Geometric Estimation with Local Affine Frames and Free-form Surfaces

Dissertation

zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
(Dr.-Ing.)
der Technischen Fakultät
der Christian-Albrechts-Universität zu Kiel

Kevin Köser

Kiel
2008

1. Gutachter _____

2. Gutachter _____

3. Gutachter _____

Datum der mündlichen Prüfung _____

Contents

List of Abbreviations

AC	Absolute Conic
CAD	Computer Aided Design
CCD	Charged Coupled Device
DIAC	Dual Image of the Absolute Conic
DLT	Direct Linear Transformation
DOF	Degrees of Freedom
DoG	Difference of Gaussians
EBR	Edge-based Region
EKF	Extended Kalman Filter
FoV	Field of View
GPU	Graphics Processing Unit
IAC	Image of the Absolute Conic
IBCA	Image Brightness Constancy Assumption
IBR	Intensity-based Region
LAF	Local Affine Frame
LDA	Linear Discriminant Analysis
MAD	Mean Absolute Difference
MDA	Multiple Discriminant Analysis
MLE	Maximum Likelihood Estimation
MSER	Maximally Stable Extremal Region
NCC	Normalized Cross Correlation
PCA	Principal Component Analysis
pdf	probability density function
PLI	Pre-image of the line at infinity
PMD	Photonic Mixer Device
RANSAC	Random Sampling Consensus

SAD	Sum of Absolute Differences
SfM	Structure from Motion
SIFT	Scale Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
SSD	Sum of Squared Differences
SVD	Singular Value Decomposition
ToF	Time of Flight
VRML	Virtual Reality Modeling Language

Symbols and Notation

To improve readability and for clarity of the equations, the following fonts, styles, and symbols have been used:

$\mathbb{R}$	the set of the real numbers
$\mathbb{C}$	the set of the complex numbers
$\mathbb{P}^n$	the projective space representing vectors of $\mathbb{R}^n$ and the ideal points
s	a scalar number (real-valued if not denoted otherwise)
$\boldsymbol{x}$	Euclidean vector from $\mathbb{R}^n$
$\mathbf{x}$	element from $\mathbb{P}^n$
R	Matrix acting on elements from $\mathbb{R}^n$
$\mathbf{K}$	Matrix acting on elements from $\mathbb{P}^n$
$\mathit{I}_{n\times n}$	$n \times n$ identity matrix
$\mathit{0}_{n\times n}$	$n \times n$ zero matrix
$\mathbf{0}_n$	zero vector of dimension n
π_∞	plane at infinity
$\mathbf{l}_\infty$	line at infinity
$[\boldsymbol{x}]_\times$	3×3 cross product matrix of $\boldsymbol{x}$
$\forall$	for all
$\exists$	there exist(s)
$\mathcal{I}[\boldsymbol{x}]$	image $\mathcal{I}$'s grey value at $\boldsymbol{x}$
$\mathtt{H}[\boldsymbol{x}]$	function $\mathtt{H}$ acting on $\boldsymbol{x}$
$\frac{\partial \mathtt{f}}{\partial x}\big\vert_b$	derivative of $\mathtt{f}$ with respect to x, evaluated at b
$\underset{[a;b]}{\mathtt{sup}}[\mathtt{f}]$	supremum of $\mathtt{f}[x]$ for $x \in [a;b]$
$\mathtt{det}[\mathit{M}]$	determinant of square matrix M
$\mathtt{trace}[\mathit{M}]$	trace of square matrix M
$\simeq$	equality up to scale, collinearity
$\mathtt{euc}[\mathbf{x}]$	homogeneous to euclidean mapping
$\mathtt{hom}[\boldsymbol{x}]$	euclidean to homogeneous mapping
$\mathtt{vec}[\mathit{M}]$	vectorization of matrix, stacking rows on top of one another
$\mathtt{dim}[\boldsymbol{x}]$	dimension of $\boldsymbol{x}$
$\mathtt{exp}[x]$	exponential function of x

Throughout the document, boldface italic serif letters $\boldsymbol{x}$ will always de-

note Euclidean vectors while boldface upright serif letters $\mathbf{x}$ denote homogeneous vectors. Matrices appear as capital letters, where those acting on Euclidean vectors are denoted by italic font without serifs (as A), while matrices acting on homogeneous vectors are denoted by upright serif letters (as $\mathbf{A}$). Functions are indicated by typewriter font `T` with the argument in square brackets $[arg]$, while matrices are indicated by round brackets.

Chapter 1

Introduction

1.1 Motivation

In the fields of photogrammetry, robotics, and computer vision often feature-based approaches are applied to obtain correspondences between different images. Such features are local image regions with special properties that depend on the type of feature detector used, e.g. corners. During the last ten years there has been tremendous progress in feature-based matching of images taken from significantly different viewpoints. Before, correspondences between unknown images could only be obtained automatically for quite restricted scenarios, e.g. if the camera position change - the baseline - between two images was small (compared to the distance to the imaged objects) and camera orientation and zoom level were approximately equal in the images.

Using the idea of intrinsic scale or shape of a local region feature, correspondence search can nowadays better adapt to the variation of a region between different images. Even in wide-baseline scenarios, i.e. where corresponding regions in two images can look significantly different, e.g. due to perspective effects, scale, shear, or rotation, correspondences can be obtained automatically. Linearly normalizing the local region and then computing robust signatures that tolerate some inevitable inaccuracies allows to handle large parts of this variation. Using these methods it is possible to search the internet for photos similar to a query image [Jegou et al., 2008], to perform automatic panorama creation from hand-held cameras[Brown and Lowe, 2007] or to reconstruct objects or scenes from web-databases, photographed by thousands of different users [Snavely et al., 2006].

Many of these approaches follow three steps: First, detect interesting regions (features) in a query image. Then, from the signature of each local region, find potential correspondences with similar signatures in other im-

ages. Third, geometrically verify correspondences and remove those that are not consistent with the majority, e.g. those that vote for a different camera motion. While there has been tremendous progress in the first two steps, in the geometrical verification and estimation often heuristics are used or they are based solely upon the pure position: In both tasks, the relative rotation, shear or scale between corresponding regions seems often to be seen as an overcome nuisance, and the information it carries has largely been ignored so far.

The main goal of this thesis is therefore to derive a geometric primitive that represents not only the position but also the other geometric properties carried in the local image features developed during the last decade, to propose a theory how these primitives are related given important image transformations and to show how the local region correspondence information can be exploited in obtaining the unknown image relation. This allows interesting applications, e.g. object pose or surface normal estimation from a single feature correspondence. Since not only the position is subject to measurement uncertainty, but also the other parameters of the feature, also a framework for embedding uncertainty is required and the estimation primitive needs to be compared to other existing primitives in geometric estimation. The first part of this thesis is dedicated to these questions.

One possible application of using feature correspondences is to track a camera, i.e. to precisely compute its position and orientation parameters during an image sequence or a video, when the camera is moved. Keeping track of these camera parameters is particularly important in the context of augmented reality, e.g. when virtual 3D objects or graphics are placed into a live broadcast and must appear as if they were in the scene. Recent systems for sports coverage for instance can already display distances in 3D or show important information on-pitch, i.e. it seems as if the information is shown within the scene. If however, this information must be shown continuously and consistently even when the camera rotates or moves, the problem is much harder. The same applies to industrial augmented reality, where assembly instructions are visualized fixed to the workpiece in 3D for the technician wearing a see-through display. In both cases there may be motion present in the scene, e.g. the hands of a technician or moving persons.

A few systems exist that can compute the pose of a camera in real-time, but most require the scene to be set up with special calibrated markers or are prone to drift on long sequences. Both in television and in industrial environments it it desirable to have a camera tracking system, which does not drift even after minutes or hours of tracking and which is flexible enough to be applied outside of a calibrated studio environment and without artificial markers in the scene. The second big contribution of this thesis is

therefore the proposition of such a markerless, drift-free camera tracking system. The idea is that in an offline phase, a free-form surface model of the environment is built up. Particularly around the most useful sparse 2D features from an initial structure from motion approach, the exact surface geometry is reconstructed, allowing a representation well-suited for analysis-by-synthesis methods on the graphics hardware. The reconstructed free-form surface model can then serve as an absolute reference in an online phase, this way preventing drift accumulation. To cope with moving scene content, the system is proposed to be based on a wide-angle camera, e.g. equipped with a fish-eye lens. This way the camera always sees large parts of the static scene and the effects of camera rotation are minor.

Main Contributions

The main contributions of this thesis can be partitioned into three groups:

First, the derivation of a novel primitive for geometric estimation, well suited for state-of-the-art local image features. The model was proposed in [Köser et al., 2008], and in [Köser and Koch, 2008a] it was related to conic correspondences and triplets of points. In this thesis the framework for the primitive is however extended further, e.g. to allow for incorporating uncertainty and statistical testing. Furthermore, an alternative, more physically motivated, interpretation is given. In [Köser and Koch, 2008b] it has been shown how to improve feature correspondences and how the model relates to affine template tracking.

Second, specific problems have been solved by application of the general constraints. Particularly camera pose estimation for a wide range of camera models can be achieved from a single feature correspondence [Köser and Koch, 2008a]. It is also shown how a general homography can be computed from two or more feature correspondences and how the surface normal directly evolves from a single feature correspondence in calibrated cameras. Using the ideas of the previous paragraph, also a minimal parameterization could be found for the conjugate rotation and it could be shown that this special infinite homography has seven degrees of freedom [Köser et al., 2008]. Also algorithms to estimate a conjugate rotation belong to this second group of important contributions.

The third group of contributions is related to the camera tracking system based on free-form surfaces. Here a good scene representation could be shown, suitable for analysis-by-synthesis methods exploiting the graphics hardware, as initially presented in [Köser et al., 2006a], evaluated with respect to model detail in [Köser et al., 2007b] and shown in context of a

large system in [Köser et al., 2007a]. Particularly the absence of drift and the independence of pre-calibrated markers are important. Also the special properties of fish-eye cameras were exploited and a system has been presented that is robust against lighting changes and moving persons although a mainly static scene is assumed.

Structure of the Thesis

The thesis is structured in the following way: In the next section, basic concepts required for understanding the work are presented. Then the previous work on image-based primitives for geometric estimation is discussed. The novel concept of the differential constraints from affine feature correspondences is then derived in chapter 4, including representation, measurement and uncertainty handling. In chapter 5 the derived concepts are applied to several problems of computer vision such as estimating surface normals, homographies or camera poses. Each of the solutions is evaluated and discussed in this chapter. In the second part of this thesis, the focus is changed to free-form surfaces, which provide a better representation for curved three-dimensional structures and are suitable for analysis-by-synthesis methods on the graphics hardware. In chapter 6 a complete system for camera tracking is described, evaluated and discussed, followed by a conclusion in chapter 7. Detailed and longer derivations for certain topics can be found in the appendix.

Chapter 2

Basic Concepts

This chapter provides the required mathematical notations and models upon which this thesis is based.

2.1 Camera Model and Geometry

Here the image creation process is described in terms of geometrical considerations in 3D space.

2.1.1 Projective Geometry and Notation

To simplify the mathematical relations, the notation of projective geometry is used, which embeds a space $\mathbb{R}^n$ into a space of one dimension higher, called $\mathbb{P}^n$. Additionally, this space contains the points which are defined by intersection of two parallel lines, called ideal points. In $\mathbb{P}^3$ all ideal points form the plane at infinity π_∞, while in $\mathbb{P}^2$ the ideal points form the line at infinity (also referred to as $\mathbf{l}_\infty$). Formally, $\mathbb{P}^n$ is defined as

$$\mathbb{P}^n = \left\{ x \in \mathbb{R}^{n+1} | x \neq 0 \right\}. \tag{2.1}$$

Each point $\boldsymbol{x} \in \mathbb{R}^n$ corresponds to an equivalence class depending on some $\lambda \neq 0$ in $\mathbb{P}^n$:

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ \dots \\ x_n \end{pmatrix} \equiv \begin{pmatrix} \lambda x_1 \\ \dots \\ \lambda x_n \\ \lambda \end{pmatrix} = \mathbf{x} \tag{2.2}$$

while the ideal points ($\mathbf{x}_{n+1} = 0$) do not have a representation in $\mathbb{R}^n$.

The left hand vector $\boldsymbol{x}$ is called Euclidean while the right hand side $\mathbf{x}$ is called a homogeneous vector and the 1-based indian-arabic subscripts indicate a specific component of a vector (e.g. the z-component of a 3D vector $\boldsymbol{x}$ would be referred to by x_3). Throughout the document, boldface italic serif letters $\boldsymbol{x}$ will always denote Euclidean vectors while boldface upright serif letters $\mathbf{x}$ denote homogeneous vectors. For matrices serifs are not used, so that matrices acting on Euclidean vectors are denoted as A, while matrices acting on homogeneous vectors are denoted as $\mathbf{A}$.

To switch between the representations and make notation easier two transformations will be defined:

$$\mathtt{hom} : \mathbb{R}^n \to \mathbb{P}^n : \mathtt{hom}\left[\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}\right] = \begin{pmatrix} x_1 \\ \vdots \\ x_n \\ 1 \end{pmatrix} \tag{2.3}$$

$$\mathtt{euc} : \{\mathbf{x} \in \mathbb{P}^n | x_{n+1} \neq 0\} \to \mathbb{R}^n : \mathtt{euc}\left[\begin{pmatrix} x_1 \\ \vdots \\ x_{n+1} \end{pmatrix}\right] = \begin{pmatrix} x_1/x_{n+1} \\ \vdots \\ x_n/x_{n+1} \end{pmatrix} \tag{2.4}$$

As a consequence, an affine transformation or central projection in euclidean space will become linear in projective space. However, the transition from projective to Euclidean space incorporates a division (equation (2.4)). This division can render a linear function in projective space non-linear in Euclidean space.

Since problems in $\mathbb{P}^n$ (e.g. equation systems) can often be solved using methods from $\mathbb{R}^{n+1}$ and vice versa, the above function definitions are used in the remainder of this thesis in an extended sense, i.e. so that $\mathbb{R}^{m+1}$ and $\mathbb{P}^m$ may both appear in domain and codomain and the interpretation - if required at all - is given by the context.

2.1.2 Geometric Objects

A point in an image can be represented by a vector of $\mathbb{R}^2$ or $\mathbb{P}^2$ and a point in 3D space by a vector of $\mathbb{R}^3$ or $\mathbb{P}^3$. Although in this section the homogeneous representation is followed, it will be the case that in later sections the Euclidean notation is sometimes preferable, such that the representation has to be switched. Nevertheless, corresponding letters $\boldsymbol{x}$ and $\mathbf{x}$ refer to the same thing and are related via equations 2.4 and 2.3.

Apart from the point used above, the most interesting geometric entities in this thesis are planes, lines and conics. Like points, lines in the plane can

be represented as 3-vectors in $\mathbb{P}^2$, where all points $\mathbf{x}$ which lie on the line $\mathbf{l}$ fulfill

$$\mathbf{l}^\mathsf{T}\mathbf{x} = 0 \tag{2.5}$$

It is easy to see that the line $\mathbf{l}$ through two points $\mathbf{x}$ and $\mathbf{y}$ is defined by the cross product of the two points:

$$\mathbf{l} = \mathbf{x} \times \mathbf{y} \tag{2.6}$$

which can equivalently be written using the cross product matrix:

$$[\mathbf{x}]_\times = \begin{pmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{pmatrix} \tag{2.7}$$

as

$$\mathbf{l} = [\mathbf{x}]_\times \mathbf{y} \tag{2.8}$$

Points are said to be dual to lines in $\mathbb{P}^2$. In $\mathbb{P}^3$, points are dual to planes and the plane representation $\pi \in \mathbb{P}^3$ is analogue to the line representation in $\mathbb{P}^2$, since all points on the plane must fulfill:

$$\pi^\mathsf{T}\mathbf{x} = 0 \tag{2.9}$$

For planes in Euclidean space the first three components $\boldsymbol{n} = (\pi_1, \pi_2, \pi_3)^\mathsf{T}$ define a vector $\boldsymbol{n}$ orthogonal to the plane. The ratio of the fourth component π_4 and $\boldsymbol{n}$'s norm defines the distance of the plane to the origin. Additionally, the plane at infinity is described as $\pi_\infty = (0, 0, 0, 1)^\mathsf{T}$ and one verifies that $\pi_\infty^\mathsf{T}\mathbf{x} = 0$ for all ideal $\mathbf{x} \in \mathbb{P}^3$.

Apart from the linearly, implicitly defined objects, there are also quadratically, implicitly defined objects, which are interesting for this thesis. In $\mathbb{P}^2$ they are called conics (e.g. ellipses, hyperbolas, ...) and in $\mathbb{P}^3$ quadrics (e.g. ellipsoids,...). Only the conics are introduced here, however, for the quadrics basically the same equations hold in the higher dimensional space. A conic $\mathbf{C}$ is defined as the set of points $\mathbf{x} \in \mathbb{P}^2$ for which the following equation is fulfilled:

$$\mathbf{x}^\mathsf{T}\mathbf{C}\mathbf{x} = 0 \tag{2.10}$$

The conic's homogeneous representation is a symmetric 3×3 matrix $\mathbf{C}$ with only five degrees of freedom because an overall scale of the matrix does not change the above equality and thus not the set of points. According to their eigenvalue structure, conics can be classified into different *proper conics* $\mathbf{C}_p$ (e.g. ellipse, parabola, hyperbola) with full rank

$$\mathtt{rank}\,[\mathbf{C}_p] = 3 \tag{2.11}$$

and *improper conics* $\mathbf{C}_i$ (e.g. single line, two lines, single point) with a rank-deficiency. Since the improper conics have rank less than three, their determinant must vanish:

$$\mathtt{det}\left[\mathbf{C}_i\right] = 0 \tag{2.12}$$

Conics that do not consist of any real points are called *virtual*, such as the *absolute conic* represented by the identity matrix. Apart from the point conics $\mathbf{C}$ considered so far, there are also line conics $\mathbf{C}^*$, which represent the set of lines tangential to the point conic $\mathbf{C}$. If $\mathbf{C}$ has full rank, then the dual conic $\mathbf{C}^*$ can be obtained by

$$\mathbf{C}^* = \mathbf{C}^{-1} \tag{2.13}$$

In $\mathbb{P}^3$ analogous concepts exists, and the quadratic equation in general implies a surface here, which is called quadric $\mathbf{Q}$. Each point on the surface must fulfill the quadric equation and the dual to this point quadric $\mathbf{Q}$ is called the plane quadric, or the dual quadric, $\mathbf{Q}^*$. For a more detailed presentation of conic properties and of projective geometry, the reader is referred to [Hartley and Zisserman, 2004].

2.1.3 Rotation

Rotations are very important operations in Euclidean space which preserve the norm of a vector and the angle between two vectors before and after the transformation. In $\mathbb{R}^2$ a rotation R_{2D} of a vector $\boldsymbol{v}_{2D}$ by an angle ϕ around the origin can be represented by the linear Euclidean matrix operation:

$$\boldsymbol{v}'_{2D} = R_{2D}\,\boldsymbol{v}_{2D} = \begin{pmatrix} \mathtt{cos}\left[\phi\right] & -\mathtt{sin}\left[\phi\right] \\ \mathtt{sin}\left[\phi\right] & \mathtt{cos}\left[\phi\right] \end{pmatrix} \boldsymbol{v}_{2D} \tag{2.14}$$

Therefore, a scalar ϕ is sufficient to uniquely determine such a rotation. In $\mathbb{R}^3$ a rotation is always described by an angle $\phi \in \mathbb{R}$ around an axis $\boldsymbol{t} \in \mathbb{R}^3$ with $\parallel \boldsymbol{t} \parallel = 1$ through the origin, leading to the Rodrigues formula for a rotation matrix ([Hartley and Zisserman, 2004], pp. 584):

$$\mathtt{R}_{\mathtt{3D}}\left[\boldsymbol{t}, \phi\right] = I_{3\times 3} + \mathtt{sin}\left[\phi\right]\,[\boldsymbol{t}]_\times + (1 - \mathtt{cos}\left[\phi\right])[\boldsymbol{t}]_\times^2 \tag{2.15}$$

A 3D rotation has eigenvalues $\{1, e^{i\phi}, e^{-i\phi}\}$, where the unit eigenvalue corresponds to the rotation axis, which is an eigenvector. All points on the rotation axis are fixpoints, which are unchanged by the rotation. A rotation in 3D is uniquely determined by the rotation axis and the rotation angle, where the rotation axis has only two degrees of freedom and can be parameterized by two angles (as in spherical coordinates). 3D rotation allows for different parameterizations, such as

1. normalized axis and angle (4 parameters, axis has norm 1)
2. normalized axis $\times$ angle
3. quaternion (4 parameters, quaternion has norm 1)
4. Euler angles (3 angles, world-fixed or object-fixed, gimbal lock problem)
5. 3×3 rotation matrix (9 parameters, rows and columns orthonormal and matrix determinant 1)
6. first two columns of 3×3 rotation matrix (6 parameters, columns orthonormal)
7. ...

Rotation matrices in 2D and 3D are always orthonormal with determinant 1. Within the above parameterizations all have their advantages and drawbacks and a suitable parameterization is application specific. For instance, the over-parameterizations with more than three parameters usually act in a simpler way on the points to be rotated than the minimal parameterizations with only three parameters. When rotating points, usually the quaternion or rotation matrix representation is used. On the other hand there exist non-linear constraints between the parameters in the over-parameterizations. When estimating the parameters these constraints have to be taken into account to guarantee a "valid" rotation. Instead of estimating a rotation matrix, it is often more convenient to estimate a minimal or small set of parameters.

2.1.4 Camera Model

To derive the camera model used in this thesis, the ideal pinhole camera is inspected first. In such a camera the light rays that fall from the scene onto the image plane all go through the pinhole. The pinhole is therefore also called the center of projection or simply the camera center. For simplicity, it is assumed that this center is the origin now, that the normal of the image plane is the z-axis and that the camera can see only objects which have positive z-components. The optical axis is defined as the ray orthogonal on the image plane which goes through the camera center. The image plane shall be at z=1 with an image coordinate system attached to this plane coincident with the x- and y-axes of the world coordinate system. Then, a Euclidean 3D world point $\boldsymbol{x}^W$ is mapped to a Euclidean 2D image point $\boldsymbol{x}^I$ by

$$\boldsymbol{x}^I = \mathsf{euc}\left[\boldsymbol{x}^W\right] \tag{2.16}$$

If the camera is now moved away from the origin to position $\boldsymbol{C}$ in 3D Euclidean space and rotated by a rotation matrix R, a rigid transformation has to be incorporated when projecting points of the world with this camera. First, the points are transferred into the camera coordinate system and then projected. In projective space such a pinhole camera can be modeled by a 3×4 projection matrix $\mathbf{P} = (R^\mathsf{T} | -R^\mathsf{T}\boldsymbol{C})$. The translation in $\mathbb{R}^3$ becomes linear in $\mathbb{P}^3$, so that now homogeneous 4-vectors $\mathbf{x}^W \in \mathbb{P}^3$ are projected onto homogeneous 3-vectors $\mathbf{x}^I \in \mathbb{P}^2$.

$$\mathbf{x}^I \simeq (R^\mathsf{T} | -R^\mathsf{T}\boldsymbol{C})\mathbf{x}^W \tag{2.17}$$

Here, the $\simeq$-sign means that these vectors are equal up to an unknown scale, representing the equivalence class (cf. to equation (2.2)). If the projection $\mathbf{x}$ is not an ideal point, the 2D Euclidean image coordinates can be obtained:

$$\boldsymbol{x}^I = \mathtt{euc}\left[\mathbf{x}^I\right] = \mathtt{euc}\left[(R^\mathsf{T} | -R^\mathsf{T}\boldsymbol{C})\mathbf{x}^W\right] \tag{2.18}$$

If the 2D coordinate system on the image plane is sheared, scaled or displaced compared to the world coordinate system, such effects can be encoded in a calibration matrix $\mathbf{K}$(cf. to [Hartley and Zisserman, 2004]). This matrix $\mathbf{K}$ is an upper triangular matrix and holds the principal point $(c_x, c_y)^\mathsf{T}$, where the optical axis intersects the image plane in image coordinates, the focal length f, a skew parameter s and the aspect ratio a:

$$\mathbf{K} = \begin{pmatrix} f & s & c_x \\ 0 & af & c_y \\ 0 & 0 & 1 \end{pmatrix} \tag{2.19}$$

Since $\mathbf{K}$ is an affine transform of the 2D image plane, its application is a linear operation in projective space and the homogeneous 3D world point $\mathbf{x}^W$ would then be mapped to a homogeneous 2D image point $\mathbf{x}^I$ by

$$\mathbf{x}^I \simeq \mathbf{K}(R^\mathsf{T} | -R^\mathsf{T}\boldsymbol{C})\mathbf{x}^W = \mathbf{P}\mathbf{x}^W \tag{2.20}$$

where $\mathbf{P}$ is called the projection matrix. Writing this completely in Euclidean coordinates, this looks like

$$\boldsymbol{x}^I = \mathtt{euc}\left[\mathbf{x}^I\right] = \mathtt{euc}\left[\mathbf{K}(R^\mathsf{T} | -R^\mathsf{T}\boldsymbol{C})\,\mathtt{hom}\left[\boldsymbol{x}^W\right]\right] \tag{2.21}$$

The simple relation of equation (2.20) can also be extended to other objects such as plane quadrics $\mathbf{Q}^*$ in space, which are mapped to line conics $\mathbf{C}^*$ in the image using the projection matrix $\mathbf{P}$:

$$\mathbf{C}^* \simeq \mathbf{P}\mathbf{Q}^*\mathbf{P}^\mathsf{T} \tag{2.22}$$

To obtain the relations for point conics or quadrics, simply the dual representation has to be used.

Unfortunately, in real cameras the pinhole cannot be infinitely small because no light would pass through. Therefore, real cameras typically contain lenses, which collect and focus the light onto the image plane. Such lenses however do not strictly follow the pinhole model and particularly for wide-angle lenses, geometric distortion can be observed. Different extended models have been proposed to describe the lens behavior, of which exemplarily the polynomial radial distortion model[Heikkilä and Silvén, 1997] is shown here. In this model the distortion is compensated phenomenologically from the observation that the distortion increases or decreases symmetrically from a center of distortion in radial direction. This is known as barrel or cushion distortion. A polynomial of degree two, four or even higher is used to correct for this distortion.

In fact there are also models for tangential [Heikkilä and Silvén, 1997] and other distortion, models for fish-eye (or equidistant) projection [Micusik, 2004, Scaramuzza et al., 2006b], for equiangular[Fleck, 1995] and other projection models [Perwass and Sommer, 2006, Tsai, 1987, Geyer and Daniilidis, 2001]. In most parts of this thesis no particular of these models is required. Rather, when talking about calibrated cameras, it is assumed that for a position in the image coordinate system the corresponding ray in the camera coordinate system can be obtained, that this mapping between image positions and rays is sufficiently smooth to be locally linearizable and invertible. It is also assumed that the camera can be modeled well by a single center of projection, which is even true for certain omnidirectional cameras.

In this case the intrinsic parameters can be calibrated beforehand and compensated, allowing to reason about rays in space. Image uncertainties can be propagated to ray uncertainties. If the intrinsic parameters are known, the remaining degrees of freedom of the camera are only in the external camera parameters, the position and orientation (the pose), of the camera.

2.1.5 Pose

The parameters of a calibrated camera as described in section 2.1.4 can be separated into the external parameters (the position and orientation in 3D space), which will be called the pose, and the internal parameters (like focal length, principal point, ...), which will be called the calibration. The pose has six degrees of freedom (three for translation and three for orientation) and can be parameterized in different ways (compare section 2.1.3 for different rotation parameterizations). The pose of a camera defines a transformation $\mathbf{T}_{c1}$ between a point's coordinates $\mathbf{X}^w \in \mathbb{P}^3$ in the world coordinate system

and $\mathbf{X}^{c1} \in \mathbb{P}^3$ in the camera coordinate system:

$$\mathbf{X}^w = \mathbf{T}_{c1}\mathbf{X}^{c1} = \begin{pmatrix} R & \boldsymbol{C} \\ & 1 \end{pmatrix} \mathbf{X}^{c1} \tag{2.23}$$

$$\mathbf{X}^{c1} = \mathbf{T}_{c1}^{-1}\mathbf{X}^w = \begin{pmatrix} R^\mathsf{T} & -R^\mathsf{T}\boldsymbol{C} \\ & 1 \end{pmatrix} \mathbf{X}^w \tag{2.24}$$

If a second camera is given by its pose transformation T_{c2} in the world, its pose transformation can also be represented in the first camera's coordinate system:

$$\mathbf{T}_{c2}^{c1} = \mathbf{T}_{c1}^{-1}\mathbf{T}_{c2} \tag{2.25}$$

The column vectors of $\mathbf{T}_{c2}$ can simply be transformed from global coordinates to local coordinates in the first camera as ordinary points. Consequently, in the first camera's coordinate system the first camera itself is at the canonic pose (represented by the identity transform $I_{4\times4}$):

$$\mathbf{T}_{c1}^{c1} = \mathbf{T}_{c1}^{-1}\mathbf{T}_{c1} = I_{4\times4} \tag{2.26}$$

2.2 Photometric Image Creation

So far only geometric considerations have been applied. The image content, the color or the grey value has not been defined so far.

2.2.1 Plenoptic Function

In [Adelson and Bergen, 1991] Adelson and Bergen introduced the plenoptic function to model what can be seen in the world. They state that objects in the world reflect light from different sources in various directions. These light rays do not interact and the plenoptic function simply represents the superpositioned intensity of all incoming light rays for each wavelength, position in space, each viewing direction and possibly also time.

$$\mathtt{Pl}\,[\theta, \phi, \lambda, t, x, y, z] : \mathbb{R}^7 \rightarrow \mathbb{R} \tag{2.27}$$

Therefore this is a very high-dimensional function and rather a theoretical construct than a practical representation of the appearance of a scene or the world. However, this way it models the visual effects that can be seen in the real world and allows for reasoning about visual phenomena.

2.2.2 Piecewise Continuous Surfaces

A more compact model often used is that of a piecewise continuous surface model. In this case a scene is not described by the set of all possible light rays but rather by all objects in the scene. If the light sources and the surface properties of the objects are known, the plenoptic function is also determined. Often the simplifying assumptions of constant lights and Lambertian surfaces are made. In this case the image brightness constancy assumption (IBCA) can be made, which states that corresponding points in images must have the same intensity. Constant lighting means that the illumination does not change over time, which may be violated if clouds move in front of the sun. Lambertian surfaces (cf. also [Jähne, 2005], p. 191) are perfectly diffuse so that points on such a surface look the same from all viewing directions. Such assumptions are often good for non-shiny objects, but they are not valid e.g. for highly reflective or partially transparent materials as metallic paints or water.

2.2.3 Camera Hardware, CCD

The cameras used for the experiments in this thesis use CCD sensors. Therefore the image creation process is described here for this camera type, although the theoretical and geometrical considerations apply also to other cameras. The image plane in a CCD camera is a finite grid of approximately rectangular CCD elements. The intersection of the optical axis with this grid is called the principal point and the ratio of width and height of one such CCD element is called the aspect ratio. If the grid-axes are orthogonal, this results in a zero skew, otherwise the skew is proportional to the scalar product of the axes (see also equation (2.19)). When light rays go through the pinhole of the camera and hit the sensor, the electro-magnetic power is integrated for some time (the shutter time) and results in charge of the CCD element.

It is assumed in this thesis that the lens system and the spatial integration across a CCD cell actually provides a low pass filter that suppresses high frequency components so that no aliasing will occur when the image signal is "sampled" at the sensor plane. If too much light energy falls into a cell and the CCD becomes saturated, charge can flow to neighboring CCD elements and can cause blooming effects, in the worst case a whole line can get saturated by this phenomenon resulting in a disturbing white line in the image (e.g. when the sun is in the image).

In the last step, the charge is read out, possibly amplified and digitized. The resulting intensity value is then stored in the image. The range from

the minimum observable light energy to the energy that saturates a CCD is called the dynamic range of the chip. More details about image creation using CCD chips can be found e.g. in [Jähne, 2005, p. 22].

2.2.4 Lens Effects

In the ideal pinhole camera model the camera center is infinitely small - a point. For light to pass through in real cameras, however, this pinhole must have a finite size, the aperture. The larger the aperture is, the more light can be measured on the sensor and the better the signal-to-noise ratio can become for the image. Unfortunately, with increasing aperture, the image would become more and more blurred because different light rays can bypass the pinhole in parallel and violate the camera model. To avoid this blurring and to collect the light, lenses are used in digital cameras, which themselves cause several deviations from the ideal model:

Focus and Photometric Distortion

Lenses collect light rays traveling through different propagation paths from a point in space to the lens and focus these onto a point behind the lens. For objects at a certain distance this point is on the chip and the object is in focus. Objects at other distances are out of focus resulting in an unsharp image of the object. In this thesis it is assumed that the lens fits the scene distance so well that such defocus distortions can be neglected. Also different wavelengths behave differently when passing through the lens, which is known as chromatic aberration and which can cause deviations from the ideal model. These effects are also considered neglectable in this thesis.

Reflection Effects

Real lenses, in particular wide angle lenses and lens systems suffer from within-lens reflections, which can create artificial structures e.g. such known as Newton's rings in the image. These effects are not modeled here, but they can be handled on a higher level as general model violations.

Vignette Effects

Due to the physical properties of the lens, it typically collects more photons near the optical axis, while the light bundles in the outer lens regions are sparser. This way a white wall does not appear with constant intensity but the intensity is typically lower towards the image border in a radially-symmetric fashion. This gives the impression of a vignette and is thus called

the vignette effect. The vignette effect can be calibrated beforehand and the image can be approximately de-vignetted before operation, so that this effect is assumed to be compensated in the following.

2.2.5 Brightness Models

A model often applied in correspondence estimation is the image brightness constancy assumption. It basically states that corresponding points in two images have the same intensity value (cf. to [Jähne, 2005, p. 425]). This is often a good assumption for controlled indoor scenes with Lambertian surfaces. In scenes with very bright and very dark parts however, the aperture and shutter of the camera are often steered dynamically so that the observed scene parts fit the dynamic range of the CCD well. Also changing illumination (e.g. due to weather conditions outdoor) and non-Lambertian surfaces can violate this assumption.

One of the simplest models to compensate for such effects is to relax the strict constancy assumption to a local affine brightness change, i.e. different intensity values of corresponding points are explained by a scale and an offset, which are constant for a whole region. Before matching regions, the intensities of each region are transformed so that their mean is zero and their standard deviation equals one. This way the matching process is invariant under affine brightness changes between the images. A similar idea has already been applied for a long time when matching with normalized cross correlation (cf. e.g. to [Lewis, 1995]).

Another strategy is to assume a model for the brightness change and estimate the parameters of the model from the image data as discussed in [Baker et al., 2003]. For instance, instead of requiring each point in a region to have the same grey value in any two images, it is assumed that there exists a constant scale a and offset o, such that the intensity in a template $\mathcal{T}$ can be transformed into the intensity of some second image $\mathcal{I}$, for all corresponding points in a region.

$$\mathcal{I}\left[\boldsymbol{x}^{I2}\right] = a\mathcal{T}\left[\boldsymbol{x}^{I1}\right] + o \tag{2.28}$$

Or, if the model contains the (relative) photometric camera parameters [Kim et al., 2007], the region may even be the whole image.

2.3 Relations between Local Regions in a Scene

This thesis deals with the estimation of the relations between views of a scene and the scene. The important relations are introduced in the following sections and their properties are described. All of these relations are parameter-

ized, i.e. the type of transformation is fixed but the actual function depends on some parameters. Each of these transformations can be parameterized by a minimal set of parameters and the number of these parameters is called degrees of freedom (DOF). Such minimal parameterizations are important concepts in estimation, because not only the parameters can be estimated but also their uncertainty, providing a measure of quality for the parameter estimate. Using more than the required number of parameters (over-parameterization), there exist constraints between the parameters, which must be considered (compare also 2.1.3). Using over-parameterizations, the concept of uncertainty estimation requires further effort and when the inter-parameter relations are not known or not enforced more data than intrinsically necessary is required to estimate a transformation.

In the next sections the most important transforms for this thesis are described starting from the simplest 2D transforms.

2.3.1 Displacement, Euclidean, Similarity and Affine Transformations

One of the simplest transformations between two images is the displacement, where a transformed point is obtained by adding an offset vector $\boldsymbol{t}$ to the original point. A slightly more powerful transform is the Euclidean transform, which additionally allows a rotation of an angle α, but which still preserves Euclidean distances between points. The similarity transform additionally allows for an isotropic scale λ of coordinates but still preserves shape so that circles stay circles. All of these transforms can be expressed by the following equation, where for the Euclidean transform λ is fixed at 1 and for the displacement additionally α is fixed at zero.

$$\mathbf{x}^{I2} = \begin{pmatrix} \lambda\cos[\alpha] & -\lambda\sin[\alpha] & \boldsymbol{t} \\ \lambda\sin[\alpha] & \lambda\cos[\alpha] & \\ \mathbf{0}_2^\mathsf{T} & & 1 \end{pmatrix} \mathbf{x}^{I1} \qquad \mathbf{x}^{I1}, \mathbf{x}^{I2} \in \mathbb{P}^2 \tag{2.29}$$

Consequently, the displacement has two DOF, the Euclidean transform has three DOF and the similarity transform has four DOF.

A full linear transform with subsequent translation is called affine transform and has six DOF. Compared to the similarity transform it allows for anisotropic stretch of some magnitude and in some direction (shear). An affine transform for instance relates two ideal perspective images when only the internal camera parameters $\mathbf{K}_i$(see equation (2.19)) differ between these images. The affine transform can be separated into linear stretch and dis-

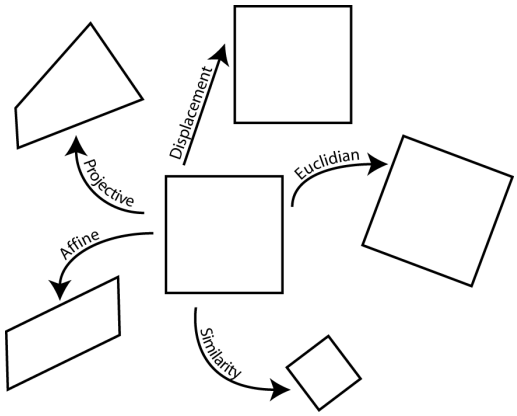

Figure 2.1: A clockwise hierarchy of transformations

placement and can efficiently be expressed in homogeneous coordinates as

$$\mathbf{x}^{I2} = \begin{pmatrix} \boldsymbol{a}_1^\mathsf{T} & \boldsymbol{t} \\ \boldsymbol{a}_2^\mathsf{T} & \\ \mathbf{0}_2^\mathsf{T} & 1 \end{pmatrix} \mathbf{x}^{I1} \tag{2.30}$$

In Euclidean coordinates this reads as

$$\boldsymbol{x}^{I2} = \mathtt{euc}\left[\begin{pmatrix} \boldsymbol{a}_1^\mathsf{T} & \boldsymbol{t} \\ \boldsymbol{a}_2^\mathsf{T} & \\ \mathbf{0}_2^\mathsf{T} & 1 \end{pmatrix} \mathtt{hom}\left[\boldsymbol{x}^{I1}\right]\right] \tag{2.31}$$

$$= \begin{pmatrix} \boldsymbol{a}_1^\mathsf{T} \\ \boldsymbol{a}_2^\mathsf{T} \end{pmatrix} \boldsymbol{x}^{I1} + \boldsymbol{t} \tag{2.32}$$

Affine transforms are also important because they cover the zero and first order parts of a Taylor series (see section A.1). Therefore they can be used to locally linearize non-linear functions. Also the displacement, the Euclidean and the similarity transform are affine transforms.

Jacobian for Transformations of the 2D Euclidean Plane

A very important aspect of a transformation in this thesis is its Jacobian, i.e. the matrix of its first partial derivatives with respect to Euclidean 2D position. Intuitively, when a small step is taken in x-direction in one image, the first column of the Jacobian at this position encodes how large a step is and in what direction it will be taken in the other image under the transformation. The second column of the Jacobian states the same for a small step in y-direction. The Jacobian therefore encodes local magnification, shear and

rotation. In computer graphics, the Jacobian of a texture transformation at a position is therefore called the *footprint* [Chen et al., 2004] and is important for anti-aliasing e.g. in projective texture mapping[Heckbert, 1989].

The Jacobian of the above affine transform in Euclidean space is

$$\frac{\partial \boldsymbol{x}^{I2}}{\partial \boldsymbol{x}^{I1}} = \begin{pmatrix} \boldsymbol{a}_1^\mathsf{T} \\ \boldsymbol{a}_2^\mathsf{T} \end{pmatrix} \tag{2.33}$$

and therefore constant in the whole Euclidean plane.

2.3.2 General Homography

A projective transform - or homography - is a linear mapping in projective space. Section A.2 characterizes 1D homographies in detail, which provides valuable insights into this type of transformation (curve sketching: poles, critical points, limits, ...) and what "linear in projective space" means in Euclidean space. Since 2D images are the basis for this thesis, in the following only homographies in $\mathbb{P}^2$ are considered:

$$\mathbf{x}^{I2} \simeq \mathbf{H}\mathbf{x}^{I1} \tag{2.34}$$

Here, $\mathbf{H}$ is a 3×3 matrix with nine entries:

$$\mathbf{H} = \begin{pmatrix} \boldsymbol{h}_1^\mathsf{T} & \boldsymbol{t} \\ \boldsymbol{h}_2^\mathsf{T} & \\ \boldsymbol{h}_3^\mathsf{T} & \lambda \end{pmatrix} \qquad \boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3, \boldsymbol{t} \in \mathbb{R}^2, \quad \lambda \in \mathbb{R} \tag{2.35}$$

Since $\mathbf{H}$ acts in projective space, it has only eight degrees of freedom and is equivalent to all other homographies, whose matrix representation equals $\mathbf{H}$ up to a non-zero scaling factor. Since $\mathbf{0}_3$ is non an element of projective space $\mathbb{P}^2$, all points that are algebraically mapped to $\mathbf{0}_3$ must be excluded from the domain. Regular 3×3 matrices, which have full rank, map only $\mathbf{0}_3$ to $\mathbf{0}_3$, so that regular homographies are defined for all elements from $\mathbb{P}^2$. The last column of such a homography is the image of the origin, i.e. the point where the origin is mapped to. The last line on the other hand is the pre-image of the line at infinity $\mathbf{l}_\infty$, i.e. the line which is mapped to infinity. For affine mappings, i.e. homographies which do not change $\mathbf{l}_\infty$, the last row is therefore already $\mathbf{l}_\infty$.

When the homography mapping is regarded in Euclidean space (excluding the ideal points), the corresponding function $\mathtt{H}$ is nonlinear (as explained in A.2):

$$\boldsymbol{x}^{I2} = \mathtt{H}\left[\boldsymbol{x}^{I1}\right] = \mathtt{euc}\left[\mathbf{H}\,\mathtt{hom}\left[\boldsymbol{x}^{I1}\right]\right] \tag{2.36}$$

Homographies preserve collinearity, i.e. if three points are collinear in one image and the image is mapped with a homography, then a line can be found on which all of the three points lie. General homographies are often used to describe mappings between images or a scene plane and an image in uncalibrated settings, i.e. when the internal camera parameters are unknown. When something is known about the camera or the setting, often specialized homographies can be used that form only a subset of all possible homographies. Therefore they depend on fewer parameters and can be estimated with more redundancy or fewer data. Often their parameterization directly provides a geometrical interpretation. The following subsections give an overview about some of these homographies that will be used in this thesis:

In detail, the inspected homographies describe relations between planes, e.g. when an uncalibrated pinhole camera is only rotated (conjugate rotation, mapping across the plane at infinity), the mapping between two planes in Euclidean space (perspectivity), or when two pinhole views observe only a single plane.

Parameterization

The simplest parameterization of a general homography is probably using the nine entries of the 3×3 matrix. This way, all possible homographies can be reached using the parameters and there is no singularity. However, since two matrices that differ only by a nonzero scaling factor define the same transformation, this parameterization is redundant and uses more parameters than necessary. Instead, it is possible to use only the ratio of the eight other matrix entries to the lower right matrix value:

$$p_{\mathtt{H}}(\mathbf{H}) = \mathtt{euc}\left[\mathtt{vec}\left[\mathbf{H}\right]\right] \in \mathbb{R}^8 \tag{2.37}$$

This parameterization is only defined if $\mathbf{H}_{33} \neq 0$, i.e. when the origin is not mapped to the line at infinity. This assumption is trivially true when the origin in one image maps to a finite position in another image when applying $\mathbf{H}$. Other parameterizations use the images of four defined points in an image (e.g. the corners) or parameterize relative to the matrix entry with the largest absolute value, which is unlikely to become zero if the parameters are changed only a little, e.g. in local optimization.

Jacobian

If $\mathtt{H}$ is considered such that the 3×3 homography matrix $\mathbf{H}$ acts as a mapping from $\mathbb{R}^2$ to $\mathbb{R}^2$, the homogenization must be taken into account and the

mapping becomes non-linear:

$$\mathtt{H} : \mathbb{R}^2 \to \mathbb{R}^2 : \mathtt{H}\left[\boldsymbol{x}\right] = \mathtt{euc}\left[\mathbf{H}\,\mathtt{hom}\left[\boldsymbol{x}\right]\right] = \begin{pmatrix} \frac{\boldsymbol{h}_1^\mathsf{T}\boldsymbol{x}+t_x}{\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda} \\ \frac{\boldsymbol{h}_2^\mathsf{T}\boldsymbol{x}+t_y}{\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda} \end{pmatrix} \tag{2.38}$$

Here $\boldsymbol{t}$ hat been substituted by $(t_x \;\; t_y)^\mathsf{T}$. The Jacobian is not constant, but a function of $\boldsymbol{x}$:

$$\frac{\partial \mathtt{H}}{\partial \boldsymbol{x}}(\boldsymbol{x}) = \frac{1}{(\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda)^2} \begin{pmatrix} \boldsymbol{h}_1^\mathsf{T}(\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda) - \boldsymbol{h}_3^\mathsf{T}(\boldsymbol{h}_1^\mathsf{T}\boldsymbol{x}+t_x) \\ \boldsymbol{h}_2^\mathsf{T}(\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda) - \boldsymbol{h}_3^\mathsf{T}(\boldsymbol{h}_2^\mathsf{T}\boldsymbol{x}+t_y) \end{pmatrix} \tag{2.39}$$

It can however easily be seen that when $\boldsymbol{h}_3$ is the zero vector and $\lambda = 1$, $\mathtt{H}$ is purely an affine transform and its Jacobian is constant (regarding $\boldsymbol{x}$) and consists solely of $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$.

2.3.3 Perspectivity

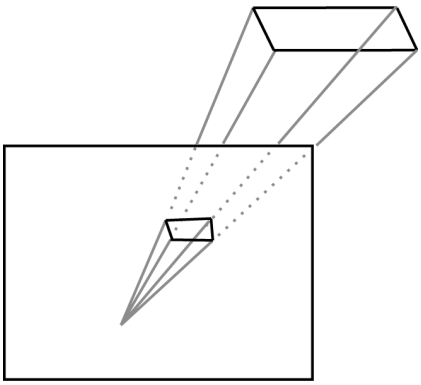

Figure 2.2: A Perspectivity maps between two planes in Euclidean Space

A 2D perspectivity is a special kind of homography (cf. also to [Hartley and Zisserman, 2004], pp. 34), which has only six degrees of freedom and which is particularly important for mappings between planes in Euclidean space. A locally planar geometry at the origin of 3D space is assumed to face into z-direction and to have x, y-coordinates attached to it, which coincide with the x, y coordinates in 3D space. If a perspective pinhole camera is now moved to position $\boldsymbol{C}$ with orientation R (which has rows $\boldsymbol{r}_i^\mathsf{T}$) and with internal camera calibration $\mathbf{K}$, a point $\mathbf{p}^S$ in space is mapped to an image point $\mathbf{p}^I$ by the camera as follows(cf. to [Hartley and Zisserman, 2004, p. 157] for details):

$$\mathbf{p}^I \simeq \mathbf{K}(R^\mathsf{T} | -R^\mathsf{T}\boldsymbol{C})\mathbf{p}^S \tag{2.40}$$

In the following $\mathbf{K} = I_{3\times3}$ is assumed, i.e. a calibrated camera (see also section 2.1.4) and R^T is replaced by its columns (i.e. the rows of R):

$$\mathbf{p}^I \simeq \begin{pmatrix} \boldsymbol{r_1} & \boldsymbol{r_2} & \boldsymbol{r_3} & -R^\mathsf{T}\boldsymbol{C} \end{pmatrix} \mathbf{p}^S \tag{2.41}$$

Now have a look at the points on the $z = 0$ plane to derive the perspectivity:

$$\mathbf{p}^I \simeq (\boldsymbol{r}_1 \;\; \boldsymbol{r}_2 \;\; \boldsymbol{r}_3 \;\; -R^\mathsf{T}\boldsymbol{C})(x\; y\; 0\; 1)^\mathsf{T} = (\boldsymbol{r}_1\; \boldsymbol{r}_2 \;\; -R^\mathsf{T}\boldsymbol{C})(x\; y\; 1)^\mathsf{T} \tag{2.42}$$

$$\simeq \begin{pmatrix} & & t_1 \\ \tilde{\boldsymbol{r_1}} & \tilde{\boldsymbol{r_2}} & t_2 \\ & & 1 \end{pmatrix} (x\; y\; 1)^\mathsf{T} = \mathbf{H}\,\mathbf{p}^P \tag{2.43}$$

Here $\mathbf{p}^P$ are homogeneous points in the plane coordinate system, which are mapped by $\mathbf{H}$ into the image, where $\mathbf{H}$ depends only on the pose of the camera. The six DOF may be parameterized in various ways, e.g. in the same way as a camera pose or as a homography with additional constraints.

2.3.4 Homography induced by a Scene Plane

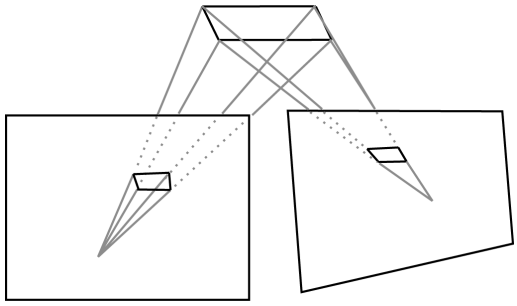

Figure 2.3: Two perspective images, which observe the same scene plane, are related by the homography induced by the plane.

If two cameras observe the same scene plane, all points that lie on this plane transform with a homography from one image to the other. It is assumed that the two cameras are given by

$$\mathbf{P}_1 = \mathbf{K}_1(I_{3\times3}|\mathbf{0}_3) \tag{2.44}$$

$$\mathbf{P}_2 = \mathbf{K}_2(R_2^\mathsf{T}|-\boldsymbol{t}) \tag{2.45}$$

Now the homography $\mathbf{H}_\pi$ between the two images induced by a plane π at a 3D point $\boldsymbol{s}$ in Euclidean space with normal $\boldsymbol{n}$ is (cf. to [Molton et al., 2004]):

$$\mathbf{H}_\pi = \mathbf{K}_2\left(\boldsymbol{s}^\mathsf{T}\boldsymbol{n}R_2^\mathsf{T} - R_2^\mathsf{T}\boldsymbol{t}\boldsymbol{n}^\mathsf{T}\right)\mathbf{K}_1^{-1} \tag{2.46}$$

The plane at infinity cannot be decomposed using a Euclidean point and a normal. In case this plane is used for mapping, the homography depends solely on the relative camera rotation and the intrinsic parameters as will be shown in the next section.

2.3.5 Infinite Homography and Conjugate Rotation

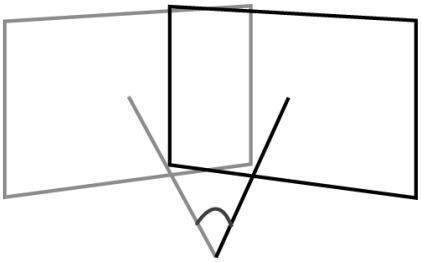

Figure 2.4: Two ideal pinhole images are related by an infinite homography if the camera center does not change.

The infinite homography $\mathtt{H}_\infty$ maps points between two images that have the same camera center or when the corresponding 3D point is infinitely far away.

$$\mathbf{H}_\infty = \mathbf{K}_2 R \mathbf{K}_1^{-1} \tag{2.47}$$

This is the limiting case for the general mapping of the previous section when the plane moves infinitely far away. It is an important concept in projective geometry in general [Hartley and Zisserman, 2004], particularly in panoramic image mosaicking [Brown and Lowe, 2007, Brown et al., 2007], self-calibration [Hartley, 1997a, 1994] or when dealing with pan-tilt-cameras [Capel and Zisserman, 1998]. If the camera calibration is constant ($\mathbf{K}_1 = \mathbf{K}_2$) $\mathbf{H}_\infty$ is algebraically a scaled conjugate rotation [Pollefeys and van Gool, 1999], i.e. it has the same eigenvalue structure as a scaled rotation matrix (cf. to section 2.1.3 for rotation). In [Köser et al., 2008] it has been shown that the conjugate rotation has seven DOF and a minimal parameterization has been proposed, which will be derived in more detail in section 5.2.

Chapter 3

Image-based Estimation Primitives in the Literature

Since a main contribution of this thesis is the introduction and application of a geometric primitive for local region features, this section reviews the most important primitives for geometric estimation used so far. Existing estimation approaches for specific relations are then reviewed in the specific section later in chapter 5.

Many people in photogrammetry, computer vision and other research fields have been dealing with correspondences to obtain view relations or scene information. Among the various types of correspondences, there are symmetric types like image-to-image correspondences, where both matching parts have the same geometric representation (e.g. 2D image positions for homography estimation) and asymmetric correspondences that relate points with lines or ellipsoids with ellipses.

The set of methods to estimate relations like homography or camera pose can be divided roughly into global and local methods. Global methods use the whole or a large portion of the image to obtain the transformation. This can even work if no interesting local structures are present or if an image contains repeated patterns or a significant amount of noise, since they exploit the whole image data. On the other hand, global methods are usually prone to local distortions, such as occlusions and dynamics, reflections or appearance changes. Consequently, the number of applications where an explicit global transformation warping one image to another in practice is limited: an example for such an application is panoramic stitching. Among the global methods are for instance Fourier- or phase-based (e.g. [Castro and Morandi, 1987, Stricker and Kettenbach, 2001, Stricker, 2002]) and color-based or histogram based methods [Felsberg and Hedborg, 2007b, Chandaria et al., 2007].

On the other hand there are the local methods, which use only small parts

of the image. Some of these parts may not be visible in another image and some may look different. Nevertheless, often many local correspondences can be established and used for estimation. Therefore such feature-based methods are less sensitive to local distortions like occlusions or local appearance changes. Due to their very local view, such local features are often less descriptive and may be similar to other local features, leading to ambiguities in the local matching. Therefore they usually require a model-based verification using more global information. The next sections show the most commonly used primitives for the estimation of geometric objects, starting with simple points and finally concluding with local regions, which are exploited in this thesis.

3.1 Points

In a way, points are among the most simple primitives to think of. Reasoning about point relations goes back to the very first known geometric considerations. Also in the beginning of modern geodesy as a precursor for photogrammetry Grunert[Grunert, 1841] and Gauss[Gauss, 1843/1844] used points for land surveying; the point correspondence is probably the most wide-spread primitive upon which view relations are estimated. A good overview on point based estimation algorithms for projective geometry is given in [Hartley and Zisserman, 2004], but point correspondences are also used in pose estimation [Haralick et al., 1994, Lu et al., 2000, Grafarend and J.Shan, 1997, Grunert, 1841, Zhang and Hu, 2005, Finsterwalder and Scheufele, 1903] or camera calibration [Scaramuzza et al., 2006a, Tsai, 1987, Micusik and Pajdla, 2006].

Points in images have to be localizable according to some criterion. Corners for instance can be localized well in two dimensional image space and often point-based estimation methods use corners, although many other interest point detectors exist. However, also repeatability is of interest, i.e. the same point needs to be detected in transformed versions of the image in an image taken from a different viewpoint or under different conditions. Pioneering analyses in this direction have been performed by Schmid et al. [Schmid et al., 1998].

In very early works, image processing was basically restricted to line drawings in more or less black and white images. In this context interest points have been defined at intersections of lines or at points on curves with high curvature, however, this required an explicit extraction of more global contour lines beforehand. A good summary of these early works can be found in [Tuytelaars and Mikolajczyk, 2008].

Among the first interest point operators based on the image signal was

the one of Moravec [1980], exploiting signal changes in x- and in y-direction in an image. Foerstner and Gülch [Förstner and Gülch, 1987] considered precise relocalization of interest points in their detector and used the roundness of the position uncertainty ellipse as a corner detection criterion. Similarly, the work of Moravec was extended to general 2D signal changes by Harris and Stevens[Harris and Stephens, 1988], improving the detector's behavior under rotation. Shi and Tomasi [Shi and Tomasi, 1994] proposed a detector that finds interest points optimal for tracking in videos. Each of the last three detectors requires two-dimensional structure in a local window, which is encoded in the *structure tensor* sometimes also called the *second moment matrix* and which resembles the outer product of the gradient summed across a window. Since then, other approaches for corner detection have been proposed, often for efficiency reasons, e.g. SUSAN[Smith and Brady, 1995] or more simple keypoints exploiting an approximation of the Laplacian [Lepetit and Fua, 2006].

To obtain precise estimates and to solve the correspondence problem, often regions around the ideal points are considered and the matching is performed by comparing these regions regarding some measure. Therefore, the mathematical concept of a point and the real-world process of measuring the parameters (the position) of such an entity have to be distinguished. If the point position is actually estimated using data from a finite region, this may or may not cause trouble in algorithms expecting infinitesimally small entities. Since the goal is usually to construct stable algorithms in the presence of noise, the disturbance from the finite support region can be neglected if they are small compared to other sources of disturbance, e.g. image noise or mis-calibration of the camera.

Aside from the way how to detect it, the geometrical information carried in points is also relevant: A point in the image has two degrees of freedom (its position), a point in space has three degrees of freedom. If a transformation maps a 2D or a 3D point to a 2D point, this correspondence imposes at most two constraints onto the transformation.

3.2 Lines and Planes

In projective geometry of the plane, lines and points are dual [Hartley and Zisserman, 2004]. Therefore, when estimating homographies between images, many algorithms working with point correspondences work in a dual way with line correspondences. Automated matching of lines has been studied in [Schmid and Zisserman, 1997], while structure-from-motion based upon lines is for example presented in [Spetsakis and Aloimonos, 1990] or [Bartoli and

Sturm, 2005].

However, compared to points, lines have a more global character because in one dimension they are infinitely extended. This can cause trouble in detection e.g. with occlusions, which local features typically avoid. Also, although most camera mappings are usually locally well-linearizable, the straight lines property may be broken in non-ideal cameras and lines in space map to curves in the image.

Lines in space have four degrees of freedom and a minimal representation in Euclidean or projective space is involved (e.g. two sphere angles for direction and 2D-displacement in the plane through the origin perpendicular to the direction). Planes have only three degrees of freedom and planes through the camera center also project to lines in the image (e.g. the epipolar plane[Hartley and Zisserman, 2004]).

Line detection can either be done locally by exploiting the image gradients [Jähne, 2005, p. 345], using e.g. the Canny edge detector[Canny, 1987] and grouping edgels or more globally by using Hough transformation [Duda and Hart, 1972] or related concepts [Thomas, 2007]. If a transformation maps a 3D line or a plane to an image line, this correspondence imposes at most two constraints on the transformation.

If the correspondence relates a point with a line this imposes only a single constraint, since the position on the line is unknown and a point being on a line is a scalar equation.

3.3 Conics, Quadrics and Convex Hull Regions

Conics are curves that result from the intersection of a cone with a plane, such as ellipses, hyperbolas, parabolas and some degenerated representations as described in section 2.1.2. Conics can be represented by symmetric 3×3 matrices in projective space $\mathbb{P}^2$ (equation (2.10)) and therefore have five degrees of freedom. In images, conic curves are usually detected using a contour line approach in an analogue way as line segments are detected, e.g. through Hough transform [de Macedo and Conci, 2007] or using grouping [Smereka, 2005].

The analogue concept in $\mathbb{P}^3$ are quadrics, which are mapped to conics in the image under the pinhole camera model as described in equation (2.22). This relationship is exploited for various geometrical problems, e.g. to estimate homographies [Kannala et al., 2006], the epipolar geometry [Kahl and Heyden, 1998], surface normal or camera pose [Ma, 1993], or in self-calibration approaches [Triggs, 1997, Pollefeys et al., 1998]. If a transformation maps a conic or a quadric to a conic, this correspondence imposes at

most five constraints onto the transformation because a conic has only five degrees of freedom.

Basri and Jacobs [Basri and Jacobs, 2001, Jacobs and Basri, 1999] worked on the concept of region correspondence, which describes each region by its convex hull. This relaxes the assumption of strict elliptically- or conically-shaped regions to a more general shape. On the other hand the direct interrelationship and explicit projection model between 2D and 3D is lost and correspondences between regions are then represented by a set of inequalities. Consequently, tailored optimization approaches are required to exploit such primitives.

3.4 Intensities

Under the image brightness constancy assumption (sometimes also "brightness change constraint equation", cf. to [Jähne, 2005, 425]), corresponding points in different images have the same grey value. Since the grey value is a scalar, this grey-value correspondence imposes only one constraint on the transformation to be estimated. However, since usually the image cannot be compactly described in an equation, often linearized versions of such constraints are used in optimization algorithms, which must start near the optimum. Then however, the intensity correspondences are algebraically equivalent to point to line correspondences and can be exploited to estimate the parameters of the warp between the two images. Each linearized intensity correspondence provides only a single constraint: the projection of the warp's Jacobian matrix onto the image intensity gradient (cf. to [Baker and Matthews, 2004]). Applications are gradient-based image registration [Lucas and Kanade, 1981, Baker and Matthews, 2004], panorama generation [Jethwa et al., 1998, Szeliski, 2006, Shum and Szeliski, 2000], local homography estimation [Astrom et al., 1998] and camera tracking[Koch, 1993]. They are usually used to optimize low-parametric models with high redundancy. However, due to their scalar nature these correspondences are extremely sensitive to noise and brightness changes and outliers can hardly be detected in practice.

3.5 Local Curves and Lines

This section briefly reviews geometric estimation based on so-called quivers, which define a set of multiple local line directions, and estimation based upon small parts of curves. Here, Schmid and Zisserman [2000] used change

of curvature of planar curves to determine homographies. As for the lines these curves have to be extracted from the image. To obtain the curvature, a larger influence region is required. The curvature is a measure of how strong the tangent changes when running along the curve. The principle exploited here is comparable to a 1D version of the primitive used in this thesis and a detailed embedding is therefore given in section 4.5.3. When small parts of curves are considered, e.g. the contours of articulated objects [Grest et al., 2006], the curve can be replaced by its local tangent allowing to work with the curve as with a line.

Johansson et al. [2002] introduced quivers, which can be imagined as a point and a direction (1-quiver) a pair of oriented lines (2-quiver) or a triplet of oriented lines (3-quiver) intersecting in a point. The 3-quiver can be imagined as being a 3D corner with its three edges. In projective geometry these quivers also impose constraints onto the multi-view relations in a scene. In an image, a 1-quiver has three DOF (position and direction), a 2-quiver has four DOF and a 3-quiver has five DOF. Since the quivers work only upon directions, they are closely related to sets of lines and do not carry information about scale or size.

3.6 Local Regions

The major drawback of the point primitive concept is that once a detector found an interesting point in an image, it did not provide information on what region around the point would be suitable to serve for comparison with point-features in other images. Typically regions of constant size and orientation were then used for matching, allowing only for small baseline matches without significant changes in scale, orientation and other warps.

Another approach was taken in the last years by authors working on robust or invariant region features. The term *region feature* is used here because all of these detectors implicitly define a local region, although the way this is achieved differs from detector to detector, and the regions are not always intuitively identifiable by a human.

In the 1990s Lindeberg proposed to exploit the scale-space representation(cf. to [Witkin, 1983]) of an image and to use a scale-normalized Laplacian to obtain features with scale invariance [Lindeberg, 1993a,b, 1998]. These structures can be thought of having an intrinsic scale and can be re-detected reliably under scale changes. This defines not only a position but also a scale and consequently a local region.

Exploiting this, Lowe suggested in a method called scale invariant feature transform (SIFT [Lowe, 1999, 2004]) that the difference of Gaussians (DoG)

can be used for a close approximation to this scheme, which can efficiently be implemented using recursive filtering with Gaussian kernels. Rejecting keypoints that are only weakly localizable in the x-y-plane, the detector produces stable features with a position and an intrinsic size (defining a local region). Furthermore, Lowe proposed to exploit the image gradients of the local region to define an intrinsic orientation of the feature. Together with the position, the scale and orientation parameters of the local features define a *local coordinate system* attached to the feature, which behaves covariantly under similarity transforms. Using the detected parameters, the local image region can then be warped into a normalized coordinate system, allowing for matching images with significant scale and rotation changes (additionally to pure translation).

Upon the *normalized regions*, Lowe computed the SIFT descriptor[Lowe, 1999, 2004], which describes the content of the local region in a robust way even in presence of intensity changes and small alignment errors. Together with the detector, the descriptor is invariant against similarity transforms of the image and affine brightness changes. The method degrades only slightly when changes are approximately explainable with these transforms. Recently, an even faster version of the SIFT principles, called SURF[Bay et al., 2008] has been proposed. In the field of robust descriptors, working on the normalized local region, a good overview is given in [Mikolajczyk and Schmid, 2005].

Some work into a different direction exploited invariants of the local image structure under different transforms [Florack et al., 1994, Flusser and Suk, 1993, Montesinos et al., 1998, Schmid and Mohr, 1997]. Here the goal was not to detect parameters of the local region (like scale or orientation), but rather to characterize the local image signal at a point in a way invariant against some transformations, e.g. by exploiting invariants of moments and derivatives. Van Gool et al. [1995] describe how such invariants against some transformation can be derived. In this approach, the transformation's parameters are not determined and then compensated (as in the SIFT approach) but the transformation's effects are rather canceled out in the formulation.

During the last 10 years several authors worked on automatically achieving full affine invariance. Lindeberg proposed to exploit local image shape for smoothing structures in affine scale space[Lindeberg and Garding, 1997]. Similarly, Baumberg [2000] examined the local image structure encoded in the second moment matrix to normalize the image regions around Harris corners. This removed skew and anisotropic stretch distortions from the local feature. In his descriptor a rotational grey-value invariant was applied, removing the need to compute a local orientation. He also pointed out that homographies can locally be explained well by affine transforms, allowing

the approach to cope with locally planar structures seen from quite different viewpoints.

Tuytelaars et al. proposed the use of edge-based (EBR[Tuytelaars and van Gool, 1999]) or intensity-based regions (IBR[Tuytelaars and van Gool, 2000]) to obtain features behaving covariantly under affine image transforms. Mikolajczyk and Schmid suggested detectors based on extensions of the Harris corner detector [Mikolajczyk and Schmid, 2001, 2002] and on the Hessian of the image function [Mikolajczyk and Schmid, 2004b]. Matas et al. proposed separated elementary cycles of the edge graph (SEC) [Matas et al., 2001] and maximally stable extremal regions (MSER), an exploitation of a watershed algorithm to compute regions that have significantly different intensity than their surrounding [Matas et al., 2001, 2002, 2004]. Kadir et al. worked on detectors based upon the statistics of the image and information theory [Kadir and Brady, 2001, Kadir et al., 2004]. A comparison of affine region detectors can be found in [Mikolajczyk et al., 2005]. Although these affine region features carry more information than simple points, in most estimation approaches they are geometrically handled as points.

To better exploit the information carried in such features, Chum et al. [Chum et al., 2003] proposed that in each local feature coordinate system, three simple points can be *detected* so that each affine feature correspondence provides three point correspondences, which could be exploited in fundamental matrix estimation. This idea was recently also adopted by Perdoch et al. [Perdoch et al., 2006] for another epipolar geometry problem. Riggi et al. [Riggi et al., 2006] on the other hand did no longer detect points in the local affine frame, but directly sampled the local affine frame into three close points, which could then be used in traditional point-based fundamental matrix estimators. In large scale image search and object retrieval [Jegou et al., 2008, Philbin et al., 2007, Chum and Matas, 2008], corresponding features are checked for geometrical consistency to re-rank the results. Here, Philbin et al. [2007] assume that all feature correspondences on a 3D query object can be explained with a single general 2D affine transform. They compare models with isotropic scale, axes-aligned scales and shear (but no 2D rotation) but do not model global perspective effects. Jegou et al. [2008] down-weight results where the features significantly differ in relative rotation or average scale, which seems to be an efficient heuristics in large-scale search.

Also in video sequences, where position, warp and appearance of local regions change only gradually from frame to frame, feature tracking based upon affine parameters has been proposed already in [Lucas and Kanade, 1981] and exploited for monitoring feature quality in [Shi and Tomasi, 1994]. An overview of the efficient algorithms for tracking through video or image alignment proposed since then is given by Baker and Matthews[Baker and

Matthews, 2004]. In [Astrom et al., 1998] local homography optimization was proposed for locally planar patches. Affine tracking is used nowadays in many applications, but the geometric information the affine warp provides is disregarded or exploited only in extended Kalman-filtering [Molton et al., 2004, Davison et al., 2007] or iterative refinement techniques. The latter has been used with an affine camera model first [Rothganger et al., 2006] and has recently been extended to a locally linearized projection matrix [Rothganger et al., 2007] for 3D planar patch tracking.

3.7 Summary and Relation to this Thesis

To summarize, several authors proposed features repeatable under scale or similarity transforms [Mikolajczyk and Schmid, 2001, Lindeberg, 1998, Lowe, 2004, Kadir and Brady, 2001, Bay et al., 2008] or even affine transforms [Tuytelaars and van Gool, 1999, 2000, Matas et al., 2001, Mikolajczyk and Schmid, 2002, 2004b, Matas et al., 2002, Kadir et al., 2004], which have a local, covariantly transforming coordinate system attached to them, allowing to normalize the local region for photometric matching. Also gradient-based tracking methods provide an affine transform between local image regions [Lucas and Kanade, 1981, Shi and Tomasi, 1994, Baker and Matthews, 2004].

However, in the literature so far, the geometric information in the affine warp has mostly been neglected or used heuristically. It was only used as a constraint in iterative optimization or sampled into a triplet of very close point correspondences. Many point-based algorithms are not designed to cope with infinitesimally differing points and the question arises whether the information cannot be represented in a more compact way than by sampling. Therefore, in the next section a novel primitive for estimation will be derived, which defines a whole new set of direct estimators and for which uncertainty can be derived in order to allow maximum likelihood estimation.

Chapter 4

Differential Constraints from Local Affine Frame Correspondences

The problem of automatically obtaining correspondences between images is complicated by the fact that parts of a scene can look significantly different when viewpoint or illumination changes. To cope with this problem, the idea of robust features appeared (compare [Tuytelaars and Mikolajczyk, 2008] for a good survey), where each describes a local, potentially continuous region of the scene. Besides the risk of introducing ambiguities due to the locality of such features, the approach has two major advantages: First, since each feature covers only a very small part of an image, there is a good chance that many of such local regions can be found fully intact in another image, even in presence of occlusion or other scene changes. Second, within a local region global effects of smooth transformations, such as perspective effects and radial distortion, are hardly observable, i.e. first order Taylor representations of distorting functions are good approximations to undistort local patches. Similar ideas also apply to photometric changes of images.

As summarized in the previous chapter, many feature detectors evolved that are able to find local regions suitable for the above approach, i.e. local regions in which affine transformations are observable and which can reliably be found also in other images (compare [Mikolajczyk et al., 2005]). In the next sections, the way to obtain such correspondences is characterized and, given a correspondence, a geometric model to exploit the information is derived. This model applies also to affine template tracking through video, and the tracking approach can thus also be exploited to refine feature correspondences. Nevertheless, in the end a measured correspondence is always afflicted with uncertainty so that a statistical framework is derived to allow

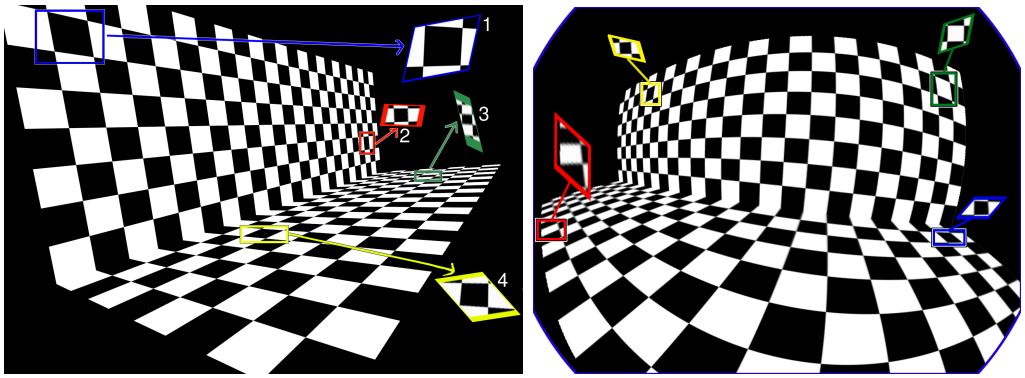

Figure 4.1: Chessboard texture on two planes observed by an ideal pinhole camera with a very wide field of view (left) and a fish-eye camera (right). In the left image the original (orthophoto) chessboard texture is related to the perspective view via the two homographies across the two planes. Four rectangular cutouts have been chosen and manually rectified using rotation, shear and scale only (affine approximation of homography). Although no projective unwarping has been done, only for the larger regions (e.g. 1 and 4) the affine approximation has a visible difference from a square. For instance, in the cutout 1 the opposite borders of the black patch are visibly not parallel. Such projective effects cannot be compensated because affine transforms preserve parallelity of lines. However, they are only observable in larger windows and locally the affine transform is a good approximation even to non-linear warps. In the right image, the nonlinear mapping of the fish-eye camera has locally been compensated by a manually selected affine transform in the same way.

for maximum-likelihood estimation and statistical testing for mismatches or outliers. The novel primitive for estimation is finally related to other estimation primitives such as conics, which concludes this chapter.

4.1 Robust Local Image Features

Progress in robust local features[1] allows matching of images in which appearance of local regions undergoes approximately affine changes of brightness and/or of shape, e.g. for automated panorama generation or scene reconstruction through wide-baseline matching. The idea is that interesting features are detected (e.g. corners, blobs, ...) in each image and that the surrounding region of each feature is normalized with respect to the local image structure in this region, leading to about the same normalized regions for correspondences in different images. Features with equal or similar normalized regions are tentative correspondences, which can be verified using a geometry/motion model (e.g. epipolar geometry check). Affine normalization of the regions is particularly interesting because an affine mapping is a first order Taylor approximation to the true (possibly unknown) mapping function. In figure 4.1 a manual affine normalization can be seen.

4.1.1 Regions of Interest

As summarized in the previous section many region detectors have been proposed for similarity [Mikolajczyk and Schmid, 2001, Lindeberg, 1998, Lowe, 2004, Kadir and Brady, 2001, Bay et al., 2008] or affine transforms [Tuytelaars and van Gool, 1999, 2000, Matas et al., 2001, Mikolajczyk and Schmid, 2002, 2004b, Matas et al., 2002, Kadir et al., 2004]. Exemplarily in figure 4.2 the MSER[Matas et al., 2002] features are shown, and figure 4.3 shows the affine normalization. Each of these features can be described by a position $(f_x, f_y)^\mathsf{T}$ in the image and its local shape and orientation, as will be described in the following section.

4.1.2 Local Affine Frame

The local affine frame (LAF, cf. also to [Obdrzalek and Matas, 2006]) is a coordinate system attached to the feature, where each point in the feature coordinate system with the center $(f_x, f_y)^\mathsf{T}$ can be described independently of the scale, the orientation or the stretch of the feature. The parameters of the

[1] While robustness in the field of estimation usually means insensitivity against gross errors (e.g. mis-matches in correspondence-based geometry estimation), in the field of local image features robustness is rather used to distinguish from invariance (compare also [Tuytelaars and Mikolajczyk, 2008]): If a feature is invariant under some transformation or disturbance, the mathematical formulation directly models and accounts for this. If a feature is only robust, this source of disturbance is usually not explicitly modeled. However, if the disturbance is small then the feature is not affected too much from this.

Figure 4.2: Two images with wide baseline, where the camera was additionally rolled by 180°. In these images MSER[Matas et al., 2002] features have been detected as indicated by the small ellipses. The ellipses represent the mean and the second central moment of the pixels segmented by the watershed algorithm.

Figure 4.3: In each of the MSER features from figure 4.2 the local gradients have been exploited to obtain a main orientation as proposed by Lowe[Lowe, 2004]. Together with the elliptical shape this provides a full affine coordinate system, the local affine frame. Two features are shown in their LAF coordinate system (after optimization). As can be seen, this normalization results in two almost identical local regions. This means that a concatenation of the two transformations warps the local region of the first image onto the local region of the second image. Mathematically, this transformation is the linearization, or the first order Taylor approximation of the true homography.

local affine frame transform covariantly when an affine transform is applied to an image. For example the region size and shape may change: these parameters are not invariant under affine transformations, but they behave accordingly to the transformation, i.e. in a downsized image the region size will also be reduced consistently. Since the affine parameters are always determined based upon regions in the image, in the remainder the detectors for such affine frames will be called affine region detectors, regardless of whether they physically represent a region or a point (e.g. a corner).

Affine covariant region detectors like Hessian or Harris affine directly compute a local texture anisotropy, which defines the scale and the elliptic shape of the feature while the MSER detector determines the affine shape using the scatter of the detected pixels. EBR, IBR and Salient Regions also have their own tailored solutions for obtaining the local shape.

The affine shape matrix $A_{\text{shape}} \in \mathbb{R}^{2\times 2}$ has already been used for the overlap error evaluation in [Mikolajczyk and Schmid, 2004a]. Depending on the type of detector used it is either the root of the second moment matrix, i.e. the gradient distribution around the feature (e.g. Harris and Hessian affine) or the root of the second central moment of the pixels belonging to the local feature (e.g. MSER). It can be decomposed as

$$A_{\text{shape}} = A_{\text{R}}^{\mathsf{T}} A_{\text{scale}} A_{\text{R}} \tag{4.1}$$

where $A_{\text{scale}} \in \mathbb{R}^{2\times 2}$ is a diagonal scale matrix holding the principal scales and the rotation $A_{\text{R}} \in \mathbb{R}^{2x2}$ defines the direction of scales. The determinant of A_{shape} is proportional to the image area covered by the local feature. Consequently, in the following $\mathtt{det}\,[A_{\text{shape}}] \neq 0$ is assumed for all measured features.

After the shape of the region is fixed, the orientation has to be defined, which can be done by computing a main gradient orientation in the region defined by the above shape matrix as originally proposed by Lowe [Lowe, 2004]. The local affine frame is then rotated according to this main orientation by $A_{\text{orientation}} \in \mathbb{R}^{2\times 2}$:

$$A_{\text{orientation}} = \begin{pmatrix} \mathtt{cos}\,[\theta] & \mathtt{sin}\,[\theta] \\ -\mathtt{sin}\,[\theta] & \mathtt{cos}\,[\theta] \end{pmatrix} \tag{4.2}$$

Therefore the overall transformation that warps between the feature and the (feature-centered) image is:

$$A_{2\times 2} = A_R^{\mathsf{T}} A_{\text{scale}} A_R A_{\text{orientation}} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \tag{4.3}$$

The linear shape of the local affine frame can efficiently be stored in a 2×2 matrix (similar to [Hartley and Zisserman, 2004, p.40]) and the factorization

is easily obtained from the singular value decomposition(SVD, cf. to [Hartley and Zisserman, 2004, p.585]) of the matrix :

$$A_{2\times 2} = USV^{\mathsf{T}} = USU^{\mathsf{T}}(UV^{\mathsf{T}}) \tag{4.4}$$

Now, the position of the feature with respect to the image coordinate system is also considered, allowing to directly transfer image coordinates into local feature coordinates and vice versa. A homogeneous point in the image $\mathbf{x}^I$ is related to a point in the local affine frame $\mathbf{x}^A$ through the 3×3 matrix $\mathbf{A}$:

$$\mathbf{x}^I = \mathbf{A}\,\mathbf{x}^A = \begin{pmatrix} a_{11} & a_{12} & f_x \\ a_{21} & a_{22} & f_y \\ 0 & 0 & 1 \end{pmatrix} \mathbf{x}^A \qquad \mathbf{x}^A, \mathbf{x}^I \in \mathbb{P}^2 \tag{4.5}$$

Since $A_{2\times 2}$ must be invertible, $\mathbf{A}$ must also be invertible:

$$\mathbf{x}^A = \mathbf{A}^{-1}\,\mathbf{x}^I \tag{4.6}$$

In the following, the matrix $\mathbf{A}$ itself will be referred to as the local affine frame (LAF) of the feature in the image.

4.1.3 Descriptors

Given the local affine frame of a feature, the local texture of the image can be warped and the normalized patch (with respect to the LAF parameters) can be constructed, similar to what Lowe [1999] proposed for the SIFT features. The normalized patch will look the same for all images which are affinely transformed versions of the first image and will still look similar when the local image warp is only approximately affine (see also figure 4.3 for an example). To obtain potential correspondences, a criterion is now required that states whether two patches look similar or not, and this can be done by computing a signature (also called a descriptor) upon the local region.

The simplest signature is a vector of grey values at fixed positions in the local affine frame. Such vectors can then be compared by means of dissimilarity measures like SSD(see e.g. [Skoglund and Felsberg, 2006]) or SAD(as used in [Skoglund and Felsberg, 2007]) or similarity measures like NCC(as exploited in [Lewis, 1995]) or any other metric which can be applied to image patches. However, since the parameters of the local affine frame are often disturbed, authors proposed to use descriptors that tolerate small errors in the parameters of the local affine frame and also some photometric distortions of the region. A good overview of such descriptors has been given by Mikolajczyk and Schmid [2005], where it was found that the SIFT[Lowe,

2004] descriptor performs particularly well. This 128-dimensional vector represents gradient orientation histograms based on a soft-binning technique and will be used in the remainder of this thesis, although the methods proposed are not limited to a specific descriptor.

4.1.4 Matching Strategies

When seeking correspondences in a video sequence, they are usually spatially close in subsequent frames. Therefore, in a feature-based approach, in each frame of the video features can be extracted and compared to features in a spatial neighborhood in the next frame. The size and shape of the spatial neighborhood depends on the relative movement of the objects in the scene and the camera. This technique is called image-space matching in the following.

When images with a totally unknown relation have to be matched, image space matching cannot make any assumptions on the size of the local neighborhood and would require the comparison of every feature in one image with every feature in another image. Particularly when multiple image are involved, this is often too complex for fast applications.

However, corresponding features will have similar descriptor vectors, and therefore the matching problem for a given feature can be posed as that of finding a cluster of similar descriptor vectors in the space of all possible descriptors. This technique is called feature-space matching and is particularly well-suited for problems where some offline processing time can be spent to prepare data structures of learnt descriptors while online lookup of a descriptor should run fast. However, for huge numbers of features feature-space matching can already be efficient for online two view matching problems. Since 128 is a very high number of dimensions, the sparsely populated feature space has to be compressed in some way or at least efficiently represented to allow storage in memory [Beis and Lowe, 1997, Köser et al., 2006b]. In the next section it is assumed that a correct correspondence has been obtained using any of the above matching strategies to derive a more powerful constraint than the traditional point-to-point correspondence.

4.2 The LAF Correspondence Constraint

4.2.1 Concatenation of Local Affine Frames

In section 4.1.2 it has been shown how the image and the local affine frame are related. Particularly equation (4.6) states how coordinates in an image can

be expressed as coordinates in the local affine frame for some local feature. Since in the following a second image and a second local affine frame will be regarded, equation (4.6) is now repeated here with more precise indices for the first image I_1 and the local affine frame $\mathbf{A}_{\mathbf{x},I_1}$ of feature $\mathbf{x}$ in this image.

$$\mathbf{x}^A = \mathbf{A}_{\mathbf{x},I_1}^{-1}\, \mathbf{x}^{I_1} \tag{4.7}$$

If a local feature $\mathbf{y}$ is now observed in another image I_2, a local affine frame for that feature can be established for that image. A point $\mathbf{y}^{I_2}$ in the corresponding region in the other image can also be transformed into its feature coordinate system.

$$\mathbf{y}^A = \mathbf{A}_{\mathbf{y},I_2}^{-1}\, \mathbf{y}^{I_2} \tag{4.8}$$

If the local region in the image I_1 is an affinely transformed version of the local region of the other image I_2 both normalized regions will look the same, i.e. the same coordinates in both normalized regions

$$\mathbf{x}^A = \mathbf{y}^A \tag{4.9}$$

will have the same intensity value (see e.g. figure 4.3). Furthermore, this correspondence together with equations (4.7) and (4.8) relate coordinates in both original images via the concatenated transformations:

$$\mathbf{A}_{\mathbf{x},I_1}^{-1}\mathbf{x}^{I_1} = \mathbf{x}^A = \mathbf{y}^A = \mathbf{A}_{\mathbf{y},I_2}^{-1}\mathbf{y}^{I_2} \tag{4.10}$$

$$\mathbf{x}^{I_1} = \mathbf{A}_{\mathbf{x},I_1}\mathbf{A}_{\mathbf{y},I_2}^{-1}\mathbf{y}^{I_2} = \mathbf{A}_{\mathbf{xy}}\mathbf{y}^{I_2} \tag{4.11}$$

Here, $\mathbf{A}_{\mathbf{xy}}$ contains the transformation from image I_2 to image I_1 in a homogeneous matrix. This can also be viewed as a mapping from 2D Euclidean image coordinates to 2D Euclidean image coordinates

$$\boldsymbol{x}^{I_1} = A_{xy}\boldsymbol{y}^{I_2} + \boldsymbol{dx} \qquad\qquad A_{xy} \in \mathbb{R}^{2\times 2}, \boldsymbol{dx} \in \mathbb{R}^2 \tag{4.12}$$

where A_{xy} represents the linear part of the image to image warp and $\boldsymbol{dx}$ the offset. The matrix A_{xy} captures local (anisotropic) stretch and rotation between both images at the correspondence and is also the Jacobian of the transformation. This relation has been exploited in matching or tracking for quite some time (at least implicitly), but in the next section the geometric constraints it imposes will be derived as originally proposed in [Köser et al., 2008].

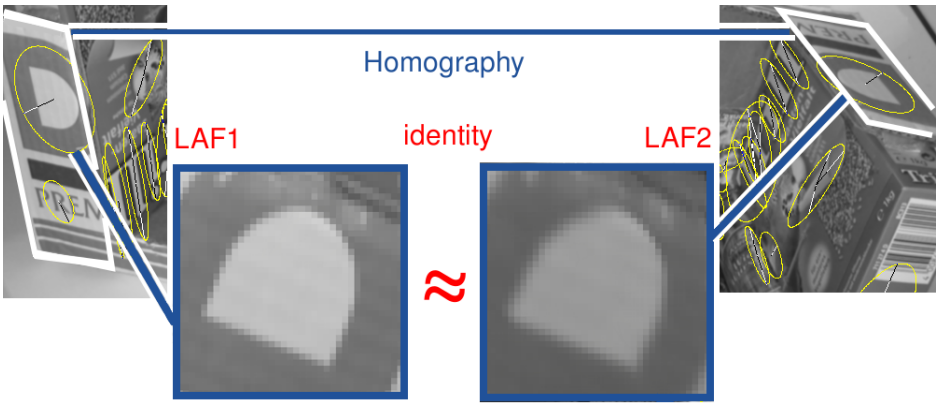

Figure 4.4: The concatenation of two corresponding LAFs yields a local linearization of the homography between the two planes on which the features lie.

4.2.2 Warp Constraints

Whenever region-based features are used to find correspondences between images, the base hypothesis is that (a transformed version of) the region can be found in both images. The implicit assumption when applying affine features is that the texture warp $\mathtt{W} : \mathbb{R}^2 \to \mathbb{R}^2$ between the two views can locally be approximated reasonably well by the first order Taylor series (an affine mapping), i.e. the warp is locally analytic and close to linear (see also figure 4.4): The centers $\boldsymbol{x}_0$ and $\boldsymbol{y}_0$ of the feature correspondence fulfill the classical point-correspondence property:

$$\boldsymbol{y}_0 = \mathtt{W}[\boldsymbol{x}_0] \quad \boldsymbol{x}_0 \in \mathbb{R}^2, \boldsymbol{y}_0 \in \mathbb{R}^2 \tag{4.13}$$

However, since $\mathtt{W}$ is required to be analytic and close to linear, the region around the point $\boldsymbol{x}_0$ in image one maps to a region around $\mathtt{W}[\boldsymbol{x}_0]$ in the other image according to the Taylor-series of $\mathtt{W}$:

$$\boldsymbol{y} = \mathtt{W}[\boldsymbol{x}] = \mathtt{W}[\boldsymbol{x}_0] + \left.\frac{\partial \mathtt{W}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_0} [\boldsymbol{x} - \boldsymbol{x}_0] + \epsilon[\boldsymbol{x} - \boldsymbol{x}_0] \tag{4.14}$$

where ϵ represents all higher order terms and vanishes if $\mathtt{W}$ is locally an affine transform. A more detailed presentation of the Taylor series and an upper bound on the error can be found in appendix A.1. In practical situations, ϵ can be considered zero if $\mathtt{W}$ is well-linearizable (such as homographies far from the (pre-)image of the line at infinity) in the region upon which the LAF correspondence is computed.

For an affine feature correspondence it is known that coordinates are related by equation (4.11). Under the assumption that $\epsilon(\boldsymbol{x} - \boldsymbol{x}_0) = 0$ for $\boldsymbol{x}_0 \approx \boldsymbol{x}$ it follows that

$$\left.\frac{\partial \mathtt{W}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_0} = A_{xy} \tag{4.15}$$

This is a matrix equation, which must hold for all four entries and consequently imposes additional constraints on the function $\mathtt{W}$. E.g. if the desired transformation $\mathtt{W}$ is a homography, equation (4.15) poses four additional constraints on the parameters of $\mathtt{W}$, i.e. each affine correspondence now contributes a total of six constraints on the components of the homography (two from the center correspondence as usual and another four from the local region deformation). Usually it is possible to apply a virtual change of coordinate systems, so that the correspondence is in the origin of the new system. Then, e.g. in case of a homography, the constraints of equation (4.15) often even become very simple and linear, depending on the parameterization.

4.2.3 Physically Motivated Interpretation

The local affine frame can also be viewed as two vectors attached to some point of the image signal spanning a local coordinate system. Under a smooth, analytic transformation $\mathtt{W}$ of the image signal $\mathcal{I}$ into a transformed version $\mathcal{T}$

$$\mathcal{I}[\boldsymbol{x}] = \mathcal{T}[\mathtt{W}[\boldsymbol{x}]] \tag{4.16}$$

these two basis vectors $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ represent a physical signal property at that point, which behaves covariantly with this transform. For instance, the local gradient of the image behaves covariantly with such a transformation of the image. Therefore, as an illustrating example, let $\boldsymbol{e}_1$ be the local image gradient. As can be seen from the chain rule, when going from $\mathcal{I}$ to $\mathcal{T}$ using the warp $\mathtt{W}$, the gradient is multiplied by the Jacobian of $\mathtt{W}$:

$$\frac{\partial \mathcal{I}[\boldsymbol{x}]}{\partial \boldsymbol{x}} = \frac{\partial \mathtt{W}}{\partial \boldsymbol{x}} \frac{\partial \mathcal{T}[\mathtt{W}[\boldsymbol{x}]]}{\partial \mathtt{W}[\boldsymbol{x}]} \tag{4.17}$$

which defines the new basis vector in the other image. If $\mathcal{I}$ is a multi-channel (e.g. color) image, $\boldsymbol{e}_2$ can be imagined as the gradient of the second channel. This gradient changes covariantly with the image transform, too. If $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ are linearly independent, they form a basis of a local affine coordinate system. The basis vectors of the transformed local affine frame can be obtained by

$$\boldsymbol{e}_1' = \frac{\partial \mathtt{W}}{\partial \boldsymbol{x}} \boldsymbol{e}_1 \qquad \boldsymbol{e}_2' = \frac{\partial \mathtt{W}}{\partial \boldsymbol{x}} \boldsymbol{e}_2 \tag{4.18}$$

These two vectors now define the rows of the 2×2 matrix $A_{2\times 2}$ from equation (4.3), the basis of the local affine frame. Therefore the origin of the local affine frame maps with the warp $\mathtt{W}$ while the two basis vectors (or the affine matrix) map with the Jacobian $\partial \mathtt{W}/\partial \boldsymbol{x}$ of $\mathtt{W}$. That means that the whole local affine frame maps into the other image with the first order Taylor representation $\mathbf{W}_{\text{Taylor}}$ of the function $\mathtt{W}$:

$$\mathbf{A}_x = \begin{pmatrix} \frac{\partial \mathtt{W}}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{x}_0} & \mathtt{W}[\boldsymbol{x}_0] - \boldsymbol{x}_0 \\ & 1 \end{pmatrix} \mathbf{A}_y = \mathbf{W}_{\text{Taylor}} \mathbf{A}_y \tag{4.19}$$

The homogeneous 3×3 matrix $\mathbf{W}_{\text{Taylor}}$ can be constructed for any analytic warp $\mathtt{W}$ and will be used to express the local linearization in the remainder of this thesis. It states a fundamental connection between local affine frames in different images.

Since the Harris-affine detector [Mikolajczyk and Schmid, 2002] is based upon the local gradients in a region (the second moment matrix), it provides a close approximation to the above requirements, if the region is small. Analogous considerations apply to the other detectors that behave covariantly with an affine transformation of the image: The regions upon which they are computed must be so small that the warp's derivative in equation (4.18) does not change (significantly) within the region. Not surprisingly, this is the same requirement as it was for establishing correspondences between these features: to allow for extraction of a good (constant) feature descriptor based upon region normalization or invariants, also no significant perspective (non-linear) effects must be visible in the patch. Since virtually all of these detectors have been designed to allow wide-baseline matching based on locally planar 3D regions, the above assumption is not a big restriction but rather a description of what has already been exploited in matching for years.

4.2.4 Triangle Decomposition

While the previous section gave another view on how the affine feature correspondence imposes constraints and under which assumptions, in this section it is shown how an affine feature correspondence can be approximated by three, spatially very close, point correspondences.

Equation (4.15) is about the derivatives of the transformation $\mathtt{W}$ at the feature position. It states that the affine transformation (as the first order Taylor approximation) is tangent to the transformation at the feature position. This can be a non-linear constraint for a particular transformation $\mathtt{W}$. Each type of transformation requires an inspection of the structure of

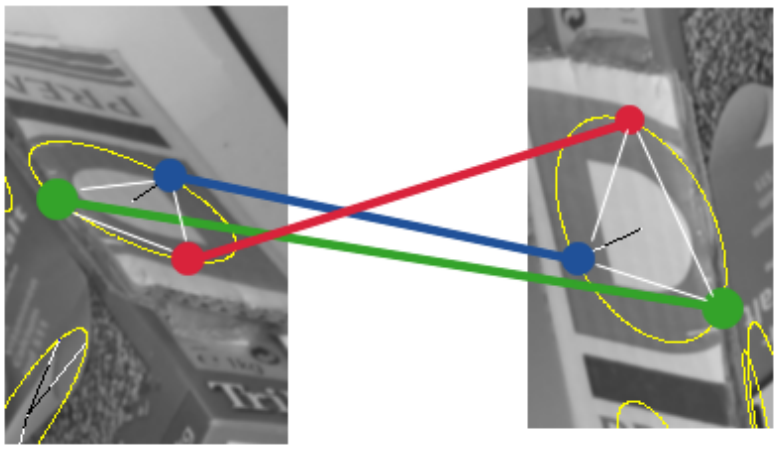

Figure 4.5: The feature of the previous figure has been cut out of both images to visualize the triangle decomposition. Each feature can be sampled into three points, which form a triangle and which each correspond to a point sampled from the other feature. The distance of the points to the center should be small enough such that the sampling does not dominate the actual transformation effects. It must however be large enough so that using the close points in the algorithm does not lead to numerical difficulties such as cancellation.

the derivative and possibly a tailored solution. However, there is also a numeric way to automatically convert these constraints into more common and well-understood point correspondences (similar to the "hallucinating points" concept [Szeliski and Torr, 1998]), where the feature center and its two basis vectors are represented by a triplet of points. This sampling will be called the triangle transform in the following and is visualized in figure 4.5.

Equation (4.15) states that the (typically measured) relative magnification, rotation, shear, etc. between the features in the two images equals the derivative of the sought transformation. In principle, the first column of the derivative is proportional to the step taken in image 1, when a small step is taken in x-direction in image 2. The second column defines the step, when a small step is taken in y-direction in image 2. When relaxing the infinitesimally small derivative in the left hand part of equation by a finite step, the differential constraint of equation (4.15) can be replaced by simple point correspondences. The finite differences are easily obtained from the feature's scale and shape as explained next, providing a straight-forward and direct approximation of the LAF correspondence by three point correspondences.

Each affine feature can be represented by a triangle, whose size and shape can automatically be constructed from the affine region. The increased information content of these features has already been used by Chum et al. [Chum et al., 2003] and the finite sampling has been proposed by Riggi[Riggi et al., 2006]. The decomposition can be understood from the defining equation of the affine matrix: Three correspondences $(\boldsymbol{x}^A, \boldsymbol{x}^I)$ define the affine matrix, since each fixes two degrees of freedom. For simplicity, the three points

$$\boldsymbol{x}_r^A = (1,0)^\mathsf{T}$$

$$\boldsymbol{x}_g^A = (\mathtt{cos}\left[\frac{2}{3}\pi\right], \mathtt{sin}\left[\frac{2}{3}\pi\right])^\mathsf{T}$$

$$\boldsymbol{x}_b^A = (\mathtt{cos}\left[\frac{4}{3}\pi\right], \mathtt{sin}\left[\frac{4}{3}\pi\right])^\mathsf{T}$$

are chosen, which divide the unit circle into three 120° segments, and there are many other possibilities. Care must however be taken that the three sampled points are not on a line, which is automatically guaranteed in the chosen method. Their coordinates in the image are computed, which lie all at the contour line of a feature ellipse, for which the affine approximation is assumed to be correct. The image coordinates can simply be computed from equation (4.5), yielding a triangle (from three sample points). A corresponding feature in the other image is decomposed in the same way, which provides a triplet of correspondences now, i.e. the corresponding triangle corners. One might as well have chosen four or more points, but this generates

only redundant data with no extra information. It is now possible to use the information of affine feature correspondences in the traditional point based geometry estimation algorithms such as direct linear transformation (DLT, cf. to [Hartley and Zisserman, 2004]). In addition to that, the triangle decomposition allows for natural integration into overdetermined systems, since residuals on point coordinates are much easier to understand, to weight, and to adjust than residuals of higher order polynomials, which do not instantly provide a geometrical real-world interpretation of what is being minimized. However, the three points all stem from a local region, which often leads to numerical difficulties such as cancellation (e.g. due to small differences of large numbers, cf. to [Press et al., 1992]) because point-based algorithms are often designed to work with well-distributed points.

4.2.5 Refinement and Upgrade from Simpler Features

The considerations so far apply to affine features. However, if matches result from similarity covariant region detectors (e.g. DoG as in [Lowe, 2004]) or even weaker features, the proposed method can also be applied. The main insight is that if a correct match has been established, the local regions are often already approximately aligned. In that case the local affine frame definition (equation (4.11)) can as well be applied to such features, resulting in a special affine transform.

However, to further improve the estimate (even for already affine features) it is possible to apply a gradient-based optimization of the correspondence using the Lucas-Kanade approach [Lucas and Kanade, 1981, Baker and Matthews, 2004]. This result is usually a much more precise region correspondence, which again provides a much more precise estimate of the local derivative of the warp. Such an optimization is inspected in the next section.

4.3 Alignment of Local Affine Frame Correspondences in Scale Space

Once a rough guess for the LAF correspondence is available, the grey values around the correspondence can be exploited to optimize it. Under the assumption that the Jacobian of the texture transform is constant within a certain surrounding of the correspondence, a local affine transform between the two features can directly be estimated using gradient-based estimation. The smaller the window for aligning the local regions is, the more likely it becomes that an affine transform between the images is sufficient to explain the local warp. Even though affine features may lie on planes that show strong

perspective in an image, in a local region the Jacobian is almost constant and this is the reason why the affine features are so successful. It is a good idea to parameterize the affine transform in terms of the Taylor expansion (with the warp center at the correspondence center), because in this case the warp and displacement parameters are less correlated than when the affine transform is parameterized using global image coordinates.

Sometimes however, the correspondence may lie in a region, where the Jacobian is not constant, so that averaging the affine transform across the region will not produce a good estimate of the Jacobian at the center. This can happen particularly when locally there is not much image structure, such that the region for refinement must be chosen quite large. In this case the next higher term from the Taylor expansion can be used for estimation or any other parameterization directly containing the parameters of the LAF correspondence. Since the linearization should still explain the warp to a large extent, the extra parameters can be initialized such that they do not change the warp (e.g. zero for the second derivatives). If the system matrix including the second derivatives has full rank, all the parameters can be estimated. Typically, the uncertainty for the higher order derivatives is quite high because they need a larger support to be measured. However, they are only exploited to obtain the correct Jacobian and can be neglected afterwards. If on the other hand the system matrix does not have full rank because the second derivatives cannot be estimated from the local data, they can safely be left out and only the first derivatives are estimated in that case.

Since typically the orientation and scale prediction is not very precise, care must be taken when using grey value linearization as inherent in gradient-based image registration. If gradient-based image alignment is directly applied to the full-resolution images, e.g. based upon affine warps, one notices that some pixels of a patch can provide contradictory information. They disturb the optimization because they are outside the valid linear environment of the true correspondence (compare figure 4.6) and such outliers have to be avoided. For instance, if the initial parameters have a rotational error of 10° (which is the orientation histogram quantization proposed in SIFT[Lowe, 2004]), this results in a position error of three pixels for the corners of a square patch with half window size 12. In presence of fine detail, this violates the grey value linearization assumption. Typically, image pyramids are used in this case or heuristic smoothing is applied, but if the resolution is reduced, one may run into a high-dimensional version of the aperture problem, where not enough data exists to estimate the parameters: the feature has a relatively good localization at its intrinsic scale in scale-space (cf. to [Lindeberg, 1994]), but does not necessarily provide much structure at significantly coarser levels. Therefore, one goal of the method proposed in [Köser

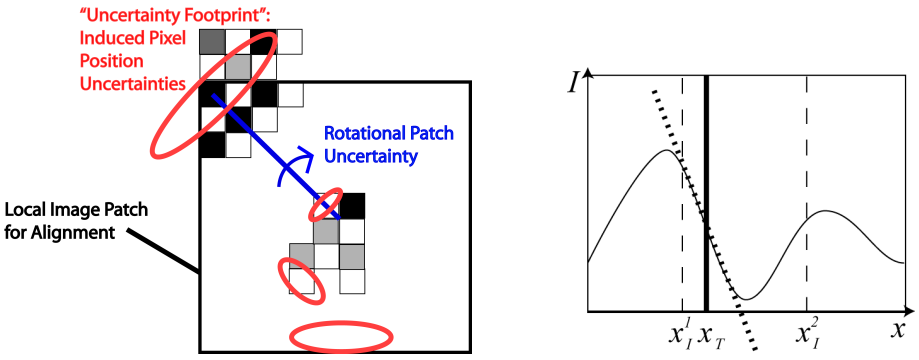

Figure 4.6: Given rotational, similarity, or affine parameter uncertainty, the 2D pixel positions within a warped patch are unequally certain (left image). The ellipses indicate the 2D position uncertainties in a patch induced by significant rotational and minor scale uncertainty. This can be viewed as the footprint of the parameter uncertainty. Outer regions (far from the warp center) are typically more uncertain and at positions near a patch border the assumption of locally linear intensities in gradient based alignment quickly gets violated with such uncertainties. The linearity assumption can be seen in the 1D intensity profile of the right image, where the intensity near x_T is approximated by the tangent (dotted line). For a small displacement, e.g. x_I^1, the linearity is quite correct, but for a larger, e.g. x_I^2, the linear extrapolation is far away from the real data.

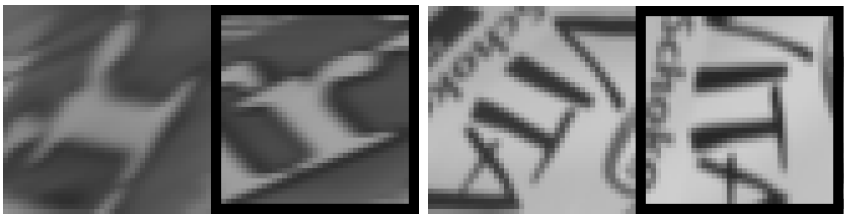

Figure 4.7: Two rough feature matches, difficult to refine for standard alignment. The left part of each pair shows an affinely normalized feature from the first image and the right part (with a black frame) shows the normalized feature from the other image. Since this is the normalized representation (in LAF coordinates), both patches should be identical. They differ however, and the main error is due to wrong 2D orientation. The missing affine transform that makes both patches look the same has to be found by gradient-based optimization. To increase the basin of convergence, traditionally pyramids have been used, while in this thesis it is proposed to use an approach based on known parameter uncertainty (e.g. for a rotation angle).

and Koch, 2008b] and summarized here, is to use as much of the data as possible and only filter out disturbing information. Also, when optimizing parameterized warps based on multiple parameters, some may allow for a better prediction than others and some may have a stronger influence on the convex environment and validity of the local linearization. Finally, not all of the pixels in a patch have to be sensitive to an incorrect start value in the same way (compare center and border pixels in figure 4.6). This section addresses all these issues and incorporates uncertainty in a unified way, thereby embedding the classical image pyramid for displacement estimation. Consequently, the proposed approach will not improve convergence in such simple displacement scenarios but its goal is to automatically exploit a heterogeneous structure of more complex warps better than constant isotropic smoothing. A difficult setting when refining features can be seen in figure 4.7.

The next section relates the contribution to previous work before section 4.3.2 presents the gradient-based alignment problem and shows how parameter uncertainty can be incorporated for affine region alignment.

4.3.1 Related Work on Gradient-based Alignment

One of the first publications on gradient based image alignment is the work by Lucas and Kanade [1981]. In a stereo setting they stated that under the *Image Brightness Constancy Assumption* the correspondence problem can

be formulated as that of minimizing the grey value difference between corresponding points using Newton's method, provided the prediction is close to the true value. They also state that smoothing the image can increase the convergence radius. Since then, a vast quantity of articles has been published on extensions, improvements, accelerations and applications of this topic, which are referred to as Lucas-Kanade-methods in the remainder. The interested reader is referred to the work of Baker and Matthews [2004], which provides an excellent overview and comparison. An approach different from the low-parametric global model is often taken in optic flow measurement [Lefebure and Cohen, 2001], where usually a 2D displacement is estimated for each pixel leading to a huge number of parameters, which are only locally important. Additional regularization terms are applied to overcome the local aperture problem. In this work, the focus is on the case of estimating the parameters of one model warp typically with high redundancy (a large number of intensity measurements but few global warp parameters), where the influence of the uncertainty of the global parameters is inspected.

At least since the work of Tomasi and Kanade [1991], alignment and tracking has been performed on image pyramids in coarse-to-fine strategies, although this was handled rather as an implementation detail. For example, [Bergen et al., 1992] mentions that parameters are propagated from one pyramid level to the next. Christmas [1997] investigated the relation between smoothing and optical flow estimation in more detail, however he provided a specialized filter analysis for pure displacement only. Later, Molton et al. [2003] examined parametric image warps in a probability-theoretic framework. They were focused on formalizing and characterizing all sources of noise and to incorporate priors on the warp parameters. Although they already give a good intuition that "smoothing should be done over a range similar to the expected change of pixel position" they do not conclude that different pixels in a patch could be subject to different amounts of smoothing or that this smoothing could be anisotropic.

Uncertainty was also handled in other works [Steele and Jaynes, 2005, Dorini and Goldenstein, 2006], however, not incorporated into the minimization but rather viewed as an outcome. So far nobody considered the influence of parameter uncertainty within the grey value difference minimization. Therefore, in contrast to previous work it is proposed to propagate parameter uncertainty to pixel position uncertainty, which helps in selecting a good filter scheme. Then an implementation is given exploiting the image's scale space to obtain local convexity with high probability.

4.3.2 Parametric Image Alignment with Uncertainty

The image brightness constancy assumption states that corresponding points in two images have the same grey value, when the images are related by some warp function $\mathtt{W}$. According to Baker and Matthews [2004] the first image is referred to as the template T and the second as the image I, where the warp depends on some parameters $\boldsymbol{p}$ to relate coordinates between T and I.

$$\boldsymbol{x}_I = \mathtt{W}\left[\boldsymbol{x}_T, \boldsymbol{p}\right] \tag{4.20}$$

The intensity in the template is represented by the function $\mathcal{T}$ and in the image by $\mathcal{I}$, such that the image brightness constancy assumption states that for the true $\boldsymbol{p}$ the intensities in both images are the same:

$$\mathcal{I}\left[\mathtt{W}\left[\boldsymbol{x}, \boldsymbol{p}\right]\right] = \mathcal{T}\left[\boldsymbol{x}\right] \tag{4.21}$$

If a parameter prediction $\tilde{\boldsymbol{p}}$ is given that is sufficiently close to the true value, one may use Newton's method to find the parameters $\hat{\boldsymbol{p}}$ that minimize the squared sum of intensity differences at positions $\boldsymbol{x}_T$ from a patch. The term *patch* is used here for intuition, but for smooth warps such as affine transforms, $\boldsymbol{x}_T$ may be from a set P of sample points in an image. For example, for affine refinement of robust image features, a fixed grid attached to the local feature is used, such that the absolute number of samples does not depend on the size of the feature. To obtain an infinite homography for a purely rotated camera, one may select a number of samples uniformly distributed across the image. Although the contribution is not restricted to a particular alignment method, the inverse compositional approach (according to Baker and Matthews [Baker and Matthews, 2004]) is used here for demonstration. They assume that the warps $\mathtt{W}$ form a group with respect to the parameter $\boldsymbol{p}$, so that

$$\forall \boldsymbol{p}_{\text{forward}}\ \exists \boldsymbol{p}_{\text{backward}} : \mathtt{W}\left[\boldsymbol{x}, \boldsymbol{p}_{\text{forward}}\right]^{-1} = \mathtt{W}\left[\boldsymbol{x}, \boldsymbol{p}_{\text{backward}}\right]\ \forall \boldsymbol{x} \tag{4.22}$$

this way allowing to avoid explicit usage of inverses. Given a prediction $\tilde{\boldsymbol{p}}$, the method obtains an inverse compositional update $\Delta\hat{\boldsymbol{p}}$ in each iteration step

$$\Delta\hat{\boldsymbol{p}} = \text{argmin}_{\Delta\boldsymbol{p}} \sum_{\boldsymbol{x}_T \in P} \left(\mathcal{T}\left[\mathtt{W}\left[\boldsymbol{x}, \Delta\boldsymbol{p}\right]\right] - \mathcal{I}\left[\mathtt{W}\left[\boldsymbol{x}, \tilde{\boldsymbol{p}}\right]\right]\right)^2 \tag{4.23}$$

which is (inversely) composed into $\tilde{\boldsymbol{p}}$ for the next iteration. The equation system is based solely upon the gradients in $\mathcal{T}$ to estimate the missing transformation, which is assumed to be close to the identity transform. If $\tilde{\boldsymbol{p}}$ is very close to the true value, this means that $\Delta\boldsymbol{p}$ is nearly zero, the prediction

is in the convex surrounding of the minimum of the error function and the above sum can be linearized

$$\sum_{\boldsymbol{x}_T \in P} \left(\mathcal{T}\left[\mathtt{W}\left[\boldsymbol{x}_T, \mathbf{0}\right]\right] + \nabla\mathcal{T}\big|_{\boldsymbol{x}_T} \left.\frac{\partial \mathtt{W}}{\partial \boldsymbol{p}}\right|_{\boldsymbol{x}=\boldsymbol{x}_T, \boldsymbol{p}=\mathbf{0}} \Delta\boldsymbol{p} - \mathcal{I}\left[\mathtt{W}\left[\boldsymbol{x}_T, \tilde{\boldsymbol{p}}\right]\right] \right)^2 \tag{4.24}$$

which (assuming $\mathtt{W}\left[\boldsymbol{x}, \mathbf{0}\right] = \boldsymbol{x}$) leads to the solution

$$\Delta\boldsymbol{p} = H^{-1} \sum_{\boldsymbol{x}_T \in P} \left(\nabla\mathcal{T}\frac{\partial \mathtt{W}}{\partial \boldsymbol{p}}\right)^{\mathsf{T}} \left(\mathcal{I}\left[\mathtt{W}\left[\boldsymbol{x}_T, \tilde{\boldsymbol{p}}\right]\right] - \mathcal{T}\left[\boldsymbol{x}_T\right]\right), \tag{4.25}$$

where

$$H = \sum_{\boldsymbol{x}_T \in P} \left(\nabla\mathcal{T}\frac{\partial \mathtt{W}}{\partial \boldsymbol{p}}\right)^{\mathsf{T}} \left(\nabla\mathcal{T}\frac{\partial \mathtt{W}}{\partial \boldsymbol{p}}\right) \tag{4.26}$$

Baker and Matthews [2004] argued that the main advantage of this formulation is that most parts on the above equation can be precomputed as they do not depend on the image intensity $\mathcal{I}$. In this thesis, the approach is however chosen just as one example of alignment and the proposed technique can as well be applied to other formulations. All Lucas-Kanade methods linearize the local intensity signal, and in the following paragraphs the prerequisites for this linearization will be inspected.

The term in (4.24) is only a valid approximation of the term in (4.23) as long as $\tilde{\boldsymbol{p}}$ is quite correct. It states that near the position $\boldsymbol{x}$ the template has the grey value $\mathcal{T}\left[\boldsymbol{x}\right] + \nabla\mathcal{T}\frac{\partial \mathtt{W}}{\partial \boldsymbol{p}}$, which is only valid in a very small neighborhood. E.g. if $\boldsymbol{p}$ parameterizes translation and $\tilde{\boldsymbol{p}}$ is 10 pixels away from the true optimum, in presence of fine detail there may be multiple local extrema in between, which are not represented by the linear approximation.

Incorporating Uncertainty

In many previous tracking applications the images were either blurred with some predefined constant Gaussian kernel or a pyramid of a certain size was used to increase the basin of convergence. The actual amount of blurring to allow convergence (or the pyramid size) is then a system parameter, either empirically set [Bleser et al., 2006] or left to choose for an expert user [Zinßer et al., 2004]. If the parameters to be estimated are different from 2D displacement, this can be rather unintuitive.

While the idea is kept, that the image brightness constancy assumption is also valid at coarser scales, an appropriate scale is now computed automatically on a per-sample basis: It is assumed that the uncertainty of the

parameter vector is unimodal and characterized well by the first two moments of its distribution[2], mean $\tilde{\boldsymbol{p}}$ and covariance $\Sigma_{\tilde{\boldsymbol{p}}\tilde{\boldsymbol{p}}}$. Since the normal distribution has the maximum entropy of all distributions for a given mean and covariance, $\boldsymbol{p}$ is assumed being normal-distributed in the following. However, qualitatively the derivation also applies to the uniform distribution or other unimodal distributions. Now, let the warp $\mathtt{W}$ map coordinates of T to I. Next, it is investigated how much the coordinates change, when the parameters $\boldsymbol{p}$ change. Under the assumption that $\mathtt{W}$ is locally approximated well by its first order Taylor approximation (compare section A.1) linear error propagation yields:

$$\Sigma_{\boldsymbol{x}_I\boldsymbol{x}_I} \approx \frac{\partial \mathtt{W}}{\partial \boldsymbol{p}} \Sigma_{\tilde{\boldsymbol{p}}\tilde{\boldsymbol{p}}} \frac{\partial \mathtt{W}}{\partial \boldsymbol{p}}^{\mathsf{T}} \tag{4.27}$$

If $\mathtt{W}$ is actually linear, then $\boldsymbol{x}_I$ is normally distributed with covariance $\Sigma_{\boldsymbol{x}_I\boldsymbol{x}_I}$. Now the iso-density curve at 2σ is chosen that comprises nearly 90% of probability inside. The enclosed area is called the *target region*. In the following it is assumed that almost always the true correspondence $\check{x}_I$ is somewhere in the target region and that therefore linear intensity is required within this region. The shape and the size depend on the projection of the parameter uncertainty $\Sigma_{\tilde{\boldsymbol{p}}\tilde{\boldsymbol{p}}}$ into the image. First the simple case is considered that $\Sigma_{\boldsymbol{x}_I\boldsymbol{x}_I}$ has two equal eigenvalues. This means that $\boldsymbol{x}_I$'s distribution is an isotropic Gaussian with circular iso-density curves and that a point $\boldsymbol{x}_T$ is mapped to a disc around $\boldsymbol{x}_I$, whose radius l computes as

$$l = 2\sqrt{0.5\,\mathtt{trace}\left[\Sigma_{\boldsymbol{x}_I\boldsymbol{x}_I}\right]} \tag{4.28}$$

Then, an appropriate scale in Gaussian *scale space* (cf. to [Witkin, 1983, Lindeberg, 1994]) must be selected such that structures of smaller size are suppressed to a large extent and the region can be considered approximately linear. This can be achieved by convolution of the image with an isotropic Gaussian having standard deviation l. The grey value is then computed at this scale.

If on the other hand $\Sigma_{\boldsymbol{x}_I\boldsymbol{x}_I}$ has two different eigenvalues, this means that $\boldsymbol{x}_I$'s position is more uncertain in some direction. In this case, imagine that the image size is normalized, so that in the transformed image the uncertainty becomes isotropic again. Then the method of above can be applied. These operations can efficiently be combined by smoothing the image with a Gaussian filter with covariance $4\Sigma_{\boldsymbol{x}_I\boldsymbol{x}_I}$ and resampling. However, in both

[2] When prior knowledge about the parameter distribution is available, it may also be of advantage to incorporate this in terms of priors in Bayesian estimation as proposed in [Molton et al., 2003]. To avoid mixing up different effects, in this contribution the focus is on the intensity-related aspects when parameter uncertainty is available.

the isotropic and the anisotropic case the desired result is a single grey value only (not a whole filtered image), so basically the image convolution can be reduced to a single weighted sum of intensities in the target region.

In the same way, also the region in the template has to be computed, where an image sample at $\boldsymbol{x}_I$ is backward-mapped given the parameter prediction and its distribution. Requiring the warp to be invertible is no restriction, since inverse compositional alignment assumes the warp to be invertible anyway.

$$\Sigma_{\boldsymbol{x}_T\boldsymbol{x}_T} \approx \frac{\partial \left(\mathtt{W}\left(\boldsymbol{x}_\mathtt{I}, \cdot\right)\right)^{-1}[\boldsymbol{p}]}{\partial \boldsymbol{p}} \Sigma_{\tilde{\boldsymbol{p}}\tilde{\boldsymbol{p}}} \frac{\partial \left(\mathtt{W}\left(\boldsymbol{x}_\mathtt{I}, \cdot\right)\right)^{-1}[\boldsymbol{p}]}{\partial \boldsymbol{p}}^\mathsf{T} \tag{4.29}$$

This represents the region around $\boldsymbol{x}_T$ where the warp prediction maps an image position $\boldsymbol{x}_I$ into the template. Since linearity is desired within this region, one can proceed in the same way as with the image. Now, also the gradient has to be calculated at the obtained scale.

To summarize, it is proposed that each grey value is obtained using an individual level of smoothing such that it is linear within the predicted parameter uncertainty. Since each pixel can be chosen from the best resolution available, it is less likely that one runs into the aperture problem, which often happens when the whole patch is lifted to a very coarse level, because more information than necessary has been suppressed. In case the warp uncertainty leads to an anisotropic position distribution anisotropic smoothing should be applied at this position, e.g. for small purely rotational uncertainty smoothing is only required tangential to the warp. The scale and the shape of the smoothing will in general vary from pixel to pixel.

In early works (e.g. [Tomasi and Kanade, 1991]), where only 1D or 2D displacement was estimated, isotropic image smoothing or the use of image pyramids was suggested. This embeds perfectly into the proposed framework because in the case of pure displacement estimation, isotropic 2D parameter uncertainty leads to a constant and isotropic pixel position uncertainty ($\Sigma_{\boldsymbol{x}_I\boldsymbol{x}_I} = \Sigma_{\boldsymbol{p}\boldsymbol{p}}$) for all positions in the patch. This results from the fact that the Jacobian of the warp with respect to the parameters (the displacement) does not depend on the pixel position. Therefore, in the novel method all intensities would be picked from the same level in scale space or the same pyramid level, which is exactly what was proposed in earlier works. In the case of more complicated warps however, the more differentiated scheme presented above is the consequent generalization.

Alignment Algorithm with Uncertainty
Select set P of measurement positions in template and repeat until all positions are taken from the best resolution:

1. For each $\boldsymbol{x}_T \in P$ propagate parameter uncertainty $\Sigma_{\boldsymbol{pp}}$ to position uncertainty $\Sigma_{\boldsymbol{x}_T\boldsymbol{x}_T}$
2. Obtain template grey value and gradient (an)isotropically from template pyramid according to $\Sigma_{\boldsymbol{x}_T\boldsymbol{x}_T}$
3. Construct Hessian and Steepest Descent Images (same as in [Baker and Matthews, 2004])
4. Repeat until no significant improvement:
   (a) For each $\boldsymbol{x}_T \in P$ obtain image coordinates $\boldsymbol{x}_I$ using $\tilde{\boldsymbol{p}}$
   (b) propagate parameter uncertainty $\Sigma_{\boldsymbol{pp}}$ to position uncertainty $\Sigma_{\boldsymbol{x}_I\boldsymbol{x}_I}$
   (c) obtain (an)isotropic grey values from image pyramid according to $\Sigma_{\boldsymbol{x}_I\boldsymbol{x}_I}$
   (d) Compute residuals, solve for $\Delta\boldsymbol{p}$ and compose $\Delta\boldsymbol{p}$ with $\tilde{\boldsymbol{p}}$
5. Update covariance $\Sigma_{\boldsymbol{pp}}$
6. If parameter update or covariance is sufficient break, otherwise go to 1

Figure 4.8: Overview of alignment with uncertainty

Algorithm and Implementation

Now some details of the implementation are given (see figure 4.8 for an overview) and additionally, a second, more efficient, approximation for the considerations presented in section 4.3.2. As stated before, an initial estimate of the uncertainty for the parameter guess $\tilde{p}$ is required. For robust feature refinement such uncertainty estimates can be obtained e.g. from an empirical feature detector evaluation [Mikolajczyk et al., 2005] or from noise models [Steele and Jaynes, 2005]. As an approximation for the image's scale space, the Gauss pyramid with width and height reduced by a factor of two per level (as e.g. used in [Lowe, 2004]) can be used. Between the pixels of a level and between the levels, linear interpolation is applied, which is also known as trilinear filtering in computer graphics (see [Williams, 1983]). If anisotropic smoothing is required, first the smaller principal vector of the pixel covariance is determined and trilinear image values are extracted from the scale space. These values are then smoothed in direction of the larger principal vector. This exploits the pyramid and avoids anisotropic filtering with huge masks at full image resolution. This method is called *anisotropic* in the following.

Since often the parameter distribution is not known exactly but only its approximate shape, since additionally the linearization of the warp is sometimes only valid in a small range and since anisotropic smoothing is expensive, it is proposed even in case of an anisotropic covariance to simply pick the grey value directly from scale space according to equation (4.28): as the trace is the sum of the eigenvalues and the eigenvalues of a covariance matrix are the variances in principal directions, the trace can be seen as a rough upper bound of the maximum variance. This approximation is called the *scale* method in the following. In case of isotropic pixel position uncertainty the anisotropic and the scale approach are the same.

Having obtained the gradients and the image intensities as described above, the inverse compositional alignment is performed. In the minimum of the error function, the parameter covariance is estimated from the Hessian and the reference variance. This new covariance is then used in the next iteration, for which the template and the image is constructed again as described above (compare figure 4.9). Convergence of the system can be declared if all measurements (or some control measurements) are picked from the highest resolution. In this case the algorithm behaves as the original inverse compositional alignment.

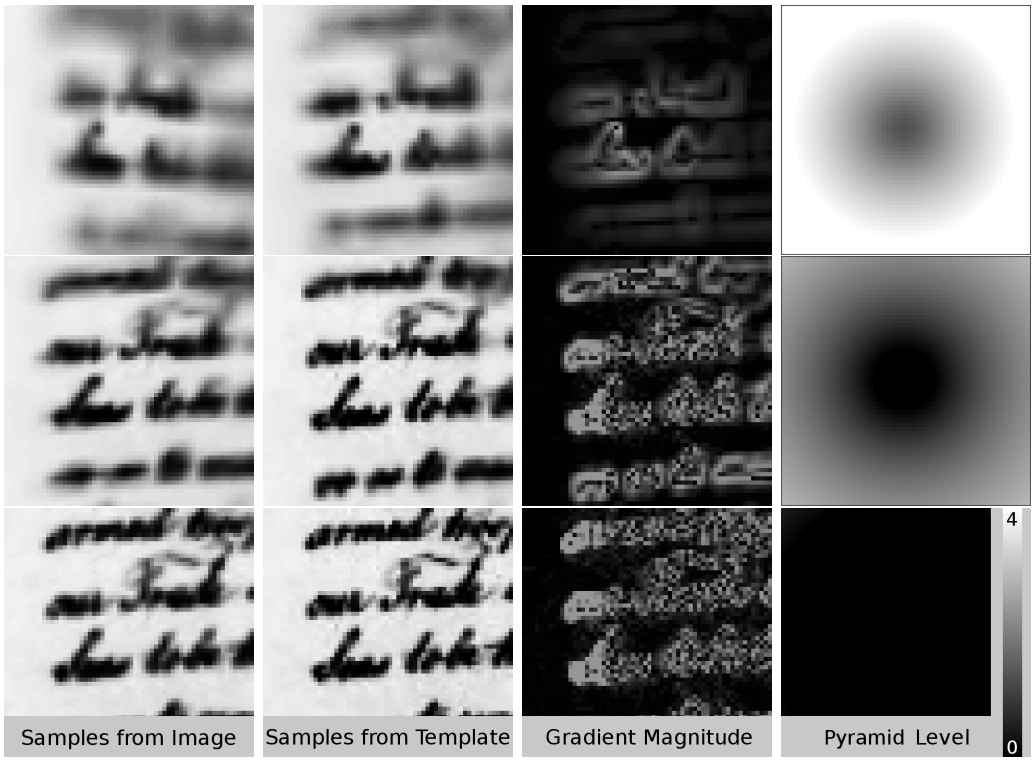

Figure 4.9: A patch containing hand-written text is aligned using the *scale* method. The initial affine warp to be estimated between the image in the first and the template in the second column contains a 15° rotation, a scale of 10% and a position offset of 1 pixel. The gradients used for estimation and the scale, where they have been taken from, can be seen in the right two columns (darker pixels represent lower values). Since initially the uncertainty is set appropriately for the missing transformation, particularly at the outer patch parts samples are picked from coarse resolutions (first row). With nearly compensated scale and rotation and improved uncertainty, finer details can be used in the second row. When all samples are taken from the highest resolution (third row) the algorithm behaves like the original inverse compositional alignment. Please note that the graphics do not show warped versions of continuous images, but rather sets of loose intensity (or gradient) measurements at some discrete positions. Instead of showing the individual intensity values in an equation system, here they are arranged in a 2D array for visualization. Because of sparse sampling these arrays are not suitable for reconstruction of the full image signal but targeted for parameter estimation only.

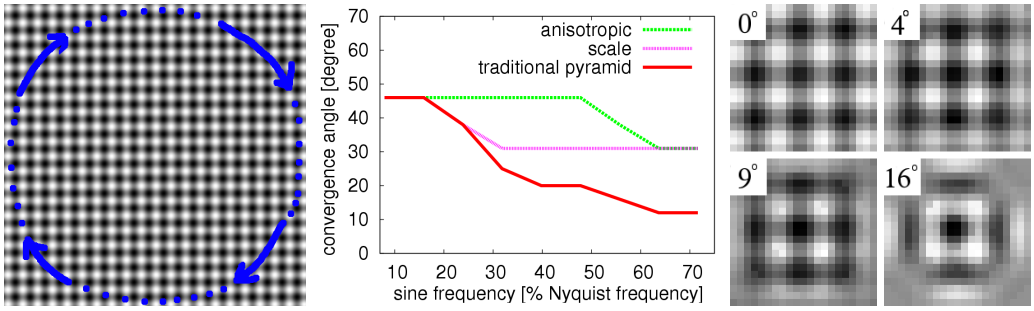

Figure 4.10: The left image shows a sum of a horizontal and a vertical sine-pattern. Such images have been created for different frequencies and each was rotated around its center as indicated by the arrows. In the center plot, the maximum angle for which gradient-based Euclidean parameter estimation converged for a 21×21 center patch is depicted in dependence of the sine-frequency. Scale and anisotropic are the new methods described in the previous sections, which are compared with a traditional pyramid approach. Particularly when very fine image structures close to the Nyquist frequency are present, both novel approaches outperform the rigid pyramid with respect to the convergence radius. The template values of the center patch computed for rotational uncertainties of 0, 4, 9 and 16 degrees with the anisotropic method can be seen in the right image.

4.3.3 Evaluation of the Optimization

In order to demonstrate the principle of the novel approach, first a very simple example is shown, where a 512×512 (floating point valued) template with intensity $\mathcal{T}[x, y] = \mathtt{sin}[\lambda\, x] + \mathtt{sin}[\lambda\, y]$ as depicted in figure 4.10 is used. This image is rotated around its center and afterwards Gaussian noise ($\sigma_I = 2\%$ of the sine amplitude) is added to each pixel. On this data a 3-parametric Euclidean warp (α, dx, dy) is estimated using the anisotropic and the scale method and a traditional pyramid-based approach for comparison, where the estimation is first performed on pyramid level three and then the results are down-propagated and refined on the next better resolution. 21×21 samples are used in a patch centered in the image and 0° is always provided as a rotation prediction, but with different uncertainties. It can be seen that with increasing sine frequency the pyramid approach converges only for smaller and smaller angles, while the anisotropic filtering almost always catches rotations of up to 45°. The scale approach has slightly worse convergence than the anisotropic but still better than the pyramid approach.

Next, the images of figure 4.11 have been chosen, where SIFT features

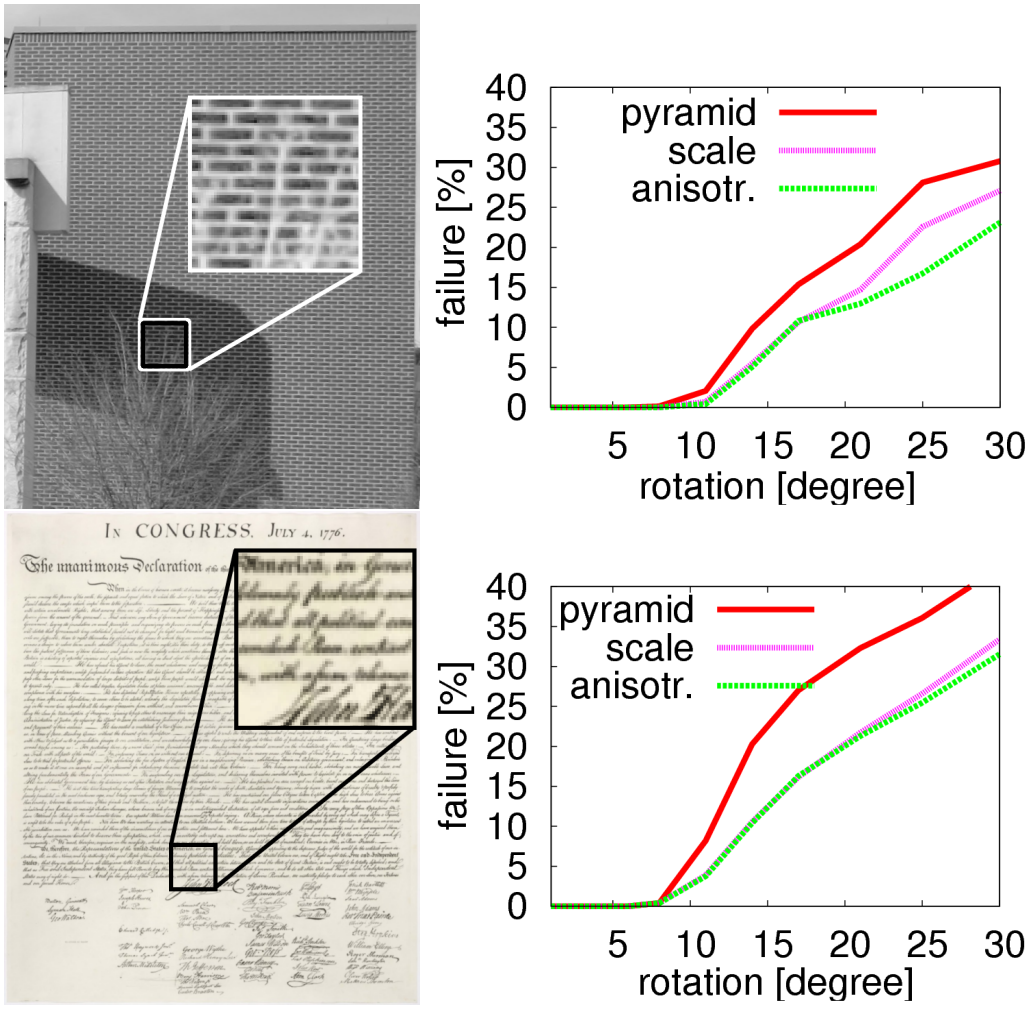

Figure 4.11: The images *bricks* and *declaration* have been rotated and the fraction of diverged alignments at SIFT-feature positions in the first image have been counted (right of each image). The starting position in the second image was correct, but the rotation was set to zero to obtain a convergence radius estimate for the rotation parameter. For the traditional pyramid method, the best pyramid layer is displayed, which provides still worse results than automatic individual smoothing. Note that these images contain very fine structures (see detail magnifications) which are almost filtered out in the classical coarse-to-fine strategy. In the novel approaches, they are used if possible (see also figure 4.9).

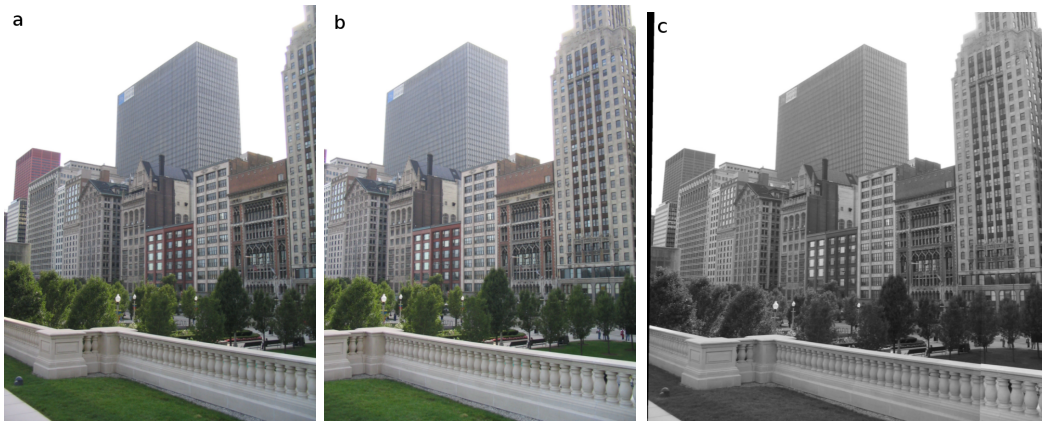

Figure 4.12: The two left images a and b (1536×2048, ≈ 40° field of view) have been taken with a digital camera, which purely rotated. An estimate of the rotation was given within 1° degree accuracy, from which an infinite homography could be predicted. Given prediction and uncertainty the homography has been optimized and the resulting parameters have been used to stitch the images (c). The optimization has been run upon 20×20 samples only, distributed uniformly across the three mega pixel image. No heuristic smoothing or manual selection of a "good" pyramid level was applied. Note that this is an extremely challenging situation because of the frequency content. Remaining errors may be due to lens distortion, camera movement and changed illumination.

were detected followed by a rotation of the images. Around each feature 21×21 samples have been used in a square window 10 times the detection σ (cf. to [Lowe, 2004]). Then Euclidean parameters have been estimated with correct position prediction but with no rotation prediction. When the rotation was estimated worse than 0.05 rad (≈ 2.9°), a failure has been recorded. The graphs show that for very small rotation errors all approaches converged, but for larger rotational errors the novel approaches diverge less frequently than the traditional pyramid approach, presumably because they are better at exploiting fine structures.

In the next experiment the automatic scale approach is demonstrated based on an extremely sparse set of samples. Gradient based homography estimation is applied for a real pair of photos containing high-frequency patterns of skyscrapers. No heuristic smoothing or *some good pyramid level* had to be selected. Instead, a prediction for the homography parameters was approximated by propagating the rotational uncertainty of 1° (see figure (4.12) for details). For such warps with higher numbers of parameters, heuristic

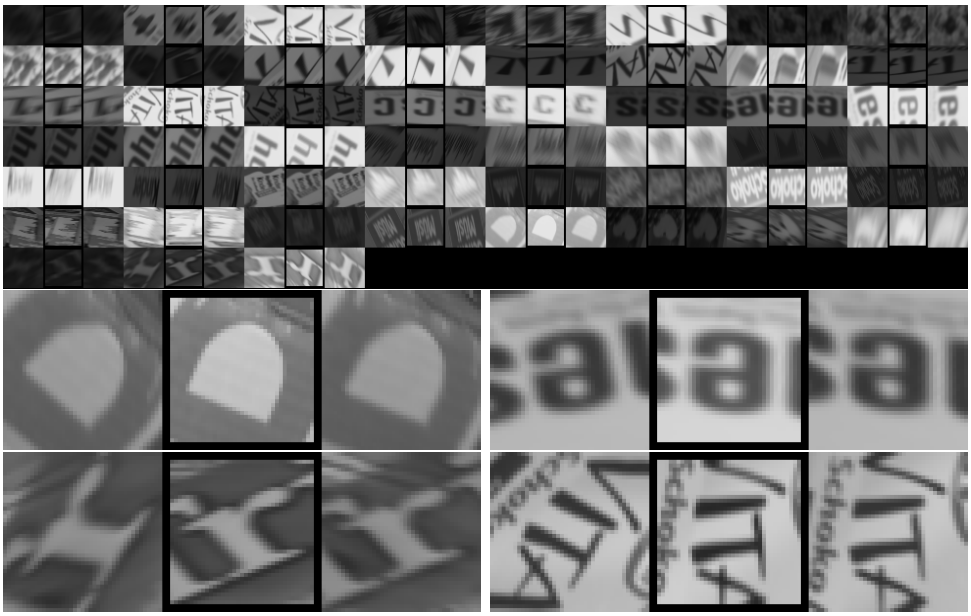

Figure 4.13: The matched features of figure 4.2 have been refined using the anisotropic approach based on an empiric covariance estimate. The top row shows all potential correspondences including mismatches. In each triple, the left image is before refinement, as the feature detector found it, the center is the feature from the other image and the right is after refinement. It can be seen that in most cases the right and the center image look very similar, even when the optimization started far away. Four correspondences are enlarged at the bottom.

smoothing becomes really involved, while the novel framework solves this problem automatically as long as the uncertainty is in the correct range.

Finally, some affine features have been detected in two images as displayed in figure 4.2, automatically matched, and the resulting correspondences have been further optimized as shown in figure 4.13. It can be seen that the optimization works for starting values quite far from the correct solution.

4.3.4 Summary on Gradient-based Optimization

Image pyramids have been used in gradient-based displacement estimation for a long time to increase the convergence radius. When more complex parametric image transformations were considered, the pyramid concept has simply been adopted in the literature so far or the images under inspection had to be provided smooth enough for convergence. In this section a framework has been presented, which incorporates parameter uncertainty into the

registration process working in scale space. The system selects the required amount of smoothing automatically on a per-sample basis, which allows to keep more detail of the original image and therefore is less susceptible to the aperture problem. It can be seen as a generalization of the pyramid concept traditionally used in displacement estimation. Although the evaluation showed superiority using local feature alignment, the concept can be applied to a much broader range of parameter estimation applications such as camera tracking or homography estimation.

4.3.5 Practical Optimization for LAF correspondences

Additionally to the novel concept for incorporating uncertainty into the grey-value optimization process, the Bayesian optimization model according to Molton et al.[Molton et al., 2003] is applied, i.e. a prior on the warp parameters is used in optimization. Furthermore, instead of strict Gauss-Newton-optimization a line-search mechanism is applied, which accepts only improvements of the error function and otherwise reduces the step size, similar to a Levenberg-Marquardt optimization. Since often the photometric appearance of local features changes, this is handled using a affine brightness model. The embedding of such appearance models in the estimation process is described in [Baker et al., 2003]. When the start value for optimization is obtained from local image features, the relative affine brightness parameters from the detectors can be used as an initialization.

When the affine approximation of the warp is not sufficient the residual grey value differences will be large. One possibility to solve this problem is to perform model selection (as proposed e.g. by Torr [1997] for two-view geometry) and, in case the affine approximation is not well suited, to estimate the next higher Taylor representation. The higher coefficients should be close to zero and can consequently be initialized with zero. Since they represent even more global properties than the first derivatives, estimating them from a local region is more uncertain than for the first derivatives. However, if they allow for correct estimation of the first derivatives they can be neglected afterwards.

4.4 The Local Affine Frame as an Uncertain Measurement

If a LAF correspondence is obtained from noisy image data, it is only an estimate of the true LAF correspondence. This section now derives the uncertainty concept. Three different cases of obtaining the correspondence are

considered here:

4.4.1 Obtaining and Representing Uncertainty

Obtained from Feature Detector

When LAFs are detected in each image independently, the parameters of each individual LAF are uncertain. In this thesis the pdf of the LAF of equation (4.5) as a 6-vector $\boldsymbol{l}_{LAF}$

$$\boldsymbol{l}_{LAF} = (f_x \; f_y \; a_{11} \; a_{12} \; a_{21} \; a_{22})^\mathsf{T} \tag{4.30}$$

is represented by two moments, a mean and a covariance. These can be obtained e.g. from noise models[Steele and Jaynes, 2005] or from empirical feature detector evaluations[Mikolajczyk and Schmid, 2004a]. If the uncertainty of the transformation between the two regions is desired, error propagation as described in appendix B.4 can be used to obtain the uncertainty of the LAF correspondence.

Obtained from Image Alignment

Given such a relative uncertainty for the LAF correspondence, the image alignment of section 4.2.5 can be applied to obtain better correspondence parameters for equation (4.12).

$$\boldsymbol{l}_{LAFC} = (d_x \; d_y \; a_{\text{rel},11} \; a_{\text{rel},12} \; a_{\text{rel},21} \; a_{\text{rel},22})^\mathsf{T} \tag{4.31}$$

In the minimum of the error function, the reference variance and the structure of the normal equation system from the Lucas-Kanade minimization directly lead to an estimate of the uncertainties (see also section C.1.3).

To obtain individual uncertainties for each LAF, error propagation can again be exploited to concatenate the first LAF's uncertainty and the transformation's uncertainty to obtain the second feature's uncertainty. If the first feature is defined to be an absolute reference its uncertainty can be set to zero.

Obtained From Three or More Point Correspondences

In the same way as three point correspondences can be sampled from a LAF correspondence using the triangle transform (compare section 4.2.4), a LAF correspondence can be approximated using three or more close point correspondences. This holds because three point correspondences define an affine transformation, which may be obtained using DLT. If uncertainty is given

for the points, then error propagation can be used to obtain an uncertainty estimate for the LAFs. For more than three points the system is overdetermined and a least squares solution can be obtained including a covariance estimate for the parameters.

Relating Displacement and Linear Warp Parameters

If the uncertainty of a patch corner is given, this allows to infer the uncertainties of the affine parameters under some reasonable assumptions:

Assume that a patch corner is mapped by the affine transform to some position in the other image according to equation (4.12), where the patch center is assumed to be in the origin. Also, its movement is disregarded here and only the linear warp of the patch is considered now. Then the corner with offset $(d, d)^\mathsf{T}$ from the patch center is mapped to a new offset $\boldsymbol{c}$ from the transformed patch center in the other image, caused by the linear warp A_{xy}:

$$\boldsymbol{c} = A_{xy}(d\ d)^\mathsf{T} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} (d\ d)^\mathsf{T} \tag{4.32}$$

Rewriting this equation in the entries a_{ij} of A yields:

$$\boldsymbol{c} = D\boldsymbol{a} \qquad \text{where} \quad \boldsymbol{a} = \begin{pmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \end{pmatrix} \quad \text{and} \quad D = \begin{pmatrix} d & d & 0 & 0 \\ 0 & 0 & d & d \end{pmatrix} \tag{4.33}$$

This shows that the linear warp parameters map linearly with D to the new corner position. Given the covariance $\Sigma_{\boldsymbol{cc}}$ of the warped corner, the question arises, whether something can be stated for the covariance $\Sigma_{\boldsymbol{aa}}$ of the affine parameters. Since the above equation is linear, linear error propagation (see section B.4) relates these uncertainties:

$$\Sigma_{\boldsymbol{cc}} = D\Sigma_{\boldsymbol{aa}}D^\mathsf{T} \tag{4.34}$$

If, for symmetry reasons, it is assumed that all four entries of A have the same uncertainty σ_a and are uncorrelated, their distribution can be expressed by a diagonal covariance $\Sigma_{\boldsymbol{aa}}$:

$$\Sigma_{\boldsymbol{aa}} = \sigma_a^2 I_{4\times 4} \tag{4.35}$$

If on the other hand an isotropic empirical uncertainty σ_c of the patch corner is assumed, linear error propagation yields:

$$\sigma_c^2 I_{2\times 2} = \sigma_a^2 D I_{4\times 4} D^\mathsf{T} = 2d^2\sigma_a^2 I_{2\times 2} \tag{4.36}$$

This leads to the relation

$$\sigma_c = \sqrt{2}\,|d|\,\sigma_a \tag{4.37}$$

between the uncertainty of a transformed position and the uncertainty of the warp parameters. This means, that for a patch of half window size $d = 30$, some isotropic uncertainty of the linear warp parameters leads to an approximately 40 times[3] larger position uncertainty of the transformed patch corners (plus a potential uncertainty of the patch center). If on the other hand it is known that the corners of the patch can be estimated (e.g. by an affine tracker) with an uncertainty of 1 pixel, the above assumptions provide 0.02 as an estimate for the upper bound of A's uncertainty, and typically some of the uncertainty is already due to an overall patch position uncertainty.

4.4.2 Empiric Covariance

The uncertainty of the LAF correspondence depends on various sources of noise and inaccuracies of the model as described in the previous paragraphs. Particularly the uncertainty of the linear warp parameters is different from the uncertainty of the displacement parameters. In the previous section the uncertainty footprints of the affine parameters have been derived, i.e. their impact on a warped patch corner uncertainty. Since the error of the corners of a patch are due to position and linear warp error, a a plausible ratio for moderately sized patches would be in the range of 1%, although in practice this certainly depends on the image data. In maximum likelihood estimation or other algorithms that require an estimate of the covariance, therefore an *empiric covariance* for the relative parameters is set to

$$\Sigma_{\text{empiric}} = \begin{pmatrix} 1 & & & & & & \\ & 1 & & & & & \\ & & 0.0001 & & & & \\ & & & 0.0001 & & & \\ & & & & 0.0001 & & \\ & & & & & 0.0001 & \end{pmatrix} \tag{4.38}$$

if no other information is available.

[3] For clarity's sake, here and in the remainder of the thesis the unit *pixel* for image distances is left out. It is assumed that all image measurements are made in pixel, i.e. that σ_c and d in equation (4.37) are given in this unit. In the same sense it is assumed that the units used when relating objects of different domains are given consistently.

4.4.3 Incidence and Outlier Detection

When automated matching techniques are applied, often RANSAC-like methods (see appendix C.1.7) are used to perform a model-based verification of the matches: given a LAF correspondence in two images, the question may arise whether these two observations are consistent with some transformation between these two images. For instance, if two images are related by a known homography, it might be asked whether the LAF correspondence can be a reasonable match or whether it is highly unlikely to measure such a correspondence (a potential outlier) given the known homography. For simple point correspondences, the distance between the predicted and the measured position can be thresholded, but for local affine frames this is more involved.

For instance, when evaluating descriptor performance in wide baseline matching, Mikolajczyk and Schmid [2005] were faced with the problem to automatically reject matches not only when the position of the feature was inconsistent with a known transformation, but also if the features did not cover the same area (e.g. a large feature and a small feature but with consistent position). To solve this problem they used a scalar measure called the overlap error, which was introduced already in the evaluation of feature detectors [Mikolajczyk et al., 2005]. This measure is based upon the ratio of the union and the intersection of two regions. The drawback of this measure is that it completely ignores the local orientation of the region. To account for this, Köser and Koch [2007] introduced the oriented overlap error, which additionally penalizes different orientations of the LAFs.

Still, in these geometrically motivated measures it is difficult to incorporate uncertainty, because some LAFs may have been measured more accurately than others and also within a LAF the uncertainty of the parameters may vary. Therefore, in the following section statistical means (as proposed in [McGlone, 2004, p.77]) will be exploited to reject matches when there is statistical evidence that they are inconsistent with a model transform.

Incidence with Respect to Affine Transforms

For simplicity first assume that two images are related by an affine transform $\mathtt{H}$ and that the parameters $\boldsymbol{l}^{I1}$ and $\boldsymbol{l}^{I2}$ (according to equation (4.30)) of the two LAFs $\mathbf{A}^{I1}$ and $\mathbf{A}^{I2}$ of both features are normally distributed with given mean and covariance. First, only the LAF in image 1 is inspected: It can be transformed to a new LAF in the second image using the linearization $\mathbf{H}_{\text{Taylor}}$ of $\mathtt{H}$ according to equation (4.19). For affine functions, the linearization of the function is the function itself, so that $\mathbf{H}_{\text{Taylor}}$ is only the matrix notation

of H here:

$$\mathbf{A}_{I1}^{I2} = \mathbf{H}_{\text{Taylor}}\, \mathbf{A}^{I1} \qquad \text{where } \mathbf{H}_{\text{Taylor}} = \begin{pmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ 0 & 0 & 1 \end{pmatrix} \tag{4.39}$$

As can easily be seen this is also an affine transformation in the parameters $\boldsymbol{l}^{I1}$ of the local affine frame $\mathbf{A}^{I1}$:

$$\boldsymbol{l}_{I1}^{I2} = M\boldsymbol{l}^{I1} + \boldsymbol{b} \tag{4.40}$$

where $\boldsymbol{b}$ and M can be computed from rearranging the previous equation. Therefore the parameter uncertainty transfers with the Jacobian (or the linear part) M of this transform, which is not an approximation because the transform is affine:

$$M = \begin{pmatrix} v_{11} & v_{12} & 0 & 0 & 0 & 0 \\ v_{21} & v_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & v_{11} & 0 & v_{12} & 0 \\ 0 & 0 & 0 & v_{11} & 0 & v_{12} \\ 0 & 0 & v_{21} & 0 & v_{22} & 0 \\ 0 & 0 & 0 & v_{21} & 0 & v_{22} \end{pmatrix} \tag{4.41}$$

Consequently, in image 2 the transformed local affine frame is normally distributed with mean $\boldsymbol{l}_{I1}^{I2}$ and covariance

$$\Sigma_{\boldsymbol{l}_{I1}^{I2}\boldsymbol{l}_{I1}^{I2}} = M\Sigma_{\boldsymbol{l}^{I1}\boldsymbol{l}^{I1}}M^{\mathsf{T}} \tag{4.42}$$

Now, the parameters $\boldsymbol{l}^{I2}$ of the original LAF in the second image are considered. The difference between $\boldsymbol{l}_{I1}^{I2}$ and $\boldsymbol{l}^{I2}$

$$\boldsymbol{l}_{\text{diff}} = \boldsymbol{l}_{I1}^{I2} - \boldsymbol{l}^{I2} \tag{4.43}$$

is then also normally distributed (see appendix B.2.1) with covariance

$$\Sigma_{\boldsymbol{l}_{\text{diff}}\boldsymbol{l}_{\text{diff}}} = \Sigma_{\boldsymbol{l}_{I1}^{I2}\boldsymbol{l}_{I1}^{I2}} + \Sigma_{\boldsymbol{l}^{I2}\boldsymbol{l}^{I2}} \tag{4.44}$$

If the two original LAFs are in correspondence, then $\boldsymbol{l}_{\text{diff}}$ should have zero mean and its squared Mahalanobis distance d_0^2 to the origin must be χ^2-distributed.

$$d_0^2 = \boldsymbol{l}_{\text{diff}}^{\mathsf{T}}\Sigma_{\boldsymbol{l}_{\text{diff}}\boldsymbol{l}_{\text{diff}}}\boldsymbol{l}_{\text{diff}} \tag{4.45}$$

If the probability of measuring a squared distance equal to or larger than d_0^2 is small, the incidence hypothesis can be rejected (compare appendix B.3.1). This means there is statistical evidence that the measured LAF correspondence is inconsistent with the assumed model and should be classified as a mismatch or an outlier.

Incidence with Respect to Homographies

A homography can locally be linearized using an affine transformation. Under the assumption that the linearization is approximately the same for the measured and the true position of the affine feature, using linear error propagation, the LAF of the first image including its covariance can approximately be transferred into the second image. Then the same method as before can be applied.

4.4.4 Maximum-Likelihood Estimation

In the last sections it has been shown how the LAF correspondence can be viewed as an uncertain measurement. Therefore, when LAF correspondences are obtained in the presence of noise, the question arises how a fair weighting between unequally reliable observations can be done. Maximum Likelihood estimation is a method to find the most likely set of model parameters that can explain such observations (details on estimation can be found in appendix C.1.2).

More formally, assume that $\boldsymbol{l}_{LAFC,i}$ are the means of n observed, normally distributed LAF correspondences (for $i \in \{1,..,n\}$) according to equation (4.31). Let the covariance of each such observation be encoded into a 6×6 covariance matrix $\Sigma_{\boldsymbol{l}_{LAFC,i}\boldsymbol{l}_{LAFC,i}}$. It is assumed now that $\mathtt{H}_{\boldsymbol{h}}[\boldsymbol{x}]$ is a model function for which the most likely parameter set $\boldsymbol{h}$ is searched. For instance, if $\mathtt{H}$ is a homography then the eight degrees of freedom of $\mathbf{H}$ can be encoded into an 8-vector $\boldsymbol{h}$ (cf. to section 2.3.2). If $\boldsymbol{l}_{LAFC,i}$ is a perfect noise-free observation at position $\boldsymbol{x}_i$, then the local linearization of $\mathtt{H}$ at that position must equal that measured linear warp, so that the residual $\boldsymbol{r}_i$ is zero

$$\boldsymbol{r}_i = \boldsymbol{l}_{LAFC,i} - \begin{pmatrix} \mathtt{H}_{\hat{\boldsymbol{h}}}[\boldsymbol{x}_i] - \boldsymbol{x}_i \\ \mathtt{vec}\left[\left.\frac{\partial \mathtt{H}_{\hat{\boldsymbol{h}}}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_i}\right] \end{pmatrix} \tag{4.46}$$

If $\boldsymbol{l}_{LAFC,i}$ is noisy however, then $\boldsymbol{r}_i$ should still be small for the correct $\boldsymbol{h}$. The approach can then be reformulated in the Gauss-Markov model (cf. to [McGlone, 2004]) as a minimization problem to find the most likely $\hat{\boldsymbol{h}}$:

$$\hat{\boldsymbol{h}} = \mathrm{argmin}_{\boldsymbol{h}} \sum_i \boldsymbol{r}_i^{\mathsf{T}} \Sigma^{-1}_{\boldsymbol{l}_{LAFC,i}\boldsymbol{l}_{LAFC,i}} \boldsymbol{r}_i \tag{4.47}$$

Since $\boldsymbol{l}_{LAFC,i}$ is normally distributed, this is the maximum likelihood estimator for h[Mikhail and Ackermann, 1976]. The minimization can be solved e.g. using any of the Newton-like algorithms of section C.1.4.

4.4.5 Evaluation: Measuring and Finite Area Approximation

In the previous section, the geometrical model for local affine frame correspondences has been derived and it was shown how a given local affine frame correspondence imposes constraints. Of course, such *given* correspondences are usually *measured* from images and in this section the principal feasibility of obtaining such correspondences is shown and how the model assumptions fit to the real world.

When classical point correspondences were determined between images, often the assumption of constant displacement has been made. This means that a constant size window is compared to candidate windows in the other image (as e.g. in [Skoglund and Felsberg, 2006]) or that in a parametric model constant displacement is assumed for a whole window to obtain the offset parameters (as e.g. in [Lucas and Kanade, 1981]). Consequently, the infinitely small point is in fact measured by a finite area.

In the same way, here also the Jacobian of the warp has to be measured using a finite area. This weakens the assumptions about the transformation compared to point matching since no constant displacement is assumed but only a constant change of displacement (the Jacobian). However, still, when a window-based method is exploited to compute a simple local 6-parametric warp as common in affine trackers, the assumption is that the higher order derivatives are neglectable in the window.

If the warp is known to be a homography, it is also possible to parameterize the homography using the six LAF correspondence parameters and two infinity parameters (see e.g. [Köser et al., 2008]). This avoids linearization of the homography at the price of estimating two additional parameters. For other transformations than homographies, instead, the second derivatives could be estimated if they are not neglectable in the window. So, even if non-linearities are observable in the window, there are ways to avoid a linearization error when the Jacobian is estimated.

Nevertheless, for small regions a constant Jacobian can be assumed and measured using a simple six-parametric affine warp. As explained in detail in section A.2 for homographies, the region should be far from the preimage of the line at infinity because the Jacobian changes very strongly near this region. To evaluate this behavior, in figure 4.14 a plane has been rotated in 3D and the texture warp between an orthophoto and an oblique view onto the surface is computed. The affine transform is then compared to the linearization of the ground truth homography. A warped local region can be seen in figure 4.15 for the views. In figure 4.16 the quality of the estimated local linearization is plotted as the Mahalanobis distance of the LAF

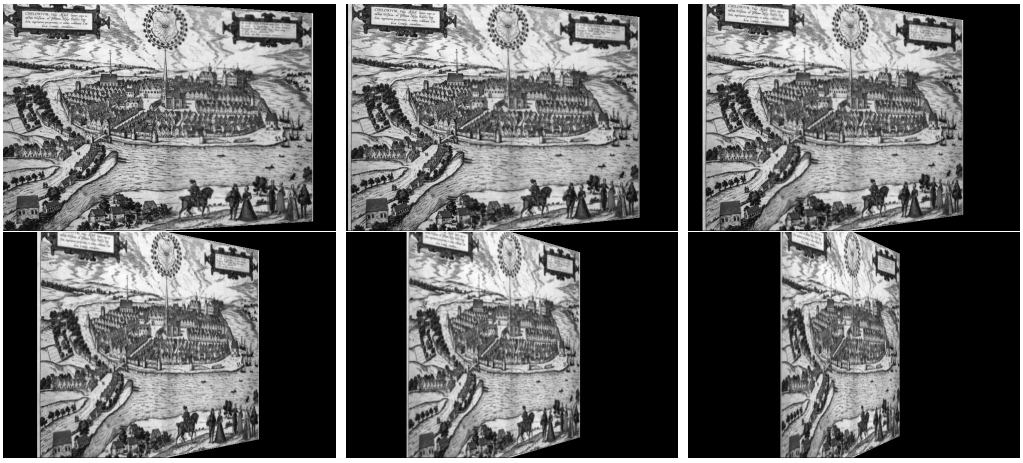

Figure 4.14: Rotated Plane in 3D by 20°, 30°, 40° (upper row) and 50°, 60°, 70° (lower row). The images are related to the orthophoto through a homography.

Figure 4.15: Affine approximation of homographies of figure 4.14. Local features have been warped from the 10° to 70° views into the orthophoto using gradient-based image registration (affine warp). The final regions are displayed for one sample feature. Apart from the decreasing x-resolution due to the oblique angle, nearly no visual differences are noticeable.

correspondence to the first order Taylor approximation of the ground truth homography using the constant empiric covariance of the previous section. In this evaluation, it can be seen that the quality of first order Taylor approximation does not depend so much on the average viewing angle but on the *change of viewing angle* and therefore change of Jacobian within a patch.

This must be seen in comparison with the small baseline matching problem (point correspondences), used e.g. in standard stereo: In such scenarios fixed rectangular windows in the two images are compared and exploited to find the correspondence. This works well as long as the offset between the images is approximately constant for the whole window. The offset is the zero order Taylor approximation of the local warp. With the LAF correspondence now this assumption is relaxed to the first order Taylor approximation. Here, the offset may vary within a patch, but only in a linear fashion, i.e. the change of the offset variation should be approximately constant within the patch.

If on the other hand the Jacobian is visibly not constant across the patch and the curving disturbs the Jacobian estimation for the center, these nonlinearities can be measured and even exploited. A way to cope with this is to apply a higher order model, e.g. to directly measure the last two degrees of freedom for a homography or to simply compute second derivatives in the patch. If the second derivatives can be measured reliably (e.g. for relatively large patches), the model presented in this chapter can be extended to second derivatives in a straightforward way. If on the other hand they can only be measured approximately this leads to extrapolation errors far from the Taylor approximation point. However, even if the second derivatives are not absolutely correct, it is still possible that they locally compensate for nonlinear effects, allowing for a better estimate of the first derivatives and thus the LAF correspondence. In that case after having stabilized estimation of the Jacobian, the exact value of the second derivatives is of no interest. The evaluation of the second derivatives is however out of scope of this thesis.

4.5 Relation to other Primitives

The presented work is actually related to the 1D curvature concept used by Schmid and Zisserman [2000]. At the same time it can also be viewed in the context of conic correspondence, where it is more powerful than traditional conics and leads to simpler equations in homography estimation. In a way the simplest relation is to a set of point correspondences which are spatially very close. This relation is analyzed first.

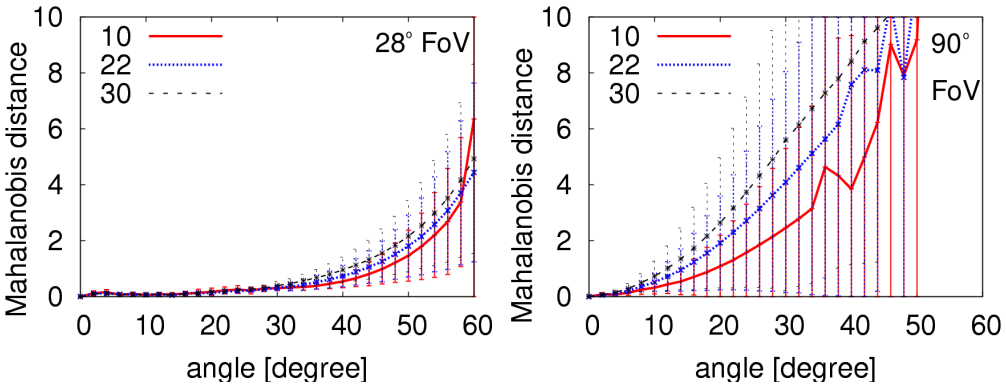

Figure 4.16: Affine approximation of oblique angle homographies using affine warps with different half window sizes. In this evaluation, a plane has been rotated in front of a camera, which is the same relative transformation as if a camera is moved along an orbit around a fixed plane. In the left figure the plane is seen from further away using a more zoomed in focal length of 3200 pixels while rotating. In the right figure the plane is imaged from a closer distance with a very wide angle (90° field of view) focal length of 400 pixels. Local windows distributed across the image have been tracked between an orthophoto and a rotated view based upon an affine warp only. The Mahalanobis distance based on a constant empiric covariance (see section 4.4.2) is plotted on the y-axis. It can be seen that the error is larger when the window is larger (compare the three curves for three different window sizes). Additionally, the affine fit to the homography degrades with increasing viewing angle (from 0° to 60°). However, the most important observation is that the affine approximation is much better in the left figure with large focal length and small field of view. A plausible reason is that in this setting the Jacobian changes much less in a window (compare also appendix A.2 and figure A.2).

4.5.1 Triple of Points

An affine transformation has six DOF and is determined by three point correspondences in general. Consequently, a LAF correspondence and three point correspondences carry comparable geometric information.

In the work of Chum et al. [2003], Perdoch et al. [2006], it was proposed to estimate epipolar geometry from only three resp. two affine feature correspondences. In each of these, three point correspondences are obtained using an extra detection step in the local affine frame. This converts the set of affine feature correspondences into a three times larger set of point correspondences. Riggi et al. [Riggi et al., 2006] on the other hand proposed to sample the affine feature directly using some fixed coordinates in the LAF, and do not require to detect and match additional points (called triangle transform in section 4.2.4). This sampling (also called triangle transform in this thesis) has the potential drawback, that if the sampling radius is chosen too large, the three point correspondences have a bias towards an affine transform. If three independent point correspondences are actually detected as in [Chum et al., 2003, Perdoch et al., 2006] this is avoided at the cost of additionally detecting and matching these points.

In any case, three points that are very close to one another can lead to numerical difficulties in practical point-based estimation algorithms. Aside from that, the LAF correspondence is equivalent to three infinitesimally close point correspondences and sampling the LAF correspondence can be thought of as a numerical derivative of the image transform.

4.5.2 Conic Correspondence

In this section the differential feature concept is shown to be a simplified version of correspondences of conics, providing more constraints in a linear (instead of quadratic) fashion.

First, it is shown how the two primitives used in the LAF correspondence can be related to conic representations: For each affine feature, e.g. MSER, there exists a local image coordinate system, the *local affine frame*, such that coordinates can be specified relative to the size, shear, position and orientation of a feature. Imagine that $\mathbf{L}$ is a local affine frame according to equation (4.5) and takes (projective) points from local feature coordinates to image coordinates:

$$\mathbf{x}_I = \mathbf{L}\mathbf{x}_{\text{LAF}} \tag{4.48}$$

If the same feature is seen in two images, points with identical feature coordinates will (ideally) have the same grey value. The local affine frames of the features in the different images are then called $\mathbf{L}_1$ and $\mathbf{L}_2$ and their

concatenation is the first order Taylor approximation $\mathbf{H}_{\text{Taylor}}$ of the texture warp (e.g. a homography) between the two images at the feature positions (compare also equation (4.19)):

$$\mathbf{H}_{\text{Taylor}} = \mathbf{L}_1\mathbf{L}_2^{-1} \iff \mathbf{L}_1 = \mathbf{H}_{\text{Taylor}}\mathbf{L}_2 \tag{4.49}$$

Now just think of a single image and imagine a small circle through the points $(0;\lambda)^\mathsf{T}$,$(\lambda;0)^\mathsf{T}$,$(0;-\lambda)^\mathsf{T}$ and $(-\lambda;0)^\mathsf{T}$ of the local feature coordinate system. It can be represented by a conic equation such that all points x at the circle contour fulfill the quadratic constraint:

$$0 = \mathbf{x}_{\text{LAF}}^\mathsf{T} \begin{pmatrix} 1 & & \\ & 1 & \\ & & -\lambda^2 \end{pmatrix} \mathbf{x}_{\text{LAF}} \tag{4.50}$$

In image coordinates this is an ellipse and the conic equation reads as

$$0 = \mathbf{x}_i^\mathsf{T}\mathbf{L}^\mathsf{T} diag(1,1,-\lambda^2)\mathbf{L}\mathbf{x}_i \tag{4.51}$$

The LAF described as a conic matrix would therefore be

$$\mathbf{C}_\lambda = \mathbf{L}^\mathsf{T} \begin{pmatrix} 1 & & \\ & 1 & \\ & & -\lambda^2 \end{pmatrix} \mathbf{L} = \mathbf{L}^\mathsf{T}\mathbf{R}^\mathsf{T} \begin{pmatrix} 1 & & \\ & 1 & \\ & & -\lambda^2 \end{pmatrix} \mathbf{R}\mathbf{L} \tag{4.52}$$

where $\mathbf{R}$ is an arbitrary (homogeneous 2D) rotation matrix, which cancels out. Therefore the first thing to observe is that 2D orientation of the feature is lost in conic representation. A conic has only five degrees of freedom and a conic correspondence therefore imposes at most five constraints on any $\mathbf{H}$.

$$\mathbf{C}_1 = \mathbf{H}^\mathsf{T}\mathbf{C}_2\mathbf{H} \tag{4.53}$$

Furthermore, these constraints are quadratic in the entries of $\mathbf{H}$ as can be seen from equation (4.53). This equation is also essentially a squared version of equation (4.49).

As a side note, the LAF correspondence is available when sufficient texture is in the image, while the conic correspondence traditionally exploits a special geometric shape (typically an ellipse contour) and ideal perspective cameras and ideal planes because conic curve estimation in distorted cameras is more involved. In contrast, the differential feature concept can also directly be applied in fish-eye or omnidirectional cameras.

Relation to the Absolute Conic

The suppression of orientation information in conic representation is a property that is sometimes desirable, for instance in self-calibration (e.g. [Hartley, 1994]). Here, the absolute conic (AC) is a virtual conic on the plane at infinity that maps into a perspective camera regardless of the orientation of the camera. Its projection into the image, the image of the absolute conic (IAC), therefore depends only on the internal camera parameters and can be exploited for self-calibration. The relation between the AC and the IAC cannot be expressed as a LAF correspondence because virtual conics do not contain any real points, at which the mapping function could be linearized. Furthermore, even for the real points in the 2D coordinate system of the plane at infinity no regular affine warp can be obtained into a pinhole camera image. The reason for this is that there would be an infinite local scaling between the two coordinate systems. However, the infinite homography between two images almost always[4] has a finite Jacobian and therefore allows for linearization and application of LAF correspondences.

Many self-calibration approaches are based upon virtual quadric-to-conic correspondences or conic-to-conic correspondences, which make the formulation independent of the camera orientation. However, most of these approaches lead to quadratic formulations (compare [Frahm, 2005]). Frahm showed [Frahm, 2005, Frahm and Koch, 2003] that the quadratic self-calibration equations arising from the rotation-invariant formulation become linear if rotation information is available and exploited. This idea is comparable to the findings of the previous section about the relation of conics and local affine frames.

4.5.3 Curves

Schmid and Zisserman [2000] inspected the behavior of planar curves under homography transformation. Compared to affine features, curves can usually be thought of as locally 1-dimensional image regions and there exists a 1D (scalar) parameter τ, which allows running along the curve $\mathcal{C}$, i.e. reaching all points on the curve:

$$\mathcal{C} : \left\{ \boldsymbol{x} \in \mathbb{R}^2 | \exists \tau \in \mathbb{R} : \mathcal{C}\left[\tau\right] = \boldsymbol{x} \right\} \tag{4.54}$$

For each point on the curve the tangent to the curve defines the local direction. In the following, for simplicity it is assumed that τ is from a natural

[4] Since the infinite homography must have full rank by construction and homographies map lines to lines, it must map a line to the line at infinity. Outside this line, which is a null set in the plane, the Jacobian is finite.

(arc length) parameterization of $\mathcal{C}$. Then the tangent direction is given by

$$\texttt{tangent}\left[\boldsymbol{x}\right] = \frac{\partial \boldsymbol{x}}{\partial \tau} \tag{4.55}$$

The rate of direction change of the curve is called the curvature κ. Therefore, for each (euclidean 2D) point on the curve, a scalar curvature measure can be obtained:

$$\kappa\left[\boldsymbol{x}\right] = \left\| \frac{\partial^2 \boldsymbol{x}}{\partial \tau^2} \right\|_2 \tag{4.56}$$

Analogous to a first order Taylor approximation, a point and the tangent of a curve are local properties. The curvature (as the change of the tangent) is already a more global property. The more global properties are taken into account (the higher order derivatives of the curve) the more information can be obtained about the unknown transformation when two curves are corresponding in two images. However, as for the second derivatives of a function, physically obtaining the curvature requires also a larger support region for measurement.

Schmid and Zisserman derived how this curvature changes when the curve is transformed by a homography. Here, they exploited the behavior of the tangent vector. Compared to the local affine frame, where two independent basis vectors are attached to the point this carries less information and therefore provides less constraints, however, it can be considered as being a one-dimensional version of the LAF correspondence.

If many points on a curve are considered with their curvature, this can fully determine the homography or transformation under consideration. The same is also true if multiple LAF correspondences are considered on a textured surface.

4.6 Summary

This chapter presented a geometrical representation for the local affine frame (LAF) and showed how a LAF correspondence imposes constraints onto a global warping function through its first order Taylor representation. These local linear shape change constraints can be explained by concatenating region-normalization transformations used in photometric matching (photometric interpretation) or by a covariantly transformed local coordinate system (physical interpretation).

In practice the LAFs can be obtained from state-of-the-art robust image features or e.g. the KLT tracker and are often readily available. Additionally, simpler correspondences (e.g. corners) can often be upgraded to such a

full first order Taylor model and also feature refinement is possible through gradient-based optimization, for which a generalization of the traditional pyramid-approach has been proposed to increase the basin of convergence. Finally, correspondences of LAFs as a geometric primitive have been related to triangle correspondences, to change of curvature and to conic correspondences, which are essentially sampled LAFs, related to 1D LAFs or a squared formulation (respectively) of the LAF approach.

In the next section the model is exemplarily applied to several estimation problems of computer vision, such as homography, pose or normal estimation. It is shown how the local shape change constraints can be exploited to obtain direct solutions for these problems with fewer feature correspondences or more reliably than when only the position information is used as in traditional point-based approaches and how maximum-likelihood estimation is possible based on uncertain LAF measurements.

Chapter 5

Applications: Geometric Estimation with Local Affine Frame Correspondences

As explained in the previous chapter, the LAF correspondence (point and local linear warp) is directly related to the Taylor representation of the warp. A major benefit of this parameterization of the correspondence is that once it is determined using image data as proposed in chapter 4, the obtained parameters can be used afterwards in a purely geometrical way: the representation is not restricted to a single special method but the same measured correspondence data can be used in various algorithms, be it for the precise 2D localization in tracking, the estimation of a conjugate rotation, the camera pose, or to obtain the 3D surface normal. This is the same as when dealing geometrically with point correspondences: it is not strictly required for each algorithm to go back to the image data and reparameterize in different sets of parameters: the six parameter representation is general and can be seen as an extension of the point correspondence.

In this chapter it is exploited now to obtain constraints onto the unknown transformation. Of particular interest in the last years were solutions that are based on very few data or even minimal solutions [Brown et al., 2007, Nistér, 2004, Chum et al., 2003, Perdoch et al., 2006, Riggi et al., 2006, Köser et al., 2008, Köser and Koch, 2008a, Kyle, 2004, Nistér and Stewénius, 2006, Stewénius, 2005]. In the case of a minimal solver, a model with n DOF is estimated from an observation providing exactly n DOF. Often multiple observations are required and each observation may be m-dimensional or imposes m constraints (e.g. $m_{\mathrm{p}} = 2$ for a two-dimensional point or $m_{\mathrm{LAF}} = 6$ for a LAF in homography estimation). From a counting argument, if n is an integer multiple of m, then a solution made up from $\frac{n}{m}$ observations is a

minimal solution. The solutions presented in the following are either minimal or quasi-minimal, which means here that the solution is based upon $\lceil \frac{n}{m} \rceil$ observations (the next full integer number), if $\frac{n}{m}$ is not an integer. Consequently, such a quasi-minimal solution uses only slightly more data than theoretically required, e.g. when models with an odd number of parameters are estimated from two-dimensional observations, theoretically only "half" of the last observation is required[1].

Solutions that require (quasi-)minimal sets of data are particularly important in RANSAC-like approaches, where the probability to select an all-inlier-set decreases exponentially with the number of correspondences required. In this field it has been shown that sampling an affine feature into three point features improves RANSAC performance for fundamental matrix estimation [Chum et al., 2003], because now only three instead of seven correspondences are required to construct a hypothesis. Consequently, in the next section such (quasi-)minimal solutions are derived based on the LAF correspondence, decreasing the number of feature correspondences required up to a factor of three.

Although these transformations are not restricted to homographies, these are very important and quite often used (see appendix A.2 for a discussion on homographies in Euclidean space). Particularly since the correspondences are small, it is often reasonable to assume planar surfaces across these small regions and also to approximate the small image part using an ideal pinhole camera, even if it is part of a fish-eye image. Consequently, in this chapter the power of the LAF correspondence is demonstrated using homographies although the concept may as well be used for estimating radial distortion or for working with curved surfaces.

Since the LAF correspondence provides six constraints, a lower bound on the number of correspondences required to obtain a model is obvious. In the following section, the estimation of a general homography (8 DOF), a conjugate rotation (7 DOF), camera or object pose (6 DOF) and 3D position and normal of a patchlet (5 DOF) is shown.

[1] An example is the estimation of the projection matrix from six 2D-3D point correspondences in the DLT algorithm [Hartley and Zisserman, 2000, pp.167]. The six observed points in the image add up to 12 measurements, while the projection matrix has only 11 DOF. In case there is no noise on these points they all agree on the true projection matrix, otherwise the redundancy can be exploited to obtain a least squares solution. One could also argue that if only the x-coordinate of the last point is used, then DLT would be a minimal solver. However, in presence of noise it is usually preferable to exploit all data available, and when points are measured each does usually have two coordinates in practice.

5.1 General Homography

5.1.1 Previous Work on Homography Estimation

According to the work summarized in section 3, a general homography has eight DOF and can be computed from four points or lines [Hartley and Zisserman, 2004] or from two conic correspondences[Kannala et al., 2006]. Since each conic can provide five constraints this is in agreement with counting. However, several matrix factorizations are required and due to the quadratic nature of the conic correspondence, the authors obtain four possible solutions. In this section, the computation of a general homography from two affine correspondences is derived using the differential constraint from equation (4.15). In the next paragraphs, a unique solution is obtained in a direct, non-iterative way.

5.1.2 Obtaining a General Homography from Two Feature Correspondences

A general homography $\mathtt{H}$ is a transformation with eight degrees of freedom, which maps points $\boldsymbol{x}$ from one (image) plane to points $\boldsymbol{y}$ in another (image) plane:

$$\mathtt{H}\left[\boldsymbol{x}\right] = \boldsymbol{y} = \mathtt{euc}\left[\mathbf{H}\ \mathtt{hom}\left[\boldsymbol{x}\right]\right]$$

In projective space, homographies are linear transformations and can be represented by 3×3 matrices:

$$\mathbf{H} = \begin{pmatrix} \boldsymbol{h}_1^\mathsf{T} & \boldsymbol{t} \\ \boldsymbol{h}_2^\mathsf{T} & \\ \boldsymbol{h}_3^\mathsf{T} & \lambda \end{pmatrix} \qquad \boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3, \boldsymbol{t} \in \mathbb{R}^2, \quad \lambda \in \mathbb{R} \tag{5.1}$$

If $\mathtt{H}$ is considered as a mapping from $\mathbb{R}^2$ to $\mathbb{R}^2$, the homogenization must be taken into account and the mapping is non-linear:

$$\mathtt{H}\left[\boldsymbol{x}\right] = \begin{pmatrix} \frac{\boldsymbol{h}_1^\mathsf{T}\boldsymbol{x}+t_x}{\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda} \\ \frac{\boldsymbol{h}_2^\mathsf{T}\boldsymbol{x}+t_y}{\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda} \end{pmatrix} \tag{5.2}$$

The derivative is therefore not constant, but a function of $\boldsymbol{x}$:

$$\frac{\partial \mathtt{H}}{\partial \boldsymbol{x}}\left[\boldsymbol{x}\right] = \frac{1}{(\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda)^2} \begin{pmatrix} \boldsymbol{h}_1^\mathsf{T}(\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda) - \boldsymbol{h}_3^\mathsf{T}(\boldsymbol{h}_1^\mathsf{T}\boldsymbol{x}+t_x) \\ \boldsymbol{h}_2^\mathsf{T}(\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x}+\lambda) - \boldsymbol{h}_3^\mathsf{T}(\boldsymbol{h}_2^\mathsf{T}\boldsymbol{x}+t_y) \end{pmatrix} \tag{5.3}$$

To estimate the parameters of such a mapping using affine feature correspondences, the differential constraint of equation (4.15) is applied. First, it is assumed that a LAF correspondence is available with local affine frames

$$\mathbf{A}_G^{I1} = \begin{pmatrix} A_G^{I1} & \boldsymbol{x}_G \\ 0 \quad 0 & 1 \end{pmatrix} \qquad \text{and} \qquad \mathbf{A}_G^{I2} = \begin{pmatrix} A_G^{I2} & \boldsymbol{y}_G \\ 0 \quad 0 & 1 \end{pmatrix} \tag{5.4}$$

However, to improve readability, initially the first image's coordinate system is changed by moving the affine feature at $\boldsymbol{x}_G$ to the origin and the second image's coordinate system by moving the corresponding feature at $\boldsymbol{y}_G$ to the origin, i.e. now a transformation $\mathtt{G}$ in which the origin is a fixpoint must be estimated:

$$\mathtt{G}\left[\mathbf{0}\right] = \mathbf{0} \tag{5.5}$$

$$\mathbf{G} = \mathbf{K}_{G2}^{-1}\mathbf{H}\mathbf{K}_{G1} \tag{5.6}$$

where the displacement matrices $\mathbf{K}_*$ look like this:

$$\mathbf{K}_{G1} = \begin{pmatrix} I_{2\times 2} & \boldsymbol{x}_G \\ 0 \quad 0 & 1 \end{pmatrix}, \mathbf{K}_{G2} = \begin{pmatrix} I_{2\times 2} & \boldsymbol{y}_G \\ 0 \quad 0 & 1 \end{pmatrix} \tag{5.7}$$

Since these matrices are triangular, it is easy to observe that their determinant equals 1. This implies that the normalized $\mathbf{G}$ and $\mathbf{H}$ from equation (5.6) have the same determinant:

$$\mathtt{det}\left[\mathbf{G}\right] = \mathtt{det}\left[\mathbf{H}\right] \tag{5.8}$$

From equation (5.5) it follows that $\mathbf{G}$ must look like this:

$$\mathbf{G} = \begin{pmatrix} \boldsymbol{g}_1^\mathsf{T} & \mathbf{0}_2 \\ \boldsymbol{g}_2^\mathsf{T} & \\ \boldsymbol{g}_3^\mathsf{T} & \lambda_G \end{pmatrix} \tag{5.9}$$

The derivative of $\mathtt{G}$ as an $\mathbb{R}^2 \to \mathbb{R}^2$ mapping is

$$\frac{\partial \mathtt{G}}{\partial \boldsymbol{x}}\left[\boldsymbol{x}\right] = \frac{1}{(\boldsymbol{g}_3^\mathsf{T}\boldsymbol{x} + \lambda_G)^2} \begin{pmatrix} \boldsymbol{g}_1^\mathsf{T}(\boldsymbol{g}_3^\mathsf{T}\boldsymbol{x} + \lambda_G) - \boldsymbol{g}_3^\mathsf{T}(\boldsymbol{g}_1^\mathsf{T}\boldsymbol{x}) \\ \boldsymbol{g}_2^\mathsf{T}(\boldsymbol{g}_3^\mathsf{T}\boldsymbol{x} + \lambda_G) - \boldsymbol{g}_3^\mathsf{T}(\boldsymbol{g}_2^\mathsf{T}\boldsymbol{x}) \end{pmatrix} \tag{5.10}$$

which, at the feature position, must equal the concatenated linear warp A_G of the local affine frames (see equations (4.11) and (4.12))

$$A_G = A_G^{I1} A_G^{I2\,-1} \tag{5.11}$$

according to equation (4.15):

$$\left.\frac{\partial \mathbb{G}}{\partial \boldsymbol{x}}\right|_{\mathbf{0}_2} = A_G \tag{5.12}$$

Substituting the parameters of $\mathbf{G}$ yields:

$$\begin{pmatrix} \boldsymbol{g}_1^\mathsf{T} \\ \boldsymbol{g}_2^\mathsf{T} \end{pmatrix} = \lambda_G A_G \tag{5.13}$$

Obviously, $\boldsymbol{g}_3$ is still undetermined, which essentially means that there is no information yet which points map to infinity. One affine feature correspondence is not sufficient to fix a general homography with its eight degrees of freedom. Analogous considerations can be made when a homography must be estimated from three close points (a small triangle), which also does not completely determine all parameters.

Now a second correspondence at $\boldsymbol{x}_F$ in image 1 and $\boldsymbol{y}_F$ in image 2 is considered to fix the remaining degrees of freedom. If a simple point-to-point correspondence is used, one might assume at first sight, that it would provide two more constraints and hence determines the homography. However, a closer look at the remaining degrees of freedom reveals that mainly the preimage of the line at infinity (see section 2.3.2) is still unconstrained because the last row of $\mathbf{G}$ has not been determined so far. One additional point correspondence can not determine the line at infinity. Algebraically, using another point correspondence creates only new equations, in which the scalar product of the new point and the last line of $\mathbf{G}$ appear, this way leaving an ambiguity for its individual entries. Geometrically this means that the feature at $\boldsymbol{x}_F$ does only contribute information in direction from $\boldsymbol{x}_G$ to $\boldsymbol{x}_F$, but does not completely determine the whole behavior.

Therefore, the second correspondence is assumed to be a LAF correspondence and is treated in the same way as the first: A normalized homography $\mathbf{F}$ is constructed using the normalization matrices $\mathbf{K}_{F0}$ and $\mathbf{K}_{F1}$, which now move $\boldsymbol{x}_F$ respective $\boldsymbol{y}_F$ to the origins:

$$\mathbf{F} = \mathbf{K}_{F2}^{-1}\mathbf{H}\mathbf{K}_{F1} \tag{5.14}$$

Since equation (5.8) applies also to $\mathbf{F}$, one may set[2] $\lambda_F = 1$, choose λ_G such

[2]Please note that choosing $\lambda_F = 1$ does not imply $\lambda \neq 0$ and therefore keeps the generality of the approach. To show that $\lambda_F = 1$ is reasonable, it is first argued that it is the ninth parameter in an up-to-scale representation of a 3×3 matrix. Since only relative sizes are of interest in projective entities, one entry can be fixed. Special care must however be taken if infinite ratios would occur, i.e. if $\lambda_F = 0$. This can however not be the case because $\lambda_F = 0$ means that $\mathbf{F}$ no longer transforms the origin ($\mathbf{x}_O = (0,0,1)^\mathsf{T}$) to a finite position. By construction of $\mathbf{F}$ however, it is known that the origin is a fixpoint of $\mathbf{F}$ (equation (5.5)), therefore λ_F cannot be zero.

that

$$\mathtt{det}\left[\mathbf{G}\right] = \lambda_G \mathtt{det}\left[\begin{pmatrix} \boldsymbol{g}_1^\mathsf{T} \\ \boldsymbol{g}_2^\mathsf{T} \end{pmatrix}\right] = \mathtt{det}\left[\begin{pmatrix} \boldsymbol{f}_1^\mathsf{T} \\ \boldsymbol{f}_2^\mathsf{T} \end{pmatrix}\right] = \mathtt{det}\left[\mathbf{F}\right] \tag{5.15}$$

Note that neither determinant of the above can become zero unless the affine feature degenerates to a line, so that λ_G can be computed

$$\lambda_G = \sqrt[3]{\frac{\mathtt{det}\left[A_F\right]}{\mathtt{det}\left[A_G\right]}} \tag{5.16}$$

Now, $\mathbf{F}$ and $\mathbf{G}$ are determined up to two parameters, and equation (5.16) constrains also λ in the searched homography $\mathbf{H}$, because all three homographies differ only by a coordinate system offset:

$$\mathbf{H} = \mathbf{K}_{G2}\mathbf{G}\mathbf{K}_{G1}^{-1} = \mathbf{K}_{F2}\mathbf{F}\mathbf{K}_{F1}^{-1} \tag{5.17}$$

Expanding the above matrix equation yields

$$\mathbf{H} = \begin{pmatrix} 1 & & \boldsymbol{y}_G \\ & 1 & \\ & & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{g}_1^\mathsf{T} & \\ \boldsymbol{g}_2^\mathsf{T} & \\ \boldsymbol{g}_3^\mathsf{T} & \lambda_G \end{pmatrix} \begin{pmatrix} 1 & & -\boldsymbol{x}_G \\ & 1 & \\ & & 1 \end{pmatrix} =$$

$$\begin{pmatrix} 1 & & \boldsymbol{y}_F \\ & 1 & \\ & & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{f}_1^\mathsf{T} & \\ \boldsymbol{f}_2^\mathsf{T} & \\ \boldsymbol{f}_3^\mathsf{T} & 1 \end{pmatrix} \begin{pmatrix} 1 & & -\boldsymbol{x}_F \\ & 1 & \\ & & 1 \end{pmatrix} \tag{5.18}$$

Using

$$\boldsymbol{x}_F = \begin{pmatrix} x_{F_1} \\ x_{F_2} \end{pmatrix} \qquad \boldsymbol{y}_F = \begin{pmatrix} y_{F_1} \\ y_{F_2} \end{pmatrix} \qquad \boldsymbol{x}_G = \begin{pmatrix} x_{G_1} \\ x_{G_2} \end{pmatrix} \qquad \boldsymbol{y}_G = \begin{pmatrix} y_{G_1} \\ y_{G_2} \end{pmatrix}$$

this can be expanded to:

$$\mathbf{H} = \begin{pmatrix} g_{11} + g_{31}y_{G1} & g_{12} + g_{32}y_{G_1} & -\boldsymbol{g}_1^\mathsf{T}\boldsymbol{x}_G + y_{G1}(-\boldsymbol{g}_3^\mathsf{T}\boldsymbol{x}_G + \lambda_G) \\ g_{21} + g_{31}y_{G2} & g_{22} + g_{32}y_{G_2} & -\boldsymbol{g}_2^\mathsf{T}\boldsymbol{x}_G + y_{G2}(-\boldsymbol{g}_3^\mathsf{T}\boldsymbol{x}_G + \lambda_G) \\ g_{31} & g_{32} & -\boldsymbol{g}_3^\mathsf{T}\boldsymbol{x}_G + \lambda_G \end{pmatrix}$$

$$= \begin{pmatrix} f_{11} + f_{31}y_{F1} & f_{12} + f_{32}y_{F1} & -\boldsymbol{f}_1^\mathsf{T}\boldsymbol{x}_F + y_{11}(-\boldsymbol{f}_3^\mathsf{T}\boldsymbol{x}_F + 1) \\ f_{21} + f_{31}y_{F2} & f_{22} + f_{32}y_{F2} & -\boldsymbol{f}_2^\mathsf{T}\boldsymbol{x}_F + y_{12}(-\boldsymbol{f}_3^\mathsf{T}\boldsymbol{x}_F + 1) \\ f_{31} & f_{32} & -\boldsymbol{f}_3^\mathsf{T}\boldsymbol{x}_F + 1 \end{pmatrix} \tag{5.19}$$

The upper left part of $\mathbf{G}$ and $\mathbf{F}$ is known from the affine matrices (compare equation (5.12)). The scale λ_F and λ_G are determined as pointed out before. Thus, the only remaining parameters are $\boldsymbol{f}_3$ and $\boldsymbol{g}_3$.

$$\begin{aligned}
g_{31}y_{01} - f_{31}y_{11} &= -g_{11} + f_{11} \\
g_{32}y_{01} - f_{32}y_{11} &= -g_{12} + f_{12} \\
y_{01}(-\boldsymbol{g}_3^\mathsf{T}\boldsymbol{x}_0 + \lambda_G) - y_{11}(-\boldsymbol{f}_3^\mathsf{T}\boldsymbol{x}_1) &= \boldsymbol{g}_1^\mathsf{T}\boldsymbol{x}_0 - \boldsymbol{f}_1^\mathsf{T}\boldsymbol{x}_1 + y_{11} \\
g_{31}y_{02} - f_{31}y_{12} &= -g_{21} + f_{21} \\
g_{32}y_{02} - f_{32}y_{12} &= -g_{22} + f_{22} \\
y_{02}(-\boldsymbol{g}_3^\mathsf{T}\boldsymbol{x}_0 + \lambda_G) - y_{12}(-\boldsymbol{f}_3^\mathsf{T}x_1) &= \boldsymbol{g}_2^\mathsf{T}\boldsymbol{x}_0 - \boldsymbol{f}_2^\mathsf{T}\boldsymbol{x}_1 + y_{12} \\
\boldsymbol{g}_3^\mathsf{T}\boldsymbol{x}_0 - \boldsymbol{f}_3^\mathsf{T}\boldsymbol{x}_1 - \lambda_G &= -1
\end{aligned}$$

The last row of a homography is the line that is mapped to infinity (pre-image of the line at infinity) because all points that lie on this line have a zero scalar product with this row. From equation (5.19) it can be seen that the pre-images of the lines at infinity under $\mathbf{G}$, $\mathbf{F}$ and $\mathbf{H}$ are parallel lines, because the line normals $\boldsymbol{f}_3^\mathsf{T}$, $\boldsymbol{g}_3^\mathsf{T}$ and $\boldsymbol{h}_3^\mathsf{T}$ are equal and this is reasonable since the coordinate systems in which these matrices work differ only by a displacement (the feature position). All that remains to obtain $\mathbf{H}$ is to determine this pre-image of $\mathbf{H}$.

$$\begin{aligned}
h_{31}y_{01} - h_{31}y_{11} &= -g_{11} + f_{11} \\
h_{32}y_{01} - h_{32}y_{11} &= -g_{12} + f_{12} \\
y_{01}\lambda_G - h_{31}y_{01}x_{01} - h_{32}y_{01}x_{02} + h_{31}y_{11}x_{11} + h_{32}y_{11}x_{12} &= \boldsymbol{g}_1^\mathsf{T}\boldsymbol{x}_0 - \boldsymbol{f}_1^\mathsf{T}\boldsymbol{x}_1 + y_{11} \\
h_{31}y_{02} - h_{31}y_{12} &= -g_{21} + f_{21} \\
h_{32}y_{02} - h_{32}y_{12} &= -g_{22} + f_{22} \\
y_{02}\lambda_G - h_{31}y_{02}x_{01} - h_{32}y_{02}x_{02} + h_{31}y_{12}x_{11} + h_{32}y_{12}x_{12} &= \boldsymbol{g}_2^\mathsf{T}\boldsymbol{x}_0 - \boldsymbol{f}_2^\mathsf{T}\boldsymbol{x}_1 + y_{12} \\
h_{31}x_{01} + h_{32}x_{02} - h_{31}x_{11} - h_{32}x_{12} - \lambda_G &= -1
\end{aligned}$$

Rewritten in matrix notation, this yields

$$\begin{pmatrix}
y_{01} - y_{11} & 0 \\
0 & y_{01} - y_{11} \\
y_{11}x_{11} - y_{01}x_{01} & y_{11}x_{12} - y_{01}x_{02} \\
y_{02} - y_{12} & 0 \\
0 & y_{02} - y_{12} \\
y_{12}x_{11} - y_{02}x_{01} & y_{12}x_{12} - y_{02}x_{02} \\
x_{01} - x_{11} & x_{02} - x_{12}
\end{pmatrix}
\begin{pmatrix} h_{31} \\ h_{32} \end{pmatrix} = \qquad (5.20)$$

$$\begin{pmatrix} f_{11} - g_{11} \\ f_{12} - g_{12} \\ \boldsymbol{g}_1^\top \boldsymbol{x}_0 - \boldsymbol{f}_1^\top \boldsymbol{x}_1 + y_{11} - \lambda_G y_{01} \\ f_{21} - g_{21} \\ f_{22} - g_{22} \\ \boldsymbol{g}_2^\top \boldsymbol{x}_0 - \boldsymbol{f}_2^\top \boldsymbol{x}_1 + y_{12} - \lambda_G y_{02} \\ \lambda_G - 1 \end{pmatrix}$$

There are seven constraints left on the two parameters, however the last one is actually a constraint on λ and refers to the fact that a total of nine parameters are used for a model with eight degrees of freedom.[3] In fact this is a redundancy of four. As a comparison, using triangle decomposition on two affine feature correspondences as proposed in the previous section, one would end up with six points and thus 12 constraints on the eight entries of $\mathbf{H}$. Now however, the triangle corners can be imagined as being infinitesimally close to the center so that the triangle collapses to a point. In both cases two affine features provide a redundancy of four constraints. With perfect noise-free data these constraints are linearly dependent because they must agree on the true solution. With noisy data however, these additional constraints can help finding a good solution, e.g. using least squares techniques.

As can be seen in equation (5.20) the matrix on the left hand side basically contains the distances between the two features in the same image, and the condition number of the least squares equation system approximately scales quadratically with the coordinate scale. Therefore, from empirical evaluation, it was found that the condition number is improved when the coordinates in both images are normalized (i.e. scaled) so that the distance of the features is about $1/\sqrt{2}$.

5.1.3 Generalizing to n Correspondences

As can be seen, each of the two LAF correspondences determines only six of the eight degrees of freedom; therefore the set of possible solutions is a 2-dimensional manifold in the space of all possible homographies. Using both correspondences, the intersection of these manifolds determines the final solution. In the same way as described above, also a third or n additional LAF correspondences can be used to obtain a direct, non-iterative solution. This is the same principle as in DLT, where each point correspondence provides two

[3] When a homography is estimated using the DLT algorithm[Hartley and Zisserman, 2004] also nine parameters are estimated and for perfect data this results in a one-dimensional solution space. Usually, the solution with Frobenius norm 1 is picked then. This "rule" can be regarded as another constraint to obtain a unique solution.

constraints and determines a 7-dimensional subspace in the 9-dimensional space of all homographies. The intersection of four of these 7-dimensional subspaces yields (in the general case) the one-dimensional solution space, where in DLT the solution with norm 1 is picked (but all solutions are equivalent). The drawback of DLT and also the least squares-solution using many LAF correspondences is that they optimize unintuitive algebraic criteria: in presence of noise they do not necessarily guarantee an optimal average projection error or weight individual uncertainties of the correspondences. These can however be considered in a subsequent maximum-likelihood estimation step, which finds the optimal solution regarding the measurements and their uncertainties, given a good parameter guess and which is discussed in section C.1.2.

5.1.4 Evaluation

In this section first the direct analytic solution (the raw version as well as the one with normalized coordinates to improve conditioning) is compared against the solution obtained through DLT based upon the triangle transform of section 4.2.4 and the sensitivity of both methods to noise is inspected. Next a closer look is taken at the sampling in the triangle transform before finally the estimation based upon multiple features is inspected.

For the first evaluation 100 general, regular homography matrices with Frobenius norm one are generated (known ground truth data) as shown in figure 5.1. For each such homography two random positions (uniformly distributed in a virtual image of size 1024×1024) are chosen 100 times, the ground truth LAF correspondence is computed and the LAFs are constructed. These are then disturbed with normally distributed noise according to scaled versions of the empiric covariance. Using this noisy correspondence data, the homography is estimated by the analytic algorithm of the section 5.1.2. The resulting homography is normalized by its Frobenius norm and difference to the ground truth homography (again Frobenius norm) is plotted in figure 5.2. Also, the triangle decomposition is applied to the two LAF correspondences and the resulting point correspondences are exploited in a DLT algorithm [Hartley and Zisserman, 2004, pp.108] using pre-normalization of points according to Hartley [1997b]. In the given setting this algorithm produced the best results, followed by the normalized differential algorithm, while the unnormalized (raw) algorithm performed worst.

In the next experiment the triangle transformation is evaluated for different triangle sizes and the resulting difference to the ground truth homography is again measured as displayed in figure 5.3. Here it can be seen that the size of the triangle has to be chosen large enough not to disturb the machine

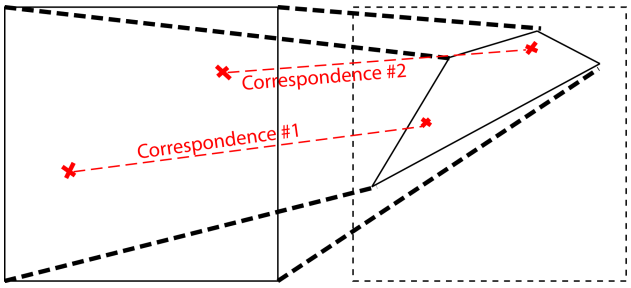

Figure 5.1: Generation of ground truth LAF correspondences. Ground truth general homographies have been constructed using the four corners of a virtual 1024×1024 image, being mapped somewhere into another image of the same size (2D uniform distribution). The transformation of the four image corners then defines a ground truth homography. For each such homography two random positions (uniformly distributed in the image) are chosen, the ground truth LAF correspondence is computed and the LAFs are constructed.

precision, which happens for the DLT algorithm not until the order of 10^{-12}. On the other hand the triangle must be within a good linearity region, such that the bias towards an affine transform induced by the triangle correspondence does not dominate potential perspective effects of the true transform, which happens here at the order of 10^3. If the triangle is significantly smaller than the position noise, the affine bias is also partially masked by the noise, i.e. the noise dominates the estimation. The acceptable range is however different for different applications and also depends on the structure of the warp function and on the numerical operations performed with the triangle corners.

The reason why the DLT works so well on such close points in presence of so much noise is that the noise is not added independently to each of the points but onto the LAF correspondence. This means that after adding noise, the shape of the triangle has slightly changed, but it is unlikely that the three corners are completely flipped or confused. Using normalization according to Hartley [1997b] makes the algorithm numerically stable, while in the differential method there may still be potential for a numerically better implementation, particularly because a cubic root is involved. However, it can be seen that the version with normalized coordinates works much better than the raw implementation.

The standard deviations being in the range of the estimated parameters means that a certain fraction of cases existed, where the algorithms produced a bad estimate. These might be because of close to degenerate ground truth

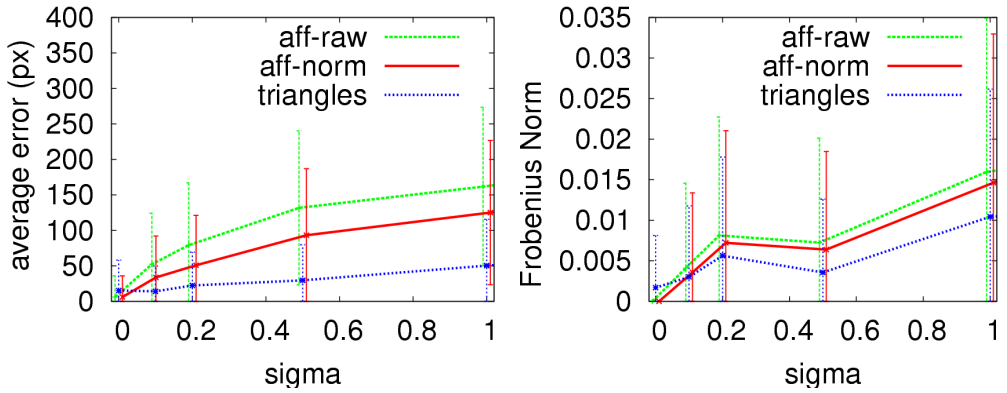

Figure 5.2: Mapping Error for noisy LAF correspondences. For each homography 100 LAFs are constructed according to figure 5.1, disturbed with normally distributed noise according to scaled versions of the empiric covariance and exploited to obtain an estimate of the homography a) using the raw differential algorithm of the previous section, b) the normalized differential algorithm and c) DLT in combination with the triangle transform. In the left graph the mapping error in pixel averaged across the overlap region is computed and in the right graph the Frobenius norm between the estimated and the ground truth homography is plotted. Qualitatively both error measures show that triangle transform in combination with DLT performs better than the unnormalized differential approach. The performance of the normalized approach is in between the other two. The error bars show only 1/3 standard deviation because the standard deviations are large compared to the estimated parameters.

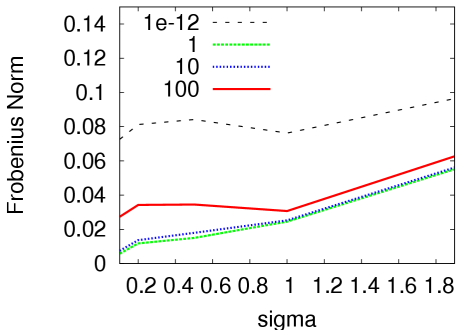

Figure 5.3: When the triangle transform is exploited to sample the affine feature into three point features the scale of the feature is important , because it determines the distance of the samples to the center. Here this distance is varied using the same input data as in figure 5.3. The plots for 10^{-1} to 10^{-9} look pretty much the same as the plot for 1 and are not displayed here. It can be seen that for linear homography estimation the sampling-ϵ for the triangle transform does not have much influence as long as it is in the range of 10^{-10} to 1 but the algorithm breaks outside this interval.

homographies, but the behavior can also be explained for the triangle decomposition with a too large triangle (see cases without noise) and for the differential formulation with numerical issues.

When multiple LAF correspondences are incorporated maximum likelihood estimation can be performed where each of the features can be seen as an (uncertain) observation. This is compared to sampling the features followed by a standard DLT algorithm (see figure 5.4). It is well known that DLT minimizes only an algebraic error within the set of unit-norm homographies (compare [Torr and Fitzgibbon, 2003] for a similar problem in fundamental matrix estimation). Therefore it is not surprising that maximum-likelihood estimation comes closer to the ground truth homography. Note also that for all correspondences the same uncertainty has been used and that the differences might get even larger when individually scaled and shaped uncertainty would have been added.

5.2 Conjugate Rotation

The *infinite homography* $\mathbf{H}_\infty = \mathbf{K}_1 R \mathbf{K}_2^{-1}$ is an image-to-image transformation that relates points in one image with points in another image if the camera has either only rotated or the corresponding 3D point is infinitely far away (compare section 2.3.5). It is a very important concept in self-

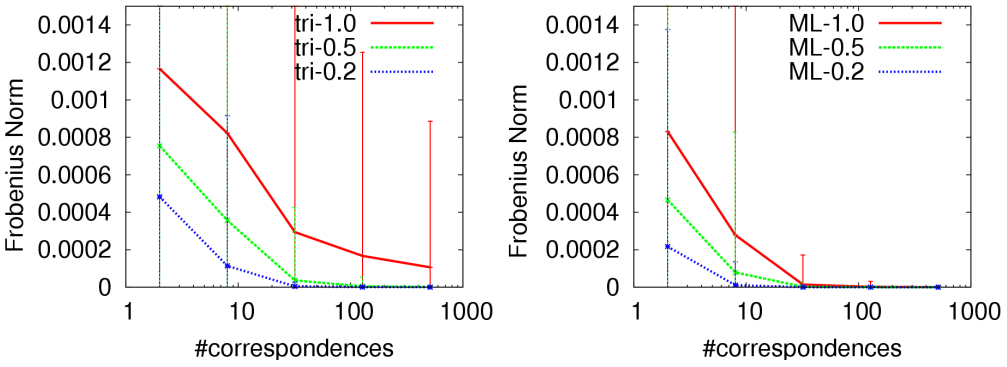

Figure 5.4: Exploitation of Multiple Features. These figures show curves for different noise levels. The remaining Frobenius norm between the estimated and the ground truth homography is plotted vs. the number of LAF correspondences exploited. The left graph shows DLT on triangle transformed LAFs and the right plot shows maximum likelihood estimation. The maximum likelihood estimation is clearly better than DLT. In the left part of the graphs nearly no redundancy exists, while in the right part the residual approaches zero for high redundancy even for larger sigma.

calibration [Hartley, 1997a], projective geometry [Hartley and Zisserman, 2004], or when dealing with purely rotating cameras, e.g. with pan-tilt-units [Capel and Zisserman, 1998], or for creating panoramic image mosaics [Brown and Lowe, 2007]. In this section the case of constant intrinsic camera parameters is inspected, i.e. $\mathbf{K}_1 = \mathbf{K}_2$ is assumed. Then, algebraically a 3×3-matrix $\mathbf{H}$ acting in the projective image space $\mathbb{P}^2$ is such an infinite homography if and only if it is proportional to a conjugate rotation, i.e. has the same eigenvalue structure as a scaled rotation matrix [Pollefeys and van Gool, 1999, Hartley and Zisserman, 2004].

5.2.1 Previous Work

The dimension and structure of the set of conjugate rotations within the space of all possible homographies has not been fully understood yet, so that, to the best knowledge of the author, no algorithm for the direct computation of general conjugate rotations existed prior to the one proposed in [Köser et al., 2008]. A solution existed for the special case when nearly all intrinsics (skew, aspect ratio, principal point) are known exactly [Brown et al., 2007]. In the general case, researchers typically estimate general homographies (e.g. using direct linear transformation [Hartley and Zisserman, 2004] on $n \geq 4$ point correspondences) and state that - in the presence of little noise - the esti-

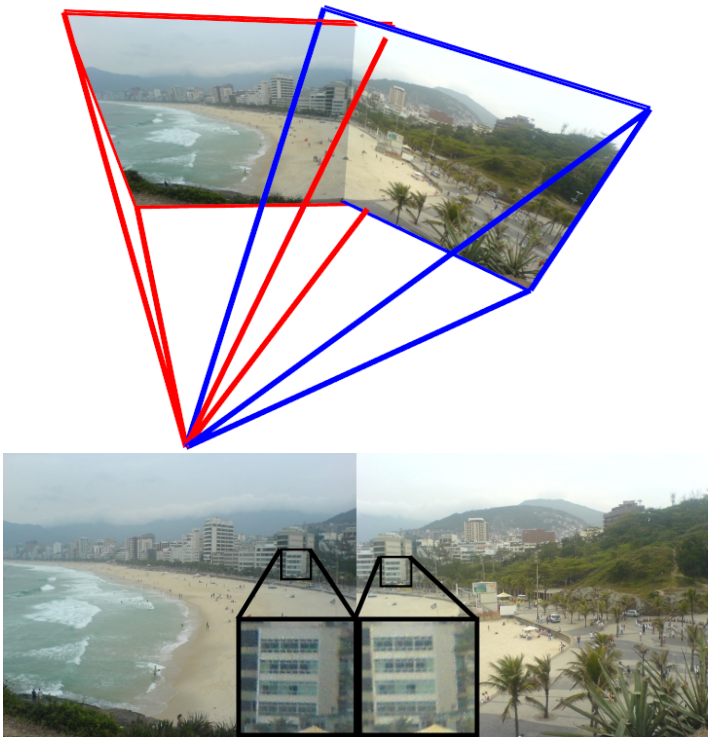

Figure 5.5: An affine correspondence in two images related by an infinite homography $\mathbf{H}_\infty$: The linear transformation (e.g. shear, rotation, magnification) between the two magnified image regions approximates the local derivative of the global image-to-image mapping $\mathbf{H}_\infty$ in the center of the window. Considering also the center shift, the resulting affine transformation can be thought of being tangent to $\mathbf{H}_\infty$, which can be exploited for estimation.

mate should not be too far from a conjugate rotation [Hartley, 1994, Capel and Zisserman, 1998]. Homography estimation approaches not using point correspondences require e.g. the identification of locally planar rectangle correspondences [Kim and Kweon, 2006] or deal with conic correspondences [Kannala et al., 2006], which typically lead to systems of quadratic equations or require lots of matrix factorizations.

However, having obtained a general conjugate rotation, enforcing the "conjugate rotation constraints" afterwards is not as straightforward as for instance in the 8-point algorithm [Hartley and Zisserman, 2004] for the fundamental matrix because the eigenvalue decomposition of $\mathbf{H}_\infty$ will in general contain complex vectors. Simply projecting onto the allowed manifold has not been possible because neither the dimension of this manifold has been known nor a suitable minimal parameterization has been available.

In the next sections such a minimal parameterization for the conjugate rotation will be proposed according to [Köser et al., 2008] and it will be shown that the set of conjugate rotations is a 7-dimensional manifold in the space of all 3×3-matrices $\mathbb{R}^9$, followed by an algorithm to estimate a conjugate rotation based upon a single LAF correspondence, which provides already six constraints. The last degree of freedom can be obtained by using another point correspondence or it can be fixed by using the additional assumption of zero skew and square pixels as described in [Köser et al., 2008]. This is in contrast to the algorithm presented in [Brown et al., 2007], which also requires the principal point to be known exactly, which is not always available. Finally, in section 5.2.4 the algorithm is evaluated and results in panoramic mosaicking with real images are shown.

5.2.2 A Minimal Parameterization

In this section a minimal parameterization will be derived for the conjugate rotation. Despite being a very important concept in multi view geometry, the number of degrees of freedom has not been investigated yet. Neither exists a parameterization with less than the eight parameters (as the naive parameterization with five intrinsic parameters and three rotation parameters). Such an over-parameterization can cause trouble in optimization, e.g. degenerate covariance matrices in maximum-likelihood estimation. Some authors have simplified $\mathbf{K}$ for the conjugate rotation to pure diagonal shape with zero skew, known aspect ratio and principal point [Brown et al., 2007]. Consequently, in this simplified model only a subset of all possible conjugate rotations is allowed. Instead, here a minimal parameterization for general conjugate rotations is proposed and estimation is discussed in the next section. A 2D

homography mapping

$$\mathbf{x}' \simeq \mathbf{H}\mathbf{x} = \begin{pmatrix} \boldsymbol{h}_1^\mathsf{T} & \boldsymbol{d} \\ \boldsymbol{h}_2^\mathsf{T} & \\ \boldsymbol{h}_3^\mathsf{T} & 1 \end{pmatrix} \mathbf{x} \tag{5.21}$$

is expressed in Euclidean coordinates as

$$\boldsymbol{x}' = \frac{\begin{pmatrix} \boldsymbol{h}_1^\mathsf{T} \\ \boldsymbol{h}_2^\mathsf{T} \end{pmatrix} \boldsymbol{x} + \boldsymbol{d}}{\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x} + 1} \tag{5.22}$$

Its derivative is

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{x}} = \tag{5.23}$$

$$\frac{(\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x} + 1)\begin{pmatrix} \boldsymbol{h}_1^\mathsf{T} \\ \boldsymbol{h}_2^\mathsf{T} \end{pmatrix} - \begin{pmatrix} \boldsymbol{h}_1^\mathsf{T} \\ \boldsymbol{h}_2^\mathsf{T} \end{pmatrix} \boldsymbol{x}\boldsymbol{h}_3^\mathsf{T} - \boldsymbol{d}\boldsymbol{h}_3^\mathsf{T}}{(\boldsymbol{h}_3^\mathsf{T}\boldsymbol{x} + 1)^2}$$

Without loss of generality the coordinate systems of both images are now moved by an offset $-\boldsymbol{x}$, such that $\boldsymbol{x} = (0,0)^\mathsf{T}$, then this simplifies to

$$A = \begin{pmatrix} \boldsymbol{h}_1^\mathsf{T} \\ \boldsymbol{h}_2^\mathsf{T} \end{pmatrix} - \boldsymbol{d}\boldsymbol{h}_3^\mathsf{T} \tag{5.24}$$

Solving this for $\boldsymbol{h}_3$ and using $\boldsymbol{d} = \boldsymbol{x}' - \boldsymbol{x}$, the homography given $\boldsymbol{x}'$ and A is therefore

$$\mathbf{H} = \begin{pmatrix} A + (\boldsymbol{x}' - \boldsymbol{x})\boldsymbol{h}_3^\mathsf{T} & \boldsymbol{x}' - \boldsymbol{x} \\ \boldsymbol{h}_3^\mathsf{T} & 1 \end{pmatrix} \tag{5.25}$$

$$= \begin{pmatrix} I_{2\times 2} & \boldsymbol{x}' - \boldsymbol{x} \\ \mathbf{0}_2^\mathsf{T} & 1 \end{pmatrix} \begin{pmatrix} A & \mathbf{0}_2 \\ \boldsymbol{h}_3^\mathsf{T} & 1 \end{pmatrix} \tag{5.26}$$

So far, $\mathbf{H}$ may be any homography[4] and no special conjugate rotation assumptions have been made. Now it is assumed that $\mathbf{H}$ is proportional to a conjugate rotation, i.e.

$$\mathbf{H} = \lambda \mathbf{K} R \mathbf{K}^{-1} \qquad \lambda \neq 0 \tag{5.27}$$

[4]Due to the assumption $\mathbf{H}_{33} = 1$ the origin (or the feature position) is mapped to a finite position. If this is not true for the given $\boldsymbol{x}$, then another $\tilde{\boldsymbol{x}}$ must be chosen that maps to a finite position.

where R is the relative camera rotation and $\mathbf{K}$ is the camera calibration matrix holding focal length f, aspect ratio a, skew s and principal point $(c_x, c_y)^\mathsf{T}$:

$$\mathbf{K} = \begin{pmatrix} f & s & c_x \\ 0 & af & c_y \\ 0 & 0 & 1 \end{pmatrix} \tag{5.28}$$

From the orthogonality of the rotation matrix R follows that its eigenvalues and therefore also the eigenvalues of $\frac{1}{\lambda}\mathbf{H}$ are $\{1, e^{i\phi}, e^{-i\phi}\}$. Exploiting that all eigenvalues have the same absolute value, Pollefeys et al. derived a fourth order polynomial constraint for self-calibration, called the modulus constraint [Pollefeys and van Gool, 1999], which is a *necessary condition* for a conjugate rotation. In contrast to this, the above parameterization now leads to a linear relation between h_{31} and h_{32} involving also the other parameters, and it is shown that this is a *sufficient condition* for conjugate rotations. The eigenvalues of $\frac{1}{\lambda}\mathbf{H}$ are the roots of the characteristic polynomial:

$$\mathtt{det}\left[\frac{1}{\lambda}\mathbf{H} - \tau\mathbf{I}_3\right] = \alpha(\tau - 1)(\tau - e^{i\phi})(\tau - e^{-i\phi}) \tag{5.29}$$

Multiplying out both sides yields a 3rd order polynomial in τ on both sides of the equation

$$\begin{aligned} c_3\tau^3 + c_2\tau^2 + c_1\tau + c_0 = \\ \alpha\tau^3 - \alpha(e^{i\phi} + e^{-i\phi} + 1)\tau^2 + \alpha(e^{i\phi} + e^{-i\phi} + 1)\tau - \alpha, \end{aligned} \tag{5.30}$$

where the coefficients c_i depend on $\mathbf{H}$ and λ and ϕ is the rotation angle of the conjugate rotation. Since different order monomials are orthogonal, corresponding coefficients must be equal for the two polynomials to become equal.

$$c_3 = \alpha \tag{5.31}$$
$$c_2 = -\alpha(e^{i\phi} + e^{-i\phi} + 1) \tag{5.32}$$
$$c_1 = \alpha(e^{i\phi} + e^{-i\phi} + 1) \tag{5.33}$$
$$c_0 = -\alpha \tag{5.34}$$

By comparison of the polynomial coefficients the unknowns α and ϕ are eliminated and two constraints are obtained, which are equivalent to

$$\lambda^3 = \mathtt{det}\,[A] \tag{5.35}$$
$$\lambda\mathtt{trace}\,[\mathbf{H}] = \frac{1}{2}((\mathtt{trace}\,[\mathbf{H}])^2 - \mathtt{trace}\,[\mathbf{H}^2]) \tag{5.36}$$

Observe that equation (5.35) eliminates the scale factor λ from subsequent computations and that all homographies fulfilling these constraints must be

conjugate rotations because they have the same eigenvalues as a rotation matrix as long as λ is not zero. Now insert equation (5.25) into those constraints and obtain the condition

$$\begin{aligned} & \left((\lambda - \mathtt{trace}\,[A])(\boldsymbol{x}' - \boldsymbol{x})^\mathsf{T} + (\boldsymbol{x}' - \boldsymbol{x})^\mathsf{T} A^\mathsf{T}\right) \boldsymbol{h}_3 \qquad (5.37) \\ = \; & -\frac{1}{2}(\mathtt{trace}\,[A])^2 - \mathtt{trace}\,[A] + \frac{1}{2}\mathtt{trace}\left[A^2\right] + \lambda\,(\mathtt{trace}\,[A] + 1) \end{aligned}$$

which is linear in $\boldsymbol{h}_3$ and can be written more compactly as

$$\boldsymbol{m}^\mathsf{T}\boldsymbol{h}_3 = b \qquad\qquad \boldsymbol{m}^\mathsf{T} = (m_1\, m_2) \qquad (5.38)$$

where $\boldsymbol{m}^\mathsf{T}$ is the row vector and b is the scalar value computed from A and $(\boldsymbol{x}' - \boldsymbol{x})$ of equation (5.37).

From a geometrical point of view, the equation above makes sure that the fixpoint of the conjugate rotation is compatible with the local offset and the local linear warp. The fixpoint of the conjugate rotation is the eigenvector corresponding to the eigenvalue 1, and geometrically this is the intersection of the rotation axis with the image plane. Particularly, when $\boldsymbol{x}$ is already a fixpoint of $\mathbf{H}$, i.e. when the last column[5] of $\mathbf{H}$ is $(0, 0, 1)^\mathsf{T}$, then the left hand side of equation (5.37) vanishes: Only A must be chosen appropriately in that case and all selections of $\boldsymbol{h}_3$ lead to valid conjugate rotations.

Parameterizing the Conjugate Rotation

The constraint of equation (5.37) does not explicitly express one of the eight parameters in terms of the other seven. Instead, it rather imposes an implicit constraint that all parameters must fulfill. Therefore, it is not straightforward to replace one parameter by the other ones. Given A and $(\boldsymbol{x}' - \boldsymbol{x})$ however, the constraint is linear in $\boldsymbol{h}_3$, and one possible parameterization is to solve for h_{31}. This is only possible if m_1 is not zero, which happens when $\boldsymbol{x}$ is a fixpoint so that $(\boldsymbol{x}' - \boldsymbol{x})$ vanishes. This case is however easily detected and avoided if some other, displaced, point is used. As a drawback, this has the effect, that the identity transform, a very special kind of conjugate rotation, cannot be parameterized using this 7-parameter model, since all points are fixpoints under the identity transform and h_{31} cannot be determined from the other parameters. In the general case however, when $m_1 \neq 0$, the consistent value for h_{31} can be obtained

$$h_{31} = \frac{m_2}{m_1} h_{32} - b \qquad (5.39)$$

[5] The last column of a homography matrix is the image of the origin because the product of the matrix with $(0, 0, 1)^\mathsf{T}$ simply yields the last column of the matrix.

Consequently, given a non-fixpoint affine correspondence there is a family of homographies, which depends on the six parameters of this correspondence and one parameter of the equation above.[6] In other words, the 3×3 matrix $\mathbf{H}$ depends on seven parameters

$$\boldsymbol{p} = (a_{11}, a_{12}, a_{21}, a_{22}, d_1, d_2, h_{32})^\mathsf{T} \tag{5.40}$$

now. By construction, $\mathbf{H}$ must be a conjugate rotation and the manifold for conjugate rotations can have at most seven dimensions, since it depends on seven parameters only.

To prove that $\mathbf{H}$ has at least seven degrees of freedom, it is sufficient to show that the tangential space of $\mathtt{vec}\,[\mathbf{H}]$ as a function of $\boldsymbol{p}$ is 7-dimensional at some position [Gray, 1994]. Intuitively, this means that for a given set of parameters, if it is possible to run into seven orthogonal directions on the manifold of conjugate rotations when changing the parameters, then the manifold has at least seven dimensions.

This can be best analyzed for a simple conjugate rotation, however, care has to be taken that the constraint of equation (5.37) is fulfilled even if the parameters are varied. This is obviously the case for the parameter set

$$\boldsymbol{p}_z = (0, 1, -1, 0, 1, 1, 0)^\mathsf{T} \tag{5.41}$$

because any of the above entries can be changed in a small interval around $\boldsymbol{p}_z$ and still a h_{31} can be found such that equation (5.37) is fulfilled. This holds, because it is a linear equation in h_{31}. In appendix D.1 it is shown that

$$\mathtt{rank}\left[\left.\frac{\partial \mathtt{vec}\,[\mathbf{H}]}{\partial \boldsymbol{p}}\right|_{\boldsymbol{p}_z}\right] = 7 \tag{5.42}$$

holds. Consequently, a conjugate rotation has 7 DOF.

This may be surprising at first sight, since knowing the eigenvalue structure seems to be more information than a single constraint. Note however, that the rotation angle ϕ is unknown and one therefore only knows the absolute value of the second and the third eigenvalue. Also, since the characteristic polynomial is holomorphic (as all polynomials), complex conjugates of

[6] Since $\lambda \neq 0$ has been assumed, equation (5.35) requires a non-zero determinant of A: Within the space of all possible selections for a_{ij} this set is not allowed. When the parameterization starts from a LAF correspondence, $\mathtt{det}\,[A] = 0$ is already guaranteed because such a correspondence cannot be measured since one of the features would have no area (e.g. a feature on the optical axis for a horizontal camera rotation of 90°). Also in practical optimization techniques like Levenberg-Marquardt these parameter configurations can easily be avoided because they form a null set.

any root must also be a root and finally, in projective space a homography is equivalent to a scaled version, so one basically ends up with the constraint "All eigenvalues have the same absolute value" (compare [Pollefeys and van Gool, 1999]). In the next section it is shown how the conjugate rotation with its seven degrees of freedom can be estimated based upon a LAF correspondence, which already provides six constraints.

5.2.3 Estimation

In the previous section a homography of the form

$$\mathbf{H}(h_{32}) = \begin{pmatrix} I_{2\times 2} & \boldsymbol{x}' - \boldsymbol{x} \\ \mathbf{0}_2^\mathsf{T} & 1 \end{pmatrix} \begin{pmatrix} A & \mathbf{0}_2 \\ \boldsymbol{h}_3^\mathsf{T} & 1 \end{pmatrix} \tag{5.43}$$

has been derived, which, given an affine feature correspondence, depends only on one parameter h_{32}. Basically this means that the affine transform locally fixes the conjugate rotation, but the pre-image of the line at infinity $\boldsymbol{h}_3$ still depends on one unknown parameter: It is unclear, what maps to infinity yet.

In order to determine this remaining parameter one additional constraint is required. This may be obtained from another point or line correspondence or from a constraint on the intrinsic camera parameters.

Additional Point or Line Correspondence

If an additional image point correspondence $(\boldsymbol{y}, \boldsymbol{y}')$ is given, it must fulfill the homography mapping (using the parameterization from equation (5.43))

$$\begin{aligned} \begin{pmatrix} \boldsymbol{y}' - \boldsymbol{x} \\ 1 \end{pmatrix} &\simeq \mathbf{H} \begin{pmatrix} \boldsymbol{y} - \boldsymbol{x} \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} I_{2\times 2} & \boldsymbol{x}' - \boldsymbol{x} \\ \mathbf{0}_2^\mathsf{T} & 1 \end{pmatrix} \begin{pmatrix} A & \mathbf{0}_2 \\ \boldsymbol{h}_3^\mathsf{T} & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{y} - \boldsymbol{x} \\ 1 \end{pmatrix} \end{aligned} \tag{5.44}$$

The displacement matrix is transferred to the left hand side so that the cross product of the left hand side and the right hand side must be zero

$$\begin{aligned} &\begin{bmatrix} \boldsymbol{y}' - \boldsymbol{x}' \\ 1 \end{bmatrix}_\times \begin{pmatrix} 0_{2\times 2} \\ (\boldsymbol{y} - \boldsymbol{x})^\mathsf{T} \end{pmatrix} \boldsymbol{h}_3 \\ &= - \begin{bmatrix} \boldsymbol{y}' - \boldsymbol{x}' \\ 1 \end{bmatrix}_\times \begin{pmatrix} A(\boldsymbol{y} - \boldsymbol{x}) \\ 1 \end{pmatrix} \end{aligned} \tag{5.45}$$

Selecting one of the first two rows yields a linear equation in $\boldsymbol{h}_3$, which

in general[7] determines the last remaining degree of freedom and therefore the conjugate rotation without any restrictions on skew, aspect ratio, focal length or principal point. Alternatively, another line correspondence might be used, e.g. if the horizon can be found in both images. Lines are dual to points and backward-map with a transposed $\mathbf{H}$, so basically the same linear algebra applies as in the point correspondence case.

Self calibration is now possible with the approach of Hartley [Hartley, 1994]. Note however, that in contrast to the homography estimation method used in [Hartley, 1994], here the estimated homography will be a perfect conjugate rotation.

If on the other hand some intrinsics of the used camera are known beforehand, no additional correspondence is required for estimation of the infinite homography as will be shown next.

Constraints on the Intrinsics

If only a single affine feature correspondence is given, the remaining unknown h_{32} may be computed using constraints on the intrinsic camera parameters. Zero skew and unit aspect ratio are reasonable assumptions for most consumer cameras on the market, so that these constraints are exploited in the following. The only other algorithm to estimate a conjugate rotation [Brown et al., 2007] additionally requires the exact principal point position (see figure 5.6 for the sensitivity of [Brown et al., 2007] to principal point deviations). Since often the principal point is only roughly known, e.g. close to the image center, our algorithm does not assume anything about the principal point. Instead, in order to compute the remaining parameter, a quadratic constraint for h_{32} can be obtained by assuming zero skew and known aspect ratio as derived in [Köser et al., 2008].

Optimization of the Conjugate Rotation

Given a good start value, it is easy in the presented formulation to optimize the conjugate rotation, given multiple measured LAF correspondences and their uncertainties. The decomposition of the homography into the parameters is straight-forward and can be done by means of coefficient comparison in the matrix of equation (5.25), e.g. by first computing $\boldsymbol{h}_3^\mathsf{T}$ and $\boldsymbol{x}' - \boldsymbol{x}$ and then A. Then, exploiting the minimal parameterization, standard Gauss-Newton methods can be applied to obtain a maximum likelihood estimate

[7] In the case that the point is on the line between the fixpoint and the affine feature, equations (5.37) and (5.45) will not be linearly independent. In this case a different point must be used.

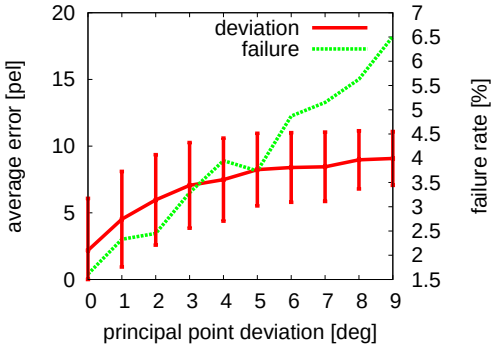

Figure 5.6: Synthetic evaluation of the sensitivity of the 2-point-algorithm [Brown et al., 2007] to principal point position (10.000 point pairs on a 50° field of view camera with width 1024 pixels), where the principal point is shifted several degrees away from the assumed position (the image center). The solid red curve shows the robust average error as evaluated by Brown et al. [Brown et al., 2007], while the dotted green curve shows the fraction of cases in which the algorithm did not come up with a solution at all. Already at 3° (5% image width) principal point error, the average error is above six pixels. Note that this is not a numerical or an implementation issue but caused by the resulting rays when the principal point varies.

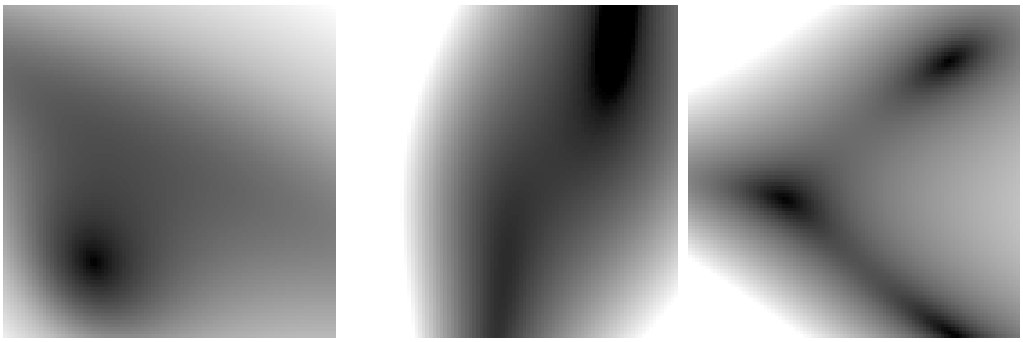

Figure 5.7: Qualitative distribution of homography mapping error (length of error vector) in a sample image, black means low error while white means large error. Left: The homography mapping error is low near the affine feature and increases outwards (result from extrapolation). Center: In the 2-point algorithm the error can have two local minima near the two feature positions. Right: Error for DLT on four point correspondences. σ was 0.5 pixels and principal point distortion for 2-point was 1 degree.

of the parameters according to section 4.4.4. Care must however be taken not to run into the case, where $\mathtt{det}\,[A] = 0$, since this is no valid conjugate rotation. However, this is a null set in the parameter space and can easily be avoided in Gauss-Newton methods by taking a smaller step.

5.2.4 Evaluation

So far it has been shown that a conjugate rotation has seven degrees of freedom. Ways to estimate it from as few data as possible were derived, which is interesting e.g. for RANSAC-like algorithms [Fischler and Bolles, 1981] or in scenarios where user initialization or interaction is required. In RANSAC-like algorithms the performance can decrease exponentially with the number of correspondences needed to estimate a solution (see also [Chum et al., 2003]). Traditionally, to obtain a conjugate rotation four point correspondences were required (general homography estimation using DLT), while recently a method has been proposed [Brown et al., 2007] to obtain a very special conjugate rotations based upon two feature correspondences (Brown et al. [2007] used only the position of SIFT[Lowe, 2004] features). The new method pushes this concept to the extreme so that only one LAF correspondence is required, while the principal point can vary freely. However, it is clear that in such a situation, where one local measurement determines a global transformation, small disturbances of the measurement can have severe

Figure 5.8: Conjugate rotation estimation from only one correspondence in each image pair. The first image is warped to the second and vice versa using the estimated homography.

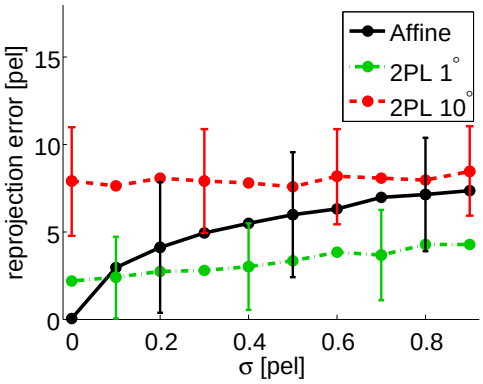

Figure 5.9: Sensitivity of the affine and the local 2-point algorithm against noise of the affine correspondence. The derivative is disturbed with 1% of the position uncertainty, which approximately evolves from the assumption that the corners of a patch for gradient-based minimization are found with the same uncertainty as the patch center. The images are of size 640×480 with 50° FOV and related by random rotations. The principal point prediction for the 2-point algorithm is disturbed with Gaussian noise of 1° (lower green curve) and 10° (upper red curve), which simulates that the principal point is only close to the image center for real cameras.

effects on the extrapolated transformation. Figure 5.7 qualitatively shows for an example that the error is small at the correspondence and slowly grows in the vicinity while it becomes larger far away from the feature. This suggests the application of a growing strategy, which first incorporates nearby correspondences for estimation of the global homography before iterating and increasing the neighborhood radius. In the two-point algorithm [Brown et al., 2007] there are two local minima because both features are forced to fit well.

To evaluate the sensitivity of the presented algorithm with respect to noise, the quality measure proposed in [Brown et al., 2007] has been used, where the average reprojection error across the overlap image region is measured and clipped at 10 pixels to ensure robustness against gross errors in the homography estimation. Figure 5.9 shows this quality measure plotted against the standard deviation of the noise on synthetic data. The proposed method is compared to the two point algorithm [Brown et al., 2007] for different principal point distributions. For a fair comparison the two correspondences are generated by obtaining two points of distance two pixels from each affine feature as described in [Riggi et al., 2006] (called *local two-point algorithm* in the following).

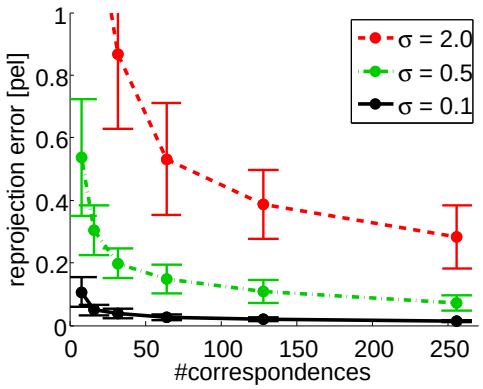

Figure 5.10: Quality of least-squares optimized conjugate rotation plotted against the number of correspondences used. The three curves represent three different standard deviations of the position noise, where again 1% Gaussian noise was added to the homography's derivative. The setting is the same as in figure 5.9.

In this evaluation it can be observed that the algorithm requires a good LAF correspondence. It is particularly sensitive to errors in the estimate of the homography's derivative. The error of the local two-point algorithm on the other hand is dominated by the principal point distortion. However, if many correspondences are available the minimal parameterization allows for a simple nonlinear least squares maximum likelihood optimization of conjugate rotations given affine feature correspondences. Figure 5.10 shows the reprojection error plotted against the number of correspondences for different noise levels. The reprojection error decreases significantly if more affine feature correspondences are used.

Finally, results on real images taken with different cameras as depicted in figure 5.8 are presented. From only one local correspondence a conjugate rotation was estimated using the proposed algorithm under the assumption of zero skew and square pixels. The images were stitched together and the results are shown in figure 5.8. Although not being subpixel correct, particularly in regions far away from the correspondence, the results look quite appealing given the minimalistic data they are based upon.

5.2.5 Discussion

It has been shown that a general conjugate rotation has seven degrees of freedom allowing for a minimal parameterization. This parameterization arises from the insight that an affine feature correspondence provides a first order

Taylor approximation to the image transformation, allowing for a differential constraint onto the homography. Another result is the first algorithm to compute a general conjugate rotation from a differential and a point or line correspondence and an algorithm for estimating a conjugate rotation from a single affine feature correspondence under the assumption of zero skew and known aspect ratio involving nothing more expensive than the solution of a quadratic equation. Also, the latter method does not require the principal point to be exactly at the image center, a crucial assumption to which previous methods are sensitive to, but which might not exactly be fulfilled in real cameras. For such real images it has finally been demonstrated that panoramic stitching is possible using only a single feature correspondence.

5.3 Triangulation and Normal Estimation

Given two calibrated cameras (assuming $\mathbf{K} = I_{3\times3}$) at different positions, observing a LAF correspondence, the 3D feature position (triangulation) and orientation (normal estimation) of the local plane is now derived. This allows for instance to use LAF correspondences to be integrated into an extended bundle adjustment[Triggs et al., 2000] also incorporating local surface orientations where often only the 3D positions are used.

5.3.1 Previous Work

Several algorithms exist to triangulate the 3D point from a 2D-2D point correspondence in two calibrated images [Hartley and Zisserman, 2004, Kanatani, 2005]. If multiple 3D points exist, e.g. from range data[Murray and Little, 2004], from photometric stereo [Murray, 2003] or from time-of-flight based cameras [Beder et al., 2007b], a local surface normal can be estimated by means of plane fitting. Such local planar patches are also called patchlets and may, given a good initial value, non-linearly be optimized based. Such an optimization can exploit the grey values of two images [Hattori and Maki, 1998, Molton et al., 2004, Pietzsch and Grossmann, 2005] or (fused) modalities from other sensors [Beder et al., 2007a]. Beder et al. [Beder et al., 2007b] obtained that the achievable accuracy in normal estimation is comparable for time-of-flight (ToF) cameras (e.g. photonic mixer devices [Xu et al., 1998]) and photometric stereo cameras, while they stated that ToF is better in obtaining the 3D position.

For purely vision-based systems that use one conic correspondence, there exist two solutions for the space conic if it is assumed that the two conics are projections of a planar conic in space [Ma, 1993]. For correspondences from

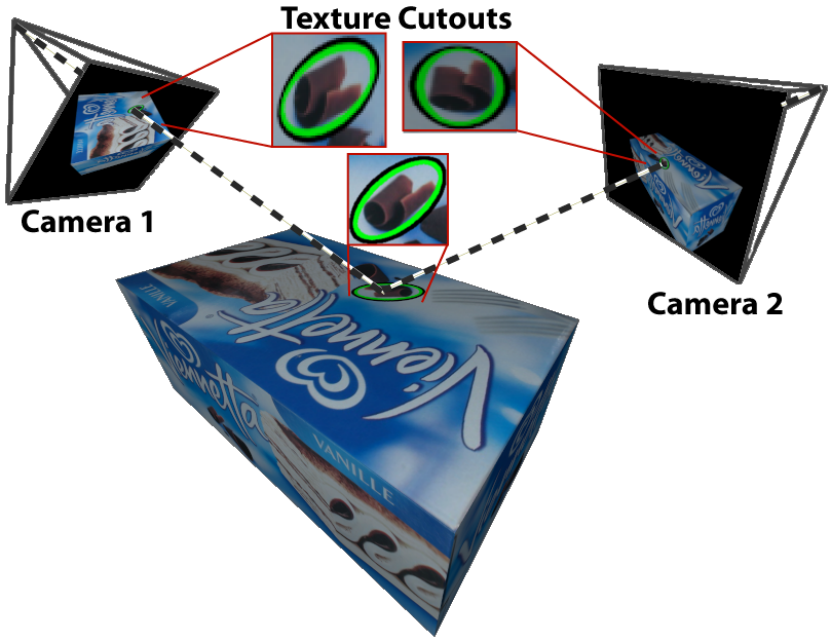

Figure 5.11: Two Cameras Observing a Textured Surface. This figure shows two cameras observing an affine feature (marked with an ellipse). The stretched versions of the feature in an orthophoto texture and in the two camera views are displayed in the top. Without knowledge about the scene, the position and the normal of the local feature can be obtained using the LAF correspondence .

affine features in videos with lots of images, Davison et al.[Davison et al., 2007, Molton et al., 2004] heuristically initialized the planar patch normal with the viewing ray of the first image and Jin et al.[Jin et al., 2003] initialize it with the optical axis of the first camera. Afterwards, both approaches update the normal in an iterative approach when new observations arrive, where [Molton et al., 2004] uses a SLAM (EKF) approach based upon affine feature correspondences while [Jin et al., 2003] exploits a generative model and works directly on the grey values. In contrast to these two filtering approaches, in the following it is derived how the point and the local surface normal can directly be obtained from a single LAF correspondence in just two images using only linear algebra.

5.3.2 Patchlet Estimation

Without loss of generality, the first camera is assumed to have the canonic pose (see equation (2.17)):

$$\mathbf{P}_{I1} = (I_{3\times 3} \,|\, \mathbf{0}_3) \tag{5.46}$$

and the second being at C with orientation R_2

$$\mathbf{P}_{I2} = (R_{I2}^{\mathsf{T}} \,|\, -R_{I2}^{\mathsf{T}}\boldsymbol{C}) \tag{5.47}$$

Let the LAF correspondence be $\mathbf{A}_{\boldsymbol{x}}^{I1}$ at $\boldsymbol{x}^{I1}$ in the first image and $\mathbf{A}_{\boldsymbol{x}}^{I2}$ at $\boldsymbol{x}^{I2}$ in the second image and let the local Jacobian (the concatenated warp according to equation (4.15)) be A. From the point correspondence, the 3D position $\mathbf{x}^W$ of the correspondence can be estimated, e.g. by DLT[Hartley and Zisserman, 2004]:

$$\left[\mathbf{x}^{I1}\right]_\times \mathbf{P}_{I1}\mathbf{x}^W = \mathbf{0}_3 \tag{5.48}$$

$$\left[\mathbf{x}^{I2}\right]_\times \mathbf{P}_{I2}\mathbf{x}^W = \mathbf{0}_3 \tag{5.49}$$

In general this homogeneous system of six equations in the four entries of $\mathbf{x}^W$ has a one-dimensional null-space (since $\mathbf{w}^W \in \mathbb{P}^3$), which can be estimated using SVD. In the presence of noise, the point triangulation problem can also be solved optimally in 3D space [Kanatani, 2005] or for projective reconstructions[Hartley and Zisserman, 2004].

The first camera is now virtually rotated around its center, so that the feature is moved onto the optical axis (according to the Rodrigues formula 2.15):

$$R_1 = \mathtt{R}_{\mathtt{3D}}\left[\frac{\mathbf{x}^{I1}}{\parallel \mathbf{x}^{I1} \parallel} \times (0\,0\,1)^{\mathsf{T}}, \cos^{-1}\left[\frac{(0\,0\,1)^{\mathsf{T}}\,\mathbf{x}^{I1}}{\parallel \mathbf{x}^{I1} \parallel}\right]\right] \tag{5.50}$$

Actually, this rotation is a homography transformation with respect to the image content and also affects the local affine frame. According to equation (4.19) by application of this homography, its linearization $\mathbf{R}_{1,\text{Taylor}}$ virtually changes the LAF in the first image to

$$\hat{\mathbf{A}}_{\boldsymbol{x}}^{I1} = \mathbf{R}_{1,\text{Taylor}} \mathbf{A}_{\boldsymbol{x}}^{I1}, \tag{5.51}$$

i.e. now $\mathbf{0}_2$ in the transformed first image corresponds to $\boldsymbol{x}^{I2}$ in the second, while the rotation-corrected Jacobian A_r is now

$$A_r = \left. \frac{\partial \mathbb{R}_1}{\partial \boldsymbol{x}} \right|_{\boldsymbol{x}^{I1}} A \tag{5.52}$$

Now the original problem has been transformed into a problem, where the feature is on the optical axis of one of the cameras. In the coordinate system of the changed first camera the projection matrices look like this

$$\hat{\mathbf{P}}_1 = (I_{3\times 3} \,|\, \mathbf{0}_3) \tag{5.53}$$

and

$$\hat{\mathbf{P}}_2 = (R_2^\mathsf{T} R_1 \,|\, -R_2^\mathsf{T} R_1 \boldsymbol{C}) = (R^\mathsf{T} | \boldsymbol{t}) \tag{5.54}$$

The patchlet normal, which is sought, is the normal of the local tangent plane to the 3D surface. The homography $\mathbf{H}_\pi$ between the two images induced by this tangent plane π at $\boldsymbol{s}$ in space with normal $\boldsymbol{n}$ is (cf. to [Molton et al., 2004]):

$$\mathbf{H}_\pi = \left(\boldsymbol{s}^\mathsf{T} \boldsymbol{n} \; R^\mathsf{T} \; - \; \boldsymbol{t} \boldsymbol{n}^\mathsf{T} \right) \tag{5.55}$$

where $\boldsymbol{s} = (0, 0, \lambda)^\mathsf{T}$.

In this representation only the surface normal $\boldsymbol{n}$ is unknown. Please observe that a scaled version of $\boldsymbol{n}$ leads to the same homography. The normal $\boldsymbol{n} \in \mathbb{R}^3$ has actually only two DOF and can be parameterized e.g. using two sphere angles. To obtain linear constraints, here it is now argued that for the feature to be visible in the first camera its surface normal must face - at least to some degree - towards the first camera and therefore have a negative z-component in the first camera's coordinate system. It is therefore set to -1 in the following, which is projectively equivalent to any other non-zero value,

$$\boldsymbol{n} = \begin{pmatrix} n_1 \\ n_2 \\ -1 \end{pmatrix} \tag{5.56}$$

so that only two parameters are left for the normal. The plane π has the following representation (according to equation (2.9)):

$$\pi = (n_1 \;\; n_2 \;\; -1 \;\; \|\boldsymbol{n}\| \lambda)^\mathsf{T} \tag{5.57}$$

$$\mathbf{H}_\pi = \left(-\lambda R^\mathsf{T} \; - \; \boldsymbol{t}(n_1 \; n_2 \; -1)\right) \tag{5.58}$$

This homography is now regarded as a mapping from $\mathbb{R}^2$ to $\mathbb{R}^2$:

$$\mathtt{H}_\pi\left[\mathbf{0}_2\right] = \boldsymbol{x}^{I2} \qquad \text{with} \qquad \left.\frac{\partial \mathtt{H}_\pi}{\partial \boldsymbol{x}}\right|_{\mathbf{0}_2} = \underbrace{\left.\frac{\partial \mathtt{R}_1}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}^{I1}} A}_{=A_r} \tag{5.59}$$

where A_r is the rotation-corrected warp already used in 5.51:

$$A_r = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \tag{5.60}$$

For convenience, the entries of the matrix are stacked into a vector (cf. to [Fusiello, 2007]):

$$\mathtt{vec}\left[\left.\frac{\partial \mathtt{H}_\pi}{\partial \boldsymbol{x}}\right|_{\mathbf{0}_2}\right] = \tag{5.61}$$

$$\frac{\lambda}{(t_3 - \lambda r_{33})^2} \begin{pmatrix} r_{33}(\lambda r_{11} + t_1 n_1) - t_3(r_{31} n_1 + r_{11}) + r_{13}(t_1 - \lambda r_{31}) \\ r_{33}(\lambda r_{21} + t_1 n_2) - t_3(r_{31} n_2 + r_{21}) + r_{23}(t_1 - \lambda r_{31}) \\ r_{33}(\lambda r_{12} + t_2 n_1) - t_3(r_{32} n_1 + r_{12}) + r_{13}(t_2 - \lambda r_{32}) \\ r_{33}(\lambda r_{22} + t_2 n_2) - t_3(r_{32} n_2 + r_{22}) + r_{23}(t_2 - \lambda r_{32}) \end{pmatrix}$$

The Jacobian is only defined if the point is not mapped to infinity, i.e. if $\hat{\mathbf{P}}_2(0 \; 0 \; \lambda \; 1)^\mathsf{T}$ has a nonzero third component $(-\lambda r_{33} + t_3)$. However, since the point is seen in the other image, finite coordinates are guaranteed and if the Jacobian is measured from feature correspondences, its full rank can also be assumed.[8] In any case, the entries of A_r can be obtained from the LAF correspondence and the camera poses and can now be exploited to reason about the normal.

$$\mathtt{vec}\left[A_r\right]\frac{(t_3 - \lambda r_{33})^2}{\lambda} = \frac{(t_3 - \lambda r_{33})^2}{\lambda} \begin{pmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \end{pmatrix} = \tag{5.62}$$

[8] In theory there are a few cases where the Jacobian does not have full rank, e.g. when one of the cameras is on the 3D surface plane or when the surface normal is orthogonal to the viewing ray of one of the cameras. However, a degenerate Jacobian between the two local regions means that in one image the feature would degenerate at least to a line, in which case it would not have been measured in practice. If there is a non-degenerate local affine frames in each of the images, the concatenation must also be non-degenerate and therefore the Jacobian must have full rank.

$$\begin{pmatrix} n_1(t_1r_{33} - r_{31}t_3) + \lambda r_{11}r_{33} - r_{11}t_3 - \lambda r_{31}r_{13} + t_1r_{13} \\ n_2(t_1r_{33} - r_{31}t_3) + \lambda r_{21}r_{33} - r_{21}t_3 - \lambda r_{31}r_{23} + t_1r_{23} \\ n_1(t_2r_{33} - r_{32}t_3) + \lambda r_{12}r_{33} - r_{12}t_3 - \lambda r_{32}r_{13} + t_2r_{13} \\ n_2(t_2r_{33} - r_{32}t_3) + \lambda r_{22}r_{33} - r_{22}t_3 - \lambda r_{32}r_{23} + t_2r_{23} \end{pmatrix} = \tag{5.63}$$

$$\begin{pmatrix} n_1(t_1r_{33} - r_{31}t_3) \\ n_2(t_1r_{33} - r_{31}t_3) \\ n_1(t_2r_{33} - r_{32}t_3) \\ n_2(t_2r_{33} - r_{32}t_3) \end{pmatrix} + \begin{pmatrix} \lambda r_{11}r_{33} - r_{11}t_3 - \lambda r_{31}r_{13} + t_1r_{13} \\ \lambda r_{21}r_{33} - r_{21}t_3 - \lambda r_{31}r_{23} + t_1r_{23} \\ \lambda r_{12}r_{33} - r_{12}t_3 - \lambda r_{32}r_{13} + t_2r_{13} \\ \lambda r_{22}r_{33} - r_{22}t_3 - \lambda r_{32}r_{23} + t_2r_{23} \end{pmatrix} \tag{5.64}$$

This provides now four linear equations in the unknown parameters n_1 and n_2, since the (nonzero) λ and the entries r_{ij} of R are known. They can be stacked into a linear equation system to estimate the unknown parts of the normal:

$$\begin{pmatrix} (t_1r_{33} - r_{31}t_3) & 0 \\ 0 & (t_1r_{33} - r_{31}t_3) \\ (t_2r_{33} - r_{32}t_3) & 0 \\ 0 & (t_2r_{33} - r_{32}t_3) \end{pmatrix} \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} = \tag{5.65}$$

$$\frac{(t_3 - \lambda r_{33})^2}{\lambda} \mathsf{vec}\left[A_r\right] + \begin{pmatrix} -\lambda r_{11}r_{33} + r_{11}t_3 + \lambda r_{31}r_{13} - t_1r_{13} \\ -\lambda r_{21}r_{33} + r_{21}t_3 + \lambda r_{31}r_{23} - t_1r_{23} \\ -\lambda r_{12}r_{33} + r_{12}t_3 + \lambda r_{32}r_{13} - t_2r_{13} \\ -\lambda r_{22}r_{33} + r_{22}t_3 + \lambda r_{32}r_{23} - t_2r_{23} \end{pmatrix}$$

As can be seen, there are cases where the left hand side matrix vanishes, so that no constraints on $\boldsymbol{n}$ are imposed. To explain this case have a look at the cross product of the epipole in the second camera $\boldsymbol{t}$ and the optical axis of the first camera (where the feature lies):

$$\boldsymbol{t} \times \boldsymbol{r}_3 = \begin{pmatrix} t_2r_{33} - r_{32}t_3 \\ t_3r_{31} - r_{33}t_1 \\ t_1r_{32} - r_{31}t_2 \end{pmatrix} \tag{5.66}$$

If the two vectors are collinear the camera has moved towards the feature and the above cross product is zero, in which case no constraints at all can be obtained for the surface normal in equation (5.65). If the camera has only moved in exact direction of the x-axis or the y-axis, two of the four constraints in equation (5.65) vanish, i.e. the system is no longer overdetermined. Equation 5.65 is now rewritten as

$$M_n \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} = B_n \tag{5.67}$$

and can be solved as

$$\begin{pmatrix} n_1 \\ n_2 \end{pmatrix} = (M_n^\mathsf{T} M_n)^{-1} M_n^\mathsf{T} B_n \tag{5.68}$$

The resulting normal vector $(n_1\ n_2\ -1)^\mathsf{T}$ has to be rotated by R_1 to compensate for the virtual rotation of the first camera onto the feature.

5.3.3 Maximum Likelihood Estimation

In the previous section a direct way of computing the normal from a LAF correspondence has been shown using only linear algebra. In the presence of noise this solution might not be optimal. However, if it provides an initial estimate for the surface normal with good accuracy it can be used to initialize a subsequent maximum likelihood estimation, which incorporates the individual uncertainties of the LAFs or the correspondence. Given the 5DOF for the patchlet, the local plane in space and therefore also the local homography can be determined, which maps between the features in the images. The linearization of the local homography should then equal the LAF correspondence . For this, noise models are available as described in section 4.4. The maximum likelihood estimation can thus be performed in the Gauss-Markov-Model as described in [McGlone, 2004]. Since only 5DOF are optimized there is redundancy even using a single correspondence.

5.3.4 Evaluation

The sensitivity of the patchlet estimation algorithm has been evaluated in figure 5.12 using synthetic ground truth data: the second camera as well as the patchlet position has been varied freely, while the patchlet normal has been set in a way that both cameras can see it well (see figure 5.12 for details). The projection of the patchlet center as well as the homography mapping across the plane has been exploited to generate a virtual LAF correspondence, which has been disturbed by scaled versions of the empiric covariance (scaled between 0 and 1). This experiment was carried out for focal lengths 400 (simulating wide-angle) and 2000 (simulating "zoomed in" camera).

Since the 3D position depends only on the position of the feature, the evaluation is restricted purely to the normal. The resulting normal error has been measured against the ground truth surface normal. The solution was then fed into a non-linear iterative maximum-likelihood estimator, whose result is also plotted in the figure. As already mentioned in the previous sections it is also possible to sample the LAF correspondence into three point correspondences, which can then each be triangulated and the triangle normal can be calculated.

The figures show that surface normal estimation is possible for moderate amounts of noise and that maximum-likelihood estimation (MLE) is better than optimizing an algebraic criterion, however the technique used for

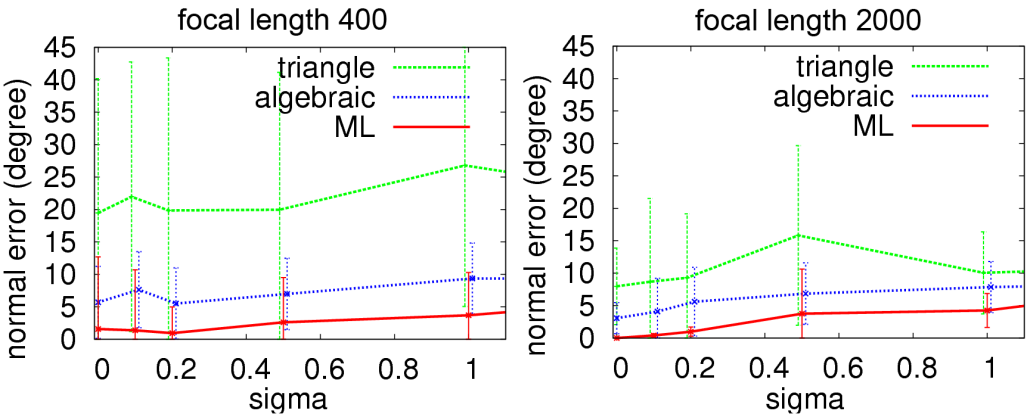

Figure 5.12: Patchlet Estimation in Presence of Correspondence Noise. In this figure a patchlet has been estimated based upon known ground truth data, which has been disturbed with gaussian noise (scaled empiric covariance) as plotted on the x-axis of the graphs. For the left figure a focal length of 400 has been used while for the right a focal length of 2000. In both cases camera 1 was at the canonic pose, camera 2 was inside a cube of ± 1 and looking towards the hemisphere with positive z-coordinate. The patchlet center was set into a box between $[-2.5, 2.5] \times [-2.5, 2.5] \times [1, 5]$. The patchlet's normal has been set to the average of the optical axes ± 0.2 in all three components (all parameters from uniform distributions). Whenever the patchlet was not in front of both cameras it was rejected. To estimate the position the algorithm given by Kanatani [Kanatani, 2005] is used. It can be seen that in the case of short focal length, the error is higher when the LAF correspondence is disturbed. The algebraic exploitation of the LAF correspondence is compared to sampling the affine feature into three points and then estimating mean and normal, further it is compared to the maximum-likelihood estimator.

MLE is Newton-like non-linear optimization, requiring a start value near the global optimum. Sampling the feature produced the worst results here. The experiments have been executed with small and large focal length, such that measuring the LAF correspondence inaccurately has different consequences on the resulting rays. As expected, errors are smaller with larger focal length because the angles do not change much with noise. The high error at almost no noise can be explained with numerical difficulties in spurious oblique normals, which would also explain the high standard deviations.

5.3.5 Discussion

This result must again be seen in relation to a conic correspondence in two views, where the quadratic formulation of the projection allows at least two interpretations [Ma, 1993]. Instead, for the LAF-based algebraic solution only linear algebra was required yielding a unique solution. In the simulations, the result of the direct (algebraic) algorithm was sufficient to initialize maximum likelihood estimation, which improves the result even on a single correspondence, since the observations are redundant because only 5DOF are estimated.

5.4 Pose Estimation

Since the description of spatial resection from three points by Grunert [1841], many people have worked on pose estimation or the so called P3P problem [Finsterwalder and Scheufele, 1903, Thompson, 1966, Fischler and Bolles, 1981, Haralick et al., 1994, Gao et al., 2003, Zhang and Hu, 2006, 2005]. PnP stands for pose estimation from n points and is underconstrained for $n < 3$ unless further information is incorporated. Here a variation of the problem is solved, namely when only a single LAF of a known 3D space surface with orthophoto texture can be found in an image. Additionally to the traditionally used point correspondence, such an image-model relation provides a local linear texture warp between the image and the model. This warp can be interpreted as the Jacobian of the perspectivity between the image and the 3D surface's tangent plane and it is shown that this determines the open degrees of freedom for the extended P1P problem to be solved. The presented approach allows to estimate a camera pose based upon only one image-model correspondence, which is particularly interesting in robot localization [Se et al., 2005], initialization or recovery in camera tracking [Davison et al., 2007, Williams et al., 2007] or determining the pose of a detected object[Skrypnyk and Lowe, 2004]. In these applications, often

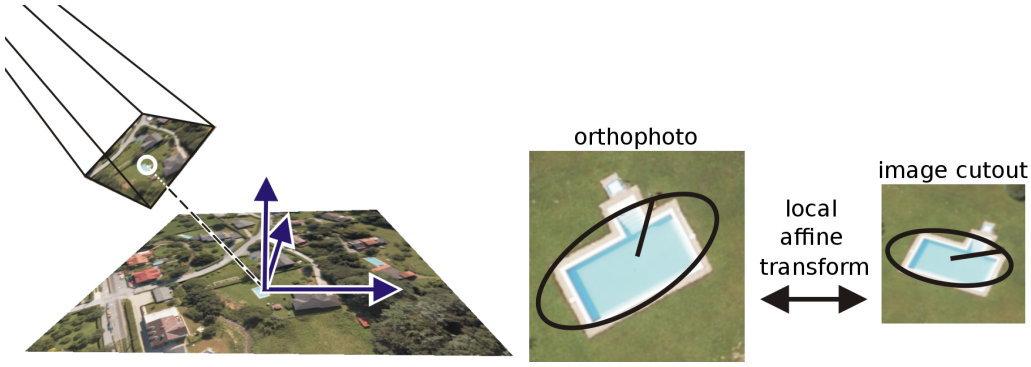

Figure 5.13: Differential Spatial Resection exploiting the Perspectivity Concept. This figure shows an aerial camera observing a ground plane (left image). If the internal camera calibration is removed, the two images are related by a perspectivity. The projection of some point on the plane and the linear transform of the surrounding region provide six constraints for the six DOF for pose estimation. In the right part an MSER feature correspondence between an orthophoto and the camera can be seen, providing an affine texture transform.

SIFT[Lowe, 2004] or MSER[Matas et al., 2002] features are used nowadays, which cover some image region ideally corresponding to a surface in the scene. In [Davison et al., 2007] even the normal of such local surface regions is estimated and also [Se et al., 2005] performs stereo from the three cameras on the robot. However, in all of the above cited approaches, the correspondences are geometrically handled as point correspondences when it comes to initialization or direct pose estimation, although they carry much more information. Therefore, by now at least three of such robust feature correspondences were required to directly estimate a camera or object pose. In contrast, in this section it is demonstrated how one such image-model correspondence is already sufficient to estimate the pose. The proposed primitive can be seen as the limiting case where three 3D points of Grunert's solution come infinitesimally close, allowing for a *differential spatial resection*.

To improve the pose estimation result, gradient based optimization techniques should be applied between the current view and a reference texture as explained in section 4.2.5. The reference texture can either be an orthophoto (cf. to [McGlone, 2004], p.758) or any other view with sufficient resolution for which the warp to an orthophoto is incorporated. When all such feature correspondences and the camera poses are optimized at once, this is similar to the approach of Jin et al.[Jin et al., 2003]. However, their approach is for-

mulated in a nonlinear fashion only and requires an initialization, comparable to the requirements for bundle adjustment.

5.4.1 Perspectivity

The contribution is based on estimating a transformation between two theoretical planes: The first plane is tangent to a textured surface in 3D and the second plane is orthogonal to the optical axis of a camera. The estimation of the pose is then formulated as the problem of obtaining a perspectivity between these two planes (see figure 5.13).

A 2D perspectivity is a special kind of homography (cf. also to [Hartley and Zisserman, 2004, p.34]), which has only six degrees of freedom and which is particularly important for mappings between planes in Euclidean space (cf. to section 2.3.3). Points on the $z = 0$ plane map into the camera as

$$\mathbf{p}_i = (R^\mathsf{T} | - R^\mathsf{T}\boldsymbol{C})\mathbf{p}_{s,z=0} = (\boldsymbol{r}_1\ \boldsymbol{r}_2\ \boldsymbol{r}_3\ -R^\mathsf{T}\boldsymbol{C})(x\ y\ 0\ 1)^\mathsf{T} \tag{5.69}$$

$$= (\boldsymbol{r}_1\ \boldsymbol{r}_2\ -R^\mathsf{T}\boldsymbol{C})(x\ y\ 1)^\mathsf{T} \simeq (\tilde{\boldsymbol{r}_1}\ \tilde{\boldsymbol{r}_2}\ \boldsymbol{t})(x\ y\ 1)^\mathsf{T} = \mathbf{H}\ \mathbf{p}_p \tag{5.70}$$

$\tilde{\boldsymbol{r}_i}$ are scaled versions of $\boldsymbol{r}_i$ so that $\boldsymbol{t}_z = 1$ and $\simeq$ means equality up to scale. Obviously, the homography $\mathbf{H}$ maps points $\mathbf{p}_p$ of the plane coordinate system to points $\mathbf{p}_i$ in the image coordinate system. $\mathbf{H}$ is a perspectivity and depends only on six parameters, the pose of the camera. Since $\mathbf{H}$ is an object of projective space, it can be scaled without changing the actual transformation. While the perspectivity $\mathbf{H}$ acts linearly in projective space $\mathbb{P}^2$, in Euclidean 2D space $\mathtt{H}$ is a nonlinear mapping from $\mathbb{R}^2 \to \mathbb{R}^2$ because of the nonlinear homogenization:

$$\mathtt{H}\left[\boldsymbol{p}_p\right] = \boldsymbol{p}_i = \mathtt{euc}\left[\mathbf{p}_i\right] = \frac{(\mathbf{H}\mathbf{p}_p)|_{1..2}}{(\mathbf{H}\mathbf{p}_p)|_3} \tag{5.71}$$

In the next section the LAF correspondence concept of section 4 is brought into context and it is shown how it can be exploited to obtain constraints on $\mathbf{H}$.

5.4.2 Previous Work

The solution of the spatial resection problem has been first mentioned in a German book for teachers from 1841 [Grunert, 1841]. Since then several more or less equivalent approaches in photogrammetry and computer vision have been proposed (cf. to [Haralick et al., 1994] for a comparison) including one in the RANSAC-article [Fischler and Bolles, 1981]. Basically, all approaches

first compute the distances to the three 3D points from the rays in the camera coordinate system. Then, using geometric reasoning on the distances and angles, they derive up to four triangle constellations, which explain the observations. In [Ma, 1993] Ma derived a way to determine the pose of a camera from two conics. He noted that a conic has only five DOF and thus a single conic is not sufficient to determine the six DOF of the camera pose uniquely. A conic $\mathbf{C}_S$ on the space plane of the previous section maps to a conic $\mathbf{C}_I$ in the image with the following equation:

$$\mathbf{C}_i = \mathbf{H}^\mathsf{T}\mathbf{C}_p\mathbf{H} \tag{5.72}$$

where $\mathbf{H}$ is the perspectivity of the previous sections and it can be seen that this leads to quadratic equations in the entries of the perspectivity. Other related work in homography estimation (e.g. [Zelnik-Manor and Irani, 2002, Kähler and Denzler, 2007, Irani et al., 1997]) and projective reconstruction [Rothganger et al., 2007] did not inspect the differential constraints on the perspectivity, because they stay projective. Camera pose estimation can however exploit the internal camera calibration and Euclidean 3D structure, this way reducing ambiguities.

In the evaluation section, the novel method will be compared to the the planar POSIT algorithm [Oberkampf et al., 1996], which requires four coplanar points and works in an iterative fashion. It already includes the parallel projection approximation proposed by Kyle [2004], who on the other hand requires 4 non-coplanar points. Basically, the last two algorithms require four points instead of three to avoid the application of the non-linear constraints from rotation parameterization. In contrast, these constraints are exploited in the differential formulation now:

5.4.3 Pose Estimation from a LAF correspondence

Having obtained a LAF correspondence between a camera image and the textured plane in the origin, the local warp equals the derivative of the perspectivity. This derivative $\partial \mathtt{H}/\partial \boldsymbol{p}_p$ tells something about the relative scaling of coordinates between the plane in the origin and the image, e.g. if $\boldsymbol{C}$ is large and the camera is far away from the origin $\partial \mathtt{H}/\partial \boldsymbol{p}_p$ will be small because a large step on the origin plane will result in a small step in the image far away. Actually, $\partial \mathtt{H}/\partial \boldsymbol{p}_p$ carries information about rotation, scale and shear through perspective effects.

Since $\mathbf{H}$ can be scaled arbitrarily without changing $\mathtt{H}$, without loss of

generality[9] let $\mathbf{H}_{3,3} = 1$ and compute the derivative at the origin:

$$\left.\frac{\partial \mathrm{H}}{\partial \boldsymbol{p}_p}\right|_{\mathbf{0}^\mathsf{T}} = \begin{pmatrix} \tilde{r}_{11} - \tilde{r}_{13}t_1 & \tilde{r}_{12} - \tilde{r}_{13}t_1 \\ \tilde{r}_{21} - \tilde{r}_{23}t_2 & \tilde{r}_{22} - \tilde{r}_{23}t_2 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \tag{5.73}$$

Also, it is inspected where the origin is projected in the image:

$$\mathbf{p}_{origin} = \mathbf{H}(0\ 0\ 1)^\mathsf{T} = -R^\mathsf{T}\boldsymbol{C} \simeq \boldsymbol{t} \tag{5.74}$$

Given a LAF correspondence, the derivative as well as the projection of the origin are given by the relative parameters of the detected features. This can determine all degrees of freedom of the camera pose, however the over-parameterization of the rotation must be resolved: Since $\tilde{R}$ is a scaled rotation matrix, $\tilde{\boldsymbol{r}}_1$ and $\tilde{\boldsymbol{r}}_2$ must be of same length and orthogonal:

$$\tilde{r}_{11}^2 + \tilde{r}_{12}^2 + \tilde{r}_{13}^2 = \tilde{r}_{21}^2 + \tilde{r}_{22}^2 + \tilde{r}_{23}^2 \ \wedge\ \tilde{\boldsymbol{r}}_1^\mathsf{T}\tilde{\boldsymbol{r}}_2 = 0 \tag{5.75}$$

Now $\mathbf{H}$ can be computed by first substituting $\boldsymbol{t}$ into equation (5.73), then solving for $\tilde{r}_{11}$, $\tilde{r}_{21}$, $\tilde{r}_{12}$ and $\tilde{r}_{22}$ and substituting into equation (5.75), leaving two quadratic equations in the two unknowns $\tilde{r}_{13}$ and $\tilde{r}_{23}$:

$$(\tilde{r}_{13}t_1 + a_{11})^2 + (\tilde{r}_{13}t_1 + a_{12})^2 + \tilde{r}_{13}^2 = (\tilde{r}_{23}t_2 + a_{21})^2 + (\tilde{r}_{23}t_2 + a_{22})^2 + \tilde{r}_{23}^2 \tag{5.76}$$

$$(\tilde{r}_{13}t_1 + a_{11})(\tilde{r}_{23}t_2 + a_{21}) + (\tilde{r}_{13}t_1 + a_{12})(\tilde{r}_{23}t_2 + a_{22}) + \tilde{r}_{13}\tilde{r}_{23} = 0 \tag{5.77}$$

The first equation is about the length and the second about the orthogonality of the $\tilde{r}$-vectors as typical for constraints on rotation matrices. It is instructive to interpret them as the intersection problem of two planar conics, the *length conic* $\mathbf{C}_l$ and the *orthogonality conic* $\mathbf{C}_o$:

$$(\tilde{r}_{13}\ \tilde{r}_{23}\ 1)\mathbf{C}_l(\tilde{r}_{13}\ \tilde{r}_{23}\ 1)^\mathsf{T} = 0 \tag{5.78}$$

$$(\tilde{r}_{13}\ \tilde{r}_{23}\ 1)\mathbf{C}_o(\tilde{r}_{13}\ \tilde{r}_{23}\ 1)^\mathsf{T} = 0 \tag{5.79}$$

$$\mathbf{C}_l = \begin{pmatrix} 2t_1^2 + 1 & 0 & t_1(a_{11} + a_{12}) \\ 0 & -2t_2^2 - 1 & -t_2(a_{21} + a_{22}) \\ t_1(a_{11} + a_{12}) & -t_2(a_{21} + a_{22}) & a_{11}^2 + a_{12}^2 - a_{21}^2 - a_{22}^2 \end{pmatrix} \tag{5.80}$$

$$\mathbf{C}_o = \begin{pmatrix} 0 & t_1t_2 + \frac{1}{2} & (a_{21} + a_{22})t_1 \\ t_1t_2 + \frac{1}{2} & 0 & (a_{11} + a_{12})t_2 \\ (a_{21} + a_{22})t_1 & (a_{11} + a_{12})t_2 & a_{11}a_{21} + a_{12}a_{22} \end{pmatrix} \tag{5.81}$$

[9] This is not a restriction because the only unrepresented value $\mathbf{H}_{3,3} = 0$ results in the origin being mapped to the plane at infinity and therefore such a feature cannot be seen in the camera.

Such a transformation of the problem is helpful because now methods from conic theory can be used to see if (and how many) real solutions exist or in which situation one might end up with an underconstrained problem (infinitely many solutions). It is well known that in spatial resection from three points there exists a surface called the danger cylinder [Finsterwalder and Scheufele, 1903, Thompson, 1966, Zhang and Hu, 2006], where the problem becomes unstable. In the given algorithm the camera pose is not observable if it is possible to change the pose parameters in a certain direction such that the LAF correspondence stays the same. Mathematically, this means that if the six affine parameters are differentiated with respect to six pose parameters, the Jacobian must have full rank, otherwise there might be a degenerate case (pose cannot be determined) or an instable case (double root) [Gruen and Huang, 2001, Thompson, 1966]. Full rank means non-zero determinant, therefore the characteristic polynomial of the Jacobian, which is of 6th degree, must not have any real root. To test for an unstable case, it is therefore possible to check the determinant of the above Jacobian.

Solving for the pose parameters

Two conics cannot have more than four intersection points, therefore, one can obtain at most four solutions for the camera pose. To solve the intersection of the two conics the elegant method of Finsterwalder and Scheufele [1903] is followed, which proved also to be the numerically most stable method of the six different 3-point algorithms for spatial resection [Haralick et al., 1994]: Since a common solution of equations (5.78) and (5.79) must also fulfill any linear combination of both, one can construct a linear combination of both conics, which does not have full rank (zero determinant), but which still holds all solutions. This creates a third order polynomial, which has at least one real root and can be solved easily (actually, this is a similar step as in the 7-point algorithm [Hartley and Zisserman, 2004] for fundamental matrix estimation):

$$\det\left[\lambda \mathbf{C}_o + (1-\lambda)\mathbf{C}_l)\right] = 0 \tag{5.82}$$

The resulting degenerate conic will in general consist of two lines. The intersection of these lines with the original conics is only a quadratic equation and determines the solutions. The resulting R and $\boldsymbol{C}$ have to be selected and normalized in such a way that an orthonormal rotation matrix (determinant $+1$) is obtained and the camera looks towards the plane. Now the pose of the camera has been obtained in the object coordinate system (relative to the feature at the $z = 0$ plane). If there is a world coordinate system, in which the plane is not at the origin, the rigid world transformation has to be appended to the computed pose of the camera. Computing the relative pose

in the object coordinate system in general also improves conditioning since the absolute numbers of the object's pose in the world become irrelevant.

5.4.4 Optimization, Tracking and Maximum Likelihood Estimation

Once the parameters are roughly known it is straightforward to use a 6-parametric gradient-based minimization technique (cf. to [Baker and Matthews, 2004, Lucas and Kanade, 1981, Köser and Koch, 2008b]) to optimize the camera pose. Note that if a pinhole camera is used and the feature in 3D is locally planar, instead of optimizing an approximate affine transform one might as well use a 6-parametric homography. Thus measurements may be incorporated from a larger region without making a mistake or an approximation. Even better, since it is possible to use global camera pose parameters, it is easy to optimize even multiple rigidly coupled features (e.g. in a rigid scene). Or, if robustness against outliers is a concern, each of the features provides an individual pose estimate and robust estimation techniques such as RANSAC[Fischler and Bolles, 1981] can be used to obtain a fused solution. If video data is available, the parameters can directly be used for tracking the regions, objects or camera pose over time similar to what is proposed in [Jin et al., 2003].

In terms of the LAF correspondence as an uncertain observation, one can find the most likely pose that led to this observation. Here, a solution obtained from the previous section can serve as a start value. The pose can then be parameterized as an offset from the initial camera center and three Euler angles for a slightly differing orientation. The Jacobian of the homography and the position of the feature is then optimized so that the Mahalanobis distance from the LAF correspondence is minimized as proposed in section 4.4.4.

5.4.5 Evaluation

In this section the LAF correspondence-based pose estimation is evaluated first using synthetic sensitivity experiments. Next, rendered images with known ground truth information are used to evaluate the real-world applicability, where everything has to be computed from image data. In the final experiments, object pose estimation from one feature is show qualitatively using non-ideal cameras.

Sensitivity to Noise and Internal Calibration Errors

The evaluation of the algorithm starts with an analysis of the sensitivity to different disturbances. Since the algorithm provides a minimal solution, which translates 6 DOF LAF correspondence into six DOF pose, the pose will adapt to noise in the correspondence. In figure (5.14) it is shown that for localization accuracies better than 1 pixel in a camera with focal length 500 pixel the camera orientation is on average better than 1 degree and also the direction of the camera center is better than 1 degree. The orientation error is computed from the axis-angle representation of the rotation that transforms the ground truth orientation into the estimated orientation and therefore incorporates all directions. The center error is the angle between the ground truth camera center and the estimated camera center as seen from the 3D feature's position. For the evaluation, a scaled version of the empiric covariance has been used according to section 4.4.2. It is remarkable that the errors in orientation and position are highly correlated. This can be explained from the fact that a slightly different LAF correspondence results in a slightly different camera orientation. However, since the feature must be projected to about the same position, the camera center has to adapt accordingly. As figure 5.14 shows, the pose estimation is stable even when the camera is not calibrated correctly, although it can be seen that the resulting pose is disturbed as inherent in minimal solutions. In particular it is clear that an error in principal point results in an error in the pose when the reference feature in 3D is correct. Keep in mind that at focal length 500 a principal point error of ten pixel means that the optical axis is more than 1 degree mis-calibrated.

Position and Orientation Errors of 3D Feature

The sensitivity to errors in the pose of the 3D feature is much more straightforward and need not to be sampled: Since the camera pose is computed in the local coordinate system of the feature, an error of this coordinate system in the world results in a concatenated error of the camera pose in the world.

Solid Angle, Approximation by Three Points and Comparison with Spatial Resection/POSIT

Using the proposed approach, the affine warp must be measured between the orthophoto and the image under inspection and this requires a region upon which this is done. If the alignment is done using an affine warp, the region should be chosen as small as possible, particularly when the feature is seen from an oblique angle, because in the affine warp model it is assumed that

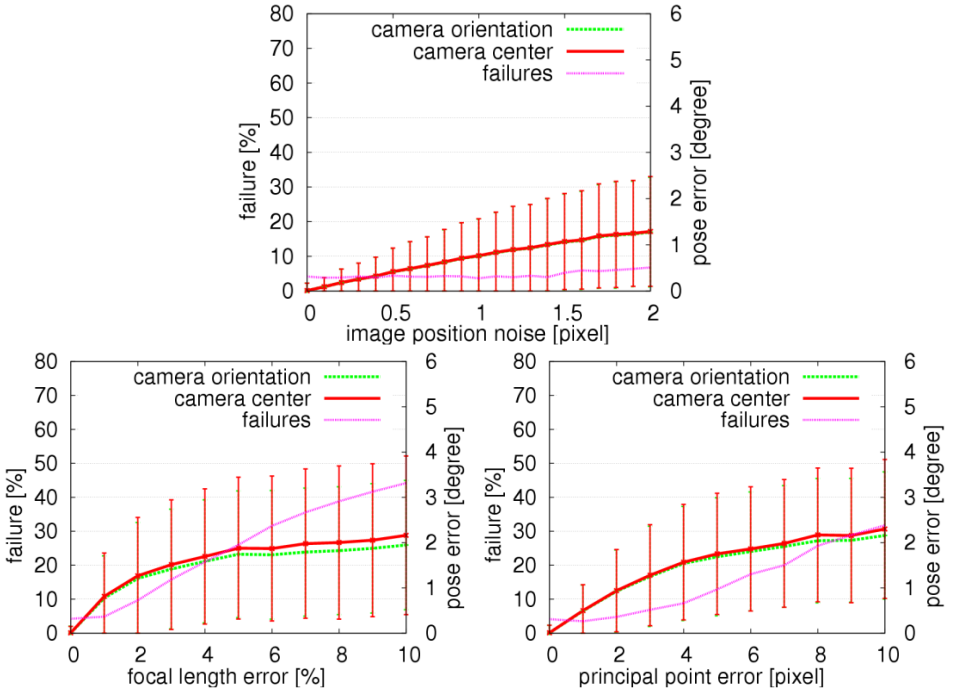

Figure 5.14: Sensitivity Analysis with Respect to Noise and Calibration Errors. In these experiments, noise has been added to the LAF correspondence (top), assumed focal length (bottom left) and principal point (bottom right). Each graph is made up from 105.000 pose estimations, for which the ground truth data is generated with a simple perspective camera with focal length 500 pixels and principal point at (100;100) with random camera poses in front of the feature. Printed are the 3D orientation error in degree and the angle between the ground truth camera center and the estimated camera center for the best of the up to four solutions. In a small fraction of cases no result can be obtained or the best result is worse than five degrees. These cases are not incorporated in the sensitivity analysis, but their occurrence can be seen in the failure curves. In the top graph the 6-parametric LAF correspondence is corrupted with additive Gaussian noise according to scaled versions of the empiric covariance (see section 4.4.2. In the bottom left graph the assumed focal length is disturbed by several percent as printed on the x-axis and in the lower right graph the principal point is disturbed by the given number of pixels (both zero mean Gaussian).

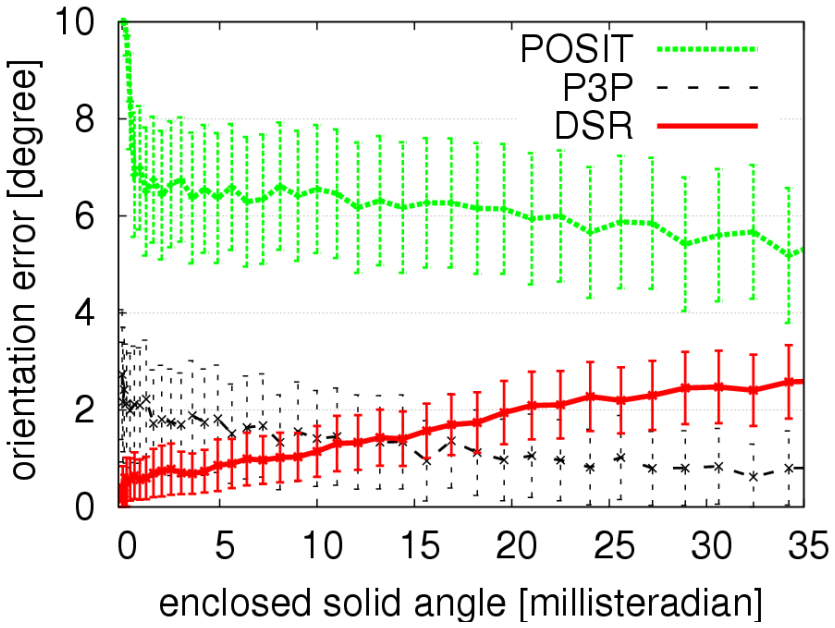

Figure 5.15: Comparison of Spatial Resection, POSIT and the LAF-based differential solution (DSR) for Small Solid Angle Patches. In this experiment a camera with focal length 500 and principal point at 100;100 has been positioned randomly in front of the z=0 plane. The corners of a patch (half window size from 10 to several hundred pixels) in the camera image have been used to estimate the camera pose, where the novel differential spatial resection solution is based upon the average affine transform of the four patch corners from the plane into the image. Each of the corners has been disturbed by additive Gaussian noise of σ_p =0.5 pixels. Due to different distances and different patch sizes, the solid angle covered by the patch also varied and is shown on the x-axis of the figure, which gives the average orientation error on the y-axis including 1/3 standard deviation. The solid angle has been obtained using the method proposed in [Van Oosterom and Strackee, 1983]. The 3-point solution proposed in the Manual of Photogrammetry[McGlone, 2004, pp.786] (P3P), the planar POSIT algorithm [Oberkampf et al., 1996] based on the four patch corners (which already includes the parallel projection approximation by [Kyle, 2004] in the POS step) are compared to the new solution, where the affine transform is approximated also based on the corners of the patch only. σ_p was fixed for the corners at 0.5. The error bars show 1/3 standard deviation. As expected, it can be seen that for large solid angle spatial resection performs best while for decreasing solid angles the novel solution gets better and better, outperforming the other approaches for very narrow constellations.

the warp (the Jacobian of the homography) does not change between the corners of the local patch.

On the other hand, when the three individual 3D points of Grunert's solution approach each other, the standard spatial resection can become unstable because it is based on the difference of the distances to the three points.

To overcome this issue, Kyle [Kyle, 2004] proposed an approximate initial guess for narrow angle images, which is the same as the POS (Pose from Orthography and Scaling) in the POSIT[DeMenthon and Davis, 1995] algorithm: Both require four non-coplanar points. For the POSIT algorithm however, there exists also a planar variant[Oberkampf et al., 1996], which copes with planar 3D points.

Therefore the LAF-based algorithm (well-suited for small solid angles) is compared to the spatial resection[Grunert, 1841, Haralick et al., 1994] implemented as proposed in the Manual of Photogrammetry [McGlone, 2004, pp. 786] Here, the size of a local square image patch is varied from ten to several hundred pixels and use the corners as individual 2D-3D correspondences in the existing algorithms. This is comparable to sampling the affine feature using triangle decomposition. For the presented method the patch corner points are used to compute a virtual local affine transform, which approximates the required Jacobian. An evaluation of the quality of the approximation can be see in the bottom right of fig. 5.14, which shows that for small solid angles the novel solution outperforms spatial resection, while for large solid angles - as expected - the affine approximation is not suitable. It is however still better in average than the orthographic approximation in the planar POSIT algorithm. Particularly, when the solid angle approaches zero, the error in the novel solution tends to zero, while for the other algorithms no solution can be obtained or the best solution is worse than the robust error threshold of 10°.

Real Texture

In the next experiment, textured views from a ground plane have been rendered and automatic matching and pose estimation has been applied. Since the ground truth poses are available the pose error can be analyzed in this case. The details of the experiment are explained in fig.5.16. Using automatic matching in presence of different intensity noise levels the pose can be estimated quite reliably, given the minimal local texture data that is used.

In the final experiment photographs of an office scene have been taken, where a cereal box is detected, which is partially occluded. As in the previous experiment, an MSER feature is selected in an orthophoto of the cereal box (here the letter "'M"') and its descriptor is recorded.

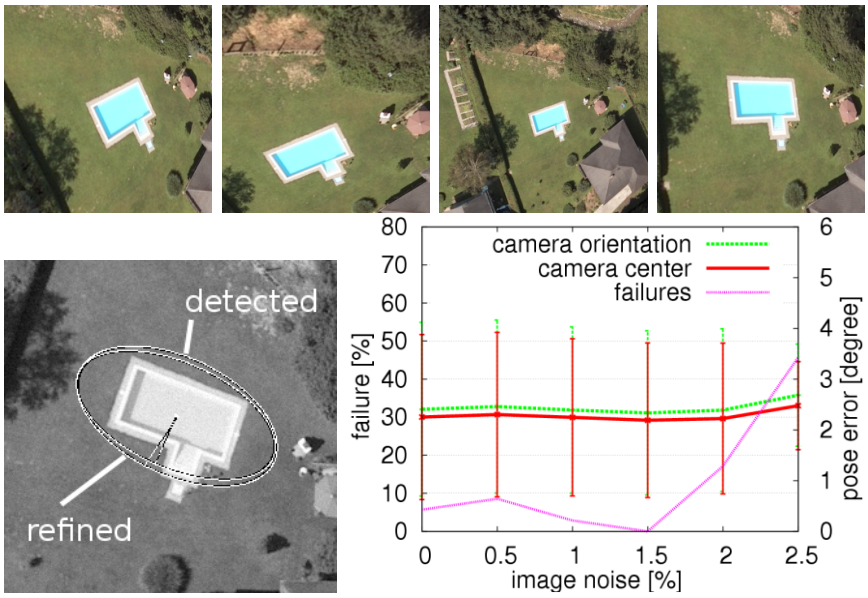

Figure 5.16: Camera Pose Estimation from Noisy Images. A ground plane has been textured with an aerial image serving as an orthophoto and a series of 40 views have been rendered with different levels of noise (upper row: sample views with low noise). A reference MSER feature with orientation has been chosen in the orthophoto. This feature is then detected in the other views and refined using a simple 6-parametric affine warp (see ellipses in bottom left image) according to [Köser and Koch, 2008b] based upon a half win size of 10 pixels. From such LAF correspondences, the camera pose is estimated and compared against the known ground truth value as explained earlier. Whenever the error was above 20° or the algorithm did not come up with a solution a failure was recorded. The bottom right graph shows the average pose errors in dependence of the added image noise. When adding much more image noise, the MSER detector is no longer able to find the feature the feature. This experiment is particularly interesting because it shows that the concept does still work when the ellipse is not infinitely small.

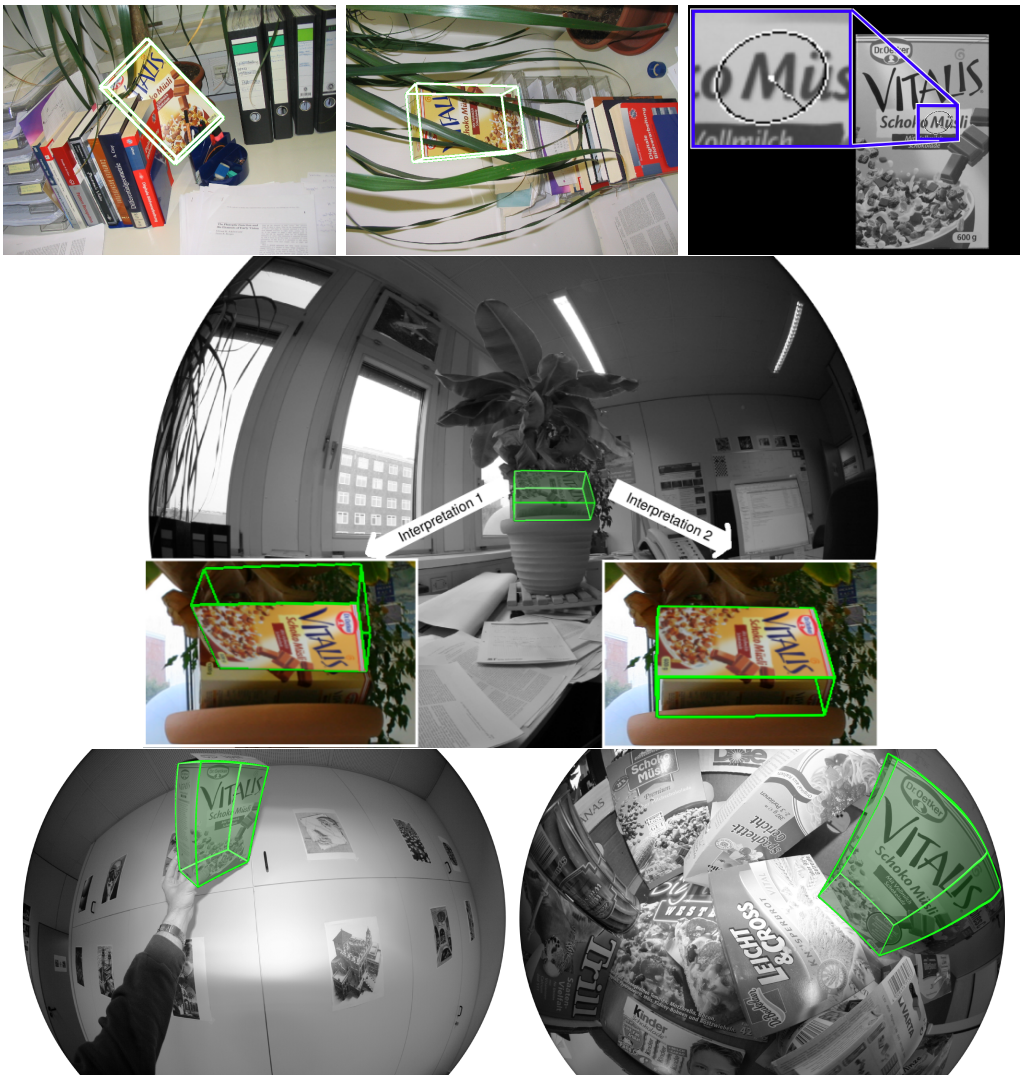

Figure 5.17: This figure shows that in a real camera even with radial distortion (upper left images) or fish-eye mapping (bottom images) object pose estimation is possible from a single feature. The orthophoto of the object is displayed in the upper right image with the local feature region enlarged. The two upper left images show cluttered views with the object partially occluded. In all views the "M" has been detected using MSER and refined, and the appropriate of the two resulting object poses from this single LAF correspondence is then displayed by augmenting a contour model. Only in the center image two possible interpretations are displayed. In both the local feature perfectly looks the same and the correct one can only be found by incorporating more global knowledge.

Then several test images are processed, where the cereal box appears. If a good match for the sample feature can be found, the LAF parameters are exploited to initialized gradient-based affine parameter optimization between the orthophoto and the test image. Finally, from this LAF correspondence the object pose is estimated. Since only the local region is used, the candidates for the object pose can be found even if large portions of the object are occluded. Figure 5.17 shows also two inferred object poses given a local warps, which are both locally plausible. As can be seen, the system works also well for fisheye cameras or cameras with other distortion.

5.4.6 Discussion

A LAF-based method for camera pose estimation has been presented, which allows estimating the pose from a single image-model correspondence using nothing more expensive than the solution of a third order polynomial. RANSAC-like approaches and applications where only few data is available or that require manual interaction can benefit from such a minimal solution. Such correspondences are often readily available from photometric matching or tracking but the information they carry has not been exploited in direct pose estimation so far. The LAF-based algorithm showed quite stable results in the sensitivity analysis and proved to be real-world applicable for camera or object pose estimation.

5.5 Summary

In this chapter geometric estimation based upon local affine frame correspondences has been demonstrated for several computer vision problems. First it has been shown how the local warp imposes constraints in general homography estimation, how multiple features can be used to obtain a unique solution and how maximum likelihood estimation can be performed based on uncertain measurements. The same principles can be applied in estimation of a conjugate rotation (e.g. the infinite homography for constant camera intrinsics), where one of the most common cases has only six DOF and allows to estimate it from a single LAF correspondence, even when the principal point of the camera is unknown. Also the first feature-based approach to estimate a general conjugate rotation and maximum-likelihood estimation with multiple features has been shown. The LAF correspondence analysis allowed also a minimal parameterization of the conjugate rotation with seven parameters.

In another application, when a LAF correspondence is observed from two

calibrated cameras at different positions, a direct solution for the position and surface normal in 3D space has been presented. Finding such a patchlet in some other view, it has been shown that the camera pose can be estimated up to a four-fold ambiguity from a single feature. This formulation is also advantageous when the three 3D points of the classical resection form a very small solid angle, i.e. when their 2D projections in the image are very close to one another. Sampling a LAF correspondence into three individual points on the other hand produces worse pose estimates for small features, but this is application specific and depends on the parameters to be estimated: In general homography estimation it yields better results, while it produced comparable or slightly worse results in patchlet estimation.

In summary this chapter showed that the geometric LAF representation can be exploited for several (and in fact for virtually all) steps in structure-from-motion and other geometrical problems in a comparable fashion as points are used nowadays. However, the LAF carries much more information: Where a simple point correspondence allows for instance to triangulate the 3D point, the LAF correspondence provides also the surface normal, and where classical spatial resection requires three point correspondences, the pose can be estimated already from a single LAF correspondence.

Chapter 6

Free-form Surface Models: Camera Tracking System

The previous chapters showed how the local 2D shape information carried in features of 2D images can be exploited. In doing so, each local feature has been considered individually and no topological connection, no occlusion and no interpolation between the small features is in the model. They provide a compact, sparse local 2D representation well-suited for geometric reasoning as shown in the previous chapters.

Given detailed 3D surface geometry for larger, curved regions, in this chapter now a scene representation for the purpose of camera tracking in reconstructed scenes is discussed: textured free-form surfaces. Besides representing also non-planar geometry of larger size and allowing for occlusion reasoning, these textured free-form surfaces can also be processed very efficiently by graphics processing units (GPUs), so that they are well-suited for analysis-by-synthesis methods.

In the following, a complete system for markerless, drift-free camera tracking based upon such surfaces is presented. Such systems can serve as visual pose sensors for various purposes in sufficiently static and textured environments. One particular application is the tracking of a camera for augmented reality applications in TV-production or industrial environments [Lepetit and Fua, 2005, Thomas et al., 1997, Koch et al., 2005]. Here, virtual 3D objects from computer graphics appear fixed to the scene even if the camera moves (see e.g. figure 6.1).

For such an augmentation to become plausible and believable, the virtual objects must not jitter or slide with respect to the scene, which requires accurate tracking of the camera pose for every single image. Therefore tracking has to be resistant to drift and the data used for tracking must be consistent and reliable. Camera tracking or reconstruction systems like [Koch et al.,

2007, Evers-Senne et al., 2006, Skrypnyk and Lowe, 2004] not using markers or any other kind of absolute reference show drift problems because the unavoidable measurement uncertainty and noise accumulate over time. In long tv shows this would degrade the augmentation, since the virtual objects would slide away from their positions.

For offline time-uncritical systems (e.g. [Pollefeys et al., 2004]) this drift can be compensated in a subsequent, global bundle adjustment step, which minimizes errors in a whole image sequence at once[Triggs et al., 2000, Hartley and Zisserman, 2004]. In interactive systems that require instant pose information however, such post-processing is not possible and therefore drift has to be avoided by other means. Concurrently with the work described here also SLAM approaches have been proposed (e.g. [Davison et al., 2007]), which basically run a simplified bundle adjustment while the camera is tracked: Using Kalman filtering techniques[Kalman, 1960], the state of 3D points and camera poses is continuously updated, when new measurements arrive under a statistical model. However, due to the limited state vector size for real-time performance and linearizations of non-linearities this system is only an approximation of a true bundle adjustment involving all data. Commercially available solutions like the VIS tracker [Foxlin and Naimark, 2003] or BBC's free-d system [Thomas et al., 1997] on the other hand use markers at predefined positions for tracking and avoid drift in this way. As a drawback, the free-d system for instance requires an expensively calibrated marker setup, which makes its application outside of prepared studio environments difficult. Another issue arises from the design of perspective cameras. The smaller the field of view, the more difficult it becomes to distinguish between translation and rotation, therefore wide angle or fisheye lenses are better suited for pose estimation [Streckel and Koch, 2005, J. Neumann, 2002, Micusik and Pajdla, 2006]. Fish-eye cameras also have the advantage that they always "see" large parts of the static scene even if objects or persons move and occlude parts of the background or when the camera rotates.

Other, e.g. purely gradient-based object tracking approaches (e.g. [Koch,

Figure 6.1: A virtual car model inserted into a scene. When the camera is tracked, virtual 3D models can be overlayed such as if they were in the scene.

1993]) cannot cope with clutter and occlusion like moving objects or persons within the scene. Purely interest point based systems tend to jitter, because they apply fast 2D feature extraction methods to every single image (e.g. SIFT in [Skrypnyk and Lowe, 2004] or Laplacian approximation in [Lepetit et al., 2005]), which can suffer from few features or poor feature localization and have to be regularized by temporal pose filtering. An overview of approaches can be found in [Lepetit and Fua, 2005].

To address the summarized issues, in the next sections a fish-eye camera tracking system, which relies on an automatically generated three-dimensional reference model, is proposed. To obtain a model, the scene is explored offline and recorded on video. That means that the tracking problem is split into two phases: an offline phase where the model is generated and an online phase where the camera is tracked in real-time. The real-time tracking is using an analysis-by-synthesis approach. Herein the reference model is used to generate an image using predicted camera parameters. The difference of the generated image and the captured image is exploited for estimation of the current pose parameters. For this approach, every part of the reference model that has sufficient texture and depth information can be used.

In the first part it is summarized how such a model can be created solely based upon image data as presented in [Bartczak et al., 2007]. In this offline phase there is enough time to perform bundle adjustment and make the model consistent. Then, when the system is used online, the camera is switched on somewhere and has to be registered with the learnt scene (initialization). When its position and orientation is known, it can be assumed that the camera moves only slightly and will have a similar pose in the next image. An approach to track the camera is then shown in the final section. Parts of the approach presented here have already been published in [Köser et al., 2007b,a, 2006a,b] and also the presented approach is part of a larger system, which can be found in [Thomas et al., 2007, Chandaria et al., 2007] and which has been demonstrated at the International Broadcasting Convention in Amsterdam[Chandaria et al., 2006].

6.1 Offline Modeling

First, a brief overview of the methods to construct a model for initialization and tracking is given. A detailed presentation of the system can be found in [Bartczak et al., 2007].

Although the applied mathematical model allows any camera with a single center of projection, it is proposed to use spherical images, in particular fish-eye lenses with a very wide field of view because of their superior proper-

ties for reconstruction[Streckel and Koch, 2005, J. Neumann, 2002, Micusik and Pajdla, 2006]. In order to be usable in scenarios that cannot be manipulated the reconstruction scheme must be flexible with respect to the extent and shape of the scene. Previously presented systems [Cornelius et al., 2004, Nistér, 2001, Pollefeys et al., 2004] achieve this flexibility by splitting the reconstruction process into structure from motion, self-calibration and dense reconstruction. In the proposed system, the self-calibration is replaced by a preceding calibration step, which proved to be more robust and avoids degeneracies. Small errors in the internal parameters can however be tolerated and these parameters can be included into optimization in a global bundle adjustment step. The resulting calibrated images are then fed into a depth estimation procedure based on a robust fusion of pairwise disparity image measurements. Since modern graphics accelerator cards are able to transform and texture huge amounts of triangles very efficiently, a model consisting of a high-resolution textured triangle mesh is generated as the final scene representation.

While complete systems for the reconstruction of models from image sequences have been proposed for ideal perspective images [Cornelius et al., 2004, Nistér, 2001, Pollefeys et al., 2004], this is not yet as advanced for general single-centered cameras. Multiple camera models exist for wide field-of-view cameras (e.g. [Perwass and Sommer, 2006, Geyer and Daniilidis, 2001, Scaramuzza et al., 2006b, Fleck, 1995, Micusik, 2004]), for which several tailored approaches for self-calibration, structure from motion, rectification or depth estimation have been proposed (e.g. [Micusik and Pajdla, 2006, Takiguchi et al., 2002, Gonzalez-Barbosa and Lacroix, 2005, Geyer and Daniilidis, 2003]). These typically exploit properties of the model e.g. in the sense that Geyer and Daniilidis [2003] assume that epipolar lines are circles in catadioptric images. Instead, the proposed system rigidly works on rays in space and assumes the internal camera calibration to be known, this way not requiring a particular model and being applicable to perspective cameras (with or without radial distortion), fish-eye lens cameras, catadioptric cameras or any other single center of projection camera that has a smooth and invertible mapping from image coordinates to unit rays in the camera coordinate system.

The benefits of using spherical rather than perspective cameras, is on the one hand the better posed problem for estimation of center and orientation [Streckel and Koch, 2005] and on the other hand the ability to consistently reconstruct large scenes. Using perspective cameras large scene reconstructions can be generated by stitching depth maps of different viewpoints. Since this is prone to errors, Kang et al. [Kang and Szeliski, 1996] proposed a system that uses cylindric panorama images from rotated cameras. Because

of the difficult image acquisition process such systems are less attractive. The UrbanScape program [Mordohai et al., 2007] fuses video from multiple perspective cameras, which together cover a very large field of view comparable to a spherical camera. Exact calibrations between the cameras and an expensive inertial sensor are required and the system is quite obtrusive. Although the speed and model size is quite impressive the models created are not intended for tracking but are targeted for prompt visualization. Also the Wägele project [Biber et al., 2006] requires the calibration between a laser scanner and a camera to construct 3D models of an environment and is not as flexible and mobile as a single camera. The major benefit in the presented work is therefore the presentation of a complete scene reconstruction system dealing with single centered (spherical) cameras. Thereby the large field of view is thoroughly exploited to improve reconstruction of large scenes.

After the internal camera parameters are calibrated, a video sequence of the scene is taken covering the area in which the camera will move later on. From this stream of spherical images, the first step in the reconstruction process is the determination of exact camera parameters for each frame. Camera parameters are typically separated into two types: On the one hand the external parameters, which describe the position and orientation of a camera and on the other the internal parameters, which define the image formation process. The kind of the internal parameters depends on the camera model assumed. In this work the camera model originally proposed in [Micusik and Pajdla, 2003] is utilized, which is described in detail in [Bartczak et al., 2007]. This model is appealing because it can represent many single centered cameras including distortions.

6.1.1 Structure from Motion from Spherical Images

Given an image sequence and an internal camera calibration, in this section the algorithm for the reconstruction of position and orientation of the camera is described. One important aspect of the algorithm is the rigorous application of decision theory and error propagation. After the motivation of uncertainty usage and its description, the computation of image-to-image correspondences is presented. These correspondences are the basic measurements for both parts of a two-stage algorithm. The first stage is called bootstrapping, used for initialization, and the second stage is tracking, used for frame-to-frame camera pose estimation from 3D points. An overview for geometry update and reference frame selection concludes this section. The underlying assumption of the algorithm is that the internal camera parameters are known throughout the sequence. This requirement of the algorithm is no major restriction because typical omnidirectional lenses have a fixed fo-

Figure 6.2: Sample fish-eye images with 190° field of view. Although the camera has been moved several meters in between, the images do not look too different and much of the surrounding of the scene can be used for tracking. As a drawback, it can be seen that the lights, or when operating outdoor, the sun is often in the field of view of the camera.

cus lenses (no zoom) and hence the internal camera parameters are constant. They can reliably be determined in an initial calibration step using a specialized scheme for spherical cameras as for instance proposed in [Scaramuzza et al., 2006b]. Knowing the internal calibration during camera path reconstruction is in particular advantageous because it allows to transfer image observations back into three dimensional Euclidean space and avoids certain pitfalls and degeneracies in uncalibrated structure from motion.

Uncertainty and Error Propagation

The underlying idea of the presented SfM algorithm is usage of information about measurement uncertainties and the rigorous application of error propagation wherever possible to avoid heuristics and hence reduce the dimension of parameter space. Unless noted otherwise, uncertainties are approximated by multivariate normal distributions parameterized by mean vectors and covariance matrices, again because the Gaussian is the least biased distribution assumption under the maximum entropy model. Error propagation is conducted either using the unscented transform or, in case of linear functions, using linear error propagation. The unscented transform can be imagined as a sparse Monte-Carlo transformation of uncertainty: It samples an isoprobability surface of a Gaussian distribution where the surface intersects the ellipsoids principal axes, transfers all samples according to an arbitrary function and computes mean and scatter of the weighted points in the new

space. For instance, when measuring a point with an uncertainty in the image, given the calibration from the previous section, the unscented transform can be applied to compute the uncertainty of the ray in the camera coordinate system. Because of the low number of well-chosen samples, it is much faster than Monte-Carlo methods and has the advantage of avoiding the analytical computation of Jacobians as needed in linear error propagation [Mikhail and Ackermann, 1976].

Many parts of the algorithm are based on projective entities and hence a representation of uncertainty in projective space is needed. Förstner [Förstner, 2005] gives an excellent overview over uncertainty representation, propagation and stochastic testing of linear and bilinear relations in projective space. One fundamental problem when dealing with uncertainties in projective space is the definition of the incidence relation between projective entities leading to an infinite number of equivalent representations of a projective vector. To circumvent this problem, Förstner suggested the exclusive usage of normalized projective entities for all operations. This can be interpreted as reducing the projective space to the surface of the unit sphere around the origin. A very convenient side effect of this approach is that uncertainties of viewing rays of a camera can be easily represented without numerical difficulties, even if they are perpendicular to the optical axis. Hence the algorithm can naturally cope with cameras with a field of view of more than 180°.

An important advantage of using uncertainties is the reduction of the parameter space. The question of incidence for example can be easily coped with using a single and easily interpretable parameter for all occurrences (instead of multiple outlier thresholds). This is conducted using the χ^2 test [McGlone, 2004] in the projective space (see also appendix B.3.1).

Correspondence Estimation

The reconstruction process is separated into a bootstrapping and a tracking stage and in such it is similar to the approach presented in [Pollefeys et al., 2004]. However, instead of matching corner features in each image, LAF correspondences can be used as well. Correspondences in subsequent video images can simply be estimated using iterative, gradient based minimization of intensity differences in a surrounding window as proposed in section 4.2.5. Of course, the approach can as well be based on simple 2D correspondences. After convergence of each minimization, the remaining grey value MAD (mean absolute difference) for the tracked window is computed: Low MAD values represent good matches of the local image window, while mismatches typically produce high MAD values. Since the MAD depends on

the camera (e.g. image noise) and the scene (e.g. lighting change), a fixed threshold to reject mismatches is hard to define. Therefore, a robust statistical model is applied to the set of all correspondences between two subsequent images, which relies on the assumption that more than half of the correspondences are correct: All correspondences that have a much worse MAD than the Median of all MADs are rejected (X84 rule [Hampel et al., 2005]). This can be seen as a dynamic threshold that automatically adapts to increased noise as proposed in [Fusiello, 1999].

To account for violations of the image brightness constancy assumption, all image patches used in the correspondence estimation process are intensity-normalized to zero mean and unit variance during the estimation process. This intensity normalization allows reliable correspondence estimation even in the presence of strong illumination changes (compare [Baker et al., 2003]). The uncertainty of the displacement vector can be approximated using the standard technique for overdetermined linear systems based on the residuum and on the Jacobian of the equation system [Mikhail and Ackermann, 1976]. This provides correspondences between the images.

Having described how correspondences and their uncertainty are obtained, in the next section the reconstruction of the camera path is discussed. This is divided into an initialization (boot-strapping) stage and a frame-to-frame tracking stage before being optimized by a global bundle adjustment.

Bootstrapping Stage

Bootstrapping of the reconstruction process is based on robust estimation of the essential matrix $\mathbf{E}$ (cf. to [Hartley and Zisserman, 2000]) from 2D-2D correspondences between the first and a second frame from the image sequence. The essential matrix captures the relative pose between two views of a scene and can be computed based on 2D-2D correspondences. Care must be taken to assure that sufficient baseline exists between the two initial images for successful initialization [Beder and Steffen, 2006]. Without further knowledge, the relative pose can be determined only up to scale from image correspondences. Image correspondences typically contain a significant amount of erroneous data (outliers) stemming, for example, from repetitive textures or measured at depth discontinuities. Also, correspondences on moving persons or objects that violate the static scene assumption must be rejected. Hence a robust approach, like for example the RANSAC [Fischler and Bolles, 1981] algorithm or the preemptive RANSAC [Nistér, 2003] algorithm in combination with the 5-point algorithm [Nistér, 2004], must be used for the estimation process. Each correspondence is classified into either inlying or outlying based on the χ^2 test in the projective space [McGlone,

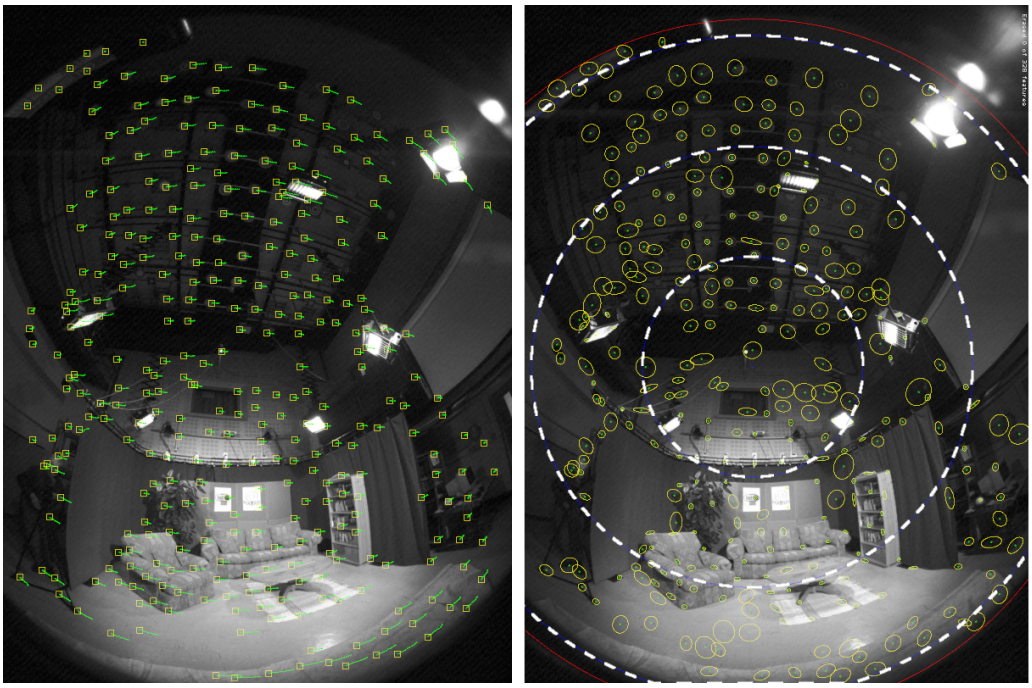

Figure 6.3: Sample fish-eye images with 190° field of view. In the left image, the trajectories of tracked regions can be seen. In the right image the concentric dashed circles show the 60°,120° and 180° field of view circles, the small ellipses show the 20 times magnified standard confidence region of each feature.

2004], where the decision is based on the fundamental constraint (cf. to [Hartley and Zisserman, 2000]). This way, a threshold for outlier detection can be determined, which depends on the uncertainty of the data itself and a probability to erroneously reject a good correspondence (false negative rate). This probability is a scene independent parameter of the system and can be chosen e.g. as 1%. Robust estimation is described in more detail in section C.1.

Afterwards the resulting essential matrix is refined by minimizing the Mahalanobis distance between inlying points and their corresponding epipolar lines. A more in-depth review of essential matrix estimation can be found in [Woelk, 2008].

Using the refined relative pose and the image correspondences, 3D features are determined including their uncertainty. To obtain two initial poses and a set of 3D points from the correspondences compatible with the essential matrix the standard methods as described in [Hartley and Zisserman, 2004] are applied. After successful bootstrapping the algorithm switches to the tracking stage.

Tracking Stage

Once this first reliable relative camera orientation and position, together with a sparse scene geometry have been found, the reconstruction scheme switches to frame-to-frame pose estimation based on 2D-3D correspondences. Again the features are tracked into subsequent frames propagating the information about the corresponding 3D structure, which is Kalman-filtered. Whenever the number of tracked features drops below some threshold (e.g. several hundred) a feature detector is applied to the most recent image and the new features are tracked along with the older ones. Due to perspective effects, lighting and image digitization, feature uncertainties in general grow with larger distances.

The pose reconstruction from 2D-3D correspondences is done by minimizing the Mahalanobis distance between the projection of the 3D features and their uncertainties and the parameters obtained by the tracker. Outliers are detected prior to the estimation using again a RANSAC [Fischler and Bolles, 1981] algorithm in combination with stochastic testing of incidence using again the χ^2 test.

Geometry Update

The underlying assumption is that of a rigid scene and hence the image sequence can be regarded as subsequent measurements of the 3D geometry of

each tracked feature. A separate Kalman filter for each point is used to integrate the information from these measurements into the 3D geometry with associated uncertainty. Each new feature measurement is used in the update step of the Kalman filter cycle. Tracked features that have no associated 3D correspondence are now triangulated if their uncertainty is below a threshold. 3D features which have not been tracked into the image under inspection are projected into that image and registered using the KLT approach. This allows to re-use temporarily occluded features and to avoid drift when features get out of and come into sight again. After one image is completed, correspondences are searched in the temporally subsequent image using the KLT tracker, and pose estimation and geometry update is run for that image.

Reference Frame Selection

Since in dense video sequences nearby frames are highly correlated but each creates vast amounts of data, a strategy is needed to discard such useless data. Therefore each computed pose is compared to the previous images' poses in the sequence and forgotten if it has too little innovation, so that only some reference frames are remembered. Innovation is understood as the amount of parallax between the camera and all other views. The parallax can be measured by the 2D flow field induced by the sparse 3D reconstruction. Before these reference views are passed to dense reconstruction their corresponding camera parameters and the sparse scene geometry is refined in a bundle adjustment step. While the bootstrapping and incremental reconstruction approach presented so far already produces a good initial estimate for each 3D point and each camera pose, the final bundle adjustment procedure is intended to obtain an optimal least-squares solution for the whole reconstruction problem.

6.1.2 Bundle Adjustment

After application of the above described SfM algorithm a sparse Euclidean model of the scene is given. However, small errors in the internal camera calibration can lead to small errors in the camera poses and the triangulated 3D features. Furthermore, after bootstrapping a first two view geometry, the reconstruction was run incrementally so that drift can occur and errors due to noise are not fairly distributed.

Since it is desirable for subsequent depth estimation and model building that drift is avoided and that all camera parameters are estimated with about the same reliability the reconstruction is made more consistent by the application of a final bundle adjustment of the sparse reconstruction, using

the Gauss-Markov model [McGlone, 2004]. A comprehensive overview of the bundle adjustment technique can be found in [Triggs et al., 2000]. Additional to the cameras' pose parameters and the 3D features, internal camera parameters can be subject to the global optimization. These internal parameters are however constrained to be the same for all views, since only one camera with fixed intrinsics has been used.

To account for remaining outliers from the SfM stage, to which simple least squares approaches are very susceptible [Hampel et al., 2005], the error of each individual measurement is clipped to λ pixels[1]. This is similar to an M-Estimator using a Huber influence function [Zhang, 1997]. The Levenberg/Marquardt method (cf. to [Triggs et al., 2000]), which blends between the Gauss-Newton method and pure gradient descend, is used to iteratively approach the minimum of the error function. This way a decrease of error for each step is guaranteed, even if the quadratic error function fit of the Newton method does not reflect the error surface well. Having reached the minimum of the error function, the computed camera poses, the intrinsic parameters and the sparse scene reconstruction can then be exploited to reconstruct scene surfaces.

6.1.3 Dense 3D Reconstruction

After having reconstructed the camera path and a sparse model of the scene, a dense textured model of the scene has to be estimated containing at least the interesting surfaces for tracking. The first step here is computing pairwise dense depth estimates from promising image pairs. Here, one reference view is fixed and for each other image that has a promising baseline with respect to the sparse scene reconstruction, a dense correspondence map is obtained. Since the camera poses of the images are known, corresponding image points directly imply the depth of the associated 3D scene points. These depths are stored in a depth map of the fixed reference view, such that for this view finally many depth maps exist. They are fused using a robust statistical approach and provide a dense approximation of the scene surface. Together with the texture from the images, a triangle mesh model is then set up from the data, which can be exploited for tracking. Further details on the reconstruction can be found in [Bartczak et al., 2007].

[1] for the presented reconstruction system it was found that the average projection error of several hundred 3D points is usually in the range of up to one pixel, therefore $\lambda = 1.0$ is chosen.

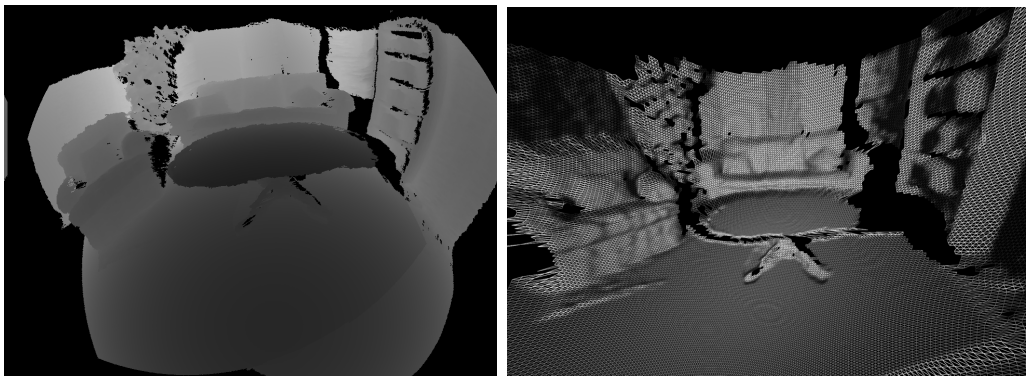

Figure 6.4: A depth map (left) for a fish-eye view and a triangle mesh model without texture (right). Dark pixels in the depth map are closer to the camera than brighter, while completely black pixels have unknown depth. The depth map and the triangle mesh have been postprocessed here for illustration purposes.

6.2 Initialization using a Descriptor Database

The 3D model constructed in the previous section is a reconstructive model of the scene. Given the pose of the camera, a model view of the scene can be synthesized, which is suitable for analysis-by-synthesis tracking. If the camera pose is however not known at all, as typical for a starting phase, a discriminative model of the scene is required that can identify which parts of the scene can be seen. The system proposed in [Köser et al., 2006b] can be used to generate and apply such a model. Also other, similar linear subspace methods became recently popular for feature lookup [Mikolajczyk and Matas, 2007, Winder and Brown, 2007, Hua et al., 2007]. Before the details of the system are presented, a short summary on other view registration techniques is given.

6.2.1 Previous Work on View Registration

In the field of camera registration in a scene, traditionally markers have been used [Thomas et al., 1997, Foxlin and Naimark, 2003, Kato and Billinghurst, 1999]. Early marker-less approaches on registering views in Augmented Reality scenarios tried to compute orientation only, e.g. using the Fourier-Mellin-Transform [Stricker and Kettenbach, 2001, Stricker, 2002] in a panorama scenario where the camera could only rotate but not change its position. The Fourier-Mellin method is a phase-based method to obtain a 2D similarity transform for the whole image, which breaks in presence of significant

perspective distortions, or if parts of the image are cluttered, occluded or look different. Other methods use histogram or P-Channel matching [Felsberg and Hedborg, 2007a,b], line-based registration [Thomas, 2007, 2006] or local features [Nistér and Stewénius, 2006, Lepetit et al., 2005, Köser et al., 2006b]. The local feature based methods have been applied successfully for image-to-image matching, panorama registration and also six DOF camera pose computation [Skrypnyk and Lowe, 2004]. The principle is that a particular detector/descriptor combination produces the same - or a slightly different - *feature vector* for the same 3D region under different conditions, while producing other vectors for differently looking regions. The feature vector can be interpreted as a signature. Among all the detector/descriptor pairs, DoG/SIFT [Lowe, 2004] is known to perform well [Mikolajczyk and Schmid, 2005] and can be computed quite fast. Although being invariant only against scale, rotation and affine brightness change of a 2D image, it is robust against mislocation, perspective effects and several other distortions. Robust means that small violations of the ideal conditions will cause only small disturbances of the feature vector. In that case the feature vectors occupy a small local area in the feature space. This makes it well-suited for the view registration purpose and it will be used in this section, though the proposed techniques can also be applied to other features with such properties like the affine features discussed in section 3.6. Often however, full affine invariance is not necessary because there is information available on the scenario under consideration. In [Köser and Koch, 2007] it has been shown that e.g. in panoramic scenarios (constant camera center) or when depth information is available, tailored approaches can lead to increased descriptiveness while providing at the same time also increased invariance. The important aspect is that for the transformations considered (here 6DOF pose in a certain range) a large fraction of the features can be redetected and obtains a similar descriptor.

For a 2D feature to provide significant information to discriminate it from others, the descriptions must be quite high-dimensional. On the other hand, when one seeks to find a similar feature vector in the space of all possible features, it is easy to run into the curse of dimensionality if the description vector is too large. One way of organizing points in high dimensional spaces is space-partitioning using kd-trees[Beis and Lowe, 1997]. In three dimensions this is comparable to a binary octree, which separates each dimension only into two segments. For the typical SIFT feature dimension of 128 a complete binary space partitioning would create a tree with 2^{128} (more than 10^{38}) leaves. Therefore the method of dimensionality reduction given in [Beis and Lowe, 1997] is compared to different methods of learning the relevant parts of the high-dimensional feature descriptions. Such methods

have been applied successfully in face recognition [Belhumeur et al., 1997] and other classification tasks, often however only on the raw image signal: Multiple discriminant analysis ("fisher-faces") and principle component analysis ("eigen-faces"). The advantages over the technique of using vector entries with largest variance, which is a common feature space matching technique today [Skrypnyk and Lowe, 2004] are also shown. In contrast to PCA-SIFT [Ke and Sukthankar, 2004], the goal is not to find out which dimensions of the SIFT descriptor are less interesting in general and to find the subspace of all feature descriptions a DoG/SIFT operator can produce on the set of all images ever possible. This encodes what all descriptors do have in common. Instead, the approach explicitly wants to learn what is *different* between the clusters of features *in a specific scene*. The learning is deliberately based on the knowledge that different representatives belong to the same class like in [Grabner and Bischof, 2005], but the approach does not only seek for one representative per class but also looks for a transformed small representation to make a fast distinction between the classes possible.

In that sense the idea is somewhat related to the Randomized Trees approach [Lepetit et al., 2005], which does not rely on high-level features but on massive simple tests. Instead of performing a nearest neighbor search in one tree and applying a decision, they propose a soft-classification by using several trees, where each tree node encodes class probabilities. The final classification is performed by combining the probabilities. While this is an interesting approach in the handling of the probabilities, the authors have proposed it only for recognition and pose computation of single objects, presumably because the simple decisions made in the trees sacrifice discriminative power for the sake of speed. They have not evaluated whether the approach does also extend to larger scale scenarios.

6.2.2 Scene Database

To register a view in the online phase a database of features is exploited, which has to be set up offline. During the creation of the free-form surface model as explained in section 6.1, also robust image features are tracked from the video sequence, and in each image their descriptor vectors are extracted and stored.

3D Features

If the descriptors for corresponding 2D points do not vary too much across several images, it can be assumed that the invariance/robustness properties of the feature type are still satisfied, e.g. for SIFT features that the 2D image

regions are projections of a three dimensional locally continuous surface from similar viewpoints and that all projections of this surface result in similar descriptors. The surface is called a 3D feature in the following. However, the surface shape in 3D is of no interest here, only the class of descriptors it produces, as it has been proposed in [Köser et al., 2006b]. It is assumed that they form a continuous area in descriptor space and their differences are e.g. due to small localisation distortions or transformations against which the descriptor is not completely invariant. Combining incremental structure from motion (in contrast to the reference image technique of [Skrypnyk and Lowe, 2004]) allows to process long image sequences with lots of descriptor measurements.

If each class of descriptors covers a coherent and relatively small part in the high-dimensional descriptor space, and any two distinct classes are at different locations in this space, the matching process can be viewed as a classification problem. For each 2D feature detected in the online phase the best matching class in descriptor space is wanted. Beis and Lowe proposed an approximate nearest neighbor search on a kd-tree partitioning the descriptor space [Beis and Lowe, 1997]. The partitioning should at best represent the distribution of the various classes, therefore some parts in the feature space are more interesting than others. To traverse a balanced binary tree of depth d (e.g. $d = 15$) one has to pass d decision hyperplanes, which divide the feature space. This tree has 2^d leaves (distinct areas in feature space). If d is too large (for instance the original vector size 128), this leads to an unmanageable number of bins (2^{128}). Even for depths not much larger than 20, the tree is over-fitted and only sparsely populated, unless one uses a huge number of features. For a small d on the other hand the question is extremely important, which is the best partitioning of the space and what are good dividing hyperplanes. Beis and Lowe solve the problem by computing the variance of each descriptor dimension across all features and select only the most variant entries. Instead, here it is proposed to apply classical methods of dimensionality reduction from pattern recognition. These methods are compared next.

Dimensionality Reduction

From the offline phase there are many 3D features that have been seen in several images. Each 3D feature defines a class with mean and scatter in feature space. Let $\boldsymbol{D}_c^i \in \mathbb{R}^h$ be the ith (of n_c) descriptor vector for class c (of a total of n classes). Since it has h entries, there is an h-dimensional descriptor space (e.g. for SIFT typically $h = 128$). A reduction transformation $\mathtt{R}\left[\boldsymbol{D}_c^i\right] = \boldsymbol{d}_c^i : \mathbb{R}^h \mapsto \mathbb{R}^l$, which shrinks the descriptor to a low dimension number l (e.g.

$l = 15$) is required now. However, the descriptor should not lose too much discriminative information needed for matching.

Principle Components Analysis The most popular approach to dimensionality reduction is principle component analysis (PCA). PCA computes the mean and scatter of all descriptors (see [Duda et al., 2001]). Different means are now defined as follows:

$$\boldsymbol{\mu}_c = \frac{1}{n_c}\sum_i \boldsymbol{D}_c^i \qquad \boldsymbol{\mu} = \frac{1}{\sum_c n_c}\sum_c\sum_i \boldsymbol{D}_c^i \qquad \boldsymbol{\mu}_{\text{Means}} = \frac{1}{n}\sum_c \boldsymbol{\mu}_c \tag{6.1}$$

$$\Sigma = \sum_c(\sum_i((\boldsymbol{D}_c^i - \boldsymbol{\mu})(\boldsymbol{D}_c^i - \boldsymbol{\mu})^\mathsf{T})) \tag{6.2}$$

The principal components are now the eigenvectors e_Σ^j of Σ according to [Duda et al., 2001]:

$$\Sigma \boldsymbol{e}_\Sigma^j = \lambda_\Sigma^j \boldsymbol{e}_\Sigma^j \tag{6.3}$$

where $\boldsymbol{e}_\Sigma^j$ are sorted according to their eigenvalues λ_Σ^j, λ_Σ^0 being the largest. Let $\hat{\boldsymbol{e}}_\Sigma^i$ be the normalized eigenvectors:

$$\hat{\boldsymbol{e}}_\Sigma^i = \frac{1}{\sqrt{\lambda_\Sigma^i}}\boldsymbol{e}_\Sigma^i \tag{6.4}$$

Finally, the reduction transformation PCA is defined as:

$$\mathtt{R}_{\text{PCA}}\left[\boldsymbol{D}_c^i\right] = (\hat{\boldsymbol{e}}_\Sigma^0\ \hat{\boldsymbol{e}}_\Sigma^1\ ...\ \hat{\boldsymbol{e}}_\Sigma^l)^\mathsf{T}(\boldsymbol{D}_c^i) \tag{6.5}$$

See [Duda et al., 2001] for a detailed derivation. A slight modification of PCA, called PCA-Means in the following, takes into account classes and is computed only using the means of the classes, which gives an equal weight to each class and does not prefer strongly populated classes over small ones. The only difference in computation is that equation (6.2) is replaced by equation (6.6), where Σ_{Means} is also called the inter class scatter matrix:

$$\Sigma_{\text{Means}} = \sum_c((\boldsymbol{\mu}_c - \boldsymbol{\mu}_{\text{Means}})(\boldsymbol{\mu}_c - \boldsymbol{\mu}_{\text{Means}})^\mathsf{T}) \tag{6.6}$$

Compared to classical PCA definition, the mean is neglected in the PCA reduction methods (see equation (6.5)). However, since the reduction transformation is linear, the mean also transforms linear and introduces a constant offset for all features, which can be ignored since only the nearest neighbor is of interest and no absolute position.

PCA is designed to minimize the reconstruction error, therefore it is suitable for compression/uncompression of similar vectors in high-dimensional space. However, it does not account for classes and does not aim at preserving separability of vectors in reduced space. In other words, PCA preserves what is common between two classes, not what is different. The goal of finding a linear transformation that maximizes class separability is the topic of discriminant analysis.

Multiple Discriminant Analysis Now an extension of multiple discriminant analysis (MDA) [Duda et al., 2001] is proposed that falls back smoothly to PCA in case only sparse within class information is available. The idea of MDA is to represent each class of descriptors by a mean and scatter and find a transformation $\mathtt{R}_{\mathrm{MDA}}$ that minimizes within class scatter while maximizing the scatter of all class means. The within class scatter Σ_c and the total scatter matrix Σ_{total} (imagine as an average within class distribution) are defined as:

$$\Sigma_c = \frac{1}{n_c - 1} \sum_i ((\boldsymbol{D}_c^i - \boldsymbol{\mu}_c)(\boldsymbol{D}_c^i - \boldsymbol{\mu}_c)^\mathsf{T}) \qquad \Sigma_{\mathrm{total}} = \frac{1}{n} \sum_c \Sigma_c \tag{6.7}$$

The rows of the reduction transformation matrix are the solutions e^j to the generalized eigenvalue problem [Duda et al., 2001]:

$$\Sigma_{\mathrm{Means}} \boldsymbol{e}^j = \lambda^j \Sigma_{\mathrm{total}} \boldsymbol{e}^j \tag{6.8}$$

If Σ_{total} is nonsingular, the system can be converted to a standard eigenvalue problem like equation (6.3). However, particularly when Σ_{total} is estimated from few samples in the high-dimensional space, it will be singular, mainly because of missing data. A full rank can be enforced by applying ridge regularization [Skurichina, 2001] to the total scatter matrix, i.e. $diag(\sigma^2)$ (a diagonal matrix with entries σ^2) is added. Small values of σ do not affect the shape of Σ_{total}, while larger ones make the diagonal dominate the matrix and very large values make it in fact a multiple of the identity matrix. In that case, equation (6.8) is the same as equation (6.3) for PCA-Means, therefore the value of sigma controls between MDA and pure PCA behavior. Since within class shape information should be preserved, a minimum noise level is computed, as the smallest existing eigenvalue of equation (6.8) clipped against a minimum empiric noise. This leads to a smooth transition from PCA-Means to MDA as soon as within class scatter is available.

Most Variant Entries The approach chosen by Beis and Lowe [1997] can also be viewed in the context of dimensionality reduction. They compute the

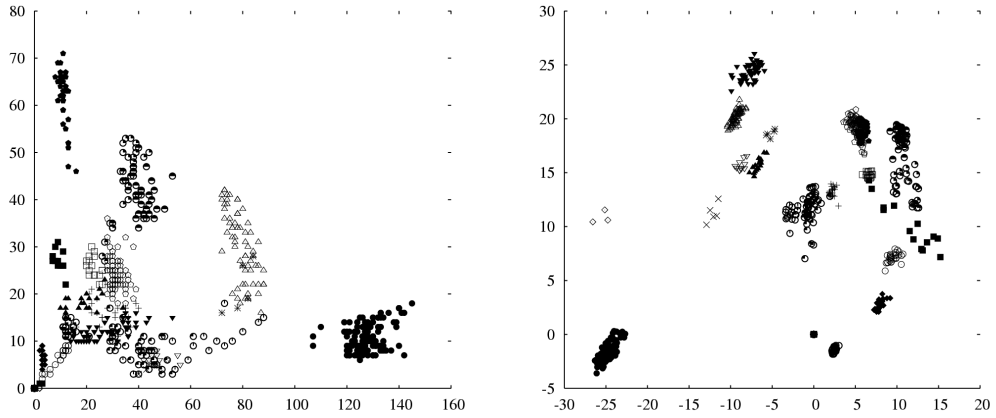

Figure 6.5: Exemplary distribution of features of 20 classes (randomly chosen) of a real video sequence of 400 images projected to the first two axes (left), the first two MDA axes (right). MDA representation shows more distinct local clusters .

variance for each vector entry separately. This corresponds to only taking into account the diagonal elements of Σ of equation (6.2) and sort vector components by these values. By disregarding the off-diagonal elements, the relations between the vector entries are thrown away. This is suboptimal for descriptors whose components are correlated, which is certainly the case for the SIFT descriptor, because the soft-binning technique distributes gradients into different vector entries upon mislocalization. In other words, the entries of the SIFT descriptor are not uncorrelated as a strictly diagonal scatter matrix would imply. However, adopting this scheme, the resulting reduction matrix is a pure permutation of the columns of the identity matrix.

Database Representation

According to the work presented in [Köser et al., 2006b], in this work the extension of the MDA method is used, since it is the most powerful transformation. Hence, the original feature vectors can be transformed into a space where the dimensions are sorted by importance. A kd-tree is then built in that space with depth $d = \texttt{log}_2[c]$ such that in average each bin holds a class. In the transformed space the method of [Beis and Lowe, 1997] is then applied.

Retrieval of Features and Camera Registration

Once the database is set up, the offline phase is finished. In the online phase, when the camera is switched on, 2D features can be extracted from an unknown image. Each feature is transformed according to the reduction and traverses the tree using the backtracking strategy [Beis and Lowe, 1997] until a better match in reduced space cannot be found, a maximum error in reduced space is reached or - if real-time is an issue - a (constant) maximum number of comparisons has been reached. The best match so far or "no match" is returned. The "no match" statement is particularly important because it decreases the false positive rate. Fewer outliers again speed up robust pose computation, which is done using RANSAC[Fischler and Bolles, 1981].

6.3 Tracking Free-form Surface Models

Given an initial pose, the expected view from the model can be rendered, the pose can be optimized according to the differences in the rendered and the captured image and a pose for the next image is predicted using a motion model. If the optimization fails, re-initialization using the robust features of the previous section 6.2.2 can be performed. Otherwise the tracking process continues with the next predicted pose and the next image. The free-form surface model used for rendering can be created with the methods of [Bartczak et al., 2007] or can be any other textured VRML or CAD model of the scene. Regions in the model that are definitely not suitable for tracking (e.g. highly reflective areas, non-rigidities like water) and that give rise to many incorrect matches should be removed from the model beforehand to avoid unnecessary additional outliers. On the other hand, it can be helpful to interpolate or adjust uncertain or missing regions (e.g. on planes) that are suitable for tracking to increase the number of trackable surfaces.

The key idea is that feature tracking is improved by compensating the features' appearances with respect to 3D viewpoint and lens effects, which can efficiently be done on graphics hardware with sub-pixel accuracy. In that sense the proposed approach is similar to the one in [Denzler et al., 2003, Heigl et al., 2000], which used plenoptic models of an environment to estimate a three DOF robot pose in a particle filter approach. The authors render multiple hypothesis images and assign a score to each, depending on a similarity between the rendered and the real image. However, with increasing dimension (e.g. full six DOF for camera pose), more samples (i.e. rendered images) are required, and the approach slows down. Instead, in the

system discussed in this thesis, a hardware-accelerated approach is proposed, which exploits depth information from a model, this way allowing to directly compute the camera pose from only one synthesized image of the model as described next:

From the predicted, approximate pose, a fish-eye image of the offline model can be synthesized using the same (intrinsic and extrinsic) camera parameters as the real fish-eye camera has (see section 6.3.1). Ideally, no intensity differences between rendered and real image should be visible. Therefore these could be minimized to obtain the correct camera pose. However, local illumination change and moving scene content violate the image brightness constancy assumption. Consequently, a global gradient-based approach as in [Koch, 1993] cannot be used to estimate the pose parameters in the given scenario. Instead, local 2D offsets of individual free-form surfaces are determined using the KLT approach [Lucas and Kanade, 1981]. From the exact locations of the surfaces in the camera image the final camera pose can be computed in a robust way as described in section 6.3.2. Section 6.3.3 is dedicated to the evaluation of the system on real and synthetic data followed by a discussion.

6.3.1 Spherical Camera

In this system a wide field-of-view camera is proposed, e.g. with a fish-eye lens, which has a nearly linear and isotropic relation between distance in pixels to the principal point and the angle between the ray and the optical axis [J. Neumann, 2002]. Fleck [1995] calls this the equidistant projection. A comparison between spherical and perspective cameras regarding tracking can be found in [Streckel and Koch, 2005], which showed that pose estimation is more accurate with a wider field of view and that the lower angular resolution of the fish-eye lens is more than compensated by its wide field of view. Furthermore, such a camera covers a larger solid angle and therefore features can be seen for a longer period of time in image sequences.

Let $\mathtt{P}$ be the function that computes a 2D image point $\boldsymbol{x}_i$ from a 3D scene point $\boldsymbol{X}_i$, which takes care of all internal and external camera parameters of the real camera (CCD size, lens distortion, camera pose $\boldsymbol{p}$, ...):

$$\mathtt{P}\left[\boldsymbol{p}, \boldsymbol{k}, \boldsymbol{X}_i\right] = \boldsymbol{x}_i \tag{6.9}$$

$\mathtt{P}$ is actually composed of extrinsic camera parameters, i.e. the pose $\boldsymbol{p}$ (position and orientation) of the camera, and intrinsic camera parameters $\boldsymbol{k}$, which describe the mapping of 3D points *in the camera coordinate system* to image coordinates, i.e. the image formation process. The internal camera

parameters do not change during tracking, since they depend only on the lens and the hardware, here only the pose is unknown. The internal camera transformation can be described with the function $\mathtt{K}_{\boldsymbol{k}}$, where $\mathtt{K}_{\boldsymbol{k}}$ maps projection rays in the camera coordinate system to 2D points in the image depending on a vector of internal parameters $\boldsymbol{k}$. Therefore when a 2D image point $\boldsymbol{x}_i$ is measured in any camera, $\mathtt{K}_{\boldsymbol{k}}^{-1}$ can be applied to compute the ray that maps the image point onto the unit sphere within the camera coordinate system. The mapping from world coordinates to *normalized camera coordinates* (a ray) by $\hat{P}$ is now defined as:

$$\hat{\mathtt{P}}\left[\boldsymbol{p}, \boldsymbol{X}_i\right] = \mathtt{K}_{\boldsymbol{k}}^{-1}\left[\boldsymbol{x}_i\right] = \hat{\boldsymbol{x}}_i$$

where $\hat{\mathtt{P}}$ is only a function of the pose and the 3D point. $\boldsymbol{k}$ can be determined by calibration [Scaramuzza et al., 2006b]. If the effects of $\mathtt{K}_{\boldsymbol{k}}$ are removed from the image measurement, one can compute on rays in the camera coordinate system, which is quite flexible and abstracts from the underlying hardware: Although the proposed wide field-of-view cameras provide the aforementioned advantages, the methods in this contribution are not restricted to a particular fish-eye lens and can in principle be applied to all calibrated cameras with a single center of projection.

Virtual View Synthesis

In a similar way to how the 2D positions and uncertainties have been normalized given an internal camera calibration, a fish-eye image can be synthesized using the graphics hardware (compare figure 6.6). The GPU already provides very efficient techniques of generating ideal perspective images of virtual scenes, e.g. scenes represented as textured triangle meshes. In order to synthesize a fish-eye image from a given camera position, it is possible to render six perspective views in the six main directions (left, front, right, back, bottom, up). In computer graphics this is known as *cube-mapping of environment* [Salomon, 2006]. Afterwards these 6 images are stitched together to form a fish-eye image by means of indirect texture look-up. This again exploits that the optical ray for each pixel in the fish-eye image is known by calibration and that therefore the relevant perspective cube face can be chosen. Furthermore the coordinates are known where the perspective camera observes this ray. This technique can be efficiently implemented using OpenGL/CG [Fernando and Kilgard, 2003] and runs directly on the graphics hardware. Furthermore the cube's z-Buffer values can be processed to produce a spherical depth map in a similar way. With the same scheme, perspective views with radial distortion or other single center of projection

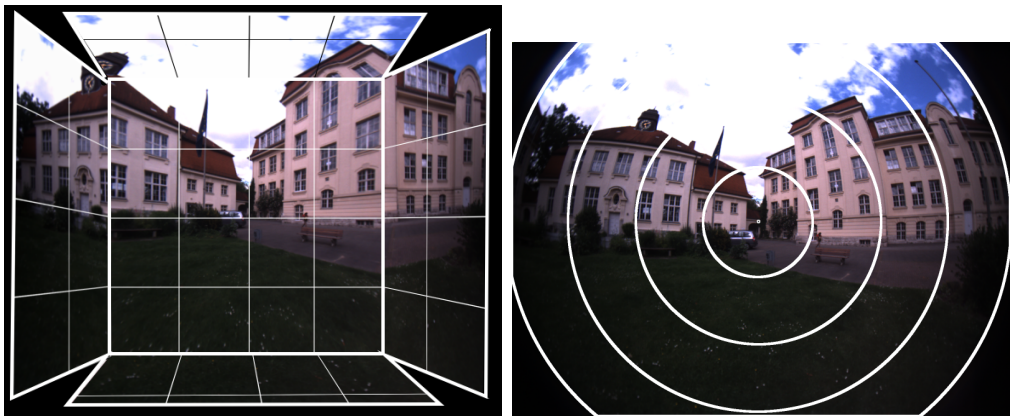

Figure 6.6: Top: 3D view of cube-mapped environment (perspective views) Bottom:Fish-eye image with center, 45°, 90°, 135° and 180° field of view circles. To synthetically generate a fish-eye image, the faces of a cube are rendered as perspective images with 90° field of view. Each cube side color image and z-map is rendered into a separate texture using frame buffer objects. The stitching is performed using a displacement texture holding the direction of the optical ray for each pixel of the final image. From these directions the accessed cube side and the texture coordinates are calculated within a fragment shader and the appropriate colors and z-values are transfered to the final frame-buffer.

cameras can be simulated. The resolution of the cube maps must be carefully chosen so that the stitched image can still contain the highest frequency components. This requires the determination of an upper bound of the local minification during stitching. Aliasing on the other hand is avoided automatically, e.g. using trilinear texture filtering on the graphics hardware.

6.3.2 Camera Tracking

Synthesizing a Model View

Given an approximate pose, the model is rendered (compare section 6.3.1) and the true pose is computed from the displacement vectors between regions of the rendered image and the real camera image. By using a fish-eye lens one has all the advantages in visibility and geometrical stability, however the appearance of the model is quite different between distant camera poses. Therefore the rendering must undistort these effects by warping the model image into the new viewpoint and allowing to establish correspondences using standard techniques like the KLT[Baker and Matthews, 2004].

The fish-eye image is synthesized from the model (cf. section 6.3.1). The assumption here is that the reconstructed surfaces are Lambertian (cf. to [Jähne, 2005, p.191]), i.e. from each viewpoint a point on the surface is observed with the same color. Real surfaces however usually have also specular properties, which are not captured in our models. Such effects can be handled better using plenoptic rendering [Heigl et al., 2000] or view-dependent texture mapping, where the model is textured from a previously stored view close to the current view (as e.g. in [Koch et al., 1999]). A drawback of such dynamic texturing techniques is that blending strategies are required when moving from one viewpoint to another. If temporal illumination changes can occur (dynamic lighting), which is typical for outdoor scenarios, none of the methods will synthesize the exact camera image, therefore the simple textured mesh approach is followed and illumination will be compensated locally as explained later.

Correspondences between Image and Model

After the free-form surfaces are rendered, simple tests such as clipping techniques can be used to check geometrically which ones are projected into the virtual image. For those, image-to-model correspondences are sought.

Each rendered free-form surface can serve as an anchor for tracking as long as it has a minimum size, so that tracking is feasible. Its center point $\boldsymbol{x}_i$ is back-projected to create its corresponding 3D point $\boldsymbol{x}_m$ on the model using the depth from the renderer. This delivers a 3D model feature.

For each surface individual gradient-based minimizations of the intensity differences at these locations $\boldsymbol{x}_i$ between the patches in the synthesized and the real image (see figure 6.7) is performed to obtain a 2D displacement vector. This is more robust than a gradient-based global optimization of the pose across the whole image [Koch, 1993], since several scene parts may be occluded by persons or other unmodeled objects and it is hard to decide within one iteration step, which pixels should be used and which not. In contrast, for a whole free-form surface the projection error can be tested to see whether it is an outlier [McGlone, 2004]. Furthermore, for each free-form surface, different local lighting changes may occur, e.g. due to temporarily illuminated regions or dynamic shadows in outdoor scenarios, and therefore it is good to have a local brightness compensation.

The difference minimization is always carried out between a synthesized image and the current camera image. This way, the offline model serves as a global reference and no drift will be accumulated as it would be the case when one tracks from camera image to camera image. The rendering can be seen as a fish-eye compensation of the free-form surface for tracking. In controlled

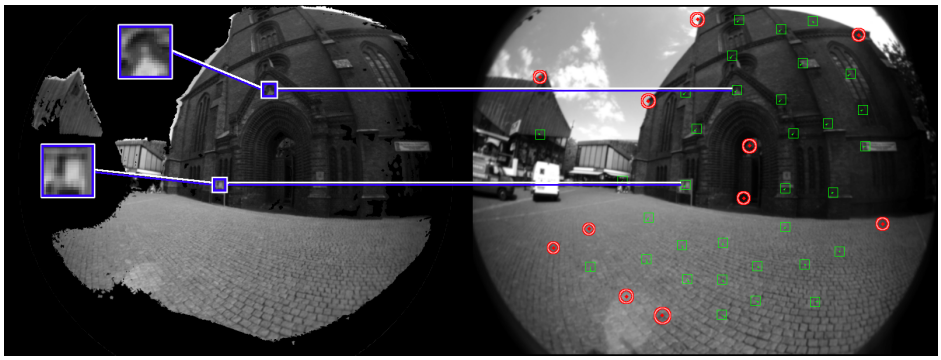

Figure 6.7: Correspondences between a rendered model view (left) and a camera image (right). Corresponding regions are shown as small green squares in the right image, while rejected features appear as red circles in the right image. For classifying the tracks into success and failure (e.g. due to moving persons) the X84 rule [Fusiello, 1999] based on the median SAD value of all features is applied. Two blue squares show magnified regions in the rendered image.

environments the standard KLT tracker can be used, which measures only 2D offsets for a region, since the prediction (the rendered image) is usually very close. A comprehensive overview of this gradient based image registration and KLT tracking can be found in [Baker and Matthews, 2004].

For outdoor sequences, which usually suffer from illumination changes depending on time of day and weather, a light insensitive version of the KLT should be used, which relaxes the *image brightness constancy assumption* of the original algorithm to an affine brightness model: It is assumed that illumination changes can locally be explained by a brightness scale and an offset that is constant for one local free-form surface. For a detailed overview of tracking with appearance change compare [Baker et al., 2003].

One difficulty which has to be addressed is that the free-form surface model contains holes, where no texture and no depth is present. For efficiency reasons only those free-form surfaces are used for tracking that project to a completely filled $n \times n$ window in the 2D view, where the window size n is a parameter, which controls between tracking speed and accuracy, e.g. $n = 9$. A different solution would be to use a mask in the window and to ignore unrendered pixels. This provides more flexibility in the features to be usable but would mean an overhead in each KLT iteration step.

A free-form surface localization has converged when a position update step is below dx pixels (e.g. $dx = 0.1$), which again controls between speed

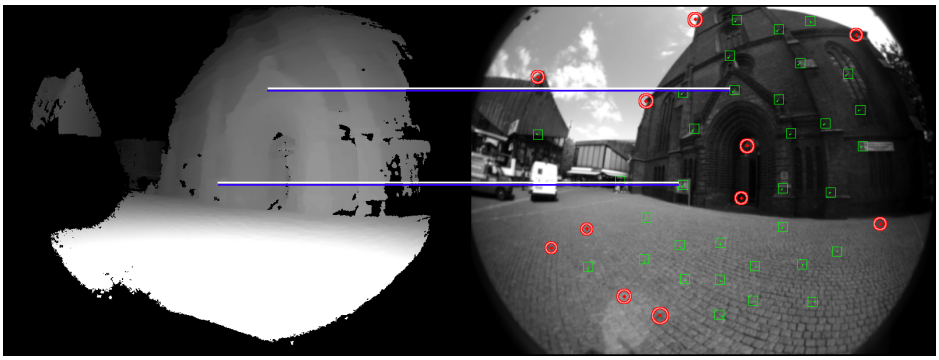

Figure 6.8: 2D positions of successfully registered free-form surfaces (right) and the depth map from the used model view (left). From this data, each 2D position is assigned a 3D model correspondence from the depth map, which can then be fed into the pose estimation.

and accuracy. After convergence, the MAD for the whole patch is computed and then the X84 rule as described in [Fusiello, 1999, Hampel et al., 2005] can be applied to the set of all tracked free-form surface MADs. This is a noise-adaptive, dynamic threshold, which rejects those correspondences whose MAD is larger than l times the Median of all MADs for that image, i.e. which behave much worse than the majority of correspondences. This way, photometric outliers are removed and many occluded features, moving persons or unmodeled objects are greedily rejected.

For the remaining (good) features 2D-3D correspondences from the patch center in the image and the back-projected 3D point from the model (compare figure 6.8) are established. This is in contrast to the plenoptic approach (cf. to [Adelson and Bergen, 1991]) for probabilistic camera tracking from [Denzler et al., 2003, Heigl et al., 2000], which does not provide depth. Using the approach of this section the rendered image is not simply scored as a whole, but the 2D-3D correspondences provide information how to correct the pose to make the rendered and the real image coincide, which is described next.

Robust Pose Estimation

The resulting 2D-3D correspondences are then processed in a robust non-linear pose estimator, which starts at the predicted pose and minimizes the ray error for all 2D-3D correspondences. To limit the influence of single remaining mismatches an M-Estimator with Huber [Zhang, 1997, Huber, 1964] error function $\mathtt{h}$ is applied. More precisely, the position $\boldsymbol{x}_i$ and the

covariance matrix $C_{\boldsymbol{x}_i\boldsymbol{x}_i}$ from the KLT tracker in the original image are transformed to a local ray $\hat{\boldsymbol{x}}_i$ and a covariance $\hat{C}_i$, which is obtained through an unscented transform[Julier and Uhlmann, 1997]. Now the Mahalanobis distance between $\hat{\boldsymbol{x}}_i$ and the ray of the 3D point is minimized, where the transformed covariance $\hat{C}_i$ of the tracked point defines the Mahalanobis error metric.

$$\sum_i \mathtt{h}\left[(\hat{\boldsymbol{x}}_i - \hat{\mathtt{P}}\left[\boldsymbol{p}, \boldsymbol{X}_i\right])^\mathsf{T} \hat{C}_i^{-1} (\hat{\boldsymbol{x}}_i - \hat{\mathtt{P}}\left[\boldsymbol{p}, \boldsymbol{X}_i\right])\right] \rightarrow min \tag{6.10}$$

The objective is the pose p that minimizes the sum of these distances for all points. After convergence of the Levenberg-Marquardt minimization (cf. [Zhang, 1997]) of the above error, the incidence test [McGlone, 2004] is used to determine for each tracked surface whether it is an inlier or an outlier. The outliers are removed and the optimization is performed again.

Looped (Iterative) Rendering

Once the pose is computed it is possible to render an updated fish-eye image from this optimized pose and perform the KLT step again. Ideally, all features would already be at the correct positions and the real and the rendered images would be identical. However, small offsets due to only few or noisy features from the previous iteration might still occur and can be exploited to iterate towards an even better pose. If and how many iterations are needed depends on the quality of the pose prediction and therefore mainly on the speed and smoothness of camera movement and the speed of computation. Within the camera movement the rotation is the most critical part because at a certain distance to the scene fast rotations change the fish-eye image more drastically than fast translations.

At first sight it is conceivable that multiple rendering iterations might have several advantages: Spurious tracking errors in high-frequency repetitive patterns could be corrected, e.g. if the wrong brick or paving stone is matched out of many. One might also wonder whether at a new rendering pass a new free-form surface snaps in, which ties a previously uncertain degree of freedom of the camera pose.

However, in practice this turned out to be a rather theoretical consideration: If the prediction is really wrong, no feature will snap in at the first iteration and another rendering pass will not help, because there is no updated pose available. In [Köser et al., 2006a] it has been shown that no significant reduction of pose error could be observed when iterating more than two or three times. If the prediction is already good and if performance is an issue, tracking with fewer or even only one rendering iteration is feasible.

Therefore the number of rendering iterations is a quality parameter, controlling between accuracy and tracking frame-rate. Interestingly, the prediction quality improves automatically when tracking frame-rate increases because pose differences become minor when images are taken frequently enough. Another way of improving the prediction is to use additional inertial or rotation sensors[Chandaria et al., 2006].

6.3.3 System Evaluation

A quite natural and intuitive evaluation of a tracking system designed for augmentation is to augment and see whether the visual impression is good, which is however not objective. Therefore here synthetic image sequences have been rendered from reconstructed models with real texture, where exact ground truth pose information is available for quantitative evaluation. Nevertheless, qualitatively it is shown that the system runs in real environments and that it copes with the difficulties usually not modeled in synthetic data (moving persons, illumination effects,..).

As discussed earlier there are several parameters of the system that control between speed (frame-rate of tracking) and accuracy of the system, e.g. accuracy of the model, convergence criterion of the KLT patch registration or number of rendering passes. The focus here lies on the principles of the overall system and on the evaluation of sensitivity and applicability to certain environments, while specific optimizations of well-known components are less interesting and can be found in the literature, compare e.g. [Bleser et al., 2006, Molton et al., 2004, Bleser et al., 2007] for implementations of KLT and 2D-3D correspondence based pose estimation in real-time.

In the next sections several problems occurring in typical tracking applications are discussed and it is shown how the proposed system behaves in presence of disturbances. First, the influence of the accuracy of the model used for tracking is discussed, afterwards the model-based wide-angle tracking is compared to other standard tracking approaches. In the second part of the evaluation, the absence of drift on long sequences and insensitivity to lighting conditions, clutter and moving scene content as well as photometric distortion of parts of the image is demonstrated on several image sequences from outdoor and indoor (compare table 6.1).

Sensitivity to Model Accuracy

The accuracy of the pose estimation depends on the goodness of the model used for tracking. Therefore, rendering speed and average pose estimation accuracy is monitored (figure 6.11) at varying resolutions of the triangle

Scene	Type	Extent (m^3)	T	R
Sofa	rendered	$5 \times 4 \times 2$	1.5m	80°
TV Studio	real, indoor	$6 \times 4 \times 6$	3m	60°
Office	real, indoor	$5 \times 5 \times 3$	2m	60°
Church	real, outdoor	$50 \times 20 \times 12$	5m	50°
School	real, outdoor	$40 \times 20 \times 10$	5m	60°

Table 6.1: Overview of tested scenarios: Reconstructed scene size as width x depth x height and extent of camera translation (T) and rotation (R) in test sequences.

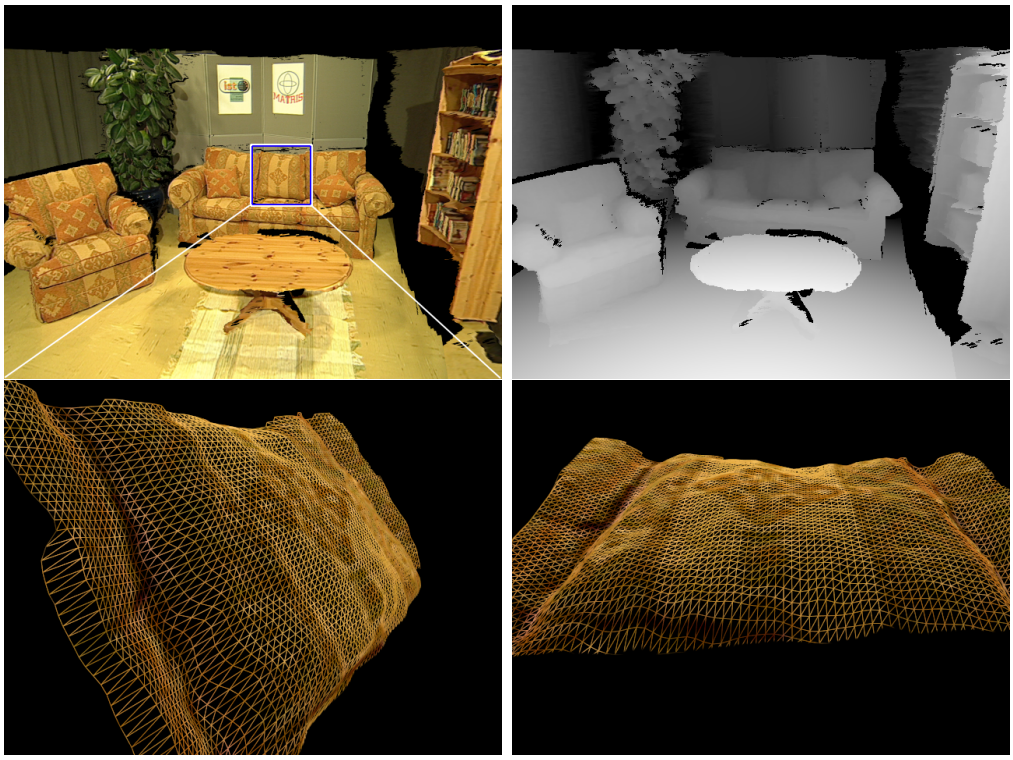

Figure 6.9: Upper Row: Perspective view and depth map of the reconstructed *Sofa Scene*, Bottom Row: Close view of full resolution mesh of the sofa region (see box in upper left image): This mesh cannot be approximated well by using just one or two planar patches because it is really free-form.

mesh for tracking (figure 6.10). As ground truth a model of a real living room scene (compare table 6.1) with real textures as reconstructed by the offline modeling part (see figure 6.9) is used. The model is fused from four perspective depth maps of the scene, consists of 1.2 million triangles with a bounding box of about 5m × 4m × 2m. A sequence of 350 fish-eye images (140° field of view, camera translation about 1.5m, rotation in all directions, where the vertical axis rotations dominates by up to 80°) with ground truth pose information has been synthesized for testing.

The number of triangles is reduced from about 1.200.000 down to 1.600 by a combination of depth map resolution reduction and quad tessellation similar to what has been proposed in [Evers-Senne and Koch, 2003]. Some of the meshes can be seen in figure 6.10.

The main result of this evaluation is not surprising: With increasing number of triangles rendering performance goes down; if the GPU resource limit is reached, real-time tracking becomes infeasible. The pose estimation error decreases about logarithmically with increasing number of triangles. In the extreme case of only a few triangles the scene is actually represented by planar patches, which turned out to be only usable as long as the underlying scene is planar. Otherwise the rendering does not fulfill the undistortion goal: the rendered and the camera image look significantly different and cannot be matched by the KLT tracker. Only those points are found that actually do lie on planes and are approximated well. A fair tradeoff for this particular scene is to choose about 100.000 triangles.

Comparison against other Algorithms

The proposed system has been compared using the Sofa scenario (compare table 6.1) against a) incremental structure-from-motion using the same camera with 140° FoV but no model and b) model-based tracking with a 40° FoV perspective camera with the same number of pixels and same number of surfaces. The pose error is given as position (translational) error in *cm* and orientation error in *degree*. The orientation error thereby incorporates deviations in all three axes because it is taken from the axis-angle representation of the rotation between the ground truth camera and the estimated camera. The sequence is run forward and backward, generating a total of 700 frames with image 1 and 700 at identical pose. The results in figure 6.12 show that the model-based fish-eye tracking outperforms both other approaches. The error is very low and constant over the complete sequence with average position error of 0.3 cm and orientation error of 0.1°.

The structure from motion algorithm a) has no prior model and generates the model *on the fly*. Therefore, the average pose error is higher than with

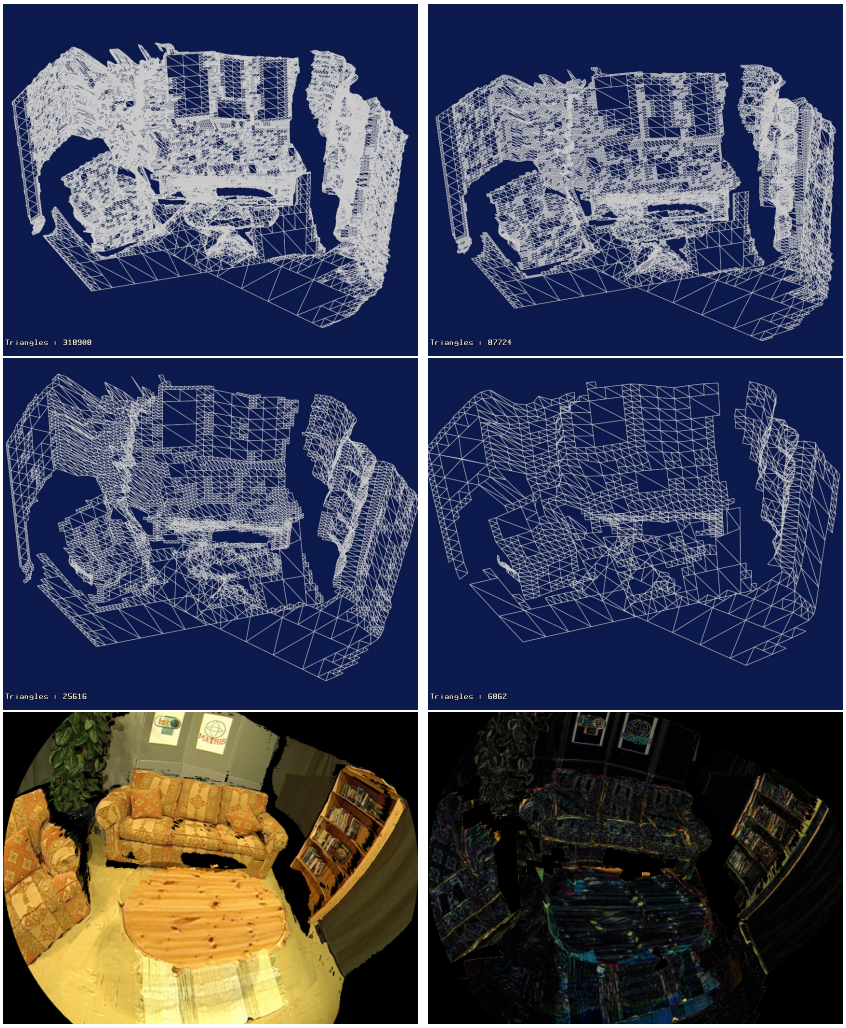

Figure 6.10: Reduced Meshes (Number of triangles: Top Left: 318908, Top Right: 87724, Center Left: 25616, Center Right: 6862), Bottom Row: Ground Truth Fish-eye Image (left) and Photometric Difference (right) for a 6862 triangles sample view from meshes above

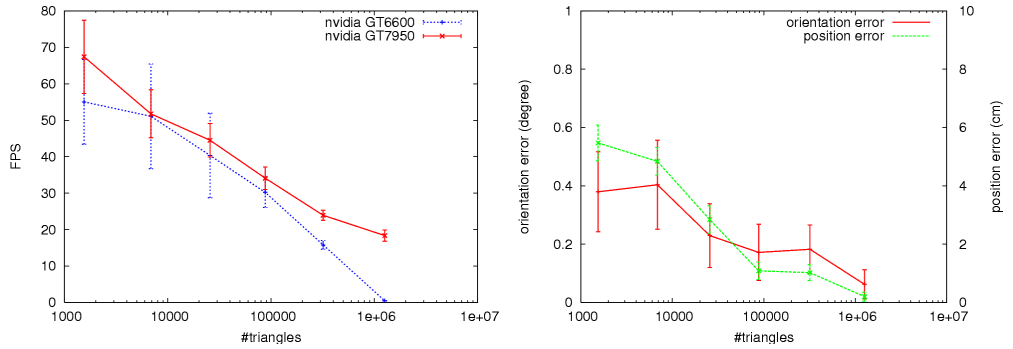

Figure 6.11: Model detail in number of triangles. Left: Pure Rendering Frame-rate, Right: Average Orientation and Position Error on Living Room Sequence

the model. Scale was fixed so that the tracking can be compared with the model-based approaches. Drift does not accumulate very much since all features are visible in most images, however an error increase is observable as the camera moves away from the initial position. A bundle adjustment, which would clearly help, but which is not feasible in real-time applications, has deliberately been left out. The average position error is about 2 cm, the average orientation error 0.3°.

The perspective model-based tracking b) on the other hand has difficulties in distinguishing between camera rotation and camera translation, which might be the reason for the high correlation between orientation and translation error in figure 6.12. Furthermore it does only see about 100 of the about 500 free-form surfaces in the model at a given time because of its limited field of view. The errors are much higher with average position error 4 cm and orientation error 0.8°.

Absence of Drift

A very important aspect of the presented analysis-by-synthesis approach is the absence of drift, which allows to track infinitely long sequences inside of the 3D model range. To prove the applicability of the method here a real sequence can be used (see also figure 6.6), which consists of 1400 images of 1600×1200 pixels, and which were taken with a fish-eye lens covering a viewing angle of 185°. The camera was moved hand-held and translated approximately 6m sidewards while panning up to 90°. The filmed buildings were up to 20m away and 12m in height. The camera path was reconstructed according to [Bartczak et al., 2007] using the full fish-eye images as explained

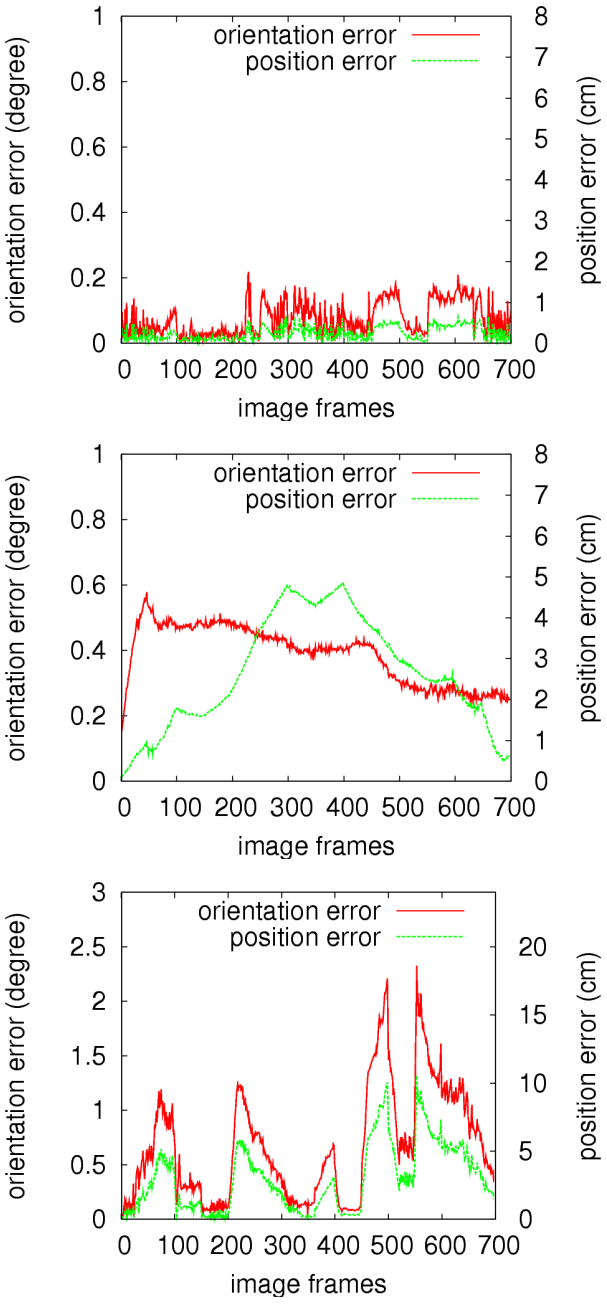

Figure 6.12: Algorithm comparison on ground truth living room sequence (350 images forward+backward). Top: Fish-eye Model Tracking, Center: Fish-eye Structure From Motion, Bottom: Perspective Model Tracking (40° FoV), which produces the worst results, printed here with triple error range.

Figure 6.13: Top: Perspective view onto reference model *School* from above and extent of camera movement used for drift measurement. The camera is going forward/backward from one end of this path to the other. It passes the green middle camera, where the pose estimation is compared to previous and following passes (see table 6.2). Bottom: augmentation of model view and real image using estimated parameters. The texture used for the augmentation is the gradient magnitude of original texture, strong gradient edges colored in red.

earlier. The resulting depth map was used to create a mesh yielding a 3D model of the scene, which consists of 90303 triangles (compare figure 6.13, top).

Without ground-truth data, the verification of the estimated camera path is difficult. One way to check for consistent model and camera path reconstruction is to augment the model into a sequence. The bottom image of figure 6.13 shows an augmentation of the model rendered with the estimated camera parameters. In order to provide an augmentation that is distinguishable from the background image, the texture of the model was replaced by its gradient magnitude. While evaluating the model tracking, the difference images between the original image and the rendered model view were monitored. This qualitative evaluation showed that the observable tracking error was in the range of one pixel.

In order to analyze potential accumulation of errors in pose estimation for long sequences, 360 consecutive images of the real sequence were processed forwards and backwards several times, starting at the middle of the sequence. The central image position is reached eight times and compared to the first pose, which should always be the same. Figure 6.13 shows the extent of this path, which is approximately 2 meters to the left and to the right of the middle camera (green, center of path). Looping through this sequence resulted in 2160 images for tracking. Given the pose for the first (central)

pass	SfM Tracking		Model Tracking	
	ΔT	$\Delta\phi$	ΔT	$\Delta\phi$
1	2.57 cm	0.098°	0.73 cm	0.047°
2	1.92 cm	0.085°	0.82 cm	0.047°
3	1.92 cm	0.085°	0.69 cm	0.046°
4	2.79 cm	0.11°	0.73 cm	0.047°
5	2.41 cm	0.11°	0.84 cm	0.048°
6	1.06 cm	0.02°	0.68 cm	0.046°
7	3.53 cm	0.13°	0.73 cm	0.047°
8	3.22 cm	0.14°	0.81 cm	0.047°

Table 6.2: Pose error evaluation for a looped image sequence, which passed the image under inspection eight times. The SfM columns show the position and orientation error using a structure from motion based tracking and how pose estimation has drifted when passing this image. The model columns show the avoidance of error accumulation when tracking is supported by a model.

image, the camera poses for this "oscillating" sequence are estimated using SfM tracking and model based tracking with 400 features for both. Model based tracking uses only one rendering iteration.

Table 6.2 compares the error development at the middle image over consecutive passes of a looped sequence using tracking on fish-eye images, but without a model. Although the error is not constantly growing with each pass, an error increase is visible. On the other hand the tracking error observed using the model (last two columns) is confined and does not increase over consecutive passes. This confirms that the system does not drift. Furthermore the pose error is smaller at all times when compared with the SfM tracking.

Occlusion and Lighting

In another outdoor sequence, results in a really challenging scenario are presented: tracking in front of a church in a busy pedestrian precinct (compare figure 6.14 and 6.7 for the model creation process). Illumination changed quite significantly between model creation and tracking and even between subsequent frames during tracking, because lots of clouds moved and let the sun appear and disappear again. This does not only change the brightness of the image but amplifies local shadows and stresses relief effects. Furthermore, when using a fish-eye lens, the sun is almost always in the camera image, leading to blooming effects, corrupted image lines and optical ring

reflections within the lens.

Apart from the lighting, people moved in front of the camera even during the reconstruction process. For visualization purposes online video has been augmented with a transparent model, which shows only some edges of the church as reference. It can be seen that the augmentation stays stable even when persons move in the scene or when the light changes (compare sample views in figure 6.14). Even in the case where the CCD blooming effects distort parts of the image, the pose can be estimated from the remaining good free-form surfaces, because the subdivision of the model allows to track each "feature" individually. The KLT residuum check already rejects the blooming outliers.

Evaluation of Further Scenarios

The tracking system has been tested in additional scenarios (see also table 6.1), of which a brief overview is given in figure 6.15.

As a qualitative evaluation, the sequences used during reconstruction have been looped several times. In both scenarios, the pose error (compared to the offline model) for the first image of the sequence is about the same as for all its other occurrences in the sequence and does not systematically increase:*TV Studio*: below 0.1° orientation and 1 cm translation error. *Office*: up to 0.2° orientation and 1.4 cm translation error. This shows that the system can cope with these scenarios and does not drift. The average pose difference with respect to the offline reconstruction of the *Office* sequence was 0.23° ($\sigma = 0.12°$) orientation and 0.95 cm ($\sigma = 0.54$ cm) translation. On the *TV Studio* sequence the orientation difference was 0.27° ($\sigma = 0.16°$) orientation and 4.1 cm ($\sigma = 2.9$ cm).

6.3.4 Discussion

A complete camera tracking system has been discussed, which first builds a textured model from the environment and afterwards uses the model in an analysis-by-synthesis approach for tracking. Initially, an automated registration is performed based on a database of robust features. Afterwards, the graphics hardware is exploited to render a distortion-compensated and perspectively warped model image with an approximate pose. Since this compensates the effects of the lens and viewpoint, now full advantage can be taken of the fish-eye properties, which proved to be superior to perspective cameras in tracking. It was shown that there is no drift accumulation over time and therefore the system is well-suited to work on infinitely long image sequences. The accuracy of the model approximation should fit well the free-

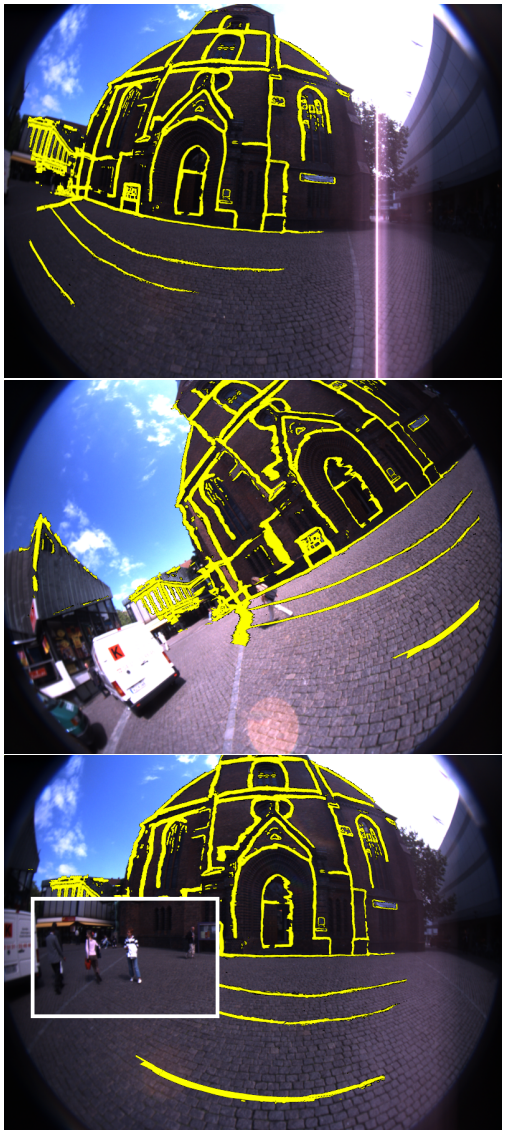

Figure 6.14: Lighting and Moving Persons in the *Church* Sequence. Sample images are augmented with an edge model only containing some yellow lines as reference for visualization. Moving persons (bottom): The rectangle in the lower image covers a horizontal field of view of more than 60 degrees, which appears rather small due to the large field of view of the proposed system. Light changes: Observe that the sun heavily degraded some of the images (blooming, particularly in top image) and changes the local appearances of some regions (e.g. the floor in the center image). Compare figure 6.7 for a rendered view from the model.

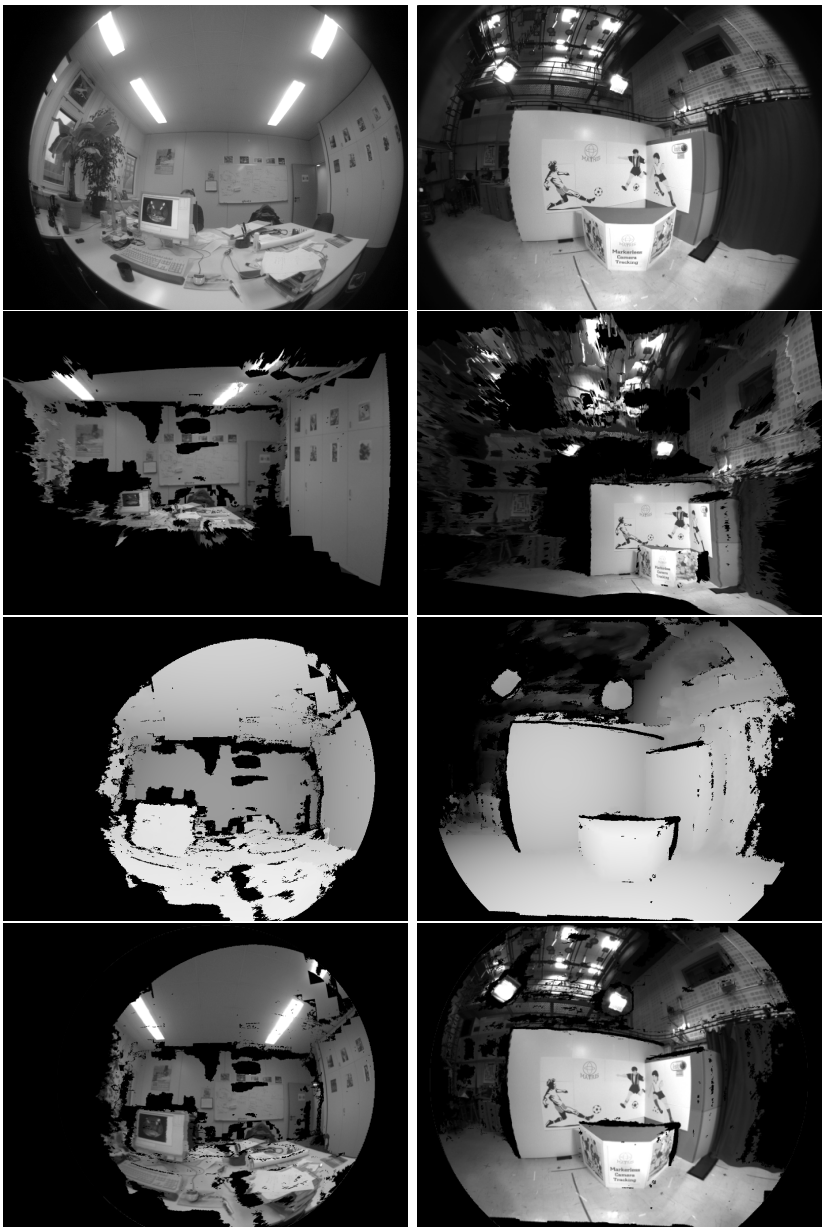

Figure 6.15: Example scenarios *Office* (left) and *TV Studio* (right), each with sample image, perspective view of 3D model, rendered depth map and rendered image (top to bottom). The approximate scene volumes are 5m × 5m × 3m for the office and 6m × 4m × 6m for the TV studio. The real camera images can be calibrated reliably only up to a certain percentage(e.g. 90%) of their field of view, therefore the rendered images use only that smaller field of view and the active area is smaller.

form surfaces, since planar approximations of curved surfaces degrade the accuracy. On current GPUs a model complexity of about 100.000 triangles is feasible. The system is robust against outliers and drastic lighting changes and works well even in challenging outdoor scenarios.

Chapter 7

Conclusion

7.1 Summary

During the last ten years, substantial advances in solving the correspondence problem for images from significantly different viewpoints have been achieved. As a result, nowadays local image regions can automatically be determined that correspond to approximately affinely warped versions in another image. As a consequence, the relative rotation, shear and scale between these regions became implicitly available. In geometric estimation however, this information has mostly been disregarded and the region correspondence was used as a simple center point correspondence only.

Therefore, the main contribution of this thesis is the introduction and mathematical derivation of a geometric primitive called the local affine frame correspondence. Such a correspondence between two images or an image and a 3D surface augments the traditional point-to-point relation with a local, linear warp of the surrounding textures. This local, linear warp is actually the derivative of the image to image transformation at the feature position and imposes constraints, when the Taylor representation of the transformation in Euclidean 2D space is inspected. In many situations the warp information is readily available or would be easy to obtain. It has been shown that conic correspondences can be seen as a squared formulation of this approach, while at the same time the LAF correspondence is more powerful in that it carries more information. Curvature of lines on the other hand is related to a 1D version (hence providing less constraints). Only triplets of points are geometrically comparable and can be understood as a sampling of the local affine frame.

Although the base local affine frame concept has been used in matching to correct for local distortions for quite some time, no algorithms existed for

explicit geometric exploitation of the local affine frame other than in non-linear optimization or by approximating the local affine frame with a triplet of points. In this thesis it has been shown, how the correspondence can be exploited to express differential constraints onto the global transformation using exemplary problems of homography, pose and normal estimation, which are important parts in scene reconstruction and photogrammetry. It has also been shown how a local affine frame is transformed using an arbitrary analytic and well-linearizable function into a local affine frame in another image, allowing to reason with local affine frames in a similar way as with simple points.

Due to the power of the primitive used, less correspondences than previously necessary have to be used for estimating the inspected transformations. This is particularly interesting in settings with few data, with manual user interaction, or in settings with high fractions of outliers, because in this case the complexity often depends exponentially on the number of samples required to construct a minimal solution. Furthermore, it has been shown that the novel primitive is numerically stable in various applications. In sensitivity analyses and practical evaluations it turned out that for reasonable noise assumptions of today's consumer cameras, all of the inspected estimation problems are remarkably stable. In some cases the sampling into three point correspondences in practice yielded comparable or slightly better results as with the differential constraints, particularly when simple algorithms like the DLT are used. On the other hand, the parameterization and estimation of the conjugate rotation is essentially based upon the differential constraint and no algorithm is known to estimate it from three point pairs so far. Using the LAF correspondence concept, also the number of degrees of freedom for the conjugate rotation (the infinite homography for constant calibration) and the first minimal parameterization have been discovered. Additionally, the first algorithm to estimate a general conjugate rotation has been proposed. Furthermore, it has been shown that pose estimation is possible from a single feature correspondence. In the same context it could be shown that the differential formulation exploiting a LAF correspondence is better-suited for small solid angles than the classical spatial resection based upon three points. This result does not only hold for exact local affine frames but also for LAFs computed from triplets of close points, because their proximity can create numerical difficulties in the traditional resection formulation.

A further useful property of the LAF correspondence is that it is applicable to non-ideal perspective cameras, too, as has been shown e.g. in pose estimation with radial or fish-eye distortion. Furthermore, since a model for uncertainty handling has been proposed, the novel primitive can also be applied as an uncertain observation in maximum-likelihood estimation or

non-linear optimization like bundle adjustment and in frameworks for statistical testing, e.g. to detect outliers. However, the described applications are only exemplary demonstrations. The LAF correspondence technique might as well be used for self-calibration, estimating camera distortion, in articulated motion tracking, or to compute other multiple view geometry, which leaves a large area for future research.

In the final part of this thesis a system has been presented that extended the sparse 2D representation of local features to a 3D model of curved surfaces. Here, a complete framework for tracking a camera in a mainly static, textured scene has been proposed. The system is marker-less, drift-free and robust against lighting changes and occlusion and is therefore suitable to work on sites where no special markers can be set up and in infinitely long sequences without drift. It exploits building a model using structure-from-motion techniques to form a dense, textured three-dimensional model of the scene in which tracking is desired later on. In an online phase, initialization can be achieved using robust techniques on a feature database, where descriptors are projected to a small, discriminative subspace using methods from pattern recognition. Once the camera pose is known approximately, the focus changes to exact, jitter-free tracking. Herefore, it has been shown that using as much 3D surface information as possible improves the accuracy, and therefore tracking is performed using an analysis-by-synthesis approach on the GPU exploiting free-form surfaces, which perform better than locally planar patches because they are a better representation oft the real world.

To summarize, it can be argued that the LAF correspondences are well-suited when as much information has to be extracted from a few image correspondences as possible and to find initial solutions. They can also be used to stabilize and optimize the estimates even using maximum likelihood estimation. However, usually only a sparse set of affine features exist in an image and no occlusion or connection information in 3D space is given in this model. If on the other hand three-dimensional information about the scene is available, e.g. the shape of curved surfaces within the scene, exploiting this additional information is useful, particularly when the surfaces cannot be represented well by planes. Although the free-form surfaces are more complex to handle in equations, they are well-suited to be processed in analysis-by-synthesis approaches using the graphics hardware.

7.2 Future Work

A natural question arising from the proposed LAF correspondence formulation is whether it would be possible to use also the second order Taylor

approximation of a function. The major difficulties would be to obtain the second derivatives of the transformation from the image data (a larger region would be required) and to solve the equation system obtained from the second derivatives. Here, also an automatic model-selection approach would be interesting, which determines whether second derivatives are observable in the region and then decides whether the first order or a higher order model should be used.

Another interesting topic of future research is the modeling of asymmetric relations for local affine frames: in homography estimation, the one-to-one correspondence of affine features can now be expressed by the LAF correspondence constraint through the linearization of the warp, but a subject to future research must be how to extend this concept to one-to-many relations (analogous to point-to-line relations), which appear for instance in epipolar geometry estimation. For fundamental matrix estimation, clearly multiple features would be required. A promising approach could be to determine a manifold of homographies that is compatible with each LAF correspondence, and then to find an epipolar geometry that is consistent with at least one homography of each LAF correspondence. Another solution would be to find another primitive that mimics the epipolar line and an adapted representation of the fundamental matrix: a set of local affine frames in the first image, derived from (and consistent with) a given epipolar geometry and a local affine frame in the other image. This would allow for a full structure-from-motion system based solely upon local affine frames.

Appendix A

Analysis

A.1 Taylor Series

Continuous functions $\mathtt{f} : \mathbb{R} \to \mathbb{R}$ which are arbitrarily often differentiable in a neighborhood of a position a can be represented in a neighborhood of a as a sum of a power series, called the Taylor-series representation $\mathtt{T_f}$ (cf. to [Bronstein et al., 1999, p. 383]):

$$\mathtt{T_f^{(n)}}[x] = \mathtt{f}[a] + \sum_{i=1}^{n} \left(\frac{1}{i!} \left. \frac{\partial^i \mathtt{f}}{\partial x^i} \right|_a (x-a)^i \right) \tag{A.1}$$

with

$$\mathtt{R^{(n)}}[x-a] = \left\| \mathtt{f}[x] - \mathtt{T_f^{(n)}}[x] \right\| \leq \left\| \sup_{[x;a]} \left[\frac{\partial^{n+1} \mathtt{f}}{\partial x^{n+1}} \right] \frac{(x-a)^{n+1}}{(n+1)!} \right\| \tag{A.2}$$

being an upper bound of the error, when only n power series elements are considered. In a neighborhood of a (defined by the convergence radius) $\mathtt{R^{(n)}}$ vanishes as n goes to infinity. For finite n however, $\mathtt{T_f^{(n)}}$ is called the n^{th} order Taylor approximation with residual $\mathtt{R^{(n)}}$. Whenever the index (n) is left out, $n = 1$ shall be assumed.

If a function with 2-dimensional domain ($\mathtt{g} : \mathbb{R}^2 \to \mathbb{R}$) is inspected, the derivatives in both directions must be considered. For the sake of clarity this is written in operator style here (cf. to [Bronstein et al., 1999, p. 411]):

$$\mathtt{T_g^{(n)}}[\boldsymbol{x}] = \mathtt{g}[\boldsymbol{a}] + \sum_{i=1}^{n} \left(\frac{1}{i!} \left(\frac{\partial}{\partial x_1}(x_1 - a_1) + \frac{\partial}{\partial x_2}(x_2 - a_2) \right)^i \mathtt{g}[\boldsymbol{a}] \right) \tag{A.3}$$

Finally, if also the codomain has more than one dimension, i.e. if a function $\mathtt{h}$ maps to a d-dimensional codomain

$$\mathtt{h} : \mathbb{R}^2 \to \mathbb{R}^d : \quad \mathtt{h}\left[\boldsymbol{x}\right] = \begin{pmatrix} \mathtt{h_1}\left[\boldsymbol{x}\right] \\ \vdots \\ \mathtt{h_d}\left[\boldsymbol{x}\right] \end{pmatrix} \tag{A.4}$$

the above concept may be generalized to multiple codomain dimensions by representing each of the codomain dimensions $\mathtt{h_i}$ separately by its own series and then combining these series together.

$$\begin{aligned} \mathtt{T}_{\mathtt{h}}^{(\mathtt{n})}\left[\boldsymbol{x}\right] &= \begin{pmatrix} \mathtt{h_1}\left[\boldsymbol{a}\right] + \sum_{i=1}^{n} \left(\frac{1}{i!} \left(\frac{\partial}{\partial x_1}(x_1 - a_1) + \frac{\partial}{\partial x_2}(x_2 - a_2) \right)^i \mathtt{h_1}\left[\boldsymbol{a}\right] \right) \\ \vdots \\ \mathtt{h_d}\left[\boldsymbol{a}\right] + \sum_{i=1}^{n} \left(\frac{1}{i!} \left(\frac{\partial}{\partial x_1}(x_1 - a_1) + \frac{\partial}{\partial x_2}(x_2 - a_2) \right)^i \mathtt{h_d}\left[\boldsymbol{a}\right] \right) \end{pmatrix} \\ &= \mathtt{h}\left[\boldsymbol{a}\right] + \sum_{i=1}^{n} \frac{1}{i!} \begin{pmatrix} \left(\frac{\partial}{\partial x_1}(x_1 - a_1) + \frac{\partial}{\partial x_2}(x_2 - a_2) \right)^i \mathtt{h_1}\left[\boldsymbol{a}\right] \\ \vdots \\ \left(\frac{\partial}{\partial x_1}(x_1 - a_1) + \frac{\partial}{\partial x_2}(x_2 - a_2) \right)^i \mathtt{h_d}\left[\boldsymbol{a}\right] \end{pmatrix} \end{aligned} \tag{A.5}$$

Of particular interest for this thesis is the first order Taylor approximation (an affine function, also called the local linearization) of a function $\mathtt{h}$ mapping from $\mathbb{R}^2$ to $\mathbb{R}^2$:

$$\begin{aligned} \mathtt{T}_{\mathtt{h}}\left[\boldsymbol{x}\right] &= \mathtt{h}\left[\boldsymbol{a}\right] + \begin{pmatrix} \left(\frac{\partial}{\partial x_1}(x_1 - a_1) + \frac{\partial}{\partial x_2}(x_2 - a_2) \right) \mathtt{h_1}\left[\boldsymbol{a}\right] \\ \left(\frac{\partial}{\partial x_1}(x_1 - a_1) + \frac{\partial}{\partial x_2}(x_2 - a_2) \right) \mathtt{h_2}\left[\boldsymbol{a}\right] \end{pmatrix} \\ &= \mathtt{h}\left[\boldsymbol{a}\right] + \left. \frac{\partial \mathtt{h}}{\partial \boldsymbol{x}} \right|_{\boldsymbol{a}} \cdot (\boldsymbol{x} - \boldsymbol{a}) \end{aligned} \tag{A.6}$$

where the residuals $\mathtt{R_i}$ depend on the second derivatives:

$$\mathtt{R} = \begin{pmatrix} \mathtt{R_1} \\ \mathtt{R_2} \end{pmatrix} = \mathtt{h}\left[\boldsymbol{x}\right] - \mathtt{T}_{\mathtt{h}}\left[\boldsymbol{x}\right] \tag{A.7}$$

with

$$\begin{aligned} &\|\mathtt{R_i}\| \leq \\ &\left\| \left(\sup_{[\boldsymbol{a};\boldsymbol{x}]} \left[\frac{\partial^2 h_i}{\partial x_1^2} \right] \frac{(x_1 - a_1)^2}{2} + \sup_{[\boldsymbol{a};\boldsymbol{x}]} \left[\frac{\partial^2 h_i}{\partial x_1 \partial x_2} \right] (x_1 - a_1)(x_2 - a_2) + \right. \end{aligned}$$

$$\sup_{[\boldsymbol{a};\boldsymbol{x}]} \left[\frac{\partial^2 h_i}{\partial x_2^2} \right] \frac{(x_2 - a_2)^2}{2} \Bigg) \Bigg\| \tag{A.8}$$

Under the assumption that all the second derivatives have their suprema at $\boldsymbol{a}_s$, this can be written more efficiently using the Hessian matrix of $\mathtt{h_i}$

$$\frac{\partial^2 \mathtt{h_i}}{\partial \boldsymbol{x}^2} = \begin{pmatrix} \frac{\partial^2 \mathtt{h_i}}{\partial x_1 \partial x_1} & \frac{\partial^2 \mathtt{h_i}}{\partial x_1 \partial x_2} \\ \frac{\partial^2 \mathtt{h_i}}{\partial x_2 \partial x_1} & \frac{\partial^2 \mathtt{h_i}}{\partial x_2 \partial x_2} \end{pmatrix} \tag{A.9}$$

as

$$|\mathtt{R_i}| \approx \frac{1}{2} (\boldsymbol{x} - \boldsymbol{a})^\mathsf{T} \left. \frac{\partial^2 \mathtt{h_i}}{\partial \boldsymbol{x}^2} \right|_{\boldsymbol{a}_s} (\boldsymbol{x} - \boldsymbol{a}) \tag{A.10}$$

For functions with locally constant second derivative, $\boldsymbol{a}_s$ can be chosen arbitrarily between $\boldsymbol{a}$ and $\boldsymbol{x}$, for locally monotonic second derivatives $\boldsymbol{a}_s$ must be set either to $\boldsymbol{x}$ or $\boldsymbol{a}$, depending on which interval border has a larger absolute value of second derivative. For smooth functions and small intervals the selection of $\boldsymbol{a}_s$ has no big impact on the error bound.

A.2 Homographies in $\mathbb{P}^1$: Rational Functions in $\mathbb{R}^1$

A homography is a linear mapping in projective space. In this section its analogon in Euclidean space is inspected, which is a rational function. For simplicity of presentation, here only the 1D case is shown: let $\mathbf{H}$ be a homography mapping points from $\mathbb{P}^1$ to $\mathbb{P}^1$:

$$\mathbf{y} = \mathbf{H}\mathbf{x} \qquad \mathtt{det}\,[\mathbf{H}] \neq 0 \tag{A.11}$$

The additional condition on the determinant states that only regular homographies are considered here, which have full rank. Then, the problem can be transferred to Euclidean space (except for the ideal points), where it appears as:

$$\boldsymbol{y} = \mathtt{euc}\,[\mathbf{y}] = \mathtt{H}\,[\mathtt{euc}\,[\mathbf{x}]] = \mathtt{H}\,[\boldsymbol{x}] \tag{A.12}$$

When $\mathbf{H}$ consists of the two rows $\mathbf{h}_1^\mathsf{T}$ and $\mathbf{h}_2^\mathsf{T}$,

$$\mathbf{H} = \begin{pmatrix} \mathbf{h}_1^\mathsf{T} \\ \mathbf{h}_2^\mathsf{T} \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix} \tag{A.13}$$

$\mathtt{H}$ is actually a rational function of $\boldsymbol{x}$:

$$\boldsymbol{y} = \mathtt{euc}\,[\mathbf{y}] = \frac{\mathbf{h}_1^\mathsf{T}\mathbf{x}}{\mathbf{h}_2^\mathsf{T}\mathbf{x}} = \frac{h_{11}\boldsymbol{x} + h_{12}}{h_{21}\boldsymbol{x} + h_{22}} \tag{A.14}$$

This rational function is linear (more exact: affine) if the denominator does not depend on $\boldsymbol{x}$ and if it does not vanish, i.e. if $\mathtt{euc}\,[\mathbf{h}_2] = 0$. In projective space $\mathbb{P}^1$, such affine transformations do not change the point at infinity, i.e. they map infinity to infinity. If on the other hand $\mathtt{H}$ is truly projective, a finite point (the pre-image of the point at infinity or the *pole*) is mapped to infinity. The standard tools for characterizing a curve in calculus can be applied to the rational functions in Euclidean space. There can only be a pole $\boldsymbol{x}_p$ if the denominator vanishes and the enumerator does not[1]:

$$\mathbf{h}_2^\mathsf{T}\mathbf{x}_p = 0\,, \qquad \mathbf{h}_1^\mathsf{T}\mathbf{x}_p \neq 0 \tag{A.15}$$

and, if it exists, this pole is at

$$\boldsymbol{x}_p = -\frac{h_{22}}{h_{21}} \tag{A.16}$$

The rational function is further characterized by its limits at infinity

$$\lim_{\boldsymbol{x}\to\infty} \mathtt{H}\,[\boldsymbol{x}] = \lim_{\boldsymbol{x}\to-\infty} \mathtt{H}\,[\boldsymbol{x}] = \frac{h_{11}}{h_{21}} \tag{A.17}$$

and by its zero crossing, which exists if $h_{11} \neq 0$:

$$\mathtt{H}\left[-\frac{h_{12}}{h_{11}}\right] = 0 \tag{A.18}$$

Furthermore, $\mathtt{H}$'s derivative is given by

$$\frac{\partial \mathtt{H}}{\partial \boldsymbol{x}} = \frac{(h_{11}\mathbf{h}_2^\mathsf{T} - h_{21}\mathbf{h}_1^\mathsf{T})\mathbf{x}}{(\mathbf{h}_2^\mathsf{T}\mathbf{x})^2} = \frac{\mathtt{det}\,[\mathbf{H}]}{(\mathbf{h}_2^\mathsf{T}\mathbf{x})^2} \tag{A.19}$$

At a critical point, the first derivative must be zero, which can only happen when the enumerator vanishes. This however, may only happen if the determinant of $\mathbf{H}$ is zero, which would imply a degenerate homography, mapping everything to the same point. Consequently, a regular homography does not have any critical points. Therefore, a sketch of a 1D homography can be given as seen in figure A.1.

The homography's second derivative, which represents the local change of the first derivative, can be derived from equation (A.19):

$$\frac{\partial^2 \mathtt{H}}{\partial \boldsymbol{x}^2} = \frac{-2h_{21}\mathtt{det}\,[\mathbf{H}]}{(\mathbf{h}_2^\mathsf{T}\mathbf{x})^3} \tag{A.20}$$

[1] If denominator and enumerator both vanish the point is no longer in projective space. This can only happen for degenerate homographies, for which $\mathtt{det}\,[\mathbf{H}] = 0$ holds and which are not considered here.

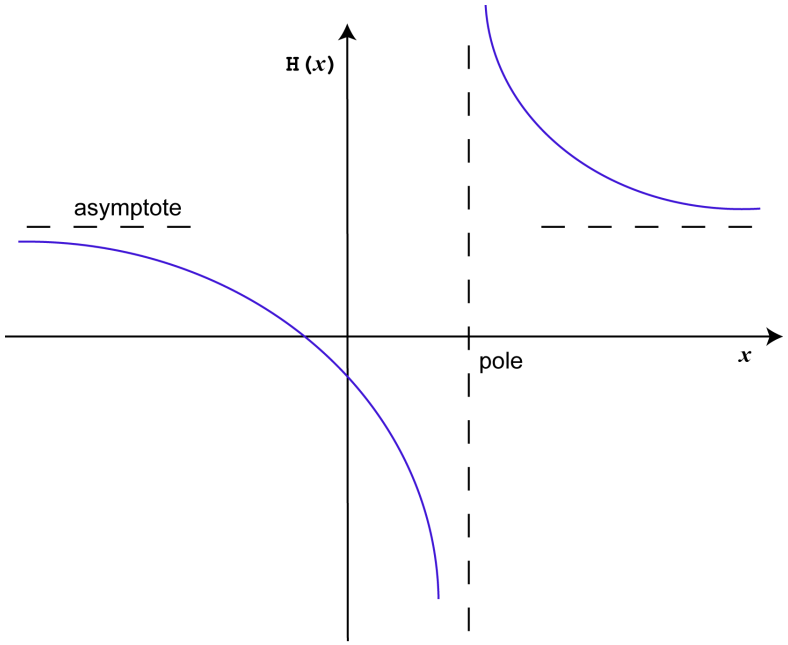

Figure A.1: Sketch of a 1D homography in euclidean space.

Using the second derivative, it is easy to see that the first derivative changes strongly near the pole but only gently far away. Therefore, the Taylor approximation using a fixed size environment represents the function better, if the linearization point is far from the pole (see figure A.2).

This becomes also clear when the error of the Taylor approximation is inspected: According to Taylor's theorem, the error is bounded by a value proportional to the absolute value of the supremum of the second derivative in the region (compare equation (A.10)). Assuming that $\boldsymbol{x}$ is in the convergence radius around the linearization point at a, the error at $\boldsymbol{x}$ is bounded by

$$|\mathtt{R}| \;\leq\; (a-\boldsymbol{x})^2 \,\mathtt{max}\left[\left\|\frac{h_{21}\mathtt{det}\,[\mathbf{H}]}{(h_{21}\boldsymbol{x}+h_{22})^3}\right\|, \left\|\frac{h_{21}\mathtt{det}\,[\mathbf{H}]}{(h_{21}a+h_{22})^3}\right\|\right] \tag{A.21}$$

This provides an estimate on the maximum difference between the affine transform using the linearization point a and the true homography.

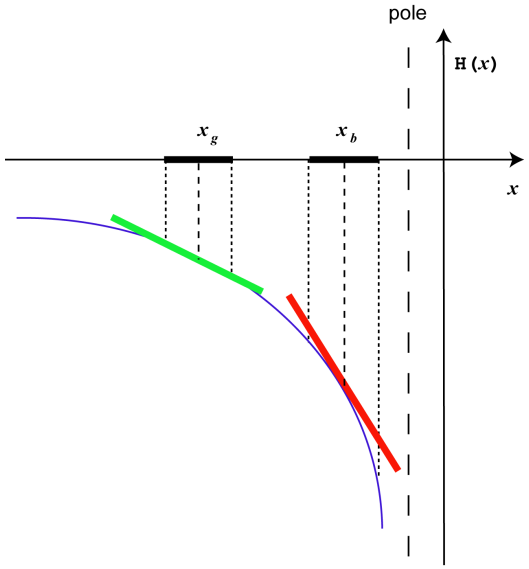

Figure A.2: Two local linearizations (first order Taylor approximation) of homography: The (green) approximation is good at $\boldsymbol{x}_g$, while the linearization is less good (red) at $\boldsymbol{x}_b$ because in the second case the derivative of the function changes more heavily in the interval.

Appendix B

Probability Theory

B.1 Basic Concepts

B.1.1 Cumulative Distribution and Density

The distribution of a continuous random variable X is characterized by its cumulative distribution function (cdf), which defines the probability that the random variable takes on a value less than some threshold x:

$$\texttt{cdf}\,[x] = P(X \leq x) \qquad x \in \mathbb{R} \tag{B.1}$$

In this thesis, only continuously differentiable cumulative distribution functions are considered. The derivative of this function is called the probability density function (pdf):

$$\texttt{pdf}\,[x] = \frac{\partial \texttt{cdf}}{\partial x} \tag{B.2}$$

Consequently, integrating the `pdf` over an interval yields the probability that the random variable takes on a value from that interval, so that

$$\int_{-\infty}^{\infty} \texttt{pdf}\,[x] = 1 \tag{B.3}$$

B.1.2 Moments

The moment representation for probability distributions can be thought of being analogous to the Taylor approximation of functions. The more higher moments exist and are incorporated into the approximation of a probability density function (pdf), the better the estimated pdf approximates the true pdf. In this thesis often unimodal distributions are used, i.e. pdfs which do not have multiple local maxima. Given only the first two moments of

such a pdf, the Gaussian distribution has the maximum entropy among all imaginable pdfs [Bishop, 2006, p.54]). In a sense this means that given the first two moments and no further information, assuming a Gaussian distribution is less biased than any arbitrarily chosen other distribution, because it provides the most surprise or uncertainty about measurements among all such distributions (maximum entropy principle [Jaynes, 1957]). This is the motivation to model measurements as normally-distributed if nothing better is known, which will be followed if not stated contrarily. Given multiple measurements on the other hand, it can be shown that the Gaussian distribution with maximum likelihood (cf. to [Bishop, 2006, p.23]) is the one with centered at the mean with standard deviation σ. These concept can also be applied to multiple dimensions, where a measurement corresponds to a pdf with a mean and a covariance matrix. In the remainder of this thesis this will also be called a measurement with uncertainty.

Probability distributions can be characterized by moments, e.g. the expectation value $\mathtt{E}$ for a random variable is the first moment of its distribution (cf. to [Bronstein et al., 1999, p.751]):

$$\mathtt{E}\left[X\right] = \int_{-\infty}^{\infty} \mathtt{pdf}\left[x\right] x dx = \mu \tag{B.4}$$

The higher (n^{th}) moments are computed as

$$\mathtt{E}\left[X^n\right] = \int_{-\infty}^{\infty} \mathtt{pdf}\left[x\right] x^n dx = \tag{B.5}$$

The higher moments are often more useful if they are computed with respect to the expectation value. This is then called the n^{th} central moment:

$$\mathtt{E}\left[(X-\mu)^n\right] = \int_{-\infty}^{\infty} \mathtt{pdf}\left[x\right] (x-\mu)^n dx \tag{B.6}$$

In the one-dimensional case the second central moment σ^2

$$\mathtt{E}\left[(X-\mu)^2\right] = \int_{-\infty}^{\infty} \mathtt{pdf}\left[x\right] (x-\mu)^2 dx = \sigma^2 \tag{B.7}$$

is also called the variance of X and its root σ is called the standard deviation.

In n dimensions, the second central moment is an $n \times n$ matrix called the covariance matrix Σ_{XX}

$$\Sigma_{XX} = \int_{-\infty}^{\infty} \mathtt{pdf}\left[\boldsymbol{x}\right] (\boldsymbol{x}-\boldsymbol{\mu})(\boldsymbol{x}-\boldsymbol{\mu})^{\mathsf{T}} d\boldsymbol{x} \tag{B.8}$$

B.1.3 Mahalanobis Distance

The L_2-norm in Euclidean vector space $\mathbb{R}^n$ measures distances between points $\boldsymbol{x}$ and $\boldsymbol{y}$ as

$$||\boldsymbol{x}-\boldsymbol{y}||_2 = \sqrt{\sum_{i=1..n}(x_i-y_i)^2} = \sqrt{(\boldsymbol{x}-\boldsymbol{y})^\mathsf{T}(\boldsymbol{x}-\boldsymbol{y})} \tag{B.9}$$

Let this space be transformed by a linear transformation L, such that

$$\hat{\boldsymbol{x}} = L\boldsymbol{x} \tag{B.10}$$

and

$$\hat{\boldsymbol{y}} = L\boldsymbol{y} \tag{B.11}$$

The distance of the original points expressed in the new space is therefore

$$||\boldsymbol{x}-\boldsymbol{y}||_2 = \sqrt{(\boldsymbol{x}-\boldsymbol{y})^\mathsf{T}(\boldsymbol{x}-\boldsymbol{y})} = \sqrt{(\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}})^\mathsf{T} L^{-\mathsf{T}}L^{-1}(\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}})} \tag{B.12}$$

If the transformation L is replaced using a matrix Σ, so that

$$\Sigma^{-1} = L^{-\mathsf{T}}L^{-1} \tag{B.13}$$

then the L_2-norm in the original space can be computed in the transformed space as

$$||\boldsymbol{x}-\boldsymbol{y}||_2 = \sqrt{(\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}})^\mathsf{T}\Sigma^{-1}(\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}})} = ||\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}||_\Sigma \tag{B.14}$$

In the transformed space, this is called the Mahalanobis distance of $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$. As it can be seen, the Euclidean distance is a special case of the Mahalanobis distance, where $\Sigma_{XX} = I_{n\times n}$ The Mahalanobis distance is important in statistical applications, where it takes into account correlations between the data, or to compute distances in (anisotropically) scaled coordinate systems.

B.2 Distributions

B.2.1 Normal Distribution

A very important probability distribution is the normal distribution, also called Gaussian distribution. One important property of the normal distribution is that given the first two moments the normal distribution has the maximum entropy of all probability distributions [Jaynes, 1957]. Consequently, given no further information than the first two moments the normal

distribution can be thought of the least biased assumption for the distribution. It is defined through its probability density function

$$\mathtt{pdf}_{\mathtt{G}}\left[\boldsymbol{x}\right] = \left((2\pi)^{\mathtt{dim}[\boldsymbol{x}]}\mathtt{det}\left[\Sigma_{\boldsymbol{xx}}\right]\right)^{-\frac{1}{2}} \mathtt{exp}\left[-\frac{1}{2}\left(\boldsymbol{x}-\boldsymbol{\mu}\right)^{\mathsf{T}}\Sigma_{\boldsymbol{xx}}^{-1}\left(\boldsymbol{x}-\boldsymbol{\mu}\right)\right] \quad \text{(B.15)}$$

where the (multidimensional) mean $\boldsymbol{\mu}$ (the first moment) and the covariance matrix $\Sigma_{\boldsymbol{xx}}$ (the second central moment) determine the position and shape of the distribution. In 2D this is

$$\mathtt{pdf}_{\mathtt{G,2D}}\left[\boldsymbol{x}\right] = \left(4\pi^2\mathtt{det}\left[\Sigma_{\boldsymbol{xx}}\right]\right)^{-\frac{1}{2}} \mathtt{exp}\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\mathsf{T}}\Sigma_{\boldsymbol{xx}}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right] \quad \text{(B.16)}$$

and in 1D it simplifies to

$$\mathtt{pdf}_{\mathtt{G,1D}}\left[\boldsymbol{x}\right] = \frac{1}{\sqrt{2\pi}\sigma}\mathtt{exp}\left[\frac{(x-\mu)^2}{-2\sigma^2}\right] \quad \text{(B.17)}$$

The interval $[\mu-\sigma;\mu+\sigma]$ is called the standard confidence region. In multiple dimensions, the corresponding region defined by the points $\boldsymbol{x}$ that fulfill

$$(\boldsymbol{x}-\boldsymbol{\mu})^{\mathsf{T}}\Sigma_{\boldsymbol{xx}}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}) \leq 1 \quad \text{(B.18)}$$

forms a hyperellipsoid, a special kind of quadric. The probability that a measurement falls inside such a hyper-ellipsoid can be computed by integrating the density across the region. For 1D it is approximately 68% but for higher dimensions this fraction quickly gets much smaller.

Difference/Sum of Two Uncorrelated Gaussians

The sum or the difference of two normally distributed random variables that are uncorrelated is also normally distributed. Let $\boldsymbol{x}_1$ be normally distributed with mean μ_1 and covariance Σ_1 and let $\boldsymbol{x}_2$ be normally distributed with mean μ_2 and covariance Σ_2. Then a joint representation for both vectors can be made in a double size vector

$$\boldsymbol{x}_{12} = \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{pmatrix} \quad \text{(B.19)}$$

which is also normally distributed, with covariance

$$\Sigma_{12} = \begin{pmatrix} \Sigma_1 & \\ & \Sigma_2 \end{pmatrix} \quad \text{(B.20)}$$

Then the sum of both

$$\boldsymbol{x} = \boldsymbol{x}_1 + \boldsymbol{x}_2 \quad \text{(B.21)}$$

can also be written as a linear operation

$$\boldsymbol{x} = S\boldsymbol{x}_{12} \tag{B.22}$$

where

$$S = \begin{pmatrix} 1 & & & 1 & & \\ & \ddots & & & \ddots & \\ & & 1 & & & 1 \end{pmatrix} \tag{B.23}$$

For linear functions, Gaussian error propagation applies and therefore

$$\Sigma_{\boldsymbol{x}\boldsymbol{x}} = S\Sigma_{12}S^{\mathsf{T}} \tag{B.24}$$

which is simply the componentwise sum of the original covariances. The same is true for the difference of two normally distributed uncorrelated variables.

B.2.2 χ^2 distribution

The χ^2-distribution is a very important probability distribution for statistical testing. It takes the form

$$\mathtt{pdf}_{\chi^2,\mathtt{n}}[x] = \frac{x^(n/2-1)e^{-y/2}}{2^{n/2}\Gamma\left(\frac{n}{2}\right)} \tag{B.25}$$

where Γ represents the Gamma function and n is a parameter for the number of degrees of freedom (cf. to [McGlone, 2004, p.64]). The sum of n independent, squared random variables X_i that are normally distributed with zero mean and variance 1 is χ^2-distributed.

$$\mathtt{pdf}_{\mathtt{X}^2}\left[\sum_i X_i^2\right] = \mathtt{pdf}_{\chi^2,\mathtt{n}}[x] \tag{B.26}$$

B.3 Statistical Testing

In several applications the question arises, whether some measured data support an assumption, the so-called null hypothesis, or whether the null hypothesis is unlikely given the data ([McGlone, 2004, p.77]).

B.3.1 Incidence Test

As a consequence from equation (B.26) it follows that the squared Mahalanobis distance (equation (B.14)) of a sample from a zero-mean Gaussian to

the origin is also χ^2-distributed. Under the assumption that a Gaussian with given covariance Σ and zero mean is given, the probability $P_{t,\Sigma}$ of obtaining a sample with a squared Mahalanobis distance larger than t can be computed as

$$P_{t,\Sigma} = \int_t^\infty \mathtt{pdf}_{\chi^2,\mathtt{n}}[x]dx \tag{B.27}$$

If, for some t_0, this probability $P_{t_0,\Sigma}$ is very small, i.e. smaller than some predefined significance level P_{sig}, then it is unlikely to draw a sample this far from the mean of the given distribution. In this case there is statistical evidence that the base hypothesis can be rejected, i.e. here that the sample is drawn from a zero mean Gaussian with covariance Σ. The smaller the threshold P_{sig} is set, the more significant the test becomes, i.e. the more likely it becomes that a rejected hypothesis was actually wrong. However, this happens at the cost of reduced sensitivity and usually a tradeoff has to be found. The testing scheme can for instance be exploited to detect outliers in sets of feature correspondences, given a geometrical model for verification: Choosing a large value for P_{sig} will detect most outliers at the risk of also misclassifying some good correspondences as outliers. Choosing a small value on the other hand will allow most of the inliers to pass the test at the risk of overlooking some of the outliers.

B.4 Uncertainty Propagation

When dealing with probability distributions for uncertain observations and parameters, often the first two moments are used to characterize the distribution of the related random variables in practice. Sometimes these variables are transformed by a function and it is necessary to reason about the moments of the transformed distribution. Here, particularly the shape and the size of the covariance matrix is interesting. This is the topic of uncertainty propagation or error propagaton.

B.4.1 Linear Error Propagation

Given a probability distribution for a random variable X, which has mean $\mu_{\boldsymbol{x}}$ and covariance $\Sigma_{\boldsymbol{xx}}$, an affine transformation of the variables

$$\hat{\boldsymbol{x}} = L\boldsymbol{x} + \boldsymbol{b} \tag{B.28}$$

will (cf. to [McGlone, 2004, p. 71]) result in a $\hat{X}$ to be distributed so that

$$\mu_{\hat{\boldsymbol{x}}} = L\mu_{\boldsymbol{x}} + \boldsymbol{b} \tag{B.29}$$

and

$$\Sigma_{\hat{\boldsymbol{x}}\hat{\boldsymbol{x}}} = L\Sigma_{\boldsymbol{x}\boldsymbol{x}}L^{\mathsf{T}} \tag{B.30}$$

This is called linear error propagation. In case X is transformed by a non-linear function $\mathtt{T}$ that is locally analytic and sufficiently smooth, then the first order Taylor approximation of $\mathtt{T}$ can be used to locally linearize the non-linear function. This local linearization can then be exploited to obtain an approximation of the second moment in the transformed space:

$$\Sigma_{\hat{\boldsymbol{x}}\hat{\boldsymbol{x}}} \approx \frac{\partial \mathtt{T}}{\partial \boldsymbol{x}} \Sigma^{\mathsf{T}}_{\boldsymbol{x}\boldsymbol{x}} \frac{\partial \mathtt{T}}{\partial \boldsymbol{x}}^{\mathsf{T}} \tag{B.31}$$

The quality of the approximation depends on the local linearity of the non-linear function.

B.4.2 Monte Carlo Methods

Another way of propagating uncertainty is using so-called Monte-Carlo methods [Metropolis and Ulam, 1949]. Here, knowledge about the underlying distribution is required. This distribution is then numerically sampled at a large number l of positions $\boldsymbol{x}_i$: these samples are then transferred using some function $\mathtt{f}$ and the moments $\hat{\mu}$ and $\Sigma_{\hat{\boldsymbol{x}}\hat{\boldsymbol{x}}}$ of the transferred samples are computed:

$$\hat{\mu} = \frac{1}{l} \sum_i \mathtt{f}\left[\boldsymbol{x}_i\right] \tag{B.32}$$

$$\Sigma_{\hat{\boldsymbol{x}}\hat{\boldsymbol{x}}} = \frac{1}{l-1} \sum_i \left(\left(\mathtt{f}\left[\boldsymbol{x}_i\right] - \hat{\mu}\right) \left(\mathtt{f}\left[\boldsymbol{x}_i\right] - \hat{\mu}\right)^{\mathsf{T}} \right) \tag{B.33}$$

The more samples are used the better the approximation of the transformed moments can become. However, using more samples also leads to an increased computational burden. Particularly when high-dimensional pdfs are considered Monte-Carlo methods quickly become slow or even computationally infeasible.

B.4.3 Unscented Transform

A more efficient approximation of uncertainty propagation has been proposed by Julier and Uhlmann [1997, 2002], the unscented transform. Here, the original distribution is assumed to be normally-distributed and is sampled at a few, well-defined positions, namely where the principal axes of the normal distribution have a certain Mahalanobis distance λ_u to the center, and at the center. These samples are then transferred using the nonlinear function, and

based on the transferred samples the transferred covariance is estimated in the same way as in the Monte-Carlo method. If the transforming function is affine, this yields the same results as linear error propagation. Otherwise, if the parameter λ_u is chosen very small, the system behaves more like linear error propagation, or if the parameter is chosen larger, the moments are transferred based on a larger surrounding of the mean. In any case, since a fixed number of samples is used, it is computationally very efficient.

Appendix C

Robust Estimation

C.1 Robust Estimation

In large parts of this thesis, observations in one or two images will be exploited to reason about the scene or the camera. Often, such observations are not perfect but contain certain sources of error or distortion.

C.1.1 Observations and Uncertainty

When observing the real world, measurements can only be obtained with a finite precision due to physical limits in sensors, digitization and transmission, and representation as well as other sources of noise. In this case the measured value is actually not the true value. However, for stable systems, small disturbances of the assumed conditions will only result in small changes of the outcome. Therefore, if the measurement method is designed well, it is possible to reason about the real world based upon measured values. Often, not only one measurement is given, but the experiment is repeated multiple times or contains redundancy. In this case, the mean and the scatter of the measurement can serve as the first two moments (cf. to [McGlone, 2004, p. 65]) of a probability distribution for the true value. Therefore, for measurements the methods of probability theory presented in appendix B can be applied.

C.1.2 Least Squares and Covariance Estimation

In this section it is assumed that there is a linear function $\boldsymbol{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, which depends on some parameters $\boldsymbol{p} \in \mathbb{R}^n$ to compute an observation $\hat{\boldsymbol{o}} \in \mathbb{R}^m, (n > m)$ according to:

$$\hat{\boldsymbol{o}} = \boldsymbol{f}(\boldsymbol{p}) \tag{C.1}$$

If $\boldsymbol{o}$ is now a normally distributed random variable with mean $\hat{\boldsymbol{o}}$ and covariance Σ_{oo}, given some measurement $\tilde{\boldsymbol{o}}$ from this distribution, the maximum likelihood estimate for $\boldsymbol{p}$ is (cf. to [McGlone, 2004, p. 84]):

$$\hat{\boldsymbol{p}} = \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}}^{\mathsf{T}} \Sigma_{\boldsymbol{oo}}^{-1} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}} \right)^{-1} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}}^{\mathsf{T}} \Sigma_{\boldsymbol{oo}}^{-1} \tilde{\boldsymbol{o}} \tag{C.2}$$

The uncertainty of the solution can be obtained through error propagation (see section B.4) as described in the following.

C.1.3 Covariance Estimation

Equation C.2 showed how to obtain an estimate for a parameter based on uncertain measurement data. Also an estimate for the uncertainty of the parameters can be obtained. First the residual vector $\boldsymbol{r}$ is defined as

$$\boldsymbol{r} = \boldsymbol{f}(\hat{\boldsymbol{p}}) - \tilde{\boldsymbol{o}} \tag{C.3}$$

and the reference variance σ_0^2 is computed as

$$\sigma_0^2 = \frac{\boldsymbol{r}^{\mathsf{T}} \Sigma_{\boldsymbol{oo}} \boldsymbol{r}}{n - m} \tag{C.4}$$

where n is the number of observations and m is the number of the degrees of freedom, i.e. $(n - m)$ is a measure of redundancy ([McGlone, 2004, 85]). The estimated covariance for the estimated parameters is then

$$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{pp}} = \sigma_0^2 \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}}^{\mathsf{T}} \Sigma_{\boldsymbol{oo}}^{-1} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}} \right)^{-1} \tag{C.5}$$

C.1.4 Newton-like methods

In the previous section only linear functions $\boldsymbol{f}$ were considered. This section now extends the method to non-linear functions that are represented well by a low order Taylor approximation. In case a parameter prediction $\tilde{\boldsymbol{p}}$ is given, $\boldsymbol{f}$ can then locally be approximated at $\tilde{\boldsymbol{p}}$ and a parameter update can be computed yielding the next prediction. This scheme can be run in an iterative fashion minimizing a positive error function $\mathtt{err}\,[\boldsymbol{p}]$. A typical convex error function is the reference variance (C.4), because of its quadratic dependence on the residuals sometimes also called a quadratic error function. Methods minimizing a function in this way are called Newton-like functions in the following:

$$\hat{p} = \mathbf{argmin}_p \mathtt{err}\,[\boldsymbol{p}], \qquad \mathtt{err}\,[\boldsymbol{p}] = \sigma_0^2 \tag{C.6}$$

Direction of Steepest Descent

Probably the simplest method to locally optimize `err` is a method called gradient descent. Here, a parameter update onto the prediction is simply obtained by using the Jacobian of the function at the predicted parameters:

$$\Delta \boldsymbol{p} = \frac{1}{\lambda} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}} \tag{C.7}$$

This means that a step is taken in direction of steepest descent of the function. However, the step size parameter λ must be provided appropriately. If a too small step is chosen, convergence is very slow. If a step is taken larger than the region of approximate local linearity, the method may not converge to the local minimum. Often, several values are tried for λ and a step is only taken if the error function improves. The advantage of steepest descent is that singular Jacobians of the error function are supported.

Newton's method and Gauss-Newton

An optimization method avoiding specification of a step size parameter is the Gauss-Newton algorithm. Here, the function $\boldsymbol{f}$ is replaced by its first order Taylor approximation. In this case the error function $\sigma_0^2(\boldsymbol{p})$ is quadratic in $\boldsymbol{p}$ and the minimum can directly be obtained in analogy to the linear case of equation (C.2):

$$\Delta \boldsymbol{p} = \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}}^{\mathsf{T}} \Sigma_{\boldsymbol{oo}}^{-1} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}} \right)^{-1} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}}^{\mathsf{T}} \Sigma_{\boldsymbol{oo}}^{-1} \tilde{\boldsymbol{r}} \tag{C.8}$$

Sometimes the Taylor approximation is only valid very locally and a large step of the algorithm may leave the local minimum and lead to divergence. The method can therefore also be combined with a line search technique, which tries different steps $\frac{1}{\lambda}\Delta \boldsymbol{p}$ and takes a step only if the error function improves. As the inverse Jacobian is computed Gauss-Newton requires the Jacobian to be non-singular.

Levenberg-Marquardt

An automatic way of handling the λ of the two previous sections is used in the Levenberg-Marquardt algorithm (cf. to [Press et al., 1992, pp. 681]). The Gauss-Newton step of the previous section exploits a second order Taylor approximation of the error function to directly jump to the minimum. The first order Taylor approximation of the error function on the other hand carries information about the direction in which the error function decreases if an infinitesimally small step is taken. The Levenberg-Marquardt method

now blends between Gauss-Newton and Steepest Descent by monitoring a weight. Initially a Gauss-Newton step is computed. If the error function increases the step size is reduced and a new step is proposed more towards the direction of steepest descent. This is repeated until the error function decreases. If a step is accepted, then the next step is started with an increased step size. Different strategies exist for the actual implementation and step handling [Hartley and Zisserman, 2004], but in this thesis the following step computation rule is used

$$\Delta \boldsymbol{p} = \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}}^{\mathsf{T}} \Sigma_{\boldsymbol{oo}}^{-1} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}} + \lambda I_{d \times d} \right)^{-1} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}}^{\mathsf{T}} \Sigma_{\boldsymbol{oo}}^{-1} \tilde{\boldsymbol{r}} \tag{C.9}$$

where d is the dimension of $\boldsymbol{p}$. This way Levenberg-Marquardt has the advantage of fast convergence where the Gauss-Newton assumptions are fulfilled but slows down in regions with higher non-linearity. As an additional benefit it works even if the Jacobian is singular.

C.1.5 Gross Errors and Breakdown Point

The previous sections dealt with different methods to estimate parameters from noisy measurements. The basic assumption of these techniques was however, that the pdfs of the observations are close to Gaussian with known covariance. This solves the problem of measurement uncertainty and inaccuracy. In real data however, there are often measurements which do not conform to this type of distribution because their deviation from the ideal model is not due to small inaccuracies but results from other sources such as relating totally wrong entities (mis-match). Such measurements are called outliers, because it is highly unlikely that they stem from a Gaussian distribution. For standard least-squares methods it is obvious that one can always disturb a single observation so that the solution will be arbitrarily far from the true solution. The maximum fraction of data for which this can not happen is called the breakdown-point of an estimator (cf. to [Hampel, 1971]). Consequently, least-squares has a break-down point of 0%. Accordingly, the Newton-like methods which do not monitor improvement of the error function also have a breakdown-point of 0%. In presence of a single (arbitrarily bad) outlier those methods that do monitor will in practice not find any good step, thus they can also be considered to have a breakdown-point of 0%.

C.1.6 Robust Error Functions

The reason, why least-squares is so susceptible to outliers is its quadratic error function. From equation (C.2) it can be seen that the residuals have a

quadratic weight in the computation of the step and therefore large residuals dominate small ones. This observation led to the introduction of robust error functions [Hampel et al., 2005]. The idea is that above a certain *outlier threshold* σ_i the influence of a residual does no longer increase quadratically. A representatives of such a function is for example the Huber function [Huber, 1964]. In this thesis the Huber error function continues differentiably and linearly at 3σ:

$$\mathtt{err}_{\text{huber}}[\boldsymbol{r}] = \Sigma_{i<n} \begin{cases} \frac{6}{\sigma}|r_i| - 9 & \text{if } r_i^2/\sigma_{ii}^2 > 3^2 \\ r_i^2/\sigma_{ii}^2 & \text{otherwise} \end{cases} \tag{C.10}$$

While such functions are more robust in the presence of outliers, they require an initial prediction close to the optimum. Such initial values have to be obtained by previous knowledge or a direct algorithm, like typical least squares or linear solvers. If only a single outlier is in the data there are simple testing strategies to find it, for up to a few outliers reweighted least squares techniques exist [Scales and Gersztenkorn, 1988], where the inverse residual is used as a weight for the next iteration, but these strategies also have a breakdown point close to zero. For higher outlier fractions also the least median squares concept exists [Rousseeuw, 1984], which expects more than half of the data to be inliers and therefore has a breakdown point of 50%. Probably the most successfull robust methods in automated correspondence-based estimation is the RANSAC family, which will be described in the next section.

C.1.7 RANSAC-like methods

When a significant fraction of outliers is present within the data and no a priori knowledge about the parameters is available, one approach is to generate lots of solution hypotheses from small subsets of the data using unrobust estimators and classify the data into inliers and outliers according to each hypothesis. If one of the small subsets was outlier-free, the obtained solution should provide a consensus for all inliers. This approach called RANSAC was published in 1981 by Fishler and Bolles [Fischler and Bolles, 1981].

Formally, it is based on the assumption that the measurements can be divided into a fraction p of inliers and $1-p$ of outliers and that this fraction is known approximately beforehand. Then, a minimal subset from the data is exploited to construct a solution hypothesis. Minimal means, that all degrees of freedom of the model can be determined but that there should be no redundancy in the estimate in order to limit the number m of measurements

the hypothesis is based upon. Then the probability of picking a set containing only inliers is:

$$P(\text{ALL-INLIER-SET}) = p^m \tag{C.11}$$

As can be seen the probability that one set is an all-inlier-set decreases exponentially with m and therefore m should be chosen as small as possible. If now s times a minimal set is sampled from the data, the probability of obtaining at least one all-inlier-set increases with s:

$$P(\text{ALL-INLIER-SET-MULTIPLE-TRIALS}) = 1 - (1 - (p^m))^s \tag{C.12}$$

This can be solved for s to obtain the number of trials required so that an all-inlier-set is picked with at least 99%:

$$s = \frac{-2}{\log_{10}\left[1 - p^m\right]} \tag{C.13}$$

For each of the sampled sets now a hypothesis is generated, and all measurements are classified as inliers or outliers according to this hypothesis. Based upon the inliers of a hypothesis, this hypothesis may be refined to obtain an improved hypothesis. The hypothesis with the highest number of inliers is then the estimated solution. Furthermore, the partitioning into inliers and outliers can be exploited for subsequent Newton-like refinement. If the outliers are statistically independent, so that they do not agree on a "wrong" parameter set, RANSAC has a breakdown point of even more than 50%, although the average run-time increases drastically with the fraction of outliers.

Since the initial RANSAC-publication several improvements, such as the Bayesian-model motivated MLESAC [Torr and Zisserman, 2000] or the QDEGSAC algorithm [Frahm and Pollefeys, 2006] for quasi-degenerate data have been proposed, however, the basic idea is the same. Whenever RANSAC is used in the following, a RANSAC-like algorithm, which uses the same principles is meant: RANSAC is a parameter estimation technique, which finds a set of parameters which is optimal according to an error function or on which the most measurements agree. In that sense it is related to Hough transform(cf. to [Jähne, 2005, p. 482]) or mode estimation for probability density functions [Bishop, 2006]. The idea is very simple: Given a set of data that includes gross errors, synthesize a hypothesis from as few samples as possible (a minimal set) and score the hypothesis. Given the approximate fraction of gross errors (e.g. 20%) the probability to select a minimal set without gross errors can be computed. Now repeat generating hypotheses and scoring them until an all-inlier set has been chosen with a high probability (e.g. 99%) or the hypothesis yields a sufficient score. After termination return the parameter set with the highest score and classify all measurements into inliers and outliers.

Appendix D

Source Code

D.1 Conjugate Rotation Parameterization

In section 5.2.2 a parameterization has been proposed for the conjugate rotation using the parameter vector

$$\boldsymbol{p} = (a_{11}, a_{12}, a_{21}, a_{22}, d_1, d_2, h_{32})^{\mathsf{T}} \tag{D.1}$$

In this section it is now shown that for a specific choice of parameters $\boldsymbol{p}_z$

$$\boldsymbol{p}_z = (0, 1, -1, 0, 1, 1, 0)^{\mathsf{T}} \tag{D.2}$$

it is possible to run into seven orthogonal direction on the manifold of conjugate rotations. To show this, the source code of figure D.1 is used in a symbolic linear algebra package [Matlab, 2008]:

As can be seen, when the code is executed the chosen set of parameters leads to a valid conjugate rotation and results in

$$\left.\frac{\partial \mathtt{vec}\,[\mathbf{H}]}{\partial \boldsymbol{p}}\right|_{\boldsymbol{p}_z} = \begin{pmatrix} 1 & 1/3 & -1/3 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 2/3 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & -1/3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{D.3}$$

This matrix has rank 7, as can easily be checked.

$$\mathtt{rank}\left[\left.\frac{\partial \mathtt{vec}\,[\mathbf{H}]}{\partial \boldsymbol{p}}\right|_{\boldsymbol{p}_z}\right] = 7 \tag{D.4}$$

```
syms a11 a12 a21 a22 real;
syms dx dy real;
syms h32 real;

% parameters from LAF correspondence
A = [a11 a12; a21 a22];
d = [dx;dy];
lambda = det(A)^(1/3);

% solve for CR constraint
m = (lambda - trace(A))*d' + d'*A';
y = lambda*(trace(A) + 1) - (trace(A)^2)/2 - trace(A) + trace(A^2)/2;
a = -m(2)/m(1);
b = - y/m(1);

% get h31
h = [a*h32+b, h32];

% construct conjugate rotation
H = [eye(2), d; 0,0,1] * [A,zeros(2,1); h,1]

% differentiate each matrix element with respect to the seven parameters
J = [diff(H(:), a11), diff(H(:), a12), diff(H(:), a21), diff(H(:), a22), ...
     diff(H(:), dx), diff(H(:), dy), diff(H(:), h32)];

% select a simple conjugate rotation with detA=1
a11 =  0;
a12 =  1;
a21 = -1;
a22 =  0;
dx  =  1;
dy  =  1;
h32 =  0;

% show how the CR looks like with these parameters
actualH = subs(H)

% just some sanity checks: conjugate rotation ?
% lhs of 5.37:
subs(((lambda-trace(A))*d'+d'*A')*h')
% rhs of 5.37:
subs(-0.5*trace(A)*trace(A) - trace(A) + 0.5*trace(A*A)+lambda*(trace(A)+1))
% show eigenvalues:
Heigenvalues = eig(actualH)
% compare eigenvalues of a specific rotation matrix (90 degree z rotation)
Reigenvalues = eig([0 1 0; -1 0 0; 0 0 1])

% show how the matrix entries depend on the 7 parameters
actualJ = subs(J)
% and print its rank
rank(actualJ)
```

Figure D.1: Matlab Source Code for Analysis of the Conjugate Rotation

Bibliography

E. H. Adelson and J. Bergen. The plenoptic function and the elements of early vision. In M. Landy and J. A. Movshon, editors, *Computation models of visual processing*, pages 3–20. MIT Press, 1991.

K. Astrom, F. Kahl, A. Heyden, and R. Berthilsson. A statistical approach to structure and motion from image features. In *International Workshops on Advances in Pattern Recognition*, pages 929–936, London, UK, 1998. Springer-Verlag. ISBN 3-540-64858-5.

S. Baker and I. Matthews. Lucas-Kanade 20 Years On: A Unifying Framework. *International Journal of Computer Vision*, 56(3):221–255, 2004. ISSN 0920-5691.

S. Baker, R. Gross, and I. Matthews. Lucas-kanade 20 years on: A unifying framework: Part 3. Technical Report CMU-RI-TR-03-35, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, November 2003.

B. Bartczak, K. Köser, F. Woelk, and R. Koch. Extraction of 3d freeform surfaces as visual landmarks for real-time tracking. *Journal of Real Time Image Processing*, 2:81–101, 2007.

A. Bartoli and P. Sturm. Structure from motion using lines: Representation, triangulation and bundle adjustment. *Computer Vision and Image Understanding*, 100(3):416–441, dec 2005.

R. Basri and D. Jacobs. Projective alignment with regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(5):519–527, 2001.

A. Baumberg. Reliable feature matching across widely separated views. In *Proceedings of CVPR*, pages 774–781, 2000.

H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110:346–359, 2008.

C. Beder and R. Steffen. Determining an initial image pair for fixing the scale of a 3d reconstruction from an image sequence. In K. Franke, K.-R. Müller, B. Nickolay, and R. Schäfer, editors, *Pattern Recognition*, number 4174 in LNCS, pages 657–666. Springer, 2006.

C. Beder, B. Bartczak, and R. Koch. A combined approach for estimating patchlets from PMD depth images and stereo intensity images. In F.A. Hamprecht, C. Schnörr, and B. Jähne, editors, *Proceedings of the DAGM 2007*, number 4713 in LNCS, pages 11–20. Springer, 2007a.

C. Beder, B. Bartczak, and R. Koch. A comparison of pmd-cameras and stereo-vision for the task of surface reconstruction using patchlets. In *IEEE/ISPRS Workshop BenCOS 2007*, 2007b.

J.S. Beis and D.G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensionalspaces. In *Proceedings of the Conference on Computer Vision and Pattern Recognition(CVPR '97)*, page 1000ff., Washington, DC, USA, 1997. IEEE Computer Society.

P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

J.R. Bergen, P. Anandan, K.j. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *Proceedings of ECCV 1992*, pages 237–252, 1992.

P. Biber, S. Fleck, and W. Strasser. The wägele: A mobile platform for acquisition of 3d models of indoor outdoor environments. In *9th Tübingen Perception Conference (TWK 2006), Tübingen*, 2006.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

G. Bleser, H. Wuest, and Stricker. Online camera pose estimation in partially known and dynamic scenes. In *Proceedings ISMAR 2006, Los Alamitos, California, pp. 56-65*, 2006.

G. Bleser, M. Becker, and D. Stricker. Real-time vision-based tracking and reconstruction. *Journal of Real-time Image Processing*, 2:161–175, 2007.

I. N. Bronstein, K. A. Semendjajew, G. Musiol, and H. Mühlig. *Taschenbuch der Mathematik*. Verlag Harri Deutsch, 4 edition, 1999.

M. Brown and D. G. Lowe. Automatic panoramic image stitching using invariant features. *International Journal of Computer Vision*, 74(1):59–73, 2007.

M. Brown, R. Hartley, and D. Nistér. Minimal solutions for panoramic stitching. In *Proceedings of CVPR 2007*, 2007.

J. F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 184–203, 1987.

D. Capel and A. Zisserman. Automatic mosaicing with super-resolution zoom. In *CVPR '98: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 885, Washington, DC, USA, 1998. IEEE Computer Society. ISBN 0-8186-8497-6.

E. De Castro and C. Morandi. Registration of translated and rotated images using finite fourier transforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):700 – 703, 1987.

J. Chandaria, G. Thomas, B. Bartczak, K. Köser, R. Koch, M. Becker, G. Bleser, D. Stricker, C. Wohlleber, M. Felsberg, J. Hol, T. Schoen, J. Skoglund, P. Slycke, and S. Smeitz. Real-time camera tracking in the matris project. In *Proceedings of International Broadcasting Convention (IBC)*, pages 321–328, Amsterdam,The Netherlands, 2006.

J. Chandaria, G. A. Thomas, and D. Stricker. The matris project: real-time markerless camera tracking for augmented reality and broadcast applications. *Journal of Real-time Image Processing*, 2, 2007.

B. Chen, F. Dachille, and A. E. Kaufman. Footprint area sampled texturing. *IEEE Transactions on Visualization and Computer Graphics*, 10(2):230–240, 2004. ISSN 1077-2626. doi: http://doi.ieeecomputersociety.org/10.1109/TVCG.2004.1260775.

W.J. Christmas. Spatial filtering requirements for gradient-based optical flow measurement. In *Proceedings of the British Machine Vision Conference*, Southampton, UK, 1997.

O. Chum and J. Matas. Web-Scale Image Clustering. Technical Report 2008-15, Center for Machine Perception, Czech Technical University, Prague, 2008.

O. Chum, J. Matas, and S. Obdrzalek. Epipolar geometry from three correspondences. In *Proc. Computer Vision Winter Workshop 2003, Prague*, pages 83–88, 2003.

H. Cornelius, R. Sara, D. Martinec, T. Pajdla, O. Chum, and J. Matas. Towards complete free-form reconstruction of complex 3d scenes from an unordered set of uncalibrated images. In *Statistical Methods in Video Processing, ECCV 2004 Workshop*, pages 1–12. Springer, 2004.

A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007. ISSN 0162-8828. doi: http://doi.ieeecomputersociety.org/10.1109/TPAMI.2007.1049.

M. Malfiza Garcia de Macedo and A. Conci. Detection of generic conic form parameters using hough transform. In *Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI)*, 2007.

D. DeMenthon and L.S. Davis. Model-based object pose in 25 lines of code. *International Journal of Computer Vision*, 15:123–141, 1995.

J. Denzler, B. Heigl, M. Zobel, and H. Niemann. Plenoptic models in robot vision. In *Künstliche Intelligenz*, pages 62–68, 2003.

L. Dorini and S. Goldenstein. Unscented klt: Nonlinear feature and uncertainty tracking. In *Symposium on Computer Graphics and Image Processing*, Brazil, 2006.

R. O. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15(1):11–15, 1972. ISSN 0001-0782. doi: http://doi.acm.org/10.1145/361237.361242.

R.O. Duda, P.E. Hart, and D.E. Stork. *Pattern Classification*. Wiley Interscience, 2nd edition, 2001.

J.-F. Evers-Senne and R. Koch. Image based rendering from handheld cameras using quad primitives. In *Vision, Modeling, and Visualization VMV: proceedings*, November 2003.

J.-F. Evers-Senne, I. Schiller, A. Petersen, and R. Koch. A mobile augmented reality system with distributed tracking. In *Proceedings of 3DPVT*, 2006.

M. Felsberg and J. Hedborg. Real-time visual recognition of objects and scenes using p-channel matching. In *Proceedings of SCIA*, pages 908–917, 2007a.

M. Felsberg and J. Hedborg. Real-time view-based scene recognition for tracking initialization. *Journal of Real-time Image Processing*, 2(2-3):103–115, November 2007b.

R. Fernando and M. J. Kilgard. *The Cg Tutorial: The Definitive Guide to Programmable Real-Time Graphics*. Addison-Wesley, 2003.

S. Finsterwalder and W. Scheufele. Das Rückwärtseinschneiden im Raum. In Königliche Bayerische Akademie der Wissenschaften, editor, *Sitzungsberichte der mathematisch-physikalischen Klasse*, volume 23/4, pages 591–614, 1903.

M. Fischler and R. Bolles. RANdom SAmpling Consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. ISSN 0001-0782. doi: http://doi.acm.org/10.1145/358669.358692.

M. Fleck. Perspective projection: The wrong imaging model. Technical Report TR 95-01, Computer Science, University of Iowa, 1995.

L. M. J. Florack, B. M. Ter Haar Romeny, J. J. Koenderink, and M. A. Viergever. General intensity transformations and differential invariants. *Journal of Mathematical Imaging and Vision*, 4(2):171–187, 1994. doi: 10.1007/BF01249895.

J. Flusser and T. Suk. Pattern recognition by affine moment invariants. *Pattern Recognition*, 26(1):167–174, 1993.

W. Förstner. Uncertainty and projective geometry. In *Handbook of Computational Geometry for Pattern Recognition, Computer Vision,Neurocomputing and Robotics*, pages 493–534. Springer, 2005.

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

E. Foxlin and L. Naimark. Vis-tracker: A wearable vision-inertial self-tracker. In *Proceedings of IEEE Conference on Virtual Reality (VR 2003), Los Angeles, CA*, 2003.

J.-M. Frahm. *Camera Self-Calibration with Known Camera Orientation*. PhD thesis, University of Kiel, 2005.

J.-M. Frahm and Reinhard Koch. Camera calibration with known rotation. In *Proceedings of IEEE Int. Conf. Computer Vision ICCV*, Nice, France, 2003.

J.-M. Frahm and M. Pollefeys. Ransac for (quasi-)degenerate data (qdegsac). In *Proceedings of CVPR2006*, pages 453– 460, 2006. doi: 10.1109/CVPR.2006.235.

A. Fusiello. A matter of notation: Several uses of the kronecker product in 3d computer vision. *Pattern Recognition Letters*, 28:2127–2132, 2007.

A. Fusiello. Improving feature tracking with robust statistics. In *Pattern Analysis and Application*, volume 2, pages 312–320. Springer-Verlag, London, 1999.

X.-S. Gao, X.-R. Hou, J.-L. Tang, and H.-F. Cheng. Complete solution classification for the perspective-three-point problem. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 25(8):930–943, 2003. doi: 10.1109/TPAMI.2003.1217599.

C. F. Gauss. Untersuchungen über gegenstände der höheren geodäsie. erste abhandlung. *Abhandlungen der Königlichen Gesellschaft der Wissenschaften in Göttingen*, 2:3–46, 1843/1844.

C. Geyer and K. Daniilidis. Catadioptric projective geometry. *International Journal of Computer Vision*, 43:223–243, 2001.

C. Geyer and K. Daniilidis. Conformal rectification of omnidirectional stereo pairs. In *Proceedings of Omnivis 2003: Workshop on Omnidirectional Vision and Camera Networks*, 2003.

J.-J. Gonzalez-Barbosa and S. Lacroix. Fast dense panoramic stereovision. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 1210–1215, 2005.

M. Grabner and H. Bischof. Object recognition based on local feature trajectories. In *1st Cognitive Vision Workshop, OCG Oesterreichische Computer Gesellschaft*, 2005.

E. Grafarend and J.Shan. Closed-form solution of p4p or the three-dimensional resection problem in terms of möbius barycentric coordinates. *Journal of Geodesy*, 71:217–231, 1997.

A. Gray. *Modern Differential Geometry of Curves and Surfaces*. CRC Press, Boca Raton, Florida, 1994.

D. Grest, D. Herzog, and R. Koch. Monocular body pose estimation by color histograms and point tracking. In *Proceedings of DAGM 2006*, 2006.

A. Gruen and T.S. Huang, editors. *Calibration and Orientation of Cameras in Computer Vision*, chapter 2, page 7ff. Springer, 2001.

J. A. Grunert. Das Pothenot'sche Problem, in erweiterter Gestalt; nebst Bemerkungen über seine Anwendung in der Geodäsie. In *Archiv der Mathematik und Physik*, volume 1, pages 238–248, Greifswald, 1841. Verlag C.A. Koch.

F. R. Hampel. A general qualitative definition of robustness. *The Annals of Mathematical Statistics*, 42(6):1887–1896, 1971. ISSN 00034851.

F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. *Robust Statistics: The Approach Based on Influence Functions (Wiley Seriesin Probability and Statistics)*. Wiley-Interscience, New York, revised edition, April 2005. ISBN 0471735779.

B. M. Haralick, C.-N. Lee, K. Ottenberg, and M. Nölle. Review and analysis of solutions of the three point perspective pose estimation problem. *International Journal of Computer Vision*, 13(3):331–356, 12 1994. doi: http://dx.doi.org/10.1007/BF02028352.

C. G. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference, Manchester*, pages 147–151, 1988.

R. Hartley. Self-calibration from multiple views with a rotating camera. In *LNCS 800 (ECCV 94)*, pages 471–478. Springer-Verlag, 1994.

R. Hartley. Self-calibration of stationary cameras. *International Journal of Computer Vision*, 22(1):5–23, 1997a.

R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.

R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision (Second Edition)*. Cambridge University Press, second edition, 2004.

R. I. Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997b. ISSN 0162-8828.

H. Hattori and A. Maki. Stereo matching with direct surface orientation recovery. In *Proceedings of the British Machine Vision Conference*, 1998.

P. S. Heckbert. Fundamentals of texture mapping and image warping. Master's thesis, CS Division, U.C. Berkeley, 06 1989. UCB/CSD 89/516.

B. Heigl, J. Denzler, and H. Niemann. Combining computer graphics and computer vision for probabilistic visual robot navigation. In Jacques G. Verly, editor, *Enhanced and Synthetic Vision 2000*, volume 4023, pages 226–235, 2000. ISBN 0819436496.

J. Heikkilä and O. Silvén. A four-step camera calibration procedure with implicit image correction. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 97)*, 1997.

G. Hua, M. Brown, and S. Winder. Discriminant embedding for local image descriptors. In *Proceedings of the 11th International Conference on Computer Vision (ICCV07)*, Rio de Janeiro, October 2007.

P. J. Huber. Robust estimation of a location parameter. *Annals of Statistics*, 35(1):73–101, 1964.

M. Irani, B. Rousso, and S. Peleg. Recovery of ego-motion using region alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(3):268–272, 1997.

Y. Aloimonos J. Neumann, C. Fermüller. Eyes from eyes: New cameras for structure from motion. In *In IEEE Workshop on Omnidirectional Vision, pp. 19-26*, 2002.

D. Jacobs and R. Basri. 3-d to 2-d pose estimation with regions. *International Journal of Computer Vision*, 34:123–145, 1999.

B. Jähne. *Digitale Bildverarbeitung*. Springer Verlag, Berlin, Heidelberg, 6th edition, 2005.

E. T. Jaynes. Information theory and statistical mechanics. *The Physical Review*, 106(4):620–630, 1957.

H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In Andrew Zisserman David Forsyth, Philip Torr, editor, *European Conference on Computer Vision*, volume I of *LNCS*, pages 304–317. Springer, oct 2008.

M. Jethwa, A. Zisserman, and A. Fitzgibbon. Real-time panoramic mosaics and augmented reality. In *Proceedings of BMVC*, pages 852–862, 1998.

H. Jin, P. Favaro, and S. Soatto. A semi-direct approach to structure from motion. *The Visual Computer*, 19(6):377–394, 2003. doi: 10.1007/s00371-003-0202-6.

B. Johansson, M. Oskarsson, and K. Åström. Structure and motion estimation from complex features in three views. In *Indian Conference on Computer Vision, Graphics and Image Processing*, 2002.

S. J. Julier and J. K. Uhlmann. A new extension of the Kalman filter to nonlinear systems. In *Int. Symp. Aerospace/Defense Sensing, Simulation and Controls, Orlando, Florida*, 1997.

S. J. Julier and J. K. Uhlmann. The scaled unscented transformation. In *Proceedings of the IEEE American Control Conference*, pages 4555–4559, Anchorage AK, USA, 8–10 May 2002. IEEE.

T. Kadir and M. Brady. Scale, saliency and image description. *International Journal of Computer Vision*, 45:83–105, 2001.

T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. In *Proceedings of the 8th European Conference on Computer Vision*, 2004.

F. Kahl and A. Heyden. Using conic correspondence in two images to estimate the epipolar geometry. In *Proceedings of ICCV*, pages 761–766, 1998.

O. Kähler and J. Denzler. Rigid motion constraints for tracking planar objects. In *Lecture Notes in Computer Science*, volume 4713, pages 102–111. Springer, 2007. doi: 10.1007/978-3-540-74936-3-11.

R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice*. Dover, 2005.

S.B. Kang and R. Szeliski. 3-d scene data recovery using omnidirectional multibaseline stereo. In *CVPR '96: Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)*, page 364, Washington, DC, USA, 1996. IEEE Computer Society. ISBN 0-8186-7258-7.

J. Kannala, M. Salo, and J. Heikkilä. Algorithms for computing a planar homography from conics in correspondence. In *Proceedings of BMVC 2006*, 2006.

H. Kato and M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proceedings of the 2nd International Workshop on Augmented Reality (IWAR 99)*, 1999.

Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 511–517, 2004.

J.-S. Kim and I. S. Kweon. Infinite homography estimation using two arbitrary planar rectangles. In *Proceedings of ACCV 2006*, pages 1–10, 2006.

S. J. Kim, J.-M. Frahm, and M. Pollefeys. Joint feature tracking and radiometric calibration from auto-exposure video. In *Proceedings of ICCV 2007*, 2007.

R. Koch. Dynamic 3d scene analysis through synthesis feedback control. *IEEE Transactions on Pattern Analysis and Machine Intelligence, Special issue on analysis and synthesis, Vol. 15 (6)*, 15(6):556–568, 1993.

R. Koch, M. Pollefeys, B. Heigl, L. van Gool, and H. Niemann. Calibration of hand-held camera sequences for plenoptic modeling. In *Proceedings of ICCV*, Korfu, Greece, Sept. 1999.

R. Koch, K. Köser, B. Streckel, and J.-F. Evers-Senne. Markerless image-based 3d tracking for real-time augmented reality applications. In *WIAMIS 2005*, Montreux, Switzerland, April 2005.

R. Koch, J.-F. Evers-Senne, I. Schiller, H. Wuest, and D. Stricker. Architecture and tracking algorithms for a distributed mobile industrial ar system. In *Proceedings of the 5th International Conference on Computer Vision Systems (ICVS07)*, 2007.

K. Köser and R. Koch. Perspectively invariant normal features. *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, Oct. 2007. ISSN 1550-5499. doi: 10.1109/ICCV.2007.4408837.

K. Köser and R. Koch. Differential spatial resection - pose estimation using a single local image feature. In *European Conference on Computer Vision (LNCS 5302-5305)*, 2008a.

K. Köser and R. Koch. Exploiting uncertainty propagation in gradient-based image registration. In *Proceedings of the British Machine Vision Conference*, pages 83–92, 2008b.

K. Köser, B. Bartczak, and R. Koch. Drift-free pose estimation with hemispherical cameras. In *IEE Conference on Visual Media Production (CVMP) 2006*, pages 20–28, London, Nov. 2006a.

K. Köser, V. Härtel, and R. Koch. Robust feature representation for efficient camera registration. In *Lecture Notes in Computer Science 4174 (DAGM06)*, pages 739–749, 2006b.

K. Köser, B. Bartczak, and R. Koch. Robust gpu-assisted camera tracking using free-form surface models. *Journal of Real Time Image Processing*, 2: 133–147, 2007a.

K. Köser, B. Bartczak, and R. Koch. An analysis-by-synthesis camera tracking approach based on free-form surfaces. In Fred A. Hamprecht, Christoph Schnörr, and Bernd Jähne, editors, *Pattern Recognition*, volume 4713 of *LNCS*, pages 122–131. Springer, 2007b. ISBN 978-3-540-74933-2.

K. Köser, C. Beder, and R. Koch. Conjugate rotation: Parameterization and estimation from an affine feature correspondence. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

S. Kyle. Using parallel projection mathematics to orient an object relative to a single image. *The Photogrammetric Record*, 19:38–50, 2004.

M. Lefebure and L.D. Cohen. Image registration, optical flow and local rigidity. *Journal of Mathematical Imaging and Vision*, 14(2):131–147, 2001.

V. Lepetit and P. Fua. Monocular model-based 3d tracking of rigid objects: A survey. *Foundations and Trends in Computer Graphics and Vision*, 1 (1):1–89, October 2005.

V. Lepetit and P. Fua. Keypoint recognition using randomized trees, vol. 28, nr. 9, pp. 1465 - 1479, 2006. *Transactions on Pattern Analysis and Machine Intelligence*, 28:1465–1479, 2006.

V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *Conference on Computer Vision and Pattern Recognition*, volume 2, pages 775–781, San Diego, CA, June 2005.

J. P. Lewis. Fast normalized cross-correlation. In *Vision Interface*, pages 120–123. Canadian Image Processing and Pattern Recognition Society, 1995.

T. Lindeberg. Scale space theorie: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, 21(2):224–270, 1994.

T. Lindeberg. Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention. *International Journal of Computer Vision*, 11:283–318, 1993a.

T. Lindeberg. On scale selection for differential operators. In *Proceedings of the 8th Scandinavian Conference on Image Analysis*, pages 857–866, 1993b.

T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):79–116, 1998.

T. Lindeberg and J. Garding. Shape-adapted smoothing in estimation of 3-d depth cues from affine distortions of local 2-d brightness structure. *Image and Vision Computing*, 15:415–434, 1997.

D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

D. G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, pages 1150–1157, 1999.

C.-P. Lu, G. D. Hager, and E. Mjolsness. Fast and globally converging pose estimation from video images. *Transactions on Pattern Analysis and Machine Intelligence*, 22(6):610–622, 2000.

B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI81*, pages 674–679, 1981.

S. D. Ma. Conics-based stereo, motion estimation, and pose determination. *International Journal of Computer Vision*, 10(1):7–25, 1993.

J. Matas, O. Chum, U. Martin, and T. Pajdla. Distinguished regions for wide-baseline stereo. Technical Report 2001-33, Center for Machine Perception, Czech Technical University, Prague, 2001.

J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proceedings of BMVC02*, 2002.

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

Matlab. Version 7.6.0.324 (R2008a), Symbolic Math Toolbox V.3.2.3, 2008. URL http://www.mathworks.com.

J. C. McGlone, editor. *Manual of Photogrammetry*. ASPRS, 5th edition, 2004.

N. Metropolis and S. Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44(247):335–341, September 1949.

B. Micusik. *Two-View Geometry of Omnidirectional Cameras.* PhD thesis, Czech Technical University in Prague, Technicka 2, 166 27 Prague 6, Czech Republic, June 2004.

B. Micusik and P. Pajdla. Estimation of omnidirectional camera model from epipolar geometry. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR'03)*, volume 1, page 485. IEEE Computer Society, June 2003.

B. Micusik and T. Pajdla. Structure from motion with wide circular field of view cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7):1135–1149, 2006. ISSN 0162-8828. doi: http://dx.doi.org/10.1109/TPAMI.2006.151.

E. M. Mikhail and F.E. Ackermann. *Observations and Least Squares.* IEP, New York, 1976.

K. Mikolajczyk and J. Matas. Improving descriptors for fast tree matching by optimal linear projection. In *Proceedings of Eleventh IEEE International Conference on Computer Vision*, 2007.

K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004a.

K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 27 (10):1615–1630, 2005.

K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004b.

K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *International Conference on Computer Vision*, volume 1, pages 525–531, 2001. doi: 10.1109/ICCV.2001.937561.

K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *European Conference on Computer Vision (ECCV)*, pages 128–142, 2002.

K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005. ISSN 0920-5691.

N. Molton, A. Davison, and I. Reid. Parameterisation and probability in image alignment. Technical Report OUEL Report 2266/03, University of Oxford, 2003.

N. Molton, A. Davison, and I. Reid. Locally planar patch features for real-time structure from motion. In *Proceedings of BMVC*, 2004.

P. Montesinos, V. Gouet, and R. Deriche. Differential invariants for color images. In *Proceedings of 14th International Conference on Pattern Recognition*, 1998.

H. Moravec. *Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover*. PhD thesis, Stanford University, 1980. Available as Stanford AIM-340, CS-80-813 and republished as a Carnegie Mellon University Robotics Institue Technical Report to increase availability.

P. Mordohai, J.-M. Frahm, A. Akbarzadeh, B. Clipp, C. Engels, D. Gallup, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewénius, H. Towles, G. Welch, R. Yang, M. Pollefeys, and D. Nistér. Real-time video-based reconstruction of urban environments. In *3D-ARCH 2007: 3D Virtual Reconstruction and Visualization of Complex Architectures*, Zurich, Switzerland, 2007.

D. Murray. *Patchlets: a method of interpreting correlation stereo 3D data*. PhD thesis, University of British Columbia, 2003.

D. Murray and J. J. Little. Segmenting correlation stereo range images using surface elements. In *Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission*, pages 656–663, 2004.

D. Nistér. *Automatic Dense Reconstruction from Uncalibrated Video Sequences*. PhD thesis, Kungl Tekniska Hogskolen, 2001.

D. Nistér. Preemptive ransac for live structure and motion estimation. In *IEEE International Conference on Computer Vision (ICCV 2003)*, pages 199–206, 2003.

D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770, 2004.

D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, volume 2, pages 2161–2168, June 2006.

D. Nistér and H. Stewénius. A minimal solution to the generalized 3-point pose problem. *Journal of Mathematical Imaging and Vision*, 2006.

S. Obdrzalek and J. Matas. *Toward Category-Level Object Recognition*, chapter 2: Object Recognition using Local Affine Frames on Maximally Stable Extremal Regions, pages 85–108. Lecture Notes in Computer Science (Vol. 4170). Springer-Verlag, Berlin, Heidelberg, 2006.

D. Oberkampf, D. DeMenthon, and L.S. Davis. Iterative pose estimation using coplanar feature points. *CVGIP*, 63(3), 1996.

M. Perdoch, J. Matas, and O. Chum. Epipolar geometry from two correspondences. In *Proceedings of ICPR 2006*, pages 215–220, 2006.

C. Perwass and G. Sommer. The inversion camera model. In *28. Symposium für Mustererkennung, DAGM 2006, Berlin, 12.-14.09.2006*, number 4174 in LNCS, pages 647–656. Springer-Verlag, Berlin, Heidelberg, 2006.

J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR 2007*, pages 1–8, 2007.

T. Pietzsch and A. Grossmann. A method of estimating oriented surface elements from stereo images. In *Proceedings of British Machine Vision Conference*, 2005.

M. Pollefeys and L. van Gool. Stratified self-calibration with the modulus constraint. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):707–724, 1999.

M. Pollefeys, R. Koch, and L. van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *ICCV*, pages 90–95, 1998.

M. Pollefeys, L. van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and Reinhard Koch. Visual modeling with a hand-held camera. *International Journal of Computer Vision*, 59(3):207–232, 2004.

W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 2nd edition, 1992.

F. Riggi, M. Toews, and T. Arbel. Fundamental matrix estimation via TIP - transfer of invariant parameters. In *Proceedings of the 18th International Conference on Pattern Recognition*, pages 21–24, Hong Kong, August 2006.

F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. 3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *Int. J. Comput. Vision*, 66(3):231–259, 2006. ISSN 0920-5691. doi: http://dx.doi.org/10.1007/s11263-005-3674-1.

F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. Segmenting, modeling, and matching video clips containing multiple moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):477–491, 2007. ISSN 0162-8828. doi: http://doi.ieeecomputersociety.org/10.1109/TPAMI.2007.57.

P. J. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984. ISSN 01621459.

D. Salomon. *Transformations and Projections in Computer Graphics*. Springer Verlag, London, 2006.

J. A. Scales and A. Gersztenkorn. Robust method in inverse theory. *Inverse Problems*, 4:1071–1091, 1988.

D. Scaramuzza, A. Martinelli, and R. Siegwart. A toolbox for easy calibrating omnidirectional cameras. In *Proceedings to IEEE International Conference on Intelligent Robots and Systems (IROS '06)*, Beijing, China, October 2006a.

D. Scaramuzza, A. Martinelli, and R. Siegwart. A flexible technique for accurate omnidirectional camera calibration and structure from motion. In *Proceedings of IEEE International Conference of Vision Systems (ICVS '06)*. IEEE, January 2006b.

C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5): 530–535, 1997. ISSN 0162-8828. doi: http://doi.ieeecomputersociety.org/10.1109/34.589215.

C. Schmid and A. Zisserman. Automatic line matching across views. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 666–671, 1997.

C. Schmid and A. Zisserman. The geometry and matching of lines and curves over multiple views. *International Journal of Computer Vision*, 40(3):199–234, 2000.

C. Schmid, R. Mohr, and C. Bauckhage. Comparing and evaluating interest points. In *International Conference on Computer Vision*, pages 230–235, 1998.

S. Se, D. Lowe, and J. Little. Vision-based global localization and mapping for mobile robots. *IEEE Transactions on Robotics*, 21(3):364–375, 2005.

J. Shi and C. Tomasi. Good features to track. In *Conference on Computer Vision and Pattern Recognition*, pages 593–600, Seattle, June 1994. IEEE.

H.-Y. Shum and R. Szeliski. Construction of panoramic mosaics with global and local alignment. *International Journal of Computer Vision*, 36(2): 101–130, February 2000.

J. Skoglund and M. Felsberg. Covariance estimation for sad block matching. In *Proceedings of SCIA*, pages 374–382, 2007.

J. Skoglund and M. Felsberg. Evaluation of subpixel tracking algorithms. In *ISVC*, pages 374–382, 2006.

I. Skrypnyk and D. G. Lowe. Scene modelling, recognition and tracking with invariant image features. In *IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 110–119, 2004.

M. Skurichina. *Stabilizing weak classifiers*. PhD thesis, Delft University of Technology, 2001.

M. Smereka. Detection of elliptical shapes using contour grouping. In *Computer Recognition Systems*, pages 443–450. Springer Berlin / Heidelberg, 2005. doi: 10.1007/3-540-32390-2.

S. M. Smith and J. M. Brady. Susan - a new approach to low level image processing. Technical report, Oxford University, 1995.

N. Snavely, S. M. Seitz, and " R. Szeliski. Photo tourism: Exploring photo collections in 3d. *ACM Transactions on Graphics*, 25(3), 2006.

M. E. Spetsakis and Y. Aloimonos. Structure from motion using line correspondences. *International Journal of Computer Vision*, 4:171–183, 1990.

R. Steele and C. Jaynes. Feature uncertainty arising from covariant image noise. In *Proceedings of CVPR 2005*, pages 1063–1070, 2005.

H. Stewénius. *Gröbner Basis Methods for Minimal Problems in Computer Vision*. PhD thesis, Lund University, April 2005.

B. Streckel and R. Koch. Lens model selection for visual tracking. In *Lecture Notes in Computer Science 3663 (DAGM 2005)*, Vienna, Austria, 2005.

D. Stricker. *Computer-Vision-basierte Tracking- und Kalibrierungsverfahren für Augmented Reality*. PhD thesis, Technische Universität Darmstadt, 2002.

D. Stricker and T. Kettenbach. Real-time markerless vision-based tracking for outdoor augmented reality applications. In *IEEE and ACM International Symposium on Augmented Reality (ISAR 2001)*, 2001.

R. Szeliski. Image alignment and stitching: A tutorial. *Foundations and Trends in Computer Graphics and Computer Vision*, 2, 2006.

R. Szeliski and P. H. S. Torr. Geometrically constrained structure from motion: Points on planes. *Lecture Notes in Computer Science*, 1506:171–186, 1998.

J.-I. Takiguchi, M. Yoshida, A. Takeya, J.-I. Eino, and T. Hashizume. High precision range estimation from an omnidirectional stereo system. In Jacques G. Verly, editor, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 263–268, 2002. ISBN 0-7803-7398-7.

G. Thomas, J . Chandaria, B. Bartczak, K. Köser, R. Koch, M. Becker, G. Bleser, D. Stricker, C. Wohlleber, M. Felsberg, F. Gustafsson, J. D. Hol, T. B. Schön, J. Skoglund, P. J. Slycke, and S. Smeitz. Real-time camera tracking in the matris project. *SMPTE Motion Imaging Journal*, 07/08, 2007.

G. A. Thomas. Real-time camera tracking using sports pitch markings. *Journal of Real-Time Image Processing*, 2:117–132, 2007. doi: 10.1007/s11554-007-0041-1.

G. A. Thomas. Real-time camera pose estimation for augmenting sports scenes. In *Proceedings of CVMP 2006*, pages 10–19, London, Nov. 2006.

G. A. Thomas, J. Jin, T. Niblett, and C. Urquhart. A versatile camera position measurement system for virtual reality. In *Proceedings of International Broadcasting Convention*, pages 284–289, 1997.

E. H. Thompson. Space resection: Failure cases. *The Photogrammetric Record*, 5(27):201–207, 1966. doi: 10.1111/j.1477-9730.1966.tb00870.x.

C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU-CS-91-132, Carnegie Mellon University, April 1991.

P. Torr. An assessment of information criteria for motion model selection. *cvpr*, 1:47–53, 1997. ISSN 1063-6919. doi: http://doi.ieeecomputersociety.org/10.1109/CVPR.1997.609296.

P. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *CVIU*, 78:138–156, 2000.

P.H.S. Torr and A. Fitzgibbon. Invariant fitting of two view geometry or "in defiance of the eight point algorithm". In *British Machine Vision Conference*, pages 83–92, 2003.

B. Triggs. Autocalibration and the absolute quadric. In *Proceedings CVPR*, pages 609–614, Puerto Rico, USA, June 1997.

B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment – a modern synthesis. In B. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, volume 1883 of *Lecture Notes in Computer Science*, pages 298–372. Springer-Verlag, 2000.

R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses, an efficient and accurate camera calibration technique. *IEEE Journal of Robotics and Automation*, RA-3(4):323–344, 1987.

T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3): 177–280, 2008. doi: http:/dx.doi.org/10.1561/0600000017.

T. Tuytelaars and L. van Gool. Wide baseline stereo matching based on local, affinely invariant regions. In *Proceedings of the British Machine Vision Conference*, pages 412–425, 2000.

T. Tuytelaars and L. van Gool. Content-based image retrieval based on local affinely invariant regions. In *International Conference on Visual Information Systems*, pages 493–500, 1999.

L. van Gool, T. Moons, E. Pauwels, and A. Oosterlinck. Vision and lie's approach to invariance. *Image and Vision Computing*, 13(4):259–277, 1995.

A. Van Oosterom and J. Strackee. The solid angle of a plane triangle. *Biomedical Engineering, IEEE Transactions on*, BME-30(2):125–126, Feb. 1983. ISSN 0018-9294. doi: 10.1109/TBME.1983.325207.

B. Williams, G. Klein, and I. Reid. Real-time slam relocalisation. In *Proceedings of 11th International Conference on Computer Vision*, pages 1–8, Rio de Janeiro, Brazil, 2007. doi: 10.1109/ICCV.2007.4409115.

L. Williams. Pyramidal parametrics. *Computer Graphics*, 17(3):1–11, 1983.

S. Winder and M. Brown. Learning local image descriptors. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR07)*, Minneapolis, June 2007.

A. P. Witkin. Scale-space filtering. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1019–1023, 1983.

F. Woelk. *Visual Detection of Independently Moving Objects by a Moving Monocular Observer*. PhD thesis, University of Kiel, 2008.

Z. Xu, R. Schwarte, H. Heinol, B. Buxbaum, and T. Ringbeck. Smart pixel - photonic mixer device (pmd). In *M2VIP '98 - International Conference on Mechatronics and Machine Vision in Practice*, pages 259 – 264, 1998.

L. Zelnik-Manor and M. Irani. Multiview constraints on homographies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):214–223, 2002.

C. Zhang and Z. Hu. Why is the danger cylinder dangerous in the p3p problem. *Acta Automatica Sinica*, 32(4):504–511, 2006.

C. Zhang and Z. Hu. A general sufficient condition of four positive solutions of the p3p problem. *Journal of Computer Science and Technology*, 20(6): 836–842, 2005.

Z. Zhang. Parameter estimation techniques: A tutorial with application to conic fitting. *IVC*, 15(1):59–76, January 1997.

T. Zinßer, C. Gräßl, and H. Niemann. Efficient Feature Tracking for Long Video Sequences. Lecture Notes in Computer Science 3175, pages 326–333, Berlin, Heidelberg, New York, 2004.