $I_y = I'_y + \delta I_y$, $x_0 = x'_0 + \delta x_0$ and $y_0 = y'_0 + \delta y_0$.

For every point (x, y) on the lines the following equation holds,

$$I'_x x + I'_y y = I'_x x'_0 + I'_y y'_0 . \quad (5.10)$$

This is fulfilled only approximately by the measurements. Let n be the number of edge elements. Each edgel measurement i defines a line given by the equation

$$I_{x_i} x + I_{y_i} y = I_{x_i} x_{0_i} + I_{y_i} y_{0_i} . \quad (5.11)$$

From the collected set of edge elements a system of equations is obtained which can be

represented in matrix form as

$$I_s \mathbf{x} = \mathbf{C} , \quad (5.12)$$

where I_s is the $n \times 2$ matrix incorporating the data in I_{x_i} and I_{y_i} from n number of edgels and \mathbf{C} is the n dimensional vector with components $I_{x_i} x_{0_i} + I_{y_i} y_{0_i}$. The vector $\mathbf{x} = (x, y)$ denotes the resulting intersection point of the lines. Using standard least squares (LS) estimation, the estimated intersection point is given by

$$\mathbf{x} = (I_s^T I_s)^{-1} I_s^T \mathbf{C} . \quad (5.13)$$

It is well known, cf. [40], that the LS solution to a linear system of the form $A\mathbf{x} = \mathbf{b}$ is biased, when errors are present in the measurement matrix A . Also the statistics of the estimation in the case of i.i.d. noise in both A and \mathbf{b} have been studied [47]. In our case, \mathbf{b} is the product of elements in A and two other noisy terms and the statistics are somewhat different.

To simplify the analysis, the variances of the noise in the spatial derivatives in the x and y directions are assumed to be the same. Denote this variance by σ_s^2 . Also, the expected values of higher order terms are assumed to be negligible. In Appendix A, the expected value of \mathbf{x} is found by expressing (5.13) using a second order Taylor expansion at zero noise. It converges in probability to

$$E(\mathbf{x}) = \mathbf{x}' + nM'^{-1}(\bar{\mathbf{x}}_0 - \mathbf{x}')\sigma_s^2 , \quad (5.14)$$

where

$$M' = I_s^T I_s = \begin{bmatrix} \sum_{i=1}^n I'_{x_i}{}^2 & \sum_{i=1}^n I'_{x_i} I'_{y_i} \\ \sum_{i=1}^n I'_{x_i} I'_{y_i} & \sum_{i=1}^n I'_{y_i}{}^2 \end{bmatrix} , \quad (5.15)$$

and $\bar{\mathbf{x}}_0$ is the mean of \mathbf{x}_{0_i} , i.e.

$$\bar{\mathbf{x}}_0 = \begin{bmatrix} \frac{1}{n} \sum_{i=1}^n x_{0_i} \\ \frac{1}{n} \sum_{i=1}^n y_{0_i} \end{bmatrix} . \quad (5.16)$$

Further, \mathbf{x}' is the actual intersection point and n is the number of edge elements. This number has no effect since the magnitude of the elements in M'^{-1} are proportional to $1/n$.

Using (5.14) allows for an interpretation of the bias. The estimated intersection point is shifted by a term which is proportional to the product of matrix M'^{-1} and the difference vector $(\bar{\mathbf{x}}_0 - \mathbf{x}')$. The vector $(\bar{\mathbf{x}}_0 - \mathbf{x}')$ extends from the actual intersection point to the mean position of the edge elements. Thus, it is the mass center of the edgels that determines this vector. The matrix M' , which depends only on the spatial gradient distribution, is a real symmetric matrix and thus its eigenvectors are orthogonal to each other.

The matrix M'^{-1} has the same eigenvectors as M' and inverse eigenvalues. The direction of the eigenvector corresponding to the larger eigenvalue of M'^{-1} is dominated by the normal to the major orientation of the image gradients and thus the product of M'^{-1} with $(\bar{\mathbf{x}}_0 - \mathbf{x}')$ is most strongly influenced by this orientation. Consider the case of two intersecting lines and thus two gradient directions. The effect of M'^{-1} is larger bias in the direction of fewer image gradients and smaller bias in the direction of more gradients. This leads to larger displacement of the intersection point in the direction perpendicular to the line with fewer edge elements.

5.2.2 Illustrative Examples

The best known illusions due to intersecting lines are the Poggendorff and Zöllner illusions. A version of the Poggendorff illusion, as described by Zöllner [146], is displayed in Figure 5.13A. The upper left portion of the interrupted, tilted straight line does not appear to be the continuation of the lower portion on the right. The upper portion seems to be positioned too high compared to the lower portion. Another version of this illusion is shown in Figure 5.13B. Here, it appears that the middle part of the interrupted line is not in the same direction as the two outer parts, but is turned clockwise with respect to the them.

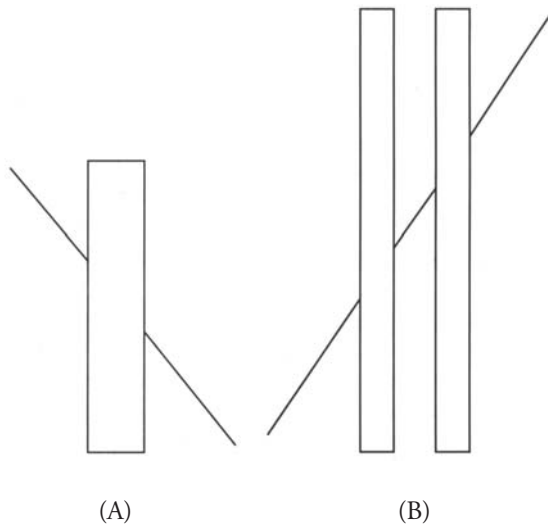


Figure 5.13: The Poggendorff illusion.

Famous vision scientists such as Hering [56] and Helmholtz [128] saw the Poggendorff and related illusions as the result of the perceptual enlargement of acute angles. It

has been found in extensive studies that acute angles are overestimated and obtuse angles are slightly underestimated, although regarding the latter there has been some controversy. The effect is larger with short line segments and seems to diminish with long intersecting lines. Our model predicts that these phenomena are due to the bias in the estimation of the intersection point.

Referring to Figure 5.13A, the intersection point of the upper left portion of the tilted line with the left vertical edge seems to be moved up and the intersection point of the lower portion of the tilted line with the right vertical edge seems to be moved down. From parametric studies it is known that the illusory effect decreases with an increase in the acute angle and reaches zero at 90 degrees. Our model predicts this. Figures 5.14B and C plots the estimated bias in the x and y directions as a function of the angle between the line elements. In these plots, the data is a tilted line which makes an angle ϕ with a vertical line, as shown in Figure 5.14A. As the vertical line is twice as long as the tilted line, the number of edge elements on the vertical line is twice as many.

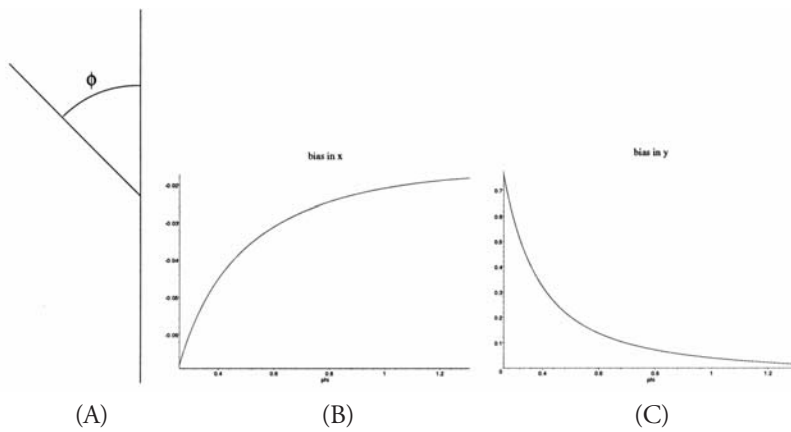


Figure 5.14: (A) The tilted line intersects the vertical line in the middle at an angle ϕ . There are twice as many edge elements on the vertical as on the tilted line. (B) Bias perpendicular to the vertical. (C) Bias parallel to the vertical.

Figure 5.15 shows two variants of the Zöllner illusion. The vertical bands in Figure 5.15A and the diagonal bands in Figure 5.15B are parallel, but they look convergent and divergent. In these patterns the biases in the estimation of the intersection points of the long lines (or the edges of the broader bands in A) with the shorter line segments cause the edge elements between the intersection points along the longer lines to be tilted, as illustrated in Figure 5.15C. In a second computational step, long lines are computed as an approximation to the small edge elements. If, as discussed in the Section 5.1.2 on this integration process, the positions and orientations of the edge elements are used in

the approximation, tilted line or bars will be computed sloping in the same direction in the same direction as perceived by the visual system. Figure 5.16 shows the estimation of the tilted line elements for a pattern such as in Figure 5.15A with an angle of 45 degrees between the vertical and the tilted bars. As input to the estimation we used edge elements uniformly distributed on the vertical and on the tilted lines. Just as for the Poggendorff illusion, it has experimentally been found that the effect increases with an increase in the acute angle between the main line and the obliques, in accordance with Figures 5.14B and C. For really small angles, however, some counteracting effect seems to take place.

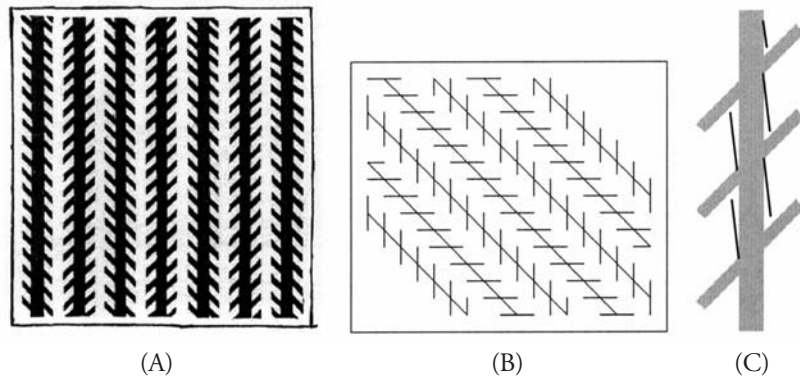


Figure 5.15: (A), (B) Zöllner patterns. (C) The bias in the intersection points of the edges causes the line elements between intersection points to be tilted.

Many other well-known illusions can be explained on the basis of biased line intersections. Examples are the Orbison figures and the patterns of Wundt, Hering and Luckiesh, cf. Figure 5.17. In these patterns geometrical objects such as straight lines, circles, triangles or squares are superimposed on differently tilted lines, which gives a distorted appearance of the objects. This distortion can be explained by erroneous estimation of the tilt in the line elements between intersection points and the subsequent fitting of curves to these line segments.

5.3 Bias in Motion Estimation

When processing image sequences some representation of image motion must be derived as a first stage. It is believed that the human visual system has the ability to compute the optical flow field from a sequence of images, cf. [134]. As mentioned in Section 2.1.3 optical flow is derived in a two stage process. First, using the optical flow constraint equation (2.10), the normal flow is calculated, i.e. the component of the optical flow field in the direction of the gradient in each point (the velocity component perpendicular to edges).

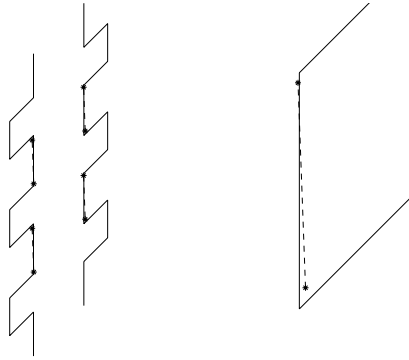


Figure 5.16: Estimation of edges in Zöllner pattern. The line elements are found by connecting two consecutive intersection points, resulting from the intersection of edges of two consecutive bars with the edge of the vertical bar (one in an obtuse and one in an acute angle).

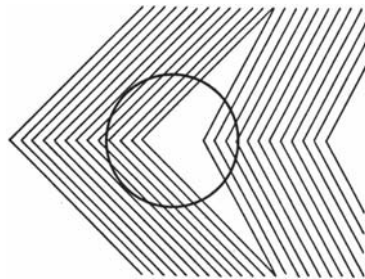


Figure 5.17: A pattern by Luckiesh [110]. The circle seems to be bumped on the left side.

In a second stage the optical flow is estimated by combining, in a small neighborhood of each image point, normal flow measurements from features in different directions. It will be shown that this estimate is biased.

To derive the optical flow from the normal flow measurements we will use the assumption of Lucas and Kanade [75], that the optical flow is approximately constant within a small neighborhood of each image point. Each of the n measurements in the neighborhood provides an equation of the form (2.10) and thus the following overdetermined system of equations is obtained,

$$I_s \mathbf{u} + \mathbf{I}_t = 0 \quad , \quad (5.17)$$

where I_s denotes, as before, the matrix of spatial gradients (I_{x_i}, I_{y_i}) . The vector of temporal derivatives is denoted by \mathbf{I}_t and $\mathbf{u} = (u, v)$ denotes the optical flow. The least squares solution to 5.17 is given by

$$\mathbf{u} = -(I_s^T I_s)^{-1} I_s^T \mathbf{I}_t \quad . \quad (5.18)$$

As noise model we consider zero mean i.i.d. noise in the spatial and temporal derivatives. As in Section 5.2.1, equal variance σ_s^2 for the noise in the spatial derivatives in the x and y directions is assumed. Also, it is assumed that terms with higher order than 2 can be ignored.

The statistics of (5.18) is well understood, as these are classical linear equations. The expected value of the flow, using a second order Taylor expansion, is derived in Appendix A. The expected value converges to

$$E(\mathbf{u}) = \mathbf{u}' - n\sigma_s^2 M'^{-1} \mathbf{u}' \quad , \quad (5.19)$$

where, as before, the actual values are denoted by primes.

Equation (5.19) is very similar to (5.14) and the interpretation given there applies here as well. It shows that the bias depends on the gradient distribution, i.e. the texture, in the neighborhood of the current image point. The estimate of the flow is always underestimated in length and its direction is biased towards the orientation of the majority of the gradients.

5.3.1 Illustrative examples

Figure 5.18 shows a variant of a pattern created by Hajime Ouchi [101]. The pattern consists of two rectangular checkerboard patterns oriented in orthogonal directions, a background orientation surrounding an inner ring with perpendicular orientation. Small retinal motions, or slight movements of the paper, cause an apparent motion of the inset relative to the background.

The tiles used to make up the pattern are longer in one direction than in the other. This leads to a gradient distribution in a small neighborhood with many more normal

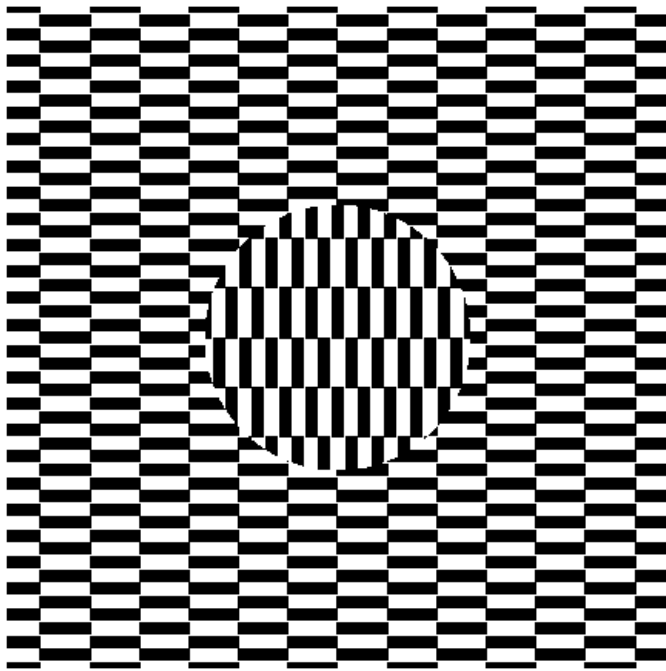


Figure 5.18: A pattern similar to one by Ouchi [101].

flow measurements in the direction perpendicular to the longer edge than in the direction of the shorter edge. Since the tiles in the two regions of the figure have different orientations, the biases in the regional optical flow estimations are different for the two regions. The difference between the bias in the inset and the bias in the background is interpreted as motion of the ring. An illustration is given in Figure 5.19 for the case of motion along the the first meridian (to the right and up). In addition to computing the flow, the visual system also performs motion segmentation, which is why a clear relative motion of the inset is seen. For a detailed analysis of this illusion, cf. [37].

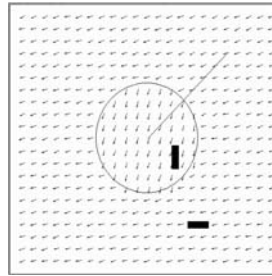


Figure 5.19: The regional motion error vector field. The vectors shown are the differences between the true motion and the calculated motion. To derive the sliding motion, compute the difference between the error in the inset and the error in the surround, and project the resulting vector on the dominant gradient direction in the inset. The line from the center is the direction of the true motion.

Another impressive illusory pattern is shown in Figure 5.20, cf. [106]. If fixating on the center and moving the page along the optical axis, back and forth, the inner circle appears to rotate. The rotation is clockwise if the figure is moved away from the eyes. If the paper is moved closer to the eyes, the optical flow vectors are directed along lines through the image center pointing away from the center. The normal flow vectors are perpendicular to the edges of the parallelograms in the rings. Thus the estimated flow vectors are biased in clockwise direction in the outer ring and in counterclockwise direction in the inner ring, as shown in Figure 5.21. The difference between the inner and outer vectors (along a line through the center) is tangential to the circles which explains the perceived rotational movement. In a similar way, one can explain the perception of the rings moving closer to each other when rotating the pattern around an axis through the center and perpendicular to the paper.

There of course exist other methods besides the Lucas-Kanade approach applied here for calculating optical flow. Besides gradient based models there are frequency domain and correlation based models [6]. But computationally they are not very different. In all models there is a stage in which smoothness assumptions are made and measurements within a neighborhood of the current point is combined to estimate the flow at this point.

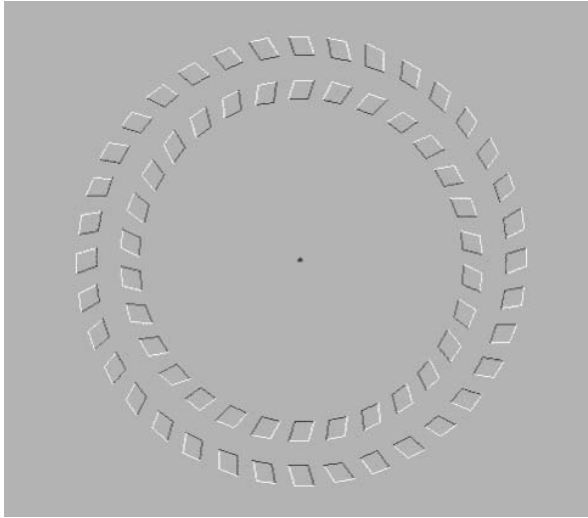


Figure 5.20: If fixating on the center and moving the paper along the line of sight, the inner circle appears to rotate [106].

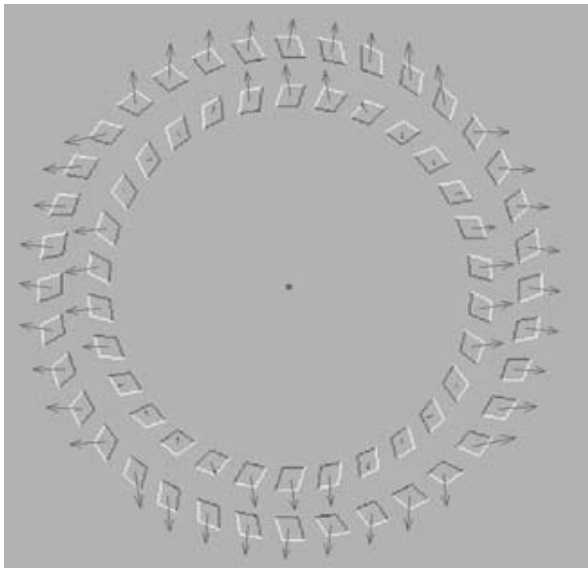


Figure 5.21: Flow field computed from motion of Figure 5.20.

At this stage, statistical difficulties occur and noisy estimates lead to bias. See [38], for an extensive study of the statistics of optical flow estimation.

5.4 Other Theories of Illusion

To put this work into perspective a short overview of previous works on geometrical optical illusions will here be given. Theories about illusions have been formulated ever since their discovery. Many of the theories are aimed only at one specific illusion and most of the early works would be seen nowadays as adding little to a mere description of the illusion.

Theories which attempt to explain a broad spectrum of illusions and which have attracted more notice in recent years are based on a specific sort of mechanism which has been suggested to explain the workings of human visual perception. These mechanisms are either hypothetical, based on physical analogies or general observations about the physiology and architecture of the brain. In contrast, the theory proposed here is of a mathematical nature, based on the geometry and statistics of capturing light rays and computing their properties, such as the gray values and spatiotemporal derivatives. Thus, it applies to any vision system, artificial or biological. However, one might find that our theory resembles some existing theories if they are put in a statistical framework. Chiang [16] proposed a retinal theory that applies to patterns in which lines running close together affect one another. He likened the light coming into the eye from two lines to the light falling onto a screen from two slit sources in the study of diffraction patterns. His explanation for this was the blurring and diffusing effect of the medium and the construction of the eye. The perceived locations of the lines are at the peaks of their distributions. Thus, two closely situated lines influence each others locations and become one when the sum of their distributions forms a single peak. According to Chiang, this leads to an overestimation of acute angles and provides an explanation of the Poggendorff, Zöllner and related illusions as well as the effect in the Müller-Lyer illusion.

Glass [45] also discussed optical blurring as a cause for the perceptual enlargement of acute angles as it fills in the angle at line intersections. He proposed this as an explanation of the Zöllner and Poggendorff effect. Ginsburg argued that neural blurring occurs because of lateral inhibition, i.e. the tendency of adjacent neural elements of the visual system to inhibit each other. This has the effect of spatial frequency filtering. In [43] he suggested that low frequency attenuation contributes to the formation of the illusory Kanizsa triangle, cf. the cover of [86] for a variation of this illusion. In [44], he discussed filtering processes as a cause for the Müller-Lyer illusion.

In a number of recent studies band-pass filtering has been proposed as cause of geometrical optical illusions. Morgan and Moulden [95] and Earle and Maskell [26] discuss linear band-pass filters as a cause of the café wall illusion. In [95], it is showed that a pattern like twisted cords, as in the Fraser illusions, is revealed in the mortar lines if the café wall figure is processed with a spatial band-pass filter. The cords then consists of

maxima and minima in the filtered image. We refer to spatial smoothing followed by edge detection and computationally these two explanations are rather similar. However, we argue that this is a special case of general principle of uncertainty and noise in visual processes which causes bias in the estimation of features, in this case edges.

Morgan and others [94, 92] have also, in the case of the twisted cord phenomenon, commented on the process of integrating the slope of the small lines in the twisted cords to an extended tilted line. In computational terms, these discussions are specific implementations of approximation processes using as input position and orientation information. In particular, they discuss the use of neural collector units [95, 94] and Gabor patches [112, 92].

Morgan and Casco [93] discuss band-pass filtering in conjunction with second order stage filters that extract features in the Zöllner and Judd illusion. In [9], Bulatov et al. propose an elaborate neurophysiological model and the Müller-Lyer and the Oppel-Kundt illusions are discussed. Morgan [91] suggests blurring in second stage filters as the main cause of the Poggendorff illusion.

The diffraction in Chiang's model and the optical and neural blurring in the other mentioned models amounts to an uncertainty in the location of the perceived gray values. They can also be interpreted as noise occurring somewhere in the image formation of the image processing. Thus, the concept of uncertainty is invoked in vague ways in these studies. The models that have been discussed are, however, very restricted. They apply to particular processes, either on the retina or in the neural elements of the vision system.

There are a number of theories in which eye movements are advanced as an important causative factor in illusions. Our theory proposes that eye movements play an important role since they are a relevant source of noise. Helmholtz suggested in [128] that ocular movements are of importance in some illusions, but he also explained doubt that they could be the main cause, as other illusions do not seem to be influenced by them. Carr [13] proposed that eyes react to accessory lines and as a result pass more freely over unfilled than filled elements. According to this theory, the eyes move more freely over the figure with outgoing fins than over the one with outgoing fins in the Müller-Lyer figure. In the Poggendorff and Zöllner figures deflections and hesitations in the eye movements are associated with the intersections of the long lines with the obliques. Piaget [105] proposed a "law of relative centrations". By centration he refers to a kind of centering of attention which is very much related to fixation. Centration on a part of the field in view is supposed to cause an overestimation of that part relative to the rest of the field. Virsu [126] suggested that the eye movements most readily made are horizontal and vertical rectilinear movements. When viewing lines with other orientations which intersect the vertical and horizontal structures, an eye movement correction must take place and this is supposed to give rise to perceptual distortion.

Other famous works include theories who discuss illusions in the context of figural after-effects [41, 69]. Figural after-effects refers to the tendency of an observer to transfer features of an observed object, such as the slope of a line, to an object observed directly

afterwards. There are also theories based on the assumption that the perceptual system interprets illusory patterns as flat projections of three-dimensional displays, cf. [121, 120, 48].

5.5 Discussion and Conclusions

In this chapter, we have discussed a major hurdle that vision systems have to deal with, noise in the image data. Noise in the gray level and its spatial and temporal derivatives causes a serious problem for early visual processes and leads to bias in the performed estimations. An artifact of the bias is illusory perceptions of images involving patterns where the bias is highly pronounced.

Linear estimation processes, in particular linear least squares, have been considered in this work. The question arises if the estimations could be corrected if more sophisticated techniques were used. If the accurate knowledge of the statistics of the noise were at hand we should be able to compensate for the bias. Inverse filters could be applied to reconstruct the gray level signal and the corrected least squares estimator could be used to remove the asymptotic bias. The method of total least squares estimation, cf. [46, 89, 10], is another alternative. This type of estimation has been shown to provide asymptotic unbiased solutions for the system $A\mathbf{x} = \mathbf{b}$ if both A and \mathbf{b} are corrupted by noise. However, for this method to work well accurate information about the noise variables must be present, e.g. the relative amounts of noise in the spatial and temporal derivatives.

For a specialized vision system working under very restricted conditions, it might be possible to estimate the statistical parameters of the noise and achieve almost unbiased estimations. However, for a system which is supposed to deal with very general and varying visual input data, such as the human visual system, the estimation of the noise characteristics seems hard. Besides the sensor noise, which might stay approximately fixed, factors that influence the noise include e.g. the lighting conditions, the physical properties of the objects being viewed, and the orientation of the viewer in 3D space. In particular, these factors have large influence on the calculation of image derivatives.

An interesting observation is that the illusory perception of some patterns weakens after extended viewing, in particular when subjects are asked to fixate. In this case the noise parameters might be fixed long enough for the visual system to acquire reasonable approximations of these parameters which allows it to improve the estimates.

The bias has been demonstrated for very general mathematical models. We have discussed possible sources of noise but have not specified e.g. the amount of uncertainty originating from eye movements compared to the amount originating from noise in the neural system in specific situations, nor in which ranges the uncertainty lies. These are issues that need to be examined. In particular, using psychophysical and physiological methods it might be feasible to conduct investigations on the amounts of noise and the relation of eye movements to noise. Also, we have only discussed the bias in the estimations, which was sufficient to demonstrate the cause of a large number of illusions and

predict some parametric studies. However, the variance of the estimators should probably also be studied. The larger the sample size, the smaller the variance of the estimator. Thus, this measure predicts slightly more accurate perception of a dense pattern compared to a less dense pattern. When it comes to estimating optical flow, a large variance in the gradient direction is important for accurate results.

There are many illusions which are classified as geometrical optical illusions that we have not taken up for discussion. These include, for example, the before-mentioned contrast effect; the illusion of filled and unfilled extents, i.e. the effect that a filled extent is overestimated when compared to an unfilled extent (an example is the Oppel-Kundt illusion); the Ponzo illusions, cf. Figure 5.22B; the vertical-horizontal illusion, i.e. the phenomenon of judging a vertical line to be longer than a horizontal line of the same extent; the framing effect, i.e. the overestimation of framed objects and the Delboeuf circles, cf. Figure 5.22A. Size parameters are of great importance in some of these illusions. For example, in the Delboeuf illusion the size of the outer circle perceptually decreases and the inner one increases for a diameter ratio up to about 2:1 (with a maximum at about 3:2). For larger ratio the smaller circle changes from being overestimated to being underestimated.

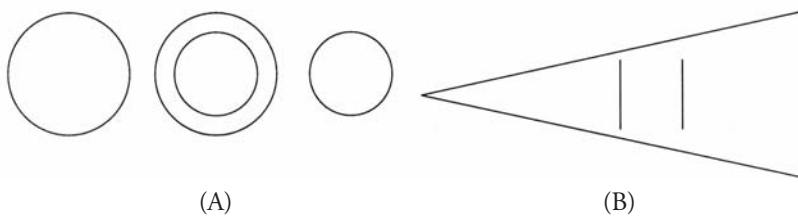


Figure 5.22: (A) Delboeuf circles. (B) A version of the Ponzo illusion.

The illusions studied in this chapter have been explained on the basis of rather small amounts of noise. These amounts of noise do not seem to be a major factor behind the illusions mentioned here. However, larger amounts of smoothing has the effect of neighboring parts in a figure to influence each other and as a result they are moved closer to each other. Consider the Delboeuf circles in Figure 5.22A. Small amounts of smoothing will cause both circles in the middle figure to contract. If the amount of noise is increased so that the two circles affect each other under smoothing, then the outer circle will contract and the inner circle will expand. Image formation processes should not give rise to such large amounts of noise to account for these large amounts of smoothing. However, a possible explanation could be a scale-space architecture in the human visual system, employing multiple levels of image resolution, cf. Section 6.1. There is evidence, cf. [139], that there are hierarchies of neurons of increasingly large receptive fields within our visual system, with neurons of smaller receptive fields providing information to neurons of larger receptive fields, and vice versa. Also, higher level processes such as segmentation

and recognition needs information from large parts of the image. This leads us to believe that there must exist some processes that combine information from local neighborhoods into representations of global information, and these processes might carry uncertainty. For example, to interpret one of the illusory figures in Figure 5.22 requires analysis of the whole pattern, as judgments of the sizes of different parts of the pattern are made.

To conclude, the findings of this research are also important for practical applications. The proposed theory is not only applicable to the human visual system, but to any visual system, natural or artificial. It points out the importance of considering the noise and uncertainty in the input data when constructing vision algorithms. This work has primarily discussed uncertainty in processes such as feature extraction and optical flow estimations, but since many other computations in visual systems are estimation processes, it is of general importance in vision research.

Part III

Nonlinear Diffusion in Image Processing

Chapter 6

Nonlinear Diffusion Filtering of Images Obtained by Planar Laser-induced Fluorescence Spectroscopy

During the last decade increasing attention has been given to the use of partial differential equations (PDE's) in image processing and computer vision. A large number of interesting applications has been reported, including e.g. 3D scene reconstruction [30], image restoration [7], and numerous medical segmentation techniques. Two large subgroups that have emerged within this field of research are the theory of level set methods [114] and scale-space theory [72, 131].

In this chapter, an application of nonlinear diffusion equations will be studied. Nonlinear diffusion filtering in image processing was pioneered by Perona and Malik [103] and further developed by e.g. Weickert [130] and is closely related to scale-space theory. The objective of the current application is to find flame contours in noisy images of turbulent flames. The imaging technique used is so-called planar laser-induced fluorescence (PLIF) spectroscopy which has proven to be one of the most valuable tools in the study of the physics and chemistry of turbulent reactive flows [53]. Nonlinear diffusion filtering is applied to the PLIF images to make the flame boundaries more apparent which simplifies preceding computations such as calculation of the flame contour length.

In the next section, nonlinear diffusion filtering and its relation to scale-space theory is introduced. The particular diffusion equation used in the current application and the forthcoming analysis is also given. This section also includes a simple scheme for the numerical implementation. In Section 6.2, planar laser-induced fluorescence spectroscopy is presented and motivation is given for the processing of the acquired images. A thorough analysis of the properties of the nonlinear diffusion filtering technique is then presented in Section 6.3, followed by particular aspects concerning the technique in the present application in Section 6.4. In Section 6.5, the influence of the filtering technique on a preceding flame front identification is analyzed. Finally, Section 6.6 summarizes the presentation and the advantages of and the problems with the technique.

6.1 Nonlinear Diffusion and Scale-Space Theory

In the following presentation an image will be defined as a function u on \mathbb{R}^2 , where $u(\mathbf{x})$ represents the gray level or intensity at point $\mathbf{x} = (x, y)$. A very common way to low-pass filter (or smooth) an image to reduce the influence of noise is to convolve it with a

Gaussian kernel $G_\sigma(\mathbf{x})$:

$$G_\sigma(\mathbf{x}) := \frac{1}{2\pi\sigma^2} e^{-\frac{|\mathbf{x}|^2}{2\sigma^2}} , \quad (6.1)$$

where σ determines the standard deviation (or width) of the kernel. It is a classical fact that the solution of the linear diffusion equation

$$\partial_t u = \operatorname{div}(d\nabla u) , \quad (6.2)$$

where d is a scalar constant, for time t is exactly the same operation as convolving the image u with a Gaussian kernel of width $\sqrt{2t}$.

The purpose of the so-called scale-space theory is to create a multi-scale representation of an image. A scale-space method can be defined as a family of filters $(T_t)_{t \geq 0}$ which when applied to the original image $u_0(x)$ yields a sequence of images $u(t, \mathbf{x}) = (T_t u_0)(\mathbf{x})$. The image $(T_t u_0)(\mathbf{x})$ represents an approximation of $u_0(\mathbf{x})$ where a neighborhood around \mathbf{x} , of a size which depends on t , has been scanned to determine the value of $(T_t u_0)(\mathbf{x})$. In this way one could argue that the image is simplified and that the information in the image is reduced for larger t , and the scale is said to be coarser. This kind of filtering can e.g. be implemented by convolution with a Gaussian kernel (6.1) with increasing width or, equivalently, by solving a linear diffusion equation (6.2) on the image. The Gaussian scale-space theory in image processing was pioneered by Witkin [137] in the beginning of the 80's.

For the scale-space to be well-founded and contain relevant and correct information about the objects in the image, it should fulfill some rather natural axioms. These should include some sort of **causality criterion**. This type of criterion guarantees that any feature at a coarse scale has a "cause" at a finer scale. In other words, no spurious detail should be generated and no new local maxima should be created when moving from finer to coarser scales. This can be formulated as a maximum-minimum principle for the scale-space generating equation. Other natural requirements are that the process should be invariant to translations and rotations of the image and constant shifts in gray scale. Koenderink [70] and Lindeberg [72] have by different methodologies argued that the Gaussian scale-space is the only well-founded linear scale-space. The latter reference contains an extensive study of the properties of Gaussian scale-space. For an example of well-founded axiomatic constructions of scale-space theories cf. Alvarez et al. [3].

In the present application we are interested in finding a good approximation of the contour of an object in a noisy image. Smoothing with a Gaussian kernel, or solving a linear diffusion equation on the image, will help to suppress noise, but it will also blur and dislocate edges and contours, cf. Figure 6.1. In [104], Perona and Malik proposed to exchange the scalar diffusion constant d in (6.2) for a scalar valued function g of the length of the gradient ∇u of the gray levels in the image. The resulting diffusion equation then reads as

$$\partial_t u = \operatorname{div}(g(|\nabla u|)\nabla u) . \quad (6.3)$$

The function g will in the following sometimes be referred to as the **diffusivity function**. The length of the gradient, $|\nabla u|$, has been empirically proven to be a good measure of the edge strength at the current location and expresses a dependence on the differential structure of the image. This dependence, of course, makes the diffusion process nonlinear.

Often the term anisotropic diffusion is used synonymously to nonlinear diffusion for the process governed by (6.3). We, however, choose to adopt the formalism used by Weickert [130], that states that the term anisotropic should be reserved for processes where the diffusion scalar d is replaced by a diffusion *tensor* D so that $D\nabla u(\mathbf{x})$ and $\nabla u(\mathbf{x})$ can be made to have different direction.

In the formulation by Perona and Malik, the function g is required to be smooth and non-increasing with $g(0) = 1$ and $g(s) \rightarrow 0$ for $s \rightarrow \infty$. This choice of g can increase the steepness of a gray level edge in an image. Two different equations proposed by Perona and Malik that fulfill the above stated requirements are

$$g(s) = \frac{1}{1 + (s/\lambda)^2} , \quad (6.4)$$

and

$$g(s) = e^{-(s/\lambda)^2} . \quad (6.5)$$

There exists some controversy whether the Perona-Malik equation (6.3) generates a true scale-space. In [104] it is argued that it fulfills a maximum-minimum principle, provided that it possesses twice differentiable solutions. However, this is questionable because of the strong gradient enhancing effect that can be obtained by the choice of diffusivity function g . Further, in some scale-space theories a so-called comparison principle [3] is formulated, requiring that the scale-space is order preserving:

$$T_t(f) \leq T_t(g) \text{ on } \mathbf{R}^2 \text{ for all } t \geq 0 \text{ and all } f, g \text{ such that } f \leq g .$$

This is not satisfied by the edge enhancing Perona-Malik equation. Still, in practical implementations the “scale-space” generated by (6.3) gives a reasonable simplification of the image when moving from finer to coarser scales and the approach is very useful in many applications.

To get insight into the behaviour of (6.3), so-called **gauge coordinates**, denoted η and ξ , will be introduced. Here, η is the normalized direction parallel to the gradient and ξ is the normalized direction orthogonal to the gradient. Using this, the right hand side of (6.3) can be written as

$$\operatorname{div}(g(|\nabla u|)\nabla u) = g(|\nabla u|)u_{\xi\xi} + \Phi'(|\nabla u|)u_{\eta\eta} , \quad (6.6)$$

where $\Phi(|\nabla u|) = |\nabla u|g(|\nabla u|)$ and $u_{\xi\xi}$ and $u_{\eta\eta}$ are the second-order derivatives in the directions ξ and η , respectively. The function Φ will hereafter be referred to as the **flux function** or simply the flux. For a detailed derivation of relation (6.6), cf. [77]. In this

new form it can be seen that the coefficient $\Phi'(|\nabla u|)$ in front of the term $u_{\eta\eta}$ governs diffusion across the edges.

For the processing of the images in the present application, a function originally proposed by Weickert [130] will be used as diffusivity function g . This function was found to be very well suited to the properties of the PLIF images recorded by our camera. Weickert's diffusivity function reads as

$$g(s) = 1 - e^{-\frac{C_m}{(\frac{s}{\lambda})^m}} . \quad (6.7)$$

Here, m is a positive integer, $C_m > 0$ and $\lambda > 0$. The constant C_m is chosen in such a way that the flux $\Phi(s)$ is increasing for $s < \lambda$ and decreasing for $s > \lambda$. That is, $\Phi'(s) > 0$ when $s < \lambda$ and $\Phi'(s) < 0$ when $s > \lambda$. Following this, C_m is uniquely determined by the choice of m . In this way, the diffusion in the direction across the edges will be a forward diffusion when $|\nabla u| < \lambda$ and a backward diffusion when $|\nabla u| > \lambda$, and the slopes at the edges will increase. The parameter λ acts as a contrast parameter that separates regions where forward diffusion occurs from those where backward diffusion takes place.

For a comparison between a linear scale-space generated by (6.2) and a "scale-space" generated by (6.3), using the diffusivity function (6.7), cf. Figure 6.1 and Figure 6.2.

Equation (6.3) is apparently rather unstable and sensitive to noise because of the backwards diffusion properties. This makes the stability of the solutions of the equation strongly dependent on the regularizing effect of the discretization and numerical implementation. Therefore, it seems natural to introduce a controlled regularization directly into the continuous formulation of the process. Catté et al. proposed in [14] to replace the intensity gradient by its smoothed version as parameter to the diffusivity function g , i.e. replace $g(|\nabla u|)$ by the estimate $g(|\nabla(G_\sigma * u)|)$, where G_σ is a Gaussian kernel of width σ . The resulting equation reads as

$$\partial_t u = \operatorname{div}(g(|\nabla(G_\sigma * u)|)\nabla u) . \quad (6.8)$$

The model of Catté et al. has the advantage that it is insensitive to noise at finer scales, determined by σ . Therefore, it does not misinterpret noise as edges which should be enhanced. This makes (6.8) much more stable than (6.3).

6.1.1 Numerical Implementation

Good results can be obtained with rather simple discretizations of (6.8). The discretization used in the examples shown in this paper is based on central finite difference approximations of the spatial derivatives with reflecting boundary conditions. The step sizes Δx and Δy are both set equal to 1 (pixel). For the differentials in the x and y directions,

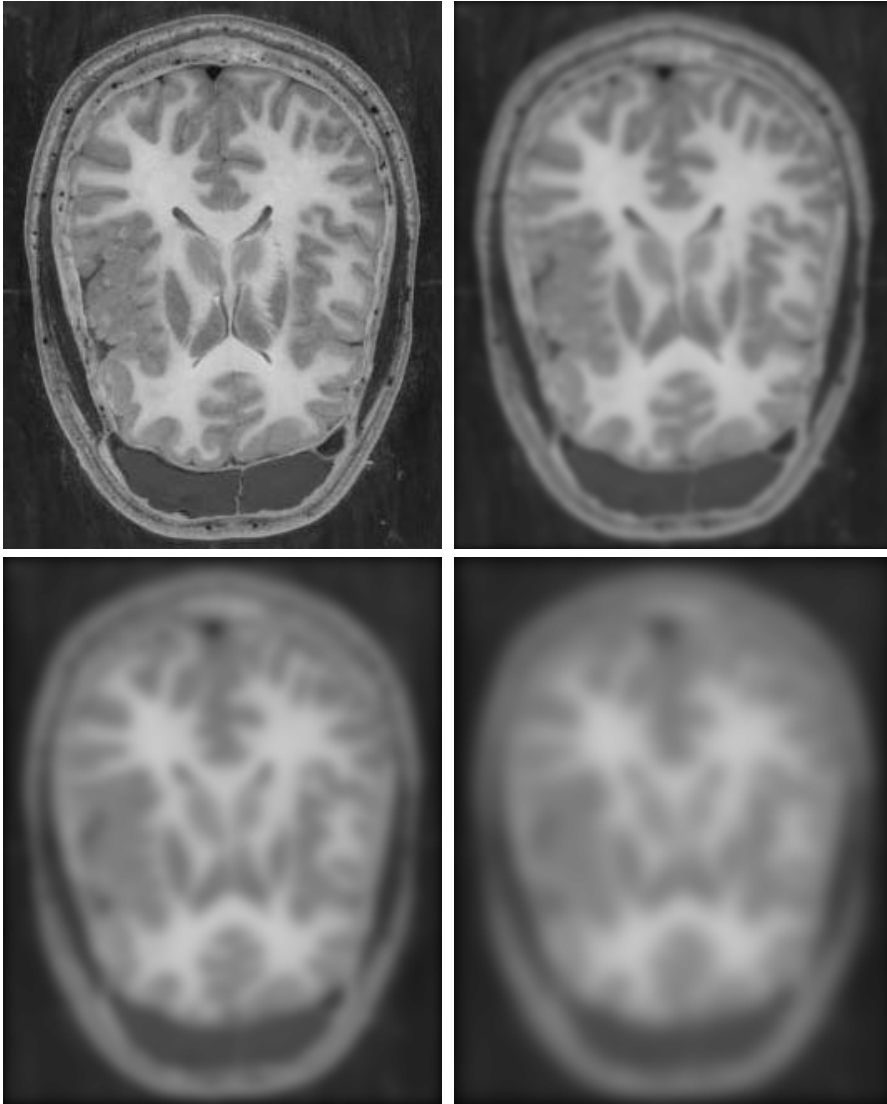


Figure 6.1: Linear diffusion scale-space. From left to right, top to bottom: $t = 0$, $t = 2$, $t = 8$ and $t = 18$.

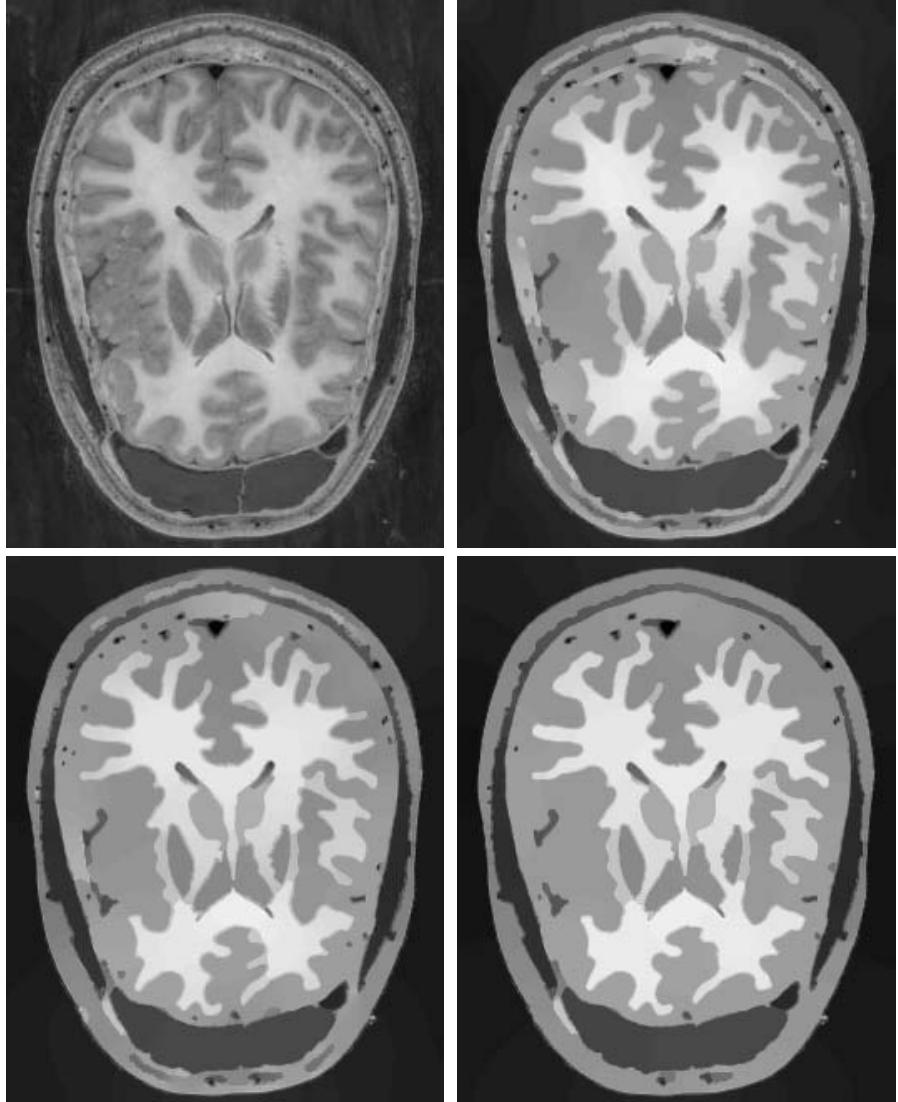


Figure 6.2: Nonlinear diffusion scale-space generated by (6.7), using the parameter values $\lambda = 3$, $\sigma = 1$ and $m = 8$. From left to right, top to bottom: $t = 0$, $t = 60$, $t = 240$ and $t = 960$.

respectively, we have:

$$\partial_x u_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2}, \quad (6.9)$$

$$\partial_y u_{i,j} = \frac{u_{i,j+1} - u_{i,j-1}}{2}. \quad (6.10)$$

The difference calculations can be implemented by multiplication by sparse matrices offering a substantial improvement in computational speed. In this manner, $\partial_x u = uX$ and $\partial_y u = Yu$ where

$$X = \frac{1}{2} \begin{bmatrix} 0 & -1 & 0 & \dots & 0 & 0 \\ 1 & 0 & -1 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & -1 \\ 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix}_{N \times N},$$

$$Y = \frac{1}{2} \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ -1 & 0 & 1 & \dots & 0 & 0 \\ 0 & -1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & -1 & 0 \end{bmatrix}_{M \times M}.$$

Here, N represents the number of the discrete points (pixels) in u along the x -axis and M the number of pixels along y .

To iterate one time step one starts by calculating $|\nabla(G_\sigma * u)|$. Since $\frac{\partial}{\partial x}(G_\sigma * u) = \frac{\partial G_\sigma}{\partial x} * u$, this can be done by smoothing the image with differentiated Gaussian kernels, cf. Section 4.6 and Section 5.1.1. The function $g(|\nabla(G_\sigma * u)|)$ is then evaluated at all discrete points and finally u is updated according to:

$$u^n = u^{n-1} + \Delta t ((g \cdot \partial_x u^{n-1})X + Y(g \cdot \partial_y u^{n-1})). \quad (6.11)$$

Here, $g \cdot \partial_x u^{n-1}$ denotes elementwise multiplication of the matrices $g(|\nabla(G_\sigma * u^{n-1})|)$ and $\partial_x u^{n-1}$.

The time step Δt has to be chosen carefully for the process to be stable. Calculations of an upper bound for stability in similar settings [42] suggest $\Delta t = 0.2$ as an appropriate choice, but simulations using the current discretization have shown that in practice we can receive stable solutions using time steps as large as $\Delta t = 0.8$.

The explicit scheme explained here is of course very basic and because of the restriction in the size of the time step it can be rather time consuming. More advanced numerical methods would most probably reduce the time of computation. Among these, the use of adaptive grid finite elements [51], semi-implicit schemes based on additive operator splitting [132], and parallel implementations [133], should be mentioned.

6.2 Planar Laser-Induced Fluorescence Spectroscopy

The application of nonlinear diffusion filtering presented here concerns reduction of noise and enhancement of contours in images of flames. Some background will here be given on the imaging technique in use, planar laser-induced fluorescence (PLIF) spectroscopy.

For the study of turbulent reactive flows two-dimensional laser spectroscopic imaging has been proven to be very valuable. A variety of spectroscopic techniques are available for two dimensional imaging of relevant scalars in combustive flows: Raman spectroscopy [138], Mie scattering, Rayleigh scattering [100], and laser-induced fluorescence [100, 125, 67] have all been applied in a two dimensional fashion in these environments. These techniques has in common that a suitable high-power laser beam is shaped into a planar sheet that cuts through the measurement volume of interest. The laser light is scattered by particles present in the measurement volume and is captured at right angles by a suitable camera, cf. Figure 6.3. By a suitable choice of laser wavelength and other parameters, one is able to measure the concentrations of chemical species present in the flame environment, the flame temperature, the flow characteristics, etc. By use of high-power laser sources with short pulse lengths (in the order of 10^{-8} s), this information can be obtained practically instantaneously on the chemical and physical time scales prevailing in combustion systems. Such data is valuable to improve the understanding of combustion processes and in the quest to design cleaner, more efficient combustion devices. An overview of spectroscopic applications to combustion diagnostics is given in [27].

Laser induced fluorescence is of particular importance in this context since it is the only technique with the sensitivity required for making concentration measurements of radical species that are produced *in situ* during the combustion process and are only present at trace levels. Of particular importance in this context is the OH radical, since it occurs in all systems in which air is used as an oxidizer. The local OH concentration is a good indicator of the efficiency of the combustion process, since it is sensitively dependent on the local flow field properties [129]. The evolution of OH concentration fields in a flame with shear flow-generated turbulence can be visualized in real time in a film-like manner by the use of planar laser-induced fluorescence [68]. The effects of convective stress on the flame front and local fuel-air mixing on the flame could be tracked in real time by these measurements. The PLIF spectrometer used for this work is unique in terms of temporal and spatial resolution attainable. However, its dynamic range is limited to 8 bits resolution, and in a typical experimental situation the signal-to-noise ratio and contrast-to-noise ratio are therefore limited.

Figure 6.4B shows an example of a low-quality OH PLIF image obtained in the turbulent diffusion flame displayed in Figure 6.4A. It represents a two-dimensional cut through the flame indicated by the rectangular box. For comparisons with model calculations of a flame like this it is necessary to extract information such as contour lengths and areas from such data.

The PLIF image shown in Figure 6.4B is an 8-bit image with a size of 261×241

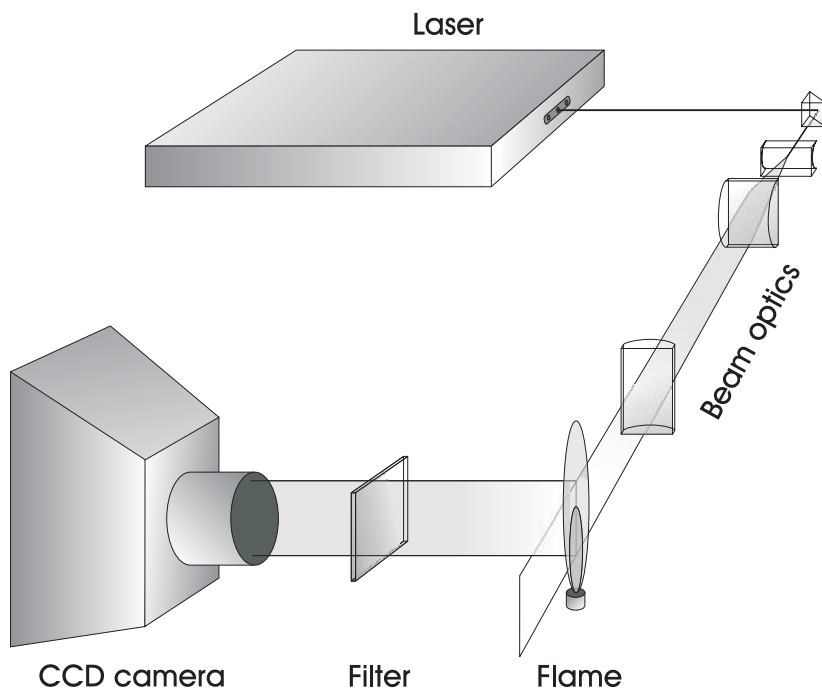


Figure 6.3: Schematic setup of planar laser light-scattering techniques.

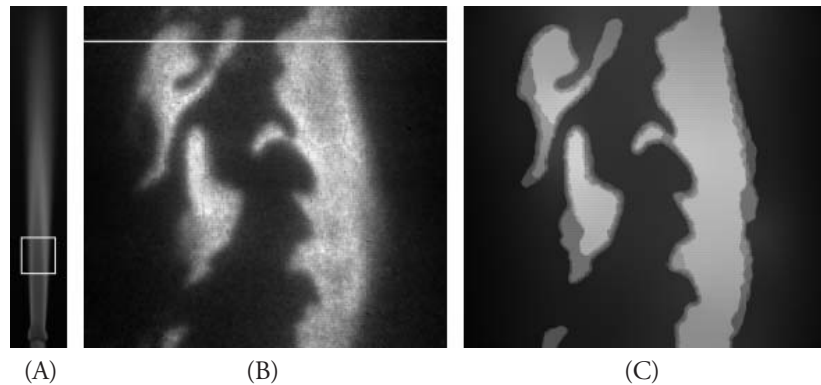


Figure 6.4: (A) Photograph of line-of-sight flame emission from a turbulent non-premixed flame. The area imaged by the PLIF experiments is indicated by the white box, corresponding to 24×23 mm, starting 40 mm above the burner nozzle. (B) Raw PLIF image showing the two-dimensional OH-radical concentrations in a cut through the flame. (C) PLIF image after nonlinear diffusion filtering. The position of the flame front is strongly enhanced.

pixels, corresponding to a region of 24×23 mm in the flame, starting 40 mm above the burner nozzle. The image corresponds to raw data from which a nearly constant background level has been subtracted and which has been compensated for laser beam profile variations. Details of the beam profile compensation scheme are given in [25]. Apart from this the image is unprocessed.

The main goal of the current research is to extract information about the position of the so-called flame front, which is where the chemical reactions in the flame are taking place. As a good approximation, the flame front lies close to the region where the maximum gradient of the OH concentration lies [65].

In the PLIF image the OH concentration boundaries are not very sharp, owing to a limited detector resolution caused by imperfect imaging and cross talk between pixels and also owing to physical phenomena such as convective and diffusive mixing of OH with ambient gases. Since the production terms for OH are highly nonlinear in nature and are strongly temperature dependent, there are also large gray-scale variations within the regions where OH occurs. Also, there are high levels of noise present, which can be attributed to several sources: scattered stray laser light that enters the detector, noise stemming from the signal amplification process in the image intensifiers of the camera, and noise associated with the CCD devices in the camera.

For quantitative comparison with results obtained from theoretical calculations of the combustion system, some form of image processing of the PLIF images for quantitative data extraction is necessary. For this processing nonlinear diffusion filtering is ideal. It

will sharpen the edges while at the same time suppress the noise stemming from the above mentioned sources. Because the length of the flame contour compared to the burned areas is of interest for comparison with theoretical findings, this is a highly desirable feature. As mentioned before simple low-pass filtering with a Gaussian smoothing kernel will remove noisy high frequency components from the images but also increase the blurring of the edges. Standard edge detection schemes based on linear filters, such as the Canny edge detector [11], inevitably suffers from localization inaccuracy when analyzing images at coarser scales where a lot of smoothing is necessary. Also, structure is often lost at meaningful regions such as edge junctions. Since the smoothing and edge detection is performed in one single process in nonlinear diffusion filtering, edges stay localized and preceding corrections such as edge linking and thinning become unnecessary.

6.3 Properties of Nonlinear Diffusion

In this section, some of the properties of nonlinear diffusion filtering will be demonstrated by application to a number of artificially generated one-dimensional test curves. These test curves simulate characteristics expected in real experimental situations using the PLIF system mentioned above. This will give us insight on how to use nonlinear diffusion filtering optimally for the present application.

The basic properties of the process are exhibited in Figure 6.5. It shows the progressive filtering of a blurred, noisy edge, which was generated by a step function and to which white noise was added. The edge intensity varies between 0 and approximately 250, corresponding to the dynamic range of the present camera system. The standard deviation of the noise was 7.5% of the peak intensity of the step function. Weickert's diffusivity function (6.7) was used in equation (6.8), with $m = 8$, $C_m = 3.315$, $\lambda = 15$ and $\sigma = 1$. After only 25 computational iterations the edge is strongly enhanced. At the same time the noise is effectively removed.

To see which edges are preserved during the process, it is useful to study the flux function $\Phi(|\nabla u|) = |\nabla u|g(|\nabla u|)$ from (6.6) and plot Φ against the gradient ∇u . In Figure 6.6 this is shown for the three different diffusivity functions presented in Section 6.1. The two dotted curves corresponds to the two functions (6.4) and (6.5), originally proposed by Perona and Malik [104], whereas the solid line corresponds to (6.7) for $m = 8$. All three curves are plotted for $\lambda = 1$. The principal behavior is the same for the three functions: Diffusion is rising to a maximum at points where $|\nabla u| = 1$, then falling off again where $|\nabla u| \geq 1$. In a given situation one must choose a parameter set for which the flux $\Phi(|\nabla u|)$ obtains maximum for gradient magnitudes exhibited by noisy features and minimum for the edges that must be preserved. Weickert's diffusivity function (6.7), for $m = 8$, is the most selective one of the three, peaked sharply around $|\nabla u| = 1$. The flux function Φ of this function for lower m values is less sharply peaked and more closely resembles the ones proposed by Perona and Malik. In the present case, Weickert's function for $m = 8$ was found to match the noise characteristics of the detector very

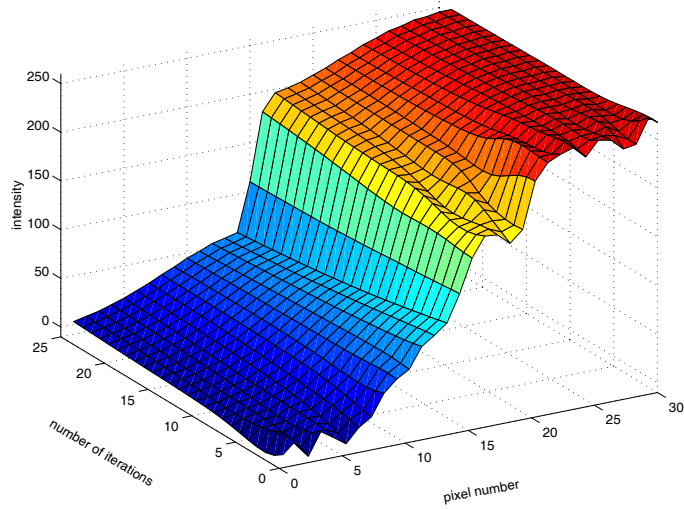


Figure 6.5: Sharpening of an edge by nonlinear diffusion. The plot shows the evolution of a noisy, blurred step function during 25 computational iterations. Clearly the edge is strongly enhanced, while noise fluctuations are effectively smoothed out by the process.

well (the noise spectrum closely resembled a normal distribution in the present case). This means that good results can be achieved after relatively few iterations (typically 100 iterations or less for a given PLIF image, cf. Section 6.4).

A drawback is that the method becomes more sensitive to the proper choice of λ and is numerically less stable than the other two with flux functions shown in Figure 6.6. Much of this instability is prevented by use of the smoothing kernel as in (6.8). In practice we have found that good results are achieved if λ is set to approximately 70% of the minimum gradient magnitudes that we wish to preserve and if a σ within the interval 0.5-1 is used. Figure 6.7 shows the behavior of different terms appearing in (6.8) when a noise free but smoothed edge function is used. This figure is an aid for qualitative understanding of the edge-sharpening process. Figure 6.7A shows the edge function u after n equal to 1, 10 and 100 computational iterations, using a one dimensional version of the scheme outlined in Section 6.1.1. The gradual steepening as n increases is clearly exhibited. In Figure 6.7B, the gradient ∇u is plotted. This shows quantitatively how the edges in u become steeper with time. Finally, Figure 6.7C displays the rate of change of the gradient, $\partial_t \nabla u$. Initially, one can see that the largest rate of change occurs at the point where the gradient ∇u is largest. $\partial_t \nabla u$ is positive here, so the edge increases its sharpness. Simultaneously, there are two negative lobes in $\partial_t \nabla u$ to the left and to the right of the center peak, giving the effect that the slope of u is flattened at these positions, which is

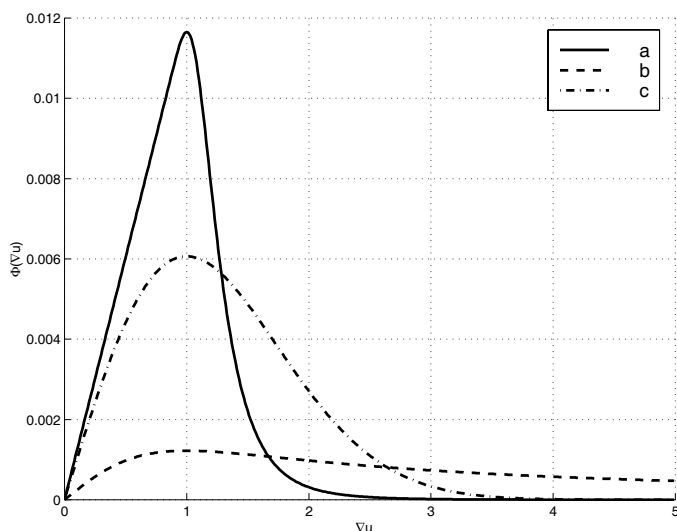


Figure 6.6: Graph displaying the flux function $\Phi(|\nabla u|) = g(|\nabla u|)|\nabla u|$ for three different diffusivity functions g , with $\lambda = 1$. Line (a) corresponds to (6.7) for $m = 8$, while (b) and (c) corresponds to (6.4) and (6.5), respectively.

reflected in Figure 6.7A. At $n = 10$, $\partial_t \nabla u$ has reached its maximum near the center. In contrast, the side lobes are decreasing again in magnitude, indicating that the edge diffusion is slower there than initially. Finally, at $n = 100$, diffusion has virtually come to a standstill over the entire region shown, although a minute amount of steepening is still active near the center of the graph. From this time on nothing much is gained by a continuation of the process. In practice, therefore, one should balance the degree of smoothing and sharpening that one wishes to achieve with the associated computational cost (number iterations required to achieve the task). We found that, using our numerical scheme, $n = 150$ was sufficient for all measurement situations that we encountered. Note, however, that there is no rule, in principle governing the maximum number of iterations, although numerical rounding errors will ultimately lead to a uniform intensity distribution in the limit $n \rightarrow \infty$.

Figure 6.8 displays the gradient selectivity of the method. In Figure 6.8A, a function u is shown, consisting of a series of steps with equal periods (40 pixels) but decreasing gradient values. In Figure 6.8B, the gradient function ∇u is shown. Figure 6.8C shows u_{fil} which is the same function after filtering using 25 iterations. In Figure 6.8D, the gradient function corresponding to the filtered function is shown. In this example, all edges are strongly enhanced except for the last feature shown, where the value of gradient is too low to be preserved by the filtering process. The sizes of the gradients preserved

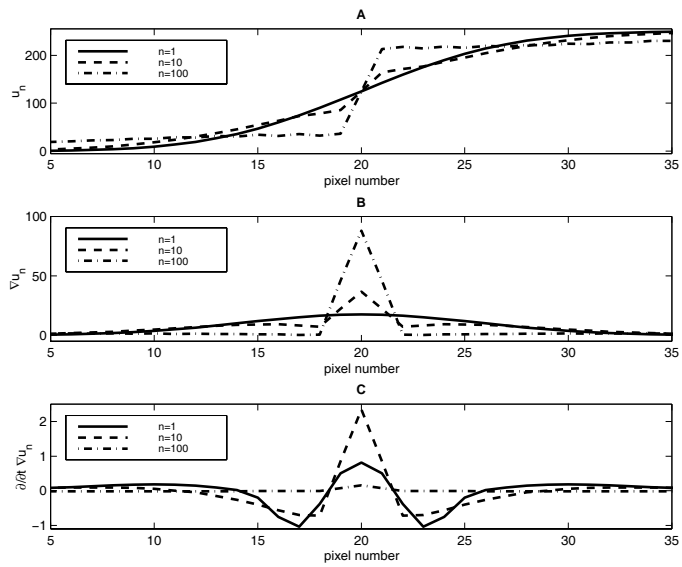


Figure 6.7: Behavior of an edge propagating through the diffusion filtering process. The evolution of u , (A); ∇u , (B); and $\partial_t \nabla u$, (C) are shown for n equal to 1, 10 and 100 iteration steps.

are governed by the shape of the diffusivity function g , and are strongly dependent on the value of the contrast parameter λ . In the present case, $\lambda = 6.4$ was chosen. A smaller λ would have enhanced all edges in the test function. In a practical situation, one has to adapt λ to the physical situation prevailing, and it is clear that the method will fail if the gradient magnitudes of the structures that one wishes to preserve are similar to a significant part of the gradient magnitudes present in the noise spectrum.

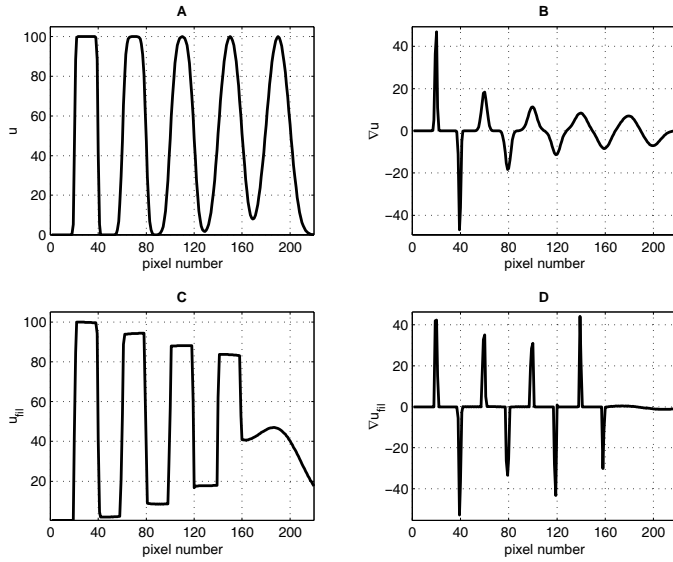


Figure 6.8: Illustration of the edge selective behavior of the diffusion process. (A) shows the unprocessed oscillating function u ; (B) its gradient function ∇u . Function u contains a large range of gradient values, successively decreasing for each period. (C) shows the filtered version u_{fil} and (D) its gradient function ∇u_{fil} . It can be seen that the peak with the least steep edges (the rightmost peak in graph (A)) is not retained in the filtered image. All other edges are strongly enhanced.

The performance of the method in the presence of various degrees of noise is seen in Figure 6.9. Again, a series of structures is shown with a period of 40 pixels and a nominal amplitude of 100 counts (on top of a DC level of 100 counts). The standard deviation of the noise added to each peak (from left to right) is 0%, 10%, 20%, 30% and 40% of the maximum amplitude. It can be seen that the edges in the original function are almost completely restored and slightly enhanced, despite the severe noise levels present.

In a given experimental situation, the best choice of λ can be obtained by a spectral analysis of the gradient magnitudes present in the noise and by adapting $g(\nabla u, \lambda)$ to maximize diffusion over the noise components, while preserving the frequency compo-

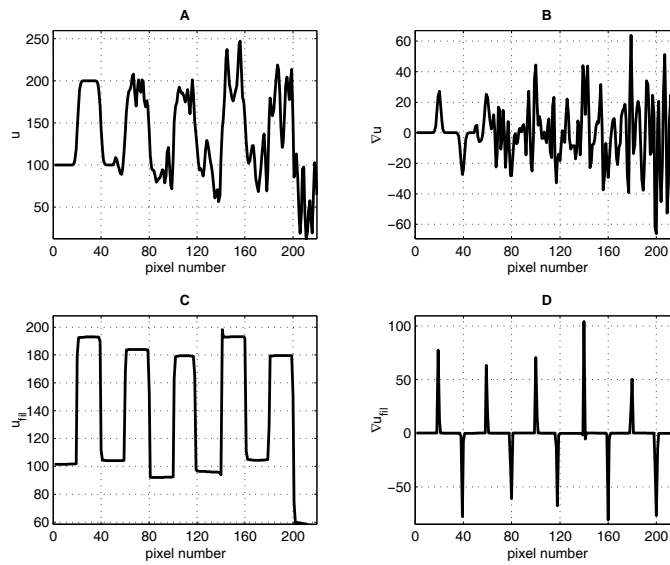


Figure 6.9: Filtering of periodic structures with increasing degrees of white noise added and consequently decreasing signal-to-noise ratios. (A) shows the unfiltered function u and (B) its gradient function. After filtering, (C), the features of the original periodic function are restored and enhanced. (D) shows the corresponding gradient function after filtering.

nents corresponding to edges that one wishes to preserve. An example of this is shown in Figure 6.10, which corresponds to a function of the type displayed in Figure 6.9. Figure 6.10A shows the gradient of the original, noise free periodic function and Figure 6.10B shows its gradient spectrum, upon which the flux function $\Phi = g(\nabla u)\nabla u$ is superimposed. Note that the absolute value of the, in this case one-dimensional, gradient ∇u is removed for this illustration. Figure 6.10C shows the gradient function corresponding to the periodic function corrupted by increasing amounts of white noise. Figure 6.10E shows the gradients in the noisy function after filtering using $n = 100$ iterations. Comparing the histograms, one can clearly see how components corresponding to noise have effectively been diffused away. Note that some of the moderate gradient component in the original also diffuse away, which leads to edge sharpening. In the presented case the optimal diffusivity function corresponds to $\lambda = 11$. For $\lambda > 11$, the original edges are not retained, but for $\lambda < 11$ some noise gradients may be amplified. In extreme situations such as in this example, where the gradient spectrum of the noise overlaps significantly with that of the original function, the correct choice of λ is crucial. As is discussed in [42] in a similar context, the optimal λ should be chosen to be as small as possible so that the largest possible range of physical gradient magnitudes can be preserved. In the context of the PLIF images that are presented below, we found $\lambda = 1.2\sigma_n$, where σ_n is the standard deviation of the noise gradient distribution, to be a good choice for almost all experimental situations encountered.

Note that the current analysis corresponds to the diffusivity function (6.7) using the parameter value $m = 8$. The choice of m defines the shape of the diffusivity function. Therefore the optimal choice of λ to suppress noise and sharpen edges might not be the same with a different choice of m and a new analysis has to be performed.

6.4 Nonlinear Diffusion Filtering of OH PLIF Images

An example of an OH PLIF image before and after nonlinear filtering is shown in Figure 6.4, where Figure 6.4B is the unprocessed OH PLIF image and Figure 6.4C shows the image after nonlinear diffusion filtering, using the diffusivity function 6.7, with $m = 8$. The image corresponds to 300 computational iterations with $\lambda = 4.5$ and $\sigma = 1$. Clearly the signal-to-noise ratio and the contrast-to-noise ratio are clearly enhanced, thus simplifying a successive segmentation of OH boundaries, cf. Section 6.5.

In Figure 6.11, the effect of the contrast parameter λ on the filtered images is illustrated. Figure 6.11A shows the intensity along one pixel row, corresponding to the line indicated in white in Figure 6.4B. In Figures 6.11B-6.11D, the filtered data is shown for λ equal to 3, 5 and 7. The edge sharpening is evident from these cuts. Note the effect of the contrast parameter on the preservation of image structure: As λ is increased, small-scale structure is lost. In Figure 6.11D, for example, the middle peak of OH has begun to disappear. It is clear, therefore, that the proper choice of the contrast parameter is crucial. However, since the noise gradient spectrum was similar in all experimental situations that

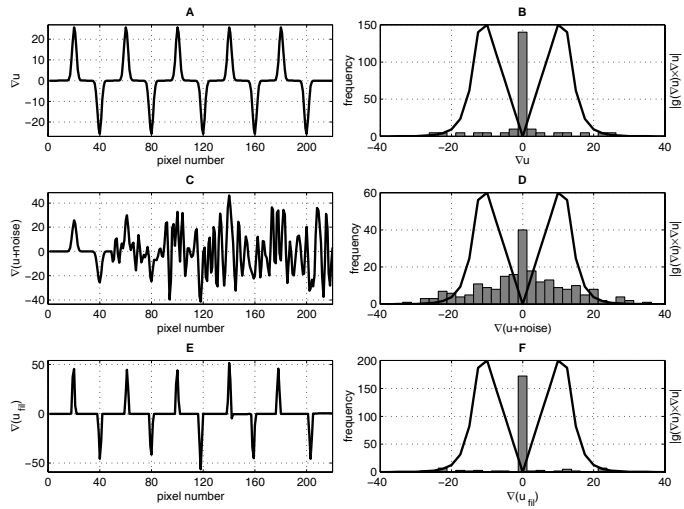


Figure 6.10: (A) shows the gradient function ∇u of an unperturbed periodic function u and (C) shows the gradient function of the same function with added noise (analogous to Figure 6.9A). (E) shows the gradient function ∇u_{fil} of the filtered structure. (B),(D) and (F) are the corresponding histograms of the gradient distributions. The magnitude of $g(|\nabla u|)\nabla u$ is superimposed on the histogram to help visualize which gradients values that are preserved by the process.

we encountered, a single λ value (obtained by the method outlined in Section 6.3) was usually sufficient to produce satisfactory results for nearly all images processed within an experimental run. The choice of width of the smoothing kernel σ and the number of iterations n is far less critical, cf. Section 6.5. Good results were achieved for $n = 150$ and $\sigma = 1$ in all cases.

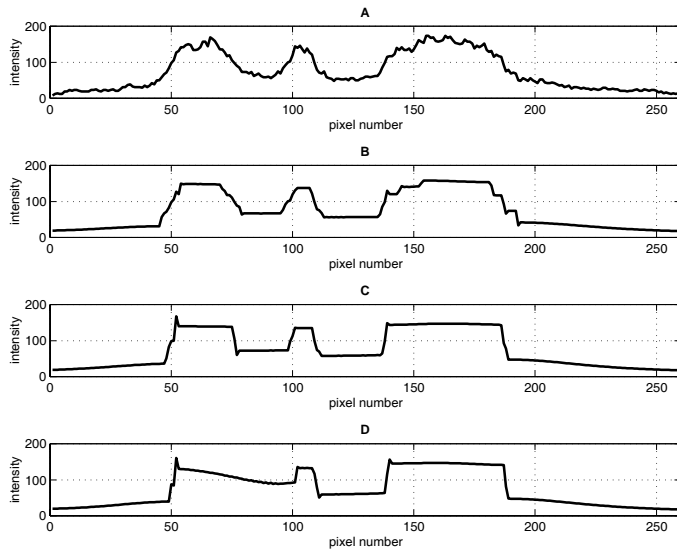


Figure 6.11: Effects of the contrast parameter λ on the retainment of image structure during nonlinear diffusion filtering. The line profile shown corresponds to the pixel row indicated in white in Figure 6.4B. (A) shows the raw intensity profile. (B),(C) and (D) shows the filtered profile for λ equal to 3, 5 and 7, respectively.

Figure 6.12 shows three-dimensional plots of Figures 6.4B and 6.4C. A comparison of the two images illustrates several attractive properties of the filtering process. Noise levels are drastically reduced without compromising overall intensities. Small structure are preserved, while OH boundaries are strongly enhanced.

6.5 Flame Front Identification

One can easily identify the OH boundary (the flame front) from the diffusion filtered images by computing the gradient of the filtered image and thresholding and binarizing the resulting gradient image. The boundary of the resulting image is then taken as the OH boundary. In Figure 6.13, this procedure has been applied to Figure 6.4C and the detected boundary has been superimposed on the raw image (Figure 6.4B). From the

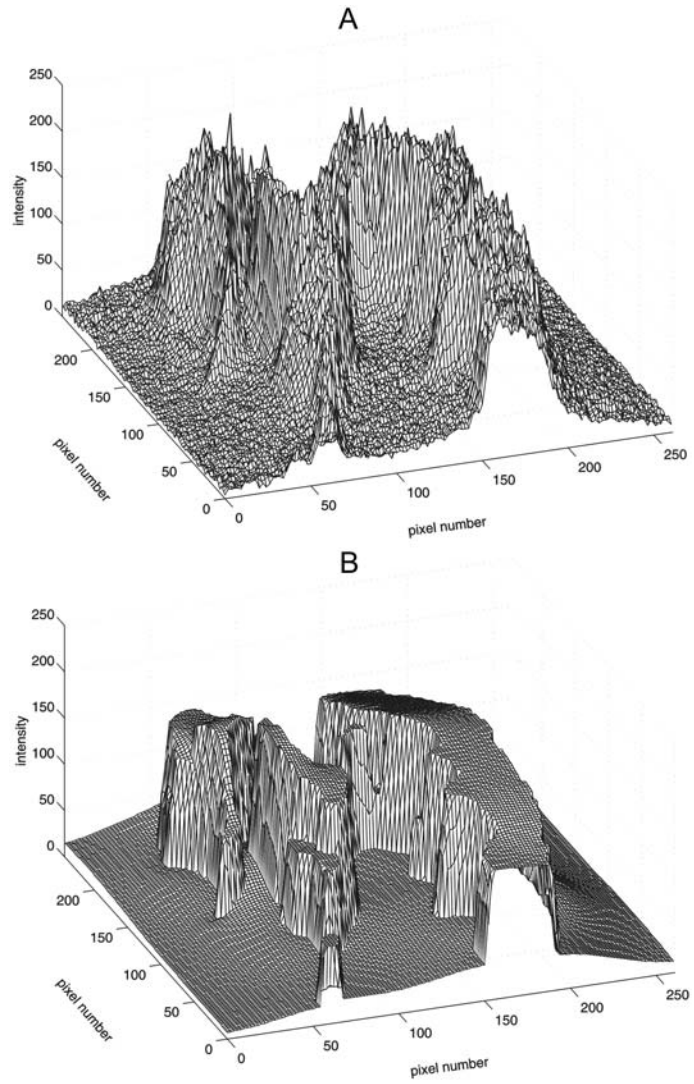


Figure 6.12: Three-dimensional plots of Figure 6.4B and Figure 6.4C, respectively.

identified flame front, the circumference-to-area ratio, η , of the burned regions in the flame can be calculated, which is a parameter that is highly suitable for comparisons with theoretical flame calculations [65].

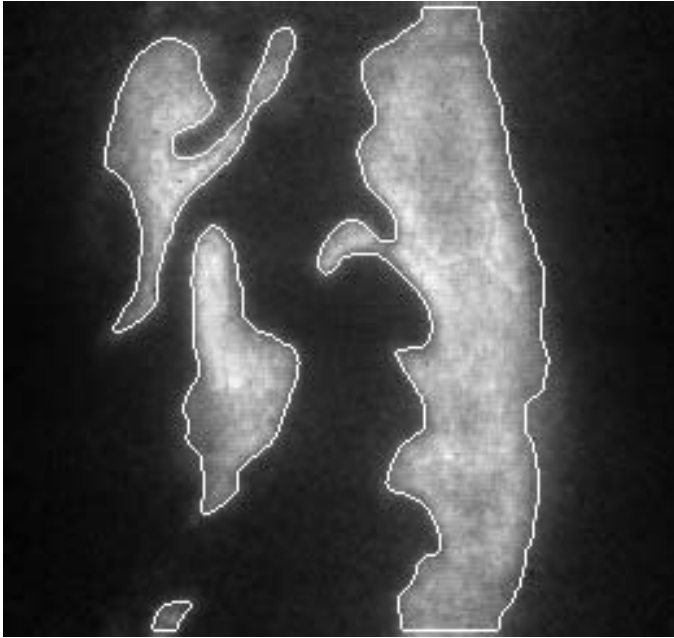


Figure 6.13: Original raw PLIF image with the identified flame front superimposed as white outlines.

An important issue in the present application is how the choice of parameters for the nonlinear filtering influences η . To investigate this quantitatively, we evaluated the sensitivity of η to the choice of the parameters in (6.8) and to the signal-to-noise level in the raw image.

For this purpose, the OH PLIF image in Figure 6.4B was first nonlinear diffusion filtered with a range of values used for λ , σ and n . The flame front was then extracted from the filtered images by the method outlined above, and η was evaluated. The variation of η in relation to these parameters is plotted in Figure 6.14A-6.14C.

In the plots in Figure 6.14, η is normalized to 1 for the parameter values used in Section 6.4, i.e. $\lambda = 4.5$, $\sigma = 1$, $m = 8$ and $n = 150$. In Figure 6.14A, η is shown as a function of the contrast parameter λ . It can be seen that η increases with increasing λ . This result can be explained by the fact that small features with low gradients are not preserved by the filter at large λ 's, cf. Figure 6.11D, and as a consequence the evaluated area decreases, thus making η larger. As can be seen, η increases by approximately 13%

when λ increase by nearly a factor 2. As was discussed above, the critical dependence on the choice of λ is a weakness of the method, which may limit its use in some applications. For larger λ than 5.5, in the present example, structures start to merge and some main edges start to diffuse away after many iterations.

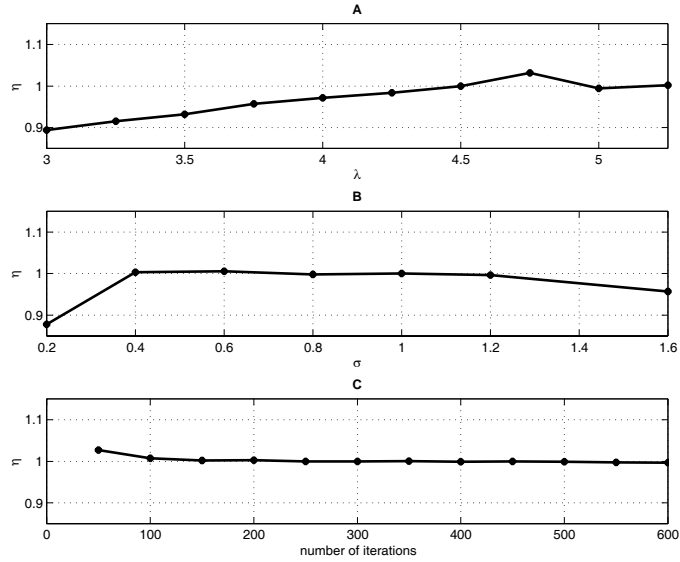


Figure 6.14: Dependence of the circumference to area ratio, η , as a function of different filter parameters. (A) shows the dependence on λ , (B) on σ and (C) on n .

In Figure 6.11B, η is plotted as a function of the width of the smoothing kernel σ . Over the range $0.4 \leq \sigma \leq 1.2$, η stays nearly constant and thus the choice of σ is not critical in the present application. For values of σ smaller than 0.4, the flame area is overestimated because noisy structures are preserved. Values of σ larger than 1.2 result in the same problem as the classical linear diffusion: Edges become blurred to a degree where physical information is lost.

In Figure 6.11C, the evolution of η with n , the number of iterations, is shown. It can be seen that η stabilizes after slightly more than 100 iterations, although there is actually a continuous small decrease caused by numerical errors.

To investigate the influence of the signal-to-noise ratio on η , a series of test images were generated with increasing degrees of noise added. For this purpose a test image was generated from the OH contour depicted in Figure 6.13. Gaussian noise, with a standard deviation ranging from 0 to 60% of the signal level was added to this test image. The ratio η was then calculated in the same manner as described above. The results are shown in Figure 6.15. For noise levels up to 10%, corresponding to a signal to noise ratio of 10,

η is constant within less than a percent, highlighting the strength of the current method. For larger noise levels, η is increasing because some artificial structure is being generated by the noise. Note, however, that even for noise levels as high as 35% (corresponding to a signal-to-noise ratio of only 3, a situation rarely present in an actual experiment) the increase in η is only approximately 4%.

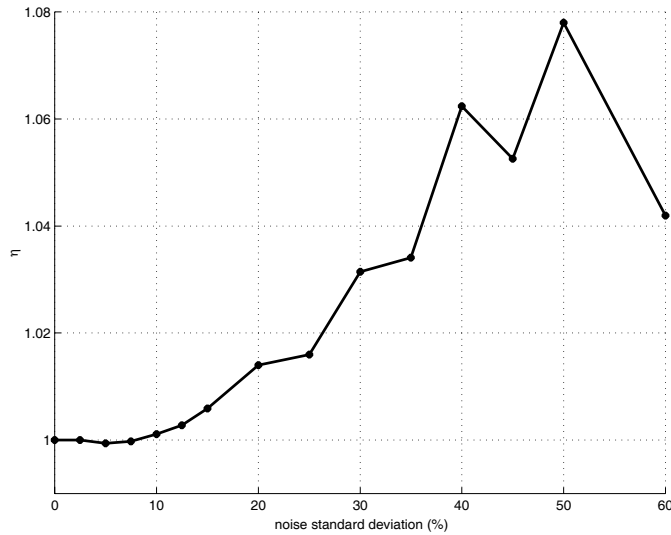


Figure 6.15: Circumference-to-area ratio, η , as a function of the noise level in an artificially generated test image.

6.6 Conclusions

In this chapter, the application and adaptation of nonlinear diffusion filtering to planar laser induced fluorescence (PLIF) spectroscopy has been presented. An introduction to both nonlinear diffusion in image processing and to the imaging technique of PLIF spectroscopy has been given. In the present application the main objective has been to extract contours in PLIF images of flames for the evaluation of reaction boundaries in turbulent flows. Nonlinear diffusion filtering is ideal for this since it suppresses the noise present in the imaging process and at the same time enhances the unsharp object boundaries owing to e.g. detector limitations. This facilitates subsequent extraction of the flame contour, and quantities interesting for comparison with theoretical findings, such as the circumference-to-area ratio η , can easily be calculated.

A thorough analysis of nonlinear diffusion filtering in general and for the processing

of PLIF images of flames in particular, is presented. The influence of the parameters in the technique is examined. Especially, it is shown that the interesting quantity η is rather insensitive to the choice of the width σ of the smoothing kernel and to the number of iterations, while the parameter λ is a bit more critical. It is also shown that the amount of noise added to the image, before the nonlinear filtering process, influence the subsequently calculated value of η only slightly up to very high noise levels.

Although the filtering technique has been applied in the particular context of laser-induced fluorescence imaging, it should be equally well suited for other planar laser diagnostic techniques, e.g. planar imaging by Mie, Rayleigh or Raman spectroscopy. For reports on successful usage of the technique for data extraction from measurements performed in turbulent flames, cf. [23], [66], [4] and [64].

Appendix A

Expected Value of Least Squares Solutions for the Estimation of Intersections and Motion

A second-order Taylor expansion of the expected values of the least squares solution for both the estimation of an intersection point, Section 5.2, and the estimation of a motion vector (optical flow), Section 5.3, will here be given.

Let \mathbf{y} be the vector to be estimated, i.e. either the intersection point \mathbf{x} or the flow \mathbf{u} . I_s is the matrix consisting of the spatial derivatives I_{x_i}, I_{y_i} . \mathbf{C} in Section 5.2 is the vector whose elements are $I_{x_i}x_{0_i} + I_{y_i}y_{0_i}$ and \mathbf{C} in Section 5.3 is the vector of temporal derivatives I_{t_i} .

The expected value $E(\mathbf{y})$ of the least squares solution is given by

$$E(\mathbf{y}) = E \left((I_s^T I_s)^{-1} (I_s^T \mathbf{C}) \right) .$$

As the noise is assumed to be independent and have zero-mean, the first-order terms as well as the second-order terms in the noise of the temporal derivatives (or the positional parameters) vanish. This means that it is only the noise in the spatial derivatives which causes bias in the mean. The expansion at points corresponding to zero noise (i.e., $\delta I_{x_i} = \delta I_{y_i} = \delta I_{t_i} = 0$ or $\delta I_{x_i} = \delta I_{y_i} = \delta x_{0_i} = \delta y_{0_i} = 0$) can be written as

$$E(\mathbf{y}) = \mathbf{y}' + \sum_i \left(\left. \frac{\partial^2 \mathbf{y}}{\partial \delta I_{x_i}^2} \right|_{N=0} \frac{E(\delta I_{x_i}^2)}{2} + \left. \frac{\partial^2 \mathbf{y}}{\partial \delta I_{y_i}^2} \right|_{N=0} \frac{E(\delta I_{y_i}^2)}{2} \right) ,$$

where $N = 0$ denotes zero noise. For notational simplicity, we define

$$\begin{aligned} M &= I_s^T I_s \quad \text{and} \quad \mathbf{b} = I_s^T \mathbf{C} , \\ M' &= I_s'^T I_s' \quad \text{and} \quad \mathbf{b}' = I_s'^T \mathbf{C}' . \end{aligned}$$

The explicit terms of the matrix M are

$$M = \begin{bmatrix} \sum_i (I'_{x_i} + \delta I_{x_i})^2 & \sum_i (I'_{x_i} + \delta I_{x_i}) (I'_{y_i} + \delta I_{y_i}) \\ \sum_i (I'_{x_i} + \delta I_{x_i}) (I'_{y_i} + \delta I_{y_i}) & \sum_i (I'_{y_i} + \delta I_{y_i})^2 \end{bmatrix} .$$

Denote by \mathbf{b}_l the vector \mathbf{b} corresponding to the estimation of line intersections and by \mathbf{b}_f the vector corresponding to estimation of flow. The explicit terms are

$$\mathbf{b}_f = \begin{bmatrix} \sum_i (I'_{x_i} + \delta I_{x_i}) (I'_{t_i} + \delta I_{t_i}) \\ \sum_i (I'_{y_i} + \delta I_{y_i}) (I'_{t_i} + \delta I_{t_i}) \end{bmatrix} ,$$

and

$$\mathbf{b}_l = \begin{bmatrix} \sum_i (I'_{x_i} + \delta I_{x_i})^2 (x'_{0_i} + \delta x_{0_i}) + (I'_{x_i} + \delta I_{x_i}) (I'_{y_i} + \delta I_{y_i}) (y'_{0_i} + \delta y_{0_i}) \\ \sum_i (I'_{x_i} + \delta I_{x_i}) (I'_{y_i} + \delta I_{y_i}) (x'_{0_i} + \delta x_{0_i}) + (I'_{y_i} + \delta I_{y_i})^2 (y'_{0_i} + \delta y_{0_i}) \end{bmatrix}.$$

Recall that for an arbitrary invertible matrix Q

$$\frac{-\partial Q^{-1}}{\partial x} = Q^{-1} \frac{\partial Q}{\partial x} Q^{-1}.$$

The first-order and second-order derivatives are

$$\begin{aligned} \frac{\partial \mathbf{y}}{\partial \delta I_{x_i}} &= -M^{-1} \begin{bmatrix} 2I_{x_i} & I_{y_i} \\ I_{y_i} & 0 \end{bmatrix} M^{-1} \mathbf{b} + M^{-1} \frac{\partial \mathbf{b}}{\partial \delta I_{x_i}}, \\ \frac{\partial^2 \mathbf{y}}{\partial \delta I_{x_i}^2} &= 2M^{-1} \begin{bmatrix} 2I_{x_i} & I_{y_i} \\ I_{y_i} & 0 \end{bmatrix} M^{-1} \begin{bmatrix} 2I_{x_i} & I_{y_i} \\ I_{y_i} & 0 \end{bmatrix} M^{-1} \mathbf{b} \\ &\quad - M^{-1} \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} M^{-1} \mathbf{b} \\ &\quad - 2M^{-1} \begin{bmatrix} 2I_{x_i} & I_{y_i} \\ I_{y_i} & 0 \end{bmatrix} M^{-1} \frac{\partial \mathbf{b}}{\partial \delta I_{x_i}} + M^{-1} \frac{\partial^2 \mathbf{b}}{\partial \delta I_{x_i}^2}, \end{aligned}$$

and, similarly, symmetric expressions for $\frac{\partial \mathbf{y}}{\partial \delta I_{y_i}}$ and $\frac{\partial^2 \mathbf{y}}{\partial \delta I_{y_i}^2}$. Since it is assumed that $E(\delta I_{x_i}^2) = E(\delta I_{y_i}^2)$, the expansion can be simplified to

$$\begin{aligned} E(\mathbf{y}) &= \mathbf{y}' - \underline{\underline{n M'^{-1} \mathbf{y}' \sigma_s^2}} + \underline{\underline{\sum_{i=1}^n M'^{-1} \left(\frac{\partial^2 \mathbf{b}'}{\partial \delta I_{x_i}^2} + \frac{\partial^2 \mathbf{b}'}{\partial \delta I_{y_i}^2} \right) \Big|_{N=0} \frac{\sigma_s^2}{2}}} \\ &\quad + \sum_i \left\{ M'^{-1} \left(\begin{bmatrix} 2I_{x_i}' & I_{y_i}' \\ I_{y_i}' & 0 \end{bmatrix} M'^{-1} \begin{bmatrix} 2I_{x_i}' & I_{y_i}' \\ I_{y_i}' & 0 \end{bmatrix} \right. \right. \\ &\quad \left. \left. + \begin{bmatrix} 0 & I_{x_i}' \\ I_{x_i}' & 2I_{y_i}' \end{bmatrix} M'^{-1} \begin{bmatrix} 0 & I_{x_i}' \\ I_{x_i}' & 2I_{y_i}' \end{bmatrix} \right) \mathbf{y}' \right. \\ &\quad \left. - M'^{-1} \left(\begin{bmatrix} 2I_{x_i}' & I_{y_i}' \\ I_{y_i}' & 0 \end{bmatrix} M'^{-1} \frac{\partial \mathbf{b}}{\partial \delta I_{x_i}} \right) \Big|_{N=0} \right. \\ &\quad \left. + \begin{bmatrix} 0 & I_{x_i}' \\ I_{x_i}' & 2I_{y_i}' \end{bmatrix} M'^{-1} \frac{\partial \mathbf{b}}{\partial \delta I_{y_i}} \Big|_{N=0} \right) \Big\} \sigma_s^2, \end{aligned}$$

where the terms that do not depend on n are underlined (where n is the number of measurements being combined in a region). These terms will give a consistent, statistically constant response. The rest of the terms diminish proportionally to $\frac{1}{n}$. Informal experiments show that these terms become negligible for $n > 5$, a number clearly smaller than the number of terms likely to be combined in any real system.

In the analysis of Section 5.2, the elements of \mathbf{C} are $I_{x_i}x_{0_i} + I_{y_i}y_{0_i}$. Thus

$$\begin{aligned} \left. \frac{\partial \mathbf{b}_i}{\partial \delta I_{x_i}} \right|_{N=0} &= \begin{bmatrix} 2I_{x_i}x_{0_i} + I_{y_i}y_{0_i} \\ I_{y_i}x_{0_i} \end{bmatrix} \left. \frac{\partial^2 \mathbf{b}_i}{\partial \delta I_{x_i}^2} \right|_{N=0} = \begin{bmatrix} 2x_{0_i} \\ 0 \end{bmatrix}, \\ \left. \frac{\partial \mathbf{b}_i}{\partial \delta I_{y_i}} \right|_{N=0} &= \begin{bmatrix} I_{x_i}y_{0_i} \\ I_{x_i}x_{0_i} + 2I_{y_i}y_{0_i} \end{bmatrix} \left. \frac{\partial^2 \mathbf{b}_i}{\partial \delta I_{y_i}^2} \right|_{N=0} = \begin{bmatrix} 0 \\ 2y_{0_i} \end{bmatrix}, \end{aligned}$$

and the main terms in the expected value of the intersection point $E(\mathbf{x})$ are

$$E(\mathbf{x}) = \mathbf{x}' - nM'^{-1}\mathbf{x}'\sigma_s^2 + M'^{-1} \begin{bmatrix} \Sigma x_{0_i} \\ \Sigma y_{0_i} \end{bmatrix} \sigma_s^2,$$

that is,

$$E(\mathbf{x}) = \mathbf{x}' + nM'^{-1}(\bar{\mathbf{x}}_0 - \mathbf{x}')\sigma_s^2,$$

where $\bar{\mathbf{x}}_0$ denotes the mean of the values \mathbf{x}_{0_i} .

In the analysis in Section 5.3 the elements of \mathbf{C} are I_{t_i} . Thus the first-order derivatives are

$$\left. \frac{\partial \mathbf{b}}{\partial \delta I_{x_i}} \right|_{N=0} = \begin{bmatrix} I_{t_i} \\ 0 \end{bmatrix} \quad \text{and} \quad \left. \frac{\partial \mathbf{b}}{\partial \delta I_{y_i}} \right|_{N=0} = \begin{bmatrix} 0 \\ I_{t_i} \end{bmatrix},$$

and the second-order derivatives vanish. The expected value of the flow $E(\mathbf{u})$ simplifies to

$$E(\mathbf{u}) = \mathbf{u}' - nM'^{-1}\mathbf{u}'\sigma_s^2.$$

Bibliography

- [1] L. De Agapito, D.Q. Huynh, and M.J. Brooks. Self-calibrating a stereo head: an error analysis in the neighbourhood of degenerate configurations. In *Proc. 6th Int. Conf. Computer Vision*, pages 747–753, Bombay, 1998.
- [2] <http://www.akita-u.ac.jp/~kmori/img/kitaoka.html>.
- [3] L. Alvarez, F. Guichard, P.-L. Lions, and J.-M. Morel. Axioms and fundamental equations in image processing. *Arch. Rational Mech. Anal.*, 29:199–257, 1993.
- [4] M. Andersson. Three-dimensional visualization of turbulent flames using shape-based interpolation. Master’s thesis, Lund Institute of Technology, 2000.
- [5] N. Andreff, R. Horaud, and B. Espiau. Robot hand-eye calibration using structure-from-motion. *Int. J. Robotics Research*, 20(3):228–248, 2001.
- [6] J.L. Barron, D.J. Fleet, and S. Beauchemin. Performance of optical flow techniques. *Int. Journal Computer Vision*, 12(1):43–77, 1994.
- [7] M. Bertalmío, A.L. Bertozzi, and G. Sapiro. Navier-stokes, fluid dynamics and image and video inpainting. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 355–362, Kauai, Hawaii, December 2001.
- [8] T. Brodsky and C. Fermüller. Self-calibration from image derivatives. *Int. J. Computer Vision*, 48(2):91–114, 2002.
- [9] A. Bulatov, A. Bertulis, and L. Mickiene. Geometrical illusions: Study and modeling. *Biological Cybernetics*, 77:395–406, 1997.
- [10] N.P. Galatsanos C.-J. Tsai and A.K. Katsaggelos. Optical flow estimation from noisy data using differential techniques. In *Proc. ICASSP*, volume VI, pages 3390–3393, Phoenix, USA, 1999.
- [11] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8:679–698, 1986.
- [12] R.H.S. Carpenter. *Movements of the Eye*. Pion, 1988.
- [13] H.A. Carr. *An Introduction to Space Perception*. Green, 1935.

BIBLIOGRAPHY

- [14] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numerical Analysis*, 29:182–193, 1992.
- [15] H.H. Chen. A screw motion approach to uniqueness analysis of head-eye geometry. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 145–151, Hawaii, 1991.
- [16] C. Chiang. A new theory to explain geometrical illusions produced by crossing lines. *Percept. Psychophys.*, 3:174–176, 1968.
- [17] J. C. K. Chou and M. Kamel. Finding the position and orientation of a sensor on a robot manipulator using quaternions. *Int. J. of Robotics Research*, 10(3):240–254, 1991.
- [18] S. Christy and R. Horaud. Euclidean shape and motion from multiple perspective views by affine iterations. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(11):1098–1104, 1996.
- [19] H. S. M. Coxeter. *The Real Projective Plane*. Springer-Verlag, New York, 3 edition, 1992.
- [20] K. Daniilidis. Hand-eye calibration using dual quaternions. *Int. J. Robotics Research*, 18(3):286–298, 1999.
- [21] D. Demirdjian, A. Zisserman, and R. Horaud. Stereo autocalibration from one plane. In *Proc. European Conf. Computer Vision*, volume 2, pages 625–639, Dublin, 2000.
- [22] F. Devernay and O.D. Faugeras. From projective to euclidean reconstruction. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 264–269, San Francisco, 1996.
- [23] T. Ding, T.H. van der Meer, M. Versluis, M. Golombok, J. Hult, M. Aldén, and C. Kaminski. Time-resolved plif measurements in turbulent diffusion flames. In *Proc. Int. Symp. Turbulence, Heat and Mass Transfer*, pages 857–864, Tokyo, 2000.
- [24] F. Dornaika and R. Horaud. Simultaneous robot-world and hand-eye calibration. *IEEE Trans. Robotics and Automation*, 14(4):617–622, 1998.
- [25] A. Dreizler, S. Lindenmaier, U. Maas, J. Hult, M. Aldén, and C.F. Kaminski. Characterization of a spark ignition system by planar laser induced fluorescence of oh at high repetition rates and comparison with chemical kinetic calculations. *Appl. Phys. B*, 70:287–294, 2000.

-
- [26] D.C. Earle and S. Maskell. Fraser cords and reversal of the café wall illusion. *Perception*, 22:383–390, 1993.
- [27] A.C. Eckbreth. *Laser Diagnostics for Combustion Temperature and Species*, volume 3 of *Combustion Science and Technology Book Series*. Gordon and Breach publishers, 1996.
- [28] C. Enroth-Cugel and J.D. Robson. The contrast sensitivity of retinal ganglion cells of the cat. *J. Physiol. (Lond.)*, 187:517–522, 1966.
- [29] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993.
- [30] O. Faugeras and R. Keriven. Variational principles, surface evolution, pdes, level set methods, and the stereo problem. *IEEE Trans. Image Processing*, 7(3):336–344, March 1998.
- [31] O. Faugeras, T. Luong, and S. Maybank. Camera self-calibration: theory and experiments. In *Proc. Euro. Conf. Computer Vision*, pages 471–478, Stockholm, May 1994.
- [32] J.A. Feldman. Four frames suffice: A provisional model of vision and space. *Behavioral and Brain Sciences*, 8:265–313, 1985.
- [33] C. Fermüller and Y. Aloimonos. Direct perception of three-dimensional motion from patterns of visual motion. *Science*, 270:1973–1976, 1995.
- [34] C. Fermüller and Y. Aloimonos. On the geometry of visual correspondence. *Int. J. Computer Vision*, 21(3):223–247, 1997.
- [35] C. Fermüller and Y. Aloimonos. Observability of 3d motion. *Int. J. Computer Vision*, 37(1):43–63, 2000.
- [36] C. Fermüller, H. Malm, and Y. Aloimonos. Uncertainty in visual processes predicts geometrical optical illusions. *Accepted for publication in Vision Research*, 2003.
- [37] C. Fermüller, R. Pless, and Y. Aloimonos. The ouchi illusion as an artifact of biased flow estimation. *Vision Research*, 40:77–96, 2000.
- [38] C. Fermüller, R. Pless, and Y. Aloimonos. The statistics of optical flow. *Computer Vision and Image Understanding*, 82:1–32, 2001.
- [39] J. Fraser. A new visual illusion of direction. *Brit. J. Psychol.*, 2:307–320, 1908.
- [40] W. Fuller. *Measurement Error Models*. Wiley, New York, 1987.
- [41] L. Ganz. Mechanism of the f.a.e.’s. *Psychol. Rev.*, 73:128–150, 1966.

BIBLIOGRAPHY

- [42] G. Gerig, O. Klüber, R. Kikinis, and F.A. Jolesz. Nonlinear anisotropic filtering of mri data. *IEEE Trans. Medical Imaging*, 11(2):221–232, 1992.
- [43] A.P. Ginsburg. Is the illusory triangle physical or imaginary? *Nature*, 257:219–220, 1975.
- [44] A.P. Ginsburg. Visual form perception based on biological filtering. In *Sensory Experience, Adaptation and Perception*, pages 53–72, New Jersey, 1984.
- [45] L. Glass. Effect of blurring on perception of a simple geometric pattern. *Nature*, 228:1341–1342, 1970.
- [46] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, 3 edition, 1996.
- [47] G.H. Golub and C.F. Van Loan. An analysis of the total least squares problem. *SIAM J. Numerical Analysis*, 17:883–893, 1980.
- [48] R.L. Gregory. Distortion of visual space as inappropriate constancy scaling. *Nature*, 119:678, 1963.
- [49] S. Grossberg and E. Mingolla. Neural dynamics of form perception: Boundary completion, illusory figures and neon color spreading. *Psychological Review*, 92(2):173–211, 1985.
- [50] S. Grossberg and E. Mingolla. Neural dynamics of perceptual grouping: Textures, boundaries and emergent segmentations. *Perception and Psychophysics*, 38(2):141–171, 1985.
- [51] A. Handlovicova, K. Mikula, and F. Sgallari. Numerical methods for nonlinear diffusion equations arising in image processing. *Preprint No. 1*, Slovak University of Technology, Bratislava, 1999.
- [52] K. Hanna. Direct multi-resolution estimation of ego-motion and structure from motion. In *Proc. IEEE Workshop on Visual Motion*, pages 156–162, Princeton, NJ, 1991.
- [53] R.K. Hanson. Combustion diagnostics: planar imaging techniques. *Proc. Combustion Institute*, 21:1677–1691, 1986.
- [54] C.J. Harris and M. Stephens. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conf*, pages 147–151, Manchester, UK, 1988.
- [55] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

-
- [56] E. Hering. *Beiträge zur Psychologie*, volume 1. Engelman, Leipzig, 1861.
- [57] A. Heyden. On the consistency of line-drawings, obtained by projections of piecewise planar objects. *J. Mathematical Imaging and Vision*, 6:393–412, 1996.
- [58] A. Heyden and K. Åström. Euclidean reconstruction from constant intrinsic parameters. In *Proc. Int. Conf. Pattern Recognition*, pages 339–343, Vienna, 1996.
- [59] A. Heyden and K. Åström. Euclidean reconstruction from image sequences with varying and unknown focal length and principal point. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 438–443, Puerto Rico, 1997.
- [60] R. Horaud and G. Csurka. Self-calibration and euclidean reconstruction using motions of a stereo rig. In *Proc. Int. Conf. Computer Vision*, pages 96–103, Bombay, India, 1998.
- [61] R. Horaud and F. Dornaika. Hand-eye calibration. *Int. J. Robotics Research*, 14(3):195–210, 1995.
- [62] B.K.P. Horn and E.J. Weldon Jr. Direct methods for recovering motion. *Int. J. Computer Vision*, 2(1):51–76, June 1988.
- [63] D.H. Hubel and T.N. Wiesel. Integrative action in the cat’s lateral geniculate body. *J. Physiol. (Lond.)*, 155:386–398, 1961.
- [64] J. Hult. *Development of Time Resolved Laser Imaging Techniques for Studies of Turbulent Reacting Flows*. PhD thesis, Lund Institute of Technology, March 2002.
- [65] C.F. Kaminski, X.S. Bai, J. Hult, A. Dreizler, and L. Fuchs. Spark ignition of turbulent methane/air mixtures revealed by time resolved planar laser induced fluorescence and direct numerical simulations. *Proc. Combustion Institute*, 28, 2000.
- [66] C.F. Kaminski, X.S. Bai, J. Hult, A. Dreizler, S. Lindenmaier, and L. Fuchs. Flame growth and wrinkling in a turbulent flow. *Appl. Phys. B*, 71:711–716, 2000.
- [67] C.F. Kaminski, J. Engström, and M. Aldén. Quasi instantaneous 2 dimensional temperature measurements in a spark ignition engine using 2-line atomic fluorescence. *The 27th Symp. on Combustion*, pages 85–93, 1998.
- [68] C.F. Kaminski, J. Hult, and M. Aldén. High repetition rate planar laser induced fluorescence of OH in a turbulent non-premixed flame. *Appl. Phys. B*, 68:757–760, 1999.
- [69] W. Köhler and H. Wallach. Figural after-effects: An investigation of visual process. *Proc. Amer. Philos. Soc.*, 88:269–357, 1944.

BIBLIOGRAPHY

- [70] J.J. Koenderink. The structure of images. *Biol. Cybern.*, 50:363–370, 1984.
- [71] A. Kundt. Untersuchungen über augenmass und optische täuschungen. *Pogg. Ann.*, 120:118–158, 1863.
- [72] T. Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer, Boston, 1994.
- [73] H. C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. In *Proc. Royal Society of London B*, volume 208, pages 385–397, 1980.
- [74] Q.-T. Loung and O.D. Faugeras. Self-calibration of a moving camera from point correspondences and fundamental matrices. *Int. J. Computer Vision*, 22(3):261–289, 1997.
- [75] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. 7th Int. Joint Conf. on AI*, pages 674–679, Vancouver, B.C., Canada, 1981.
- [76] S.D. Ma. A self-calibration technique for active vision systems. *IEEE Trans. Robotics and Automation*, 12(1):114–120, February 1996.
- [77] H. Malm. Multiscale image analysis and image enhancement based on partial differential equations. Master’s thesis, Lund Institute of Technology, April 1998.
- [78] H. Malm and A. Heyden. Hand-eye calibration from image derivatives. In *Proc. Euro. Conf. Computer Vision*, volume II, pages 493–507, Dublin, June 2000.
- [79] H. Malm and A. Heyden. A new approach to hand-eye calibration. In *Proc. Int. Conf. Pattern Recognition*, volume I, pages 525–529, Barcelona, September 2000.
- [80] H. Malm and A. Heyden. Stereo head calibration from a planar object. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, volume 2, pages 657–662, Kauai, Hawaii, December 2001.
- [81] H. Malm and A. Heyden. Plane-based camera calibration: The case of pure translation. In *Proc. Workshop Machine Vision Applications*, pages 146–149, Nara, Japan, December 2002.
- [82] H. Malm and A. Heyden. Self-calibration from image derivatives for active vision systems. In *Proc. Int. Conf. Automation, Robotics and Computer Vision*, Singapore, December 2002.
- [83] H. Malm and A. Heyden. Simplified intrinsic camera calibration and hand-eye calibration for robot vision. In *Proc. Int. Conf. Intelligent Robots and Systems*, Las Vegas, NV, October 2003. In press.

-
- [84] H. Malm, G. Sparr, J. Hult, and C.F. Kaminski. Nonlinear diffusion filtering of images obtained by planar laser induced fluorescence spectroscopy. *J. Optical Society America A*, 17:2148–2156, 2001.
- [85] D.W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Industrial and Applied Mathematics*, 11:431–441, 1963.
- [86] D. Marr. *Vision*. W.H. Freeman and Company, New York, 1982.
- [87] D. Marr and E.C. Hildreth. A theory of edge detection. *Proc. Royal Society London B*, 207(187-217), 1980.
- [88] S. Maybank and O. Faugeras. A theory of self-calibration of a moving camera. *Int. J. Computer Vision*, 8(2):123–152, August 1992.
- [89] M. Mühlich and R. Mester. The role of total least squares in motion analysis. In *Proc. Euro. Conf. Computer Vision*, pages 305–321, Freiburg, June 1998.
- [90] F.C. Müller-Lyer. Zur lehre von den optischen täuschungen: Über kontrast und konfluxion. *Z. Psychologie*, 9:1–16, 1896.
- [91] M.J. Morgan. The poggendorff illusion: a bias in the estimation of the orientation of virtual lines by second-stage filters. *Vision Research*, 39:2361–2380, 1999.
- [92] M.J. Morgan and S. Baldassi. How the human visual system encodes the orientation of a texture and why it makes mistakes. *Current Biology*, 7:999–1002, 1997.
- [93] M.J. Morgan and C. Casco. Spatial filtering and spatial primitives in early vision: An explanation of the zöllner-judd class of geometrical illusions. *Proc. Royal Society London B*, 242:1–10, 1990.
- [94] M.J. Morgan and N.H. Hoptopf. Perceived diagonals in grids and lattices. *Vision Research*, 29:1005–1015, 1989.
- [95] M.J. Morgan and B. Moulden. The münsterberg figure and twisted cords. *Vision Research*, 26(11):1793–1800, 1986.
- [96] S. Negahdaripour and B.K.P. Horn. Direct passive navigation. *IEEE Trans. Pattern Analysis Machine Intelligence*, 9(1):168–176, January 1987.
- [97] J. Oliensis. Direct multi-frame structure from motion for hand-held cameras. In *Proc. Int. Conf. Pattern Recognition*, volume 1, pages 889–895, Barcelona, 2000.
- [98] T. Olsson, J. Bengtsson, A. Robertsson, and R. Johansson. Visual position tracking using dual quaternions with hand-eye motion constraints. In *Proc. IEEE Int. Conf. Robotics and Automation*, Taipei, Taiwan, 2003.

BIBLIOGRAPHY

- [99] J.J. Oppel. Über geometrisch-optische täuschungen. *Jahrebericht phys. ver. Frankfurt*, pages 37–47, 1855.
- [100] A. Orth, V. Sick, J. Wolfrum, R. Maly, and M. Zahn. Simultaneous 2d single-shot imaging of oh concentrations and temperature fields in an si engine simulator. *The 25th Symp. on Combustion*, pages 143–150, 1994.
- [101] H. Ouchi. *Japanese and Geometrical Art*. Dover, New York, 1977.
- [102] T. Oyama. Japanese studies on the so-called geometrical-optical illusions. *Psychologia*, 3:7–20, 1960.
- [103] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. In *Proc. IEEE Workshop on Computer Vision*, pages 16–27, Miami, FL, 1987.
- [104] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. on Patt. Analysis and Machine Intelligence*, 12(7):629–639, 1990.
- [105] J. Piaget. *The Mechanisms of Perception*. Routledge & Kegan Paul, London, 1969. Translated by G.N. Seagram.
- [106] B. Pinna and G.J. Brelstaff. A new visual illusion of relative motion. *Vision Research*, 40(16):2091–2096, 2000.
- [107] M. Pollefeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown intrinsic camera parameters. *Int. J. Computer Vision*, 32(1):7–25, 1999.
- [108] O. Reutersvärd. *Omöjliga figurer i färg*. Doxa, Lund, 1985.
- [109] S. Rémy, M. Dhome, J.M. Lavest, and N. Daucher. Hand-eye calibration. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pages 1057–1065, 1997.
- [110] J.O. Robinson. *The Psychology of Visual Illusion*. Dover, Mineola, NY, 1998.
- [111] R.W. Rodieck. Analysis of receptive fields of cat retinal ganglion cells. *J. Neurophysiology*, 28:833–849, 1965.
- [112] C.B. Williams S.C. Dakin and R.F. Hess. The interaction of first- and second-order cues to orientation. *Vision Research*, 39:2867–2884, 1999.
- [113] J. G. Semple and G. T. Kneebone. *Algebraic Projective Geometry*. Clarendon Press, Oxford, 1952.
- [114] J.A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, Cambridge, UK, 2 edition, 1999.

-
- [115] Y. C. Shiu and S. Ahmad. Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form $ax = xb$. *IEEE Trans. Robotics and Automation*, 5(1):16–29, 1989.
- [116] G. Sparr. On the "reconstruction" of impossible objects. In *Proc. SSAB Symposium on Image Analysis*, pages 109–112, Uppsala, Sweden, March 1992.
- [117] G. P. Stein and A. Shashua. Model-based brightness constraints: On direct estimation of structure and motion. *IEEE Trans. Pattern Analysis Machine Intelligence*, 22(9):992–1015, September 2000.
- [118] G.P. Stein and A. Shashua. Direct estimation of motion and extended scene structure from a moving stereo rig. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 211–218, 1998.
- [119] P. Sturm and S. Maybank. On plane-based camera calibration: A general algorithm, singularities, applications. In *Proc. Conf. Computer Vision and Patterns Recognition*, volume 1, pages 432–437, USA, 1999.
- [120] R. Tausch. Optische täuschungen als artifizielle effekte der gestaltungsprozesse von grössen und formkonstanz in der natürlichen raumwahrnehmung. *Psychol. Forsch.*, 24:299–348, 1954.
- [121] A. Thiéry. Über geometrisch optische täuschungen. *Phil. Stud.*, 12:67–126, 1896.
- [122] B. Triggs. Autocalibration and the absolute quadric. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 609–614, Puerto Rico, 1997.
- [123] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-self tv cameras and lenses. *J. Robotics and Automation*, 3(4):323–344, August 1987.
- [124] R. Y. Tsai and R. K. Lenz. A new technique for fully autonomous and efficient 3d robotics hand/eye calibration. *IEEE Trans. Robotics and Automation*, 5(3):345–358, 1989.
- [125] I. van Cruyningen, A. Lozano, and R.K. Hanson. Planar imaging of concentration by planar laser induced fluorescence. *Experiments in Fluids*, 10:41–49, 1990.
- [126] V. Virsu. Tendencies to eye movement, and misperception of curvature, direction and length. *Percept. Psychophys.*, 9:65–72, 1971.
- [127] T. Viéville and O.D. Faugeras. The first order expansion of motion equations in the uncalibrated case. *Computer Vision and Image Understanding*, 64(1):128–146, 1996.

BIBLIOGRAPHY

- [128] H.L.F. von Helmholtz. *Treatise on Physiological Optics*, volume III. Dover, New York, 1962. Translated from the third german edition by J.P.C. Southall.
- [129] J. Warnatz, U. Maas, and R.W. Dibble. *Combustion*. Springer, Berlin, 1996.
- [130] J. Weickert. *Anisotropic Diffusion in Image Processing*. PhD thesis, Universität Kaiserslautern, January 1996.
- [131] J. Weickert. *Anisotropic Diffusion in Image Processing*. Teubner-Verlag, Stuttgart, 1998.
- [132] J. Weickert, B.M. ter Haar Romeny, and M.A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Trans. on Image Proc.*, 7(3):398–410, 1998.
- [133] J. Weickert, K.J. Zuiderveld, B.M. ter Haar Romeny, and W.J. Niessen. Parallel implementation of aos schemes: A fast way of nonlinear diffusion filtering. *Proc. IEEE Int. Conf. on Image Proc.*, 3:396–399, 1997.
- [134] L. Welch. The perception of moving plaids reveals two motion-processing stages. *Nature*, 337:735–737, February 1989.
- [135] H.R. Wilson and J.R. Bergen. A four mechanism model for spatial vision. *Vision Research*, 19:19–32, 1979.
- [136] H.R. Wilson and S.C. Giese. Threshold visibility of frequency gradient patterns. *Vision Research*, 17:1177–1190, 1977.
- [137] A.P. Witkin. Scale-space filtering. In *Proc. 8th Int. Joint Conf. on AI*, pages 1019–1022, Karlsruhe, West Germany, 1983.
- [138] B. Yip, J.K. Lam, M. Winter, and M.B. Long. Time resolved three-dimensional concentration measurements in a gas jet. *Science*, 235:1209–1211, 1987.
- [139] S.M. Zeki. *A Vision of the Brain*. Blackwell, London, 1993.
- [140] Z. Zhang. A flexible new technique for camera calibration. Technical Report MSR-TR-98-71, Microsoft Research, December 1998.
- [141] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Proc. Int. Conf. Computer Vision*, volume 1, pages 666–679, Kerkyra, Greece, September 1999.
- [142] Z. Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

- [143] Z. Zhang, Q.-T. Loung, and O.D. Faugeras. Motion of an uncalibrated stereo rig: Self-calibration and metric reconstruction. *IEEE Trans. Robotics and Automation*, 12(1):103–113, 1996.
- [144] H. Zhuang, Z. S. Roth, and R. Sudhakar. Simultaneous robot/world and tool/flange calibration by solving homogeneous transformation equation of the form $ax = yb$. *IEEE Trans. Robotics and Automation*, 10(4):549–554, 1994.
- [145] A. Zisserman, P. Beardsley, and I. Reid. Metric calibration of a stereo rig. In *Proc. Workshop on Representation of Visual Scenes*, pages 93–100, Cambridge, Massachusetts, USA, June 1995.
- [146] F. Zöllner. Über eine neue art von pseudokopie und ihre beziehungen zu den von plateau und oppel beschriebenen bewegungsphänomenen. *Ann. Phys. Chem.*, 186:500–523, 1860.