UNIVERSIDADE TÉCNICA DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

APPLICATION OF ROBUST ESTIMATION TO COMPUTER VISION: VIDEO MOSAICS AND 3–D RECONSTRUCTION

NUNO RICARDO ESTRELA GRACIAS
(Licenciado)

Dissertação para obtenção do Grau de Mestre em Engenharia Electrotécnica e de Computadores

Orientador Científico:
Doutor José Alberto Rosado dos Santos Victor

Constituição do Júri:
Doutor Jorge dos Santos Salvador Marques
Doutor Helder de Jesus Araújo
Doutor José Alberto Rosado dos Santos Victor

Lisboa, Dezembro de 1997

Resumo

Esta tese insere-se na área da visão por computador e trata o problema da estimação robusta e automática do movimento entre imagens. Um dos problemas fundamentais na análise do movimento consiste na determinação de regiões correspondentes entre imagens. Este problema é por vezes subestimado, pelo que nesta tese são aplicadas e desenvolvidas técnicas robustas de selecção de correspondências correctas.

Usando o formalismo da geometria projectiva, são estabelecidos vários modelos de movimento, que podem ser divididos em duas classes consoante o conteúdo de informação tridimensional presente nas imagens. Caso não haja paralaxe pode-se estabelecer uma relação unívoca entre a localização dos pontos. Caso contrário, é possível recuperar a estrutura projectiva da cena só por análise de correspondências. Um conjunto de técnicas estabelecidas de estimação robusta são analisadas, incluindo o clássico Least Median of Squares. Baseado neste, propõe-se um novo algoritmo que apresenta um ganho em termos de peso computacional para desempenho equivalente.

O modelo de movimento de imagem para o caso de inexistência de paralaxe foi utilizado no registo e composição de sequências de imagens - os mosaicos vídeo. Exemplos de mosaicos são apresentados, cobrindo áreas de aplicação distintas, tais como a cartografia aérea e submarina, representação e compressão de vídeo e realidade virtual. Para o caso alternativo de existência de paralaxe, delineou-se um processo de recuperação da estrutura projectiva. A estimação da matriz fundamental é tratada, comparando-se técnicas baseadas em critérios lineares e não lineares, com parametrizações distintas. São apresentados resultados práticos, com imagens sintéticas e condições controladas, e com imagens reais.

Palavras Chave: Visão por Computador, Análise de Movimento, Estimação Robusta, Mosaicos Vídeo, Matriz Fundamental, Reconstrução não Calibrada.

Abstract

This thesis, in the area of Computer Vision, deals with the problem of image motion estimation in a robust and automatic way. One of the main problems in motion analysis lies on the difficulty of the matching process between corresponding image areas. As this is a commonly overlooked problem, this thesis evolves around the use and applications of robust matching techniques.

Several motion models are established, under the projective geometry framework. These can be divided into two main classes according to the 3-D information content of the images. If there is no parallax then a one-to-one relation can be established between the point locations. Conversely, the presence of parallax allows the recovery of the projective structure of the scene, just by the analysis of a set of point correspondences.

The most commonly used robust estimation techniques are reviewed and compared, including the classic Least Median of Squares. Based on this method, a new algorithm is proposed that compares favorably in terms of computational cost, while attaining the same performance level.

The parallax-free motion models are used in the registration and composition of image sequences for the creation of video mosaics. Results on mosaicing are given in the context of different applications such as aerial and underwater cartography, video coding and virtual reality. For the alternative case, where there are parallax effects, a projective structure recovery method is described. The estimation of the Fundamental Matrix is addressed by comparing techniques based on linear and non-linear criteria, and different parameterisations. Results are presented using synthetic images under controlled conditions, and real images.

Key Words: Computer Vision, Motion Analysis, Robust Estimation, Video Mosaics, Fundamental Matrix, Uncalibrated Reconstruction.

Agradecimentos

O trabalho apresentado nesta tese foi desenvolvido ao longo de um ano e dois meses. Sendo esta uma área na qual não tinha antes trabalhado, gostaria aqui de agradecer a um conjunto de pessoas, cuja atenção, disponibilidade e paciência para com um 'inexperiente' foram de sobremaneira importantes.

Em primeiro lugar o meu mais sincero e sentido agradecimento ao Prof. José Alberto Santos Victor, orientador desta tese. Pelo apoio e disponibilidade constante. Pelo empenho e atenção. Pelo rigor e sinceridade nas discussões. Pelo convívio e a amizade de todos os momentos.

Aos meus colegas do Laboratório de Visão, que se tornaram amigos de todas as ocasiões. Ao Etienne pelo seu espírito matemático e ajuda empenhada. Ao César pelas discussões frutuosas e apoio no Matlab. Ao António *bombeiro* por todos os fogos que apagou! Ao Alexandre e ao Gaspar pela disponibilização de *software* de imagem e material de estudo. Ao Carlos pela companhia fora de horas. À Raquel pela alegria contagiante. Ao Luis Jordão e ao Vitor pelo convívio. A ajuda e o espírito de grupo do *VisLab* vai muito para além do que é aqui enumerável ou descritível. A todos gostaria de expressar o meu agradecimento.

A todos os colegas e amigos do Instituto de Sistemas e Robótica do Pólo de Lisboa. Ao Prof. João Sentieiro pela organização e liderança de uma instituição de investigação de excelência, e pelo empenho na criação de condições de trabalho para os investigadores mais jovens. Ao Prof. Agostinho Rosa pelo disponibilidade, incentivo e amizade de mais de 6 anos.

Ao meu amigo Henrique Miguel Pereira, companheiro de sempre de aventuras científicas, ecológicas e recreativas! Pela amizade e carinho de muitos anos. Ao Vasco Ribeiro por todas as longas conversas e desabafos. Ao José Miguel Lima pela companhia de sempre.

Ao Sr. Joaquim Garcia pelo exemplo de trabalho e de perseverança. Pelo gosto pela ciência que me contagiou, e por mostrar que a qualidade do trabalho só depende do empenho de cada um.

À Junta Nacional de Investigação Científica e Tecnológica e à recém-criada Fundação para a Ciência e a Tecnologia, pelo apoio financeiro na realização do meu Curso de Mestrado através da bolsa PRAXIS XXI/BM/6840/95.

À minha mana Maria João, pela companhia, conversa, passeios, piadas e desenhos. Por toda a atenção e amizade que tornam a nossa vivência muito valiosa e querida mesmo quando nenhum de nós tem tempo para nada. Força com a Anatomia!

Aos meus pais pelas condições que me levaram a este mestrado e à sua conclusão. Por sempre acreditarem e por sempre apoiarem. Esta tese é-vos dedicada.

À Nela pelo que ela sabe que representou.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Vision is one of the our most important sensing capabilities, and the one that provides the highest information content to our brain.

It is used by many biological systems as the principal mechanism for perceiving the surrounding space, and identifying items and phenomena essential for their survival. The effectiveness illustrated by many animals of the use of visual sensing on the perception and action, has become a source of inspiration for many problems in the field of robotics. It comes as no surprise that in this field, vision is considered as the most promising and the most challenging of the artificial senses.

Over the last decades, computer vision has emerged as discipline which focuses on issues such as visual information extraction, representation and use. Since the real world is in constant motion, the analysis of time-varying imagery can reveal valuable information about the environment [56], and allow a machine or organism to meaningfully interact with it. For this reason the analysis of image sequences for the extraction of three-dimensional structure and motion has been at the core of computer vision from the early days [15].

The research on motion perception and analysis can be divided into three time periods [61]. In the first period, research focused in trying to find whether it was possible to infer three dimensional information on motion and structure from the analysis of image projections. The subject was treated in the broadest terms. Once it was certain that a solution existed, the next period was concerned with devising ways to find it and prove its uniqueness. Researchers managed to achieve this on many sub-problems, by theoretical analysis, usually dealing with the minimum required amount of data [31]. This 'minimalistic' approach, often leading to highly noise sensitive algorithms, gave rise to the idea that the problem of structure and motion recovery was such an ill-posed problem that only qualitative solutions were attainable [67]. Therefore, in the last period, research was concerned in using as much information as possible in an optimal manner, in order to promote low error sensitivity through redundancy. The redundancy was achieved by using

more feature correspondences [60] or more image frames [61] than theoretically required for noise-free situations.

By combining multiple observations, noise sensitivity can be reduced. This is traditionally performed under the least-squares framework. Contributing factors are the ease of use, low algorithmic complexity and the availability of efficient methods for finding the solution, such as the singular value decomposition. The major drawback, however, lies on its inability to deal with gross errors. The errors in motion estimation can be informally divided into two main classes. The first includes the errors due to image quantization and limited resolution of the methods for the extraction of primitive information, such as point features and matches. The second class includes the errors corresponding to data in gross disagreement with the assumed underlying model. These *outliers* may have been originated by model shortcomings in describing the data or by the ill-posed nature of some of the problems being solved. When combining multiple observations, outliers are usually included in the initial fit. Under least-squares, outliers can make the fitting process so distorted as to have an arbitrary result.

A great deal of research effort is currently been put into the development of robust methods for computer vision. It is with the goal of providing good performance for real imaging in real applications that robust methods are studied and used in this thesis.

The main objectives of this thesis are:

- **The use of projective geometry for the derivation of image motion models**. In the last few years, significant progress has been made on fundamental problems in computer vision due to the use of classical projective geometry. Projective geometry embeds our familiar Euclidean geometry and is better suited for describing the relations between scene structure and its image projections. Its generality allows irrelevant details to be ignored. By the use of projective geometry, image motion models can be easily obtained.

- **The development of a framework for model-based motion estimation**. The methods used for motion estimation should be able to cope with several motion models in a similar way, thus allowing for the most appropriate model to be used.

- **The creation of high quality video mosaics through the application of the motion estimation to image registration.** Models for parallax-free scenes allow the creation of video mosaics. These mosaics can be used in a multitude of applications ranging from visually elucidative panoramas to efficient video coding and cartography.

- **The recovery of structure from uncalibrated images**. When parallax is present, then scene structure information becomes available. Projective geometry

provides the adequate tools for the analysis of structure and for reconstruction, even when no camera calibration is available. Three dimensional reconstruction is still one of the cornerstones of robotics applications where spatial modelling of the environment is required.

1.1 Related Work

A wide range of topics are covered in this thesis, such as robust estimation, feature matching, image registration, mosaic rendering, fundamental matrix estimation and uncalibrated reconstruction. These aspects are seldom treated in a uniform manner in the literature. However, the work of some authors relates closely to various of the individual issues. We will now briefly describe selected work, bearing in mind that relevant related work will be referred to throughout this thesis, when appropriate.

An in-depth study of robust estimation in the context of motion segmentation is presented by Torr in [67]. Here, corner features and matched correspondences are used as data for the clustering of features consistent with rigid world objects. The constraints arising from two and three views are considered. The detection of outliers and of data degeneracy are treated simultaneously, and a method for dealing with both is implemented.

A system for image registration and mosaic creation of underwater sequences is presented in [45]. Mosaicing is performed in real-time, solely from visual information. This is attained at the cost of using a very simple motion model, which restricts the range of applications.

The issue of applying robust techniques for the estimation of the fundamental matrix has been dealt with by a number of authors [42, 11, 21]. In this context, comprehensive review of robust methods is given in [66]. In [75] an approach is proposed for the recovery of the epipolar geometry from two uncalibrated views, which bears a close resemblance to the work described in this thesis for the topic. However, the issue of structure recovery is not tackled. The effect of the choice of the parameterization in the estimation of the fundamental matrix is studied in [43].

Uncalibrated reconstruction is currently a topic of intensive research. A simple procedure for Euclidean structure recovery from two views using ground-truth information was put forward by Hartley in [24]. This approach was further extended in [48]. Recent work is now focused on Euclidean reconstruction using just image information [68, 29, 12]. One of the main differences of the work presented in this thesis is that no perfect feature matching is assumed whereas most of these approaches consider the matching problem to be solved.

1.2 Thesis Outline

Chapter 2 introduces some concepts and properties of projective geometry. The projective camera model is presented, together with its calibration process. For planar scenes, the relation between image projections is analyzed and a class of motion models is presented. Next, the two-view geometry for arbitrary scenes in introduced. Key issues are the epipolar geometry and the fundamental matrix. An uncalibrated reconstruction procedure is shown where Euclidean recovery is attained through the use of some ground-truth points.

Chapter 3 describes the two main approaches to motion estimation, namely feature-based methods and optical flow. Robust estimators are reviewed, including the M-estimators, case-deletion diagnostics and random sampling algorithms. A variant of the least median of squares is proposed.

Chapter 4 is devoted to the application to video mosaicing. The two stages of mosaic creation, registration and rendering, are described separately. On the registration stage the motion parameters between frames are estimated, then individual frames are fit to a global model of the sequence. The rendering stage deals with the creation of a single mosaic, by applying a temporal operator over the registered and aligned images. Several examples of mosaics are given and applications are discussed.

Chapter 5 reports the application of robust techniques to the estimation of the epipolar geometry, and to 3-D structure recovery. The fundamental matrix is estimated under different minimization criteria and parameterization. Results of the reconstruction procedure described in Chapter 2 are given and discussed, using both synthetic and real images.

Finally, Chapter 6 summarizes the work presented on this thesis and draws the main conclusions. Directions for future developments are also given.

Chapter 2

Projective Geometry

In this chapter we will introduce some concepts and properties of projective geometry. These will be required for the understanding of the formulas presented further on, when we address the creation of video mosaics and 3-D reconstruction.

Some of the basic properties of projective space will be described in section 2.1, such as the notions of projective space and collineation, followed by the perspective camera model, widely used in the Computer Vision literature.

We then move on to the study of the planar transformations. Since these transformations can relate 3-D points lying on planes, they will be extensively used on the mosaicing of planar scenes.

Finally section 2.4 considers the problem of 3-D reconstruction using uncalibrated cameras. It introduces the fundamental matrix which encapsulates all the geometric information that can be extracted just by image analysis about a setup of two cameras.

2.1 Basic Properties of the Projective Space

Definition 1 (Affine Space and Projective Space) *The set of points parameterized by the set of all real valued n-vector* $(x_1, \ldots, x_n)^T \in \mathbb{R}^n$ *is called* Affine Space.

The set of points represented by a $n+1$ *vector* $(x_1, \ldots, x_n, x_{n+1})^T \in \mathbb{R}^{n+1}$ *is called a* Projective Space $\mathbb{P}^n$ *if the following condition and property are considered:*

1. *At least one of the* $n+1$ *vector coordinates is different from zero.*
2. *Two vectors* $(x_1, \ldots, x_n, x_{n+1})^T$ *and* $(\lambda x_1, \ldots, \lambda x_n, \lambda x_{n+1})^T$ *represent the same point for any* $\lambda \neq 0$.

The elements x_i of a projective space vector are usually called *homogeneous coordinates* or *projective coordinates*. The affine space $\mathbb{R}^n$ can be considered to be embedded in $\mathbb{P}^n$

by the use of the canonical injection $(x_1, \ldots, x_n)^T \rightarrow (x_1, \ldots, x_n, 1)^T$. Conversely, one can recover the affine coordinates of a point from its homogeneous ones by the mapping,

$(x_1, \ldots, x_n, x_{n+1})^T \doteq \left(\frac{x_1}{x_{n+1}}, \ldots, \frac{x_n}{x_{n+1}}, 1\right)^T \rightarrow \left(\frac{x_1}{x_{n+1}}, \ldots, \frac{x_n}{x_{n+1}}\right)^T$ for $x_{n+1} \neq 0$,

where $\doteq$ denotes the equality-up-to-scale property of the projective coordinates. From this chapter on, we will be using the tilde symbol on top of a vector letter (ex. $\widetilde{\mathbf{x}}$) to denote the projective coordinates of a given point. Eventually this notation will not be used if there is no risk of confusion with the affine counterparts.

If the last coordinate of a point $\mathbf{x} \in I\!P^n$ is null, i.e., $x_{n+1} = 0$, then $\mathbf{x}$ is called *point at infinity*. The direction of such point is given in the affine space by $(x_1, \ldots, x_n)^T$. Under the framework of projective geometry, the set of all points at infinity behaves like any other hyperplane, thus called hyperplane at infinity.

Definition 2 (Collineation) *A linear transformation or* collineation *of a projective space $I\!P^n$ is defined by a non-singular $(n+1) \times (n+1)$ matrix A.*

The matrix A performs an invertible mapping of $I\!P^{n+1}$ onto itself, and is defined up to a non zero scale factor. The usual representations for a collineation are $\lambda\mathbf{y} = A\mathbf{x}$ or $\mathbf{x} \overline{\wedge} \mathbf{y}$.

Definition 3 (Projective Bases) *A* projective basis *of a n-dimensional projective space, is a set of $n+2$ vectors of $I\!P^n$ such that $n+1$ of them are linearly independent.*

Any point $\mathbf{x}$ of $I\!P^n$ can be described as a linear combination of a given basis e_i:

$$\mathbf{x} = \sum_{i=1}^{n+1} x_i \mathbf{e}_i$$

Particularly, the set $\{(1,0,\ldots,0)^T, (0,1,\ldots,0)^T, \ldots, (0,\ldots,1,0)^T, (1,\ldots,1)^T\}$ forms the *canonical basis*. It can easily be seen that this is indeed a basis, and that it contains the points at infinity along each of the n dimensions, plus the *unit point* with all coordinates equal to one.

We will now present a proposition which characterizes the change of projective basis. A proof for this can be found in [15] and in [55].

Proposition 1 *Let $\mathbf{x}_1, \ldots, \mathbf{x}_{n+2}$ and $\mathbf{y}_1, \ldots, \mathbf{y}_{n+2}$ be two sets of vectors with at least $n+1$ linearly independent vectors each, thus forming two projective bases. Then, there exists a non-singular $(n+1) \times (n+1)$ matrix A, such that $\lambda_i \mathbf{y}_i = A\mathbf{x}_i$ for $i = 1, \ldots, n+2$ and a set of scalars λ_i. The matrix A is a collineation, therefore unique up to a scale factor.*

This proposition states an important property of the projective space: a collineation on $I\!P^n$ is completely defined by a set of $n+2$ pairs of corresponding points.

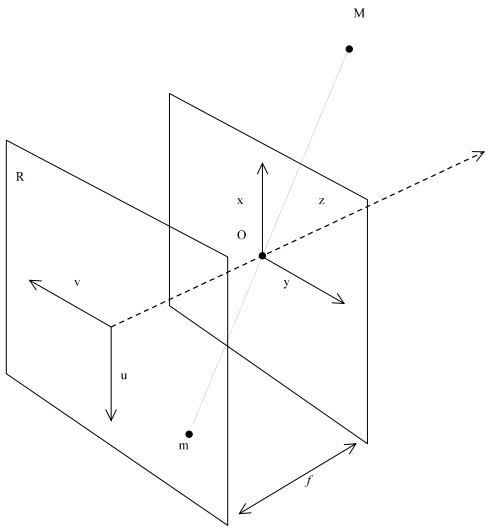


Figure 2.1: Perpective Camera Projection

2.2 Perspective Camera Model

The most commonly used camera model in computer vision is the pinhole model. This a simple and effective way of modelling most of the modern CCD cameras by considering the projection of rays of light passing through a small hole and being projected on a flat surface.

A geometrical model is now presented, based on the system depicted on Figure 2.1. It contains a plane R where the image is formed, called *retinal* or *image plane*. The image of the 3-D point M undergoes a *perspective projection*, passing through the *optical center* O, and is projected in m. The distance of the optical center to the retinal plane is called *focal distance*. The line passing through the optical center and orthogonal to the retinal plane is called *optical axis*. The optical axis intersects the image plane in the *principal point*.

2.2.1 The Perspective Projection Matrix

We will now consider the origin of the 3-D reference frame to be at the camera optical center O, and its $\vec{z}$ axis to be along the optical axis. For the retinal plane, we consider a 2-D reference frame as depicted on Figure 2.1, with the origin in the principal point. Let (u, v) be the 2-D coordinates of $\mathbf{m}$ and (x, y, z) the 3-D coordinates of $\mathbf{M}$. It can easily be seen that the following equations hold.

$$u = \frac{f \cdot x}{z} \qquad v = \frac{f \cdot y}{z}$$

This can be written as a linear relation, by the use of homogeneous coordinates as

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad \Leftrightarrow \qquad \widetilde{\mathbf{m}} \doteq P\widetilde{\mathbf{M}}$$

where P is usually referred as the *perspective projection matrix.*

It is worth noting that the use of projective geometry allows the perspective projection model to be described by a linear equation, which makes the model much easier to deal with. A camera can be considered to perform a linear projective mapping from the projective space $I\!P^3$ to the projective plane $I\!P^2$.

Commonly, the origin of the image reference frame is not the principal point, but the upper left corner of the image. Moreover, the scaling along the u and v axis is not necessarily the same. We can account for this, by performing a coordinate transformation on the image and rewriting the camera mapping as:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k_u & k_\theta & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

where k_u and k_v are scaling factors (along u and v), and (u_0, v_0) is the location of the principal point in the new image referential. The additional parameter k_θ gives the skew between axes. For most CCD cameras k_θ can be considered zero, on applications not relying on high accuracy calibration.

When the 3-D reference frame (world frame) is not the camera frame, the above equation holds if we consider the 3-D points undergo a rigid transformation, *i.e.*, rotation and translation. This transformation that can be easily expressed as a collineation relating the two coordinate systems:

$$\lambda \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} & R & & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \Leftrightarrow \widetilde{\mathbf{x}}' \doteq G\widetilde{\mathbf{x}}$$

The matrix R is a (3×3) rotation matrix and the (3×1) vector $\mathbf{t}$ contains the coordinates of the origin of the world frame expressed in the camera frame.

We can now express the general form of the perspective camera mapping as:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k_u & k_\theta & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} & R & & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

or equivalently,

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f\,k_u & f\,k_\theta & u_0 \\ 0 & f\,k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} & R & & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} G \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.1}$$

The A matrix depends only on parameters internal to the camera, thus called *intrinsic parameters*. Conversely, the G matrix depends only on the chosen external reference frame. The transformation parameters are called the *extrinsic parameters*.

Let us now introduce the notion of *normalized coordinates* of a 3-D point projection. Let $\mathbf{m}$ be a point projection such that $\widetilde{\mathbf{m}} \doteq P\widetilde{\mathbf{M}}$ where P can be expressed in the form of equation (2.1). Let P' be a camera matrix with the same extrinsic parameters but with intrinsic parameters such that A is the identity matrix. Then $\widetilde{\mathbf{n}} \doteq P'\widetilde{\mathbf{M}}$ are the normalized coordinates of $\widetilde{\mathbf{m}}$. It is easy to see that P' corresponds to a camera with unit focal length, principal point coincident with the origin of the image frame and no scaling or skewing along the axes.

For a given camera matrix, the determination of the camera optical center in world coordinates is straightforward, as the following proposition shows.

Proposition 2 (Coordinates of the camera optical center) *Let P be a finite focal length camera matrix. Let L be the (3×3) matrix formed by the first three columns of P, and $\mathbf{p}$ its last column, such that P can be written as $P = [L \quad \mathbf{p}]$. Then the null space of P gives the world coordinates of the camera optical center. The coordinates are $\mathbf{o} = -L^{-1}\mathbf{p}$.*

Proof. Can be found in [15].

Using this proposition, it easy to see that P can also be expressed as $P = [L \quad -L\mathbf{o}]$.

Restricted camera models

The camera model that we have described in this subsection is the commonly used general projection model of 11 independent parameters. However it is worth saying that, on

some applications such as the estimation of scene structure from image motion, restricted camera models should be considered. The restricted models may be better suited if the data are degenerate [74]. For a discussion on camera models and their appropriateness, refer to [74].

2.2.2 Camera Calibration

The process of finding the intrinsic and extrinsic parameters of a cameras is called *camera calibration*. This process can be divided in two major steps[15]:

1. Estimating the matrix P.
2. Recovering the explicit intrinsic and extrinsic parameters from P.

In this thesis we will only deal with the first, as it is all that is required for the 3-D reconstruction procedures presented on chapter 5.

A simple linear calibration procedure will now be presented, based on least-squares minimization. We assume that a set of 3-D points $\mathbf{x}_i$ are available together with their projections on the retinal plane m_i. The perspective camera mapping can be written as:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \mathbf{p}_3^T \end{bmatrix} \widetilde{\mathbf{x}}$$

where $\mathbf{p}_1^T$, $\mathbf{p}_2^T$ and $\mathbf{p}_3^T$ are the row vectors of each of the lines of P, and $\widetilde{\mathbf{x}} = (x, y, z, 1)^T$. Let us re-arrange the equations as:

$$\lambda = \mathbf{p}_3^T \cdot \widetilde{\mathbf{x}} \qquad \Rightarrow$$

$$\mathbf{p}_3^T \cdot \widetilde{\mathbf{x}} \cdot u = \mathbf{p}_1^T \cdot \widetilde{\mathbf{x}}$$
$$\mathbf{p}_3^T \cdot \widetilde{\mathbf{x}} \cdot v = \mathbf{p}_2^T \cdot \widetilde{\mathbf{x}}$$

For each point $\mathbf{x}_i$ and projection m_i, we have two of the above equations. Let $\mathbf{p}_l = (\mathbf{p}_1^T\ \mathbf{p}_2^T\ \mathbf{p}_3^T)^T$ be the column vector containing all the 12 parameters of P. The following homogeneous system can be formed:

$$H \cdot \mathbf{p}_l = 0 \tag{2.2}$$

For a set of n points, H is the $(n \times 12)$ matrix:

$$H = \begin{bmatrix} -x_1 & -y_1 & -z_1 & -1 & 0 & 0 & 0 & 0 & x_1u_1 & y_1u_1 & z_1u_1 & u_1 \\ 0 & 0 & 0 & 0 & -x_1 & -y_1 & -z_1 & -1 & x_1v_1 & y_1v_1 & z_1v_1 & v_1 \\ & & & & & & \vdots & & & & & \\ -x_n & -y_n & -z_n & -1 & 0 & 0 & 0 & 0 & x_nu_n & y_nu_n & z_nu_n & u_n \\ 0 & 0 & 0 & 0 & -x_n & -y_n & -z_n & -1 & x_nv_n & y_nv_n & z_nv_n & v_n \end{bmatrix}$$

If six or more 3-D points are on a general configuration, and their projections are known with sufficient high accuracy, then H will have exactly rank 11. By a general configuration we mean that no four of the points are coplanar, nor they all lie on a twisted cubic as described Faugeras in [15], although this later situation is very unlikely to occur on practical setups.

From Equation (2.2) it can be seen that $\mathbf{p}_l$ is the null space of H, thus defined up to scale. To avoid the trivial solution $\mathbf{p}_l = 0$, one has to impose an additional constraint on P, usually $\|\mathbf{p}_l\| = 1$[1]. Furthermore, real applications are prone to inaccuracies on the measurements of point locations and H will not be rank deficient. In order to find a least-squares solution for this equation, we can formulate the classical minimization problem:

$$\min_{\mathbf{p}_l} \| H \cdot \mathbf{p}_l \| \text{ constrained to } \| \mathbf{p}_l \| = 1 \tag{2.3}$$

By the use of the Lagrange multipliers it can be easily shown that the solution to this problem is the eigenvector associated with the smallest singular value of H. A suitable algorithm for finding the eigenvector is the Singular Value Decomposition (SVD)[51]. This solution has the advantage of being non iterative, thus allowing the implementation of a fast calibration procedure. For a short description of the SVD, refer to Appendix A.

2.3 Planar Transformations

This section is devoted to the presentation of 2-D projective transformations. The importance of the study of these collineations is emphasised by the fact that they can be used as models for image motion with an enormously vast field of application in Computer Vision. We will now show that two different views of the same planar scene in 3-D space are related by a collineation in $I\!P^2$, and that this collineation can be computed by the use of four pairs of matched points on the two images.

Proposition 3 *Let P and P' be two camera projection matrices. Let $\mathbf{n}$ be the (4×1) coefficient vector of a plane $\tilde{\mathbf{x}}_P \in I\!P^3$ not containing the cameras optical centers, such that the plane can be expressed as the inner product $\mathbf{n}^T.\tilde{\mathbf{x}}_P = 0$. Then the coordinates of $P\tilde{\mathbf{x}}_P$ and $P'\tilde{\mathbf{x}}_P$ in the two image frames are related by a projective transformation in $I\!P^2$.*

We will not formally prove this proposition, but present just a proof outline. Let us start by explicitly expressing the set of points which project on a single point $\mathbf{u}$ of retinal plane. The locus of all theses points is a 3-D line called the *optical ray* of $\mathbf{u}$.

[1] Alternatively, one can set $\mathbf{p}_{34} = 1$ and rewrite equation 2.2, where H will be a $(n \times 11)$ matrix and solve it accordingly.

Consider $\tilde{\mathbf{u}} \doteq P\tilde{\mathbf{x}}$. The projection matrix can be written as $P = [L\ \mathbf{p}]$, where L is the square matrix formed by the first three columns of P, and $\mathbf{p}$ is the fourth column. It can be shown [24, 15]that if a camera is not located at infinity, then L is non-singular. The equation for the optical ray can now be found:

$$\lambda\tilde{\mathbf{u}} = [L\ \mathbf{p}] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\lambda\tilde{\mathbf{u}} = L\mathbf{x} + \mathbf{p}$$

$$\mathbf{x} = \lambda L^{-1}\tilde{\mathbf{u}} - L^{-1}\mathbf{p} \tag{2.4}$$

For each value of λ, a point of the ray is defined. We can write an homogeneous equation for $\tilde{\mathbf{x}}$ by the use of $N = \begin{bmatrix} L^{-1} \\ 0\ \ 0\ \ 0 \end{bmatrix}$ and $\mathbf{q} = \begin{bmatrix} -L^{-1}\mathbf{p} \\ 1 \end{bmatrix}$

$$\tilde{\mathbf{x}} \doteq \lambda N\tilde{\mathbf{u}} + \mathbf{q}$$

The vector $\mathbf{q}$ is the null space of P, thus containing the optical center homogeneous coordinates, as seen earlier. For notation simplicity we will now replace λ in the following equations, by the parameter φ, such that $\lambda = -\varphi^{-1}$.The above equation is equivalent to

$$\tilde{\mathbf{x}} \doteq N\tilde{\mathbf{u}} - \varphi\mathbf{q}$$

The equation of the plane in the world frame can be written as the inner product $\mathbf{n} \cdot \tilde{\mathbf{x}}_P = 0$. The intersection of the optical ray and the plane imposes

$$\mathbf{n}^T N\tilde{\mathbf{u}} - \varphi\mathbf{n}^T\mathbf{q} = \mathbf{0}$$

$$\varphi = \frac{\mathbf{n}^T N\tilde{\mathbf{u}}}{\mathbf{n}^T\mathbf{q}}$$

The point of intersection therefore is

$$\tilde{\mathbf{x}} \doteq N\tilde{\mathbf{u}} - \frac{\mathbf{n}^T N\tilde{\mathbf{u}}}{\mathbf{n}^T\mathbf{q}}\mathbf{q} = \left(I - \frac{\mathbf{q}\ \mathbf{n}^T}{\mathbf{n}^T\mathbf{q}}\right) N\tilde{\mathbf{u}}$$

This point is projected on the other camera as

$$\tilde{\mathbf{u}'} \doteq P'\tilde{\mathbf{x}} = P'\left(I - \frac{\mathbf{q}\ \mathbf{n}^T}{\mathbf{n}^T\mathbf{q}}\right) N\tilde{\mathbf{u}} = T_{2D}\tilde{\mathbf{u}}$$

From this we can conclude that the coordinates of the two camera frames are related by the (3×3) collineation T_{2D}.

2.3.1 Linear computation of planar transformations

According to proposition 1, the computation of a planar collineation requires at least four pairs of corresponding points. If we have more than four correspondences, least-square minimization can then be accomplished in a way close to the one outlined above, in the camera calibration subsection. Let T_{2D} be the collineation relating two image planes from which we have a set of n correspondences such that $\widetilde{\mathbf{u}'_i} \doteq T_{2D}\widetilde{\mathbf{u}_i}$, for $i = 1, \ldots, n$. For each pair we will have two linear constraints on the elements of T_{2D}. An homogeneous system of equations can thus be assembled in the form $H.\mathbf{t}_l = 0$, where $\mathbf{t}_l$ is the column vector containing the elements of T_{2D} in a row-wise fashion, and H is a $(2n \times 9)$ matrix

$$H = \begin{bmatrix} u_1 & v_1 & 1 & 0 & 0 & 0 & -u'_1u_1 & -u'_1v_1 & -u'_1 \\ 0 & 0 & 0 & u_1 & v_1 & 1 & -v'_1u_1 & -v'_1v_1 & -v'_1 \\ & & & & & \vdots & & & \\ u_n & v_n & 1 & 0 & 0 & 0 & -u'_nu_n & -u'_nv_n & -u'_n \\ 0 & 0 & 0 & u_n & v_n & 1 & -v'_nu_n & -v'_nv_n & -v'_n \end{bmatrix}$$

The system can now be solved by the means of the Singular Value Decomposition, after imposing the additional constraint of unit norm for $\mathbf{t}_l$, *i.e.*, $\|\mathbf{t}_l\| = 1$.

The use of planar transformations on the correction of radial distortion

The camera model presented in section 2.2 is a linear perspective projection model, which is sufficiently accurate for most commercially available and computer vision applications. Even so, for some applications such as high-accuracy metrology a more complex model is required, capable of dealing with the real-world camera non-linearities.

During the experimental part of the work reported in this thesis, we came to realize that some of the wide-angle lenses that were used for image acquisition caused noticeable radial distortion. For this reason a radial distortion calibration technique was developed. It is based on the estimation of the planar transformation between a planar calibration grid and its projection the image plane. The equations used for the distortion modelling are given on appendix B, together with a description of the algorithm and some test images.

2.3.2 Restricted planar transformations

The most general collineation in $I\!P^2$ has eight independent parameters. As it has been shown, it accounts for the perspective mapping of a planar scene to the image plane of a camera. It can therefore be used to merge two different images of the same plane, just by using a set of point correspondences. The subject of image composition will be addressed in chapter 4, where an in-depth discussion will be presented. For now on let us say that if the scene is static but not planar, then there will be image misalignements due to parallax,

except for the case where the cameras have the same optical center. This is the same as to say that the cameras are free to rotate in any direction and to zoom, but not to translate. For this later case, the relation between camera rotation and the collineation can be found in [63]. Table 2.1 summarizes the two cases where the general collineation captures the exact coordinate transform.

	Scene assumptions	Camera assumptions
Case 1	Arbitrary 3-D	free to rotate on any directions and to zoom
Case 2	Planar	no restrictions on camera movement

Table 2.1: The two parallax-free cases for static scenes.

If additional information is available on the camera setup, such has camera motion constraints, then the coordinate transformation $\widetilde{\mathbf{u}}_i' \doteq T_{2D}\widetilde{\mathbf{u}}_i$ might not need the eight independent parameters of the general case to accurately describe the image motion. As an example we can point out the case where the camera is just panning, thus inducing a simple sideways image translation. If we know beforehand which is the simplest model that can explain the data equally well, then there will be no reason for using the most general. Table 2.2 illustrates some restricted models. These are used in chapter 4 for the construction of video mosaics.

2.4 Projective Stereo Vision

In this section we will present some concepts and methods useful for the analysis of a static scene using a pair of cameras. We will describe the important epipolar constraint by giving a geometric interpretation and an algebraic interpretation, that leads to the presentation of the Fundamental Matrix. It will also be shown how the structure of the scene can be recovered up to a projective transformation by the use of pairs of matched projected points. Furthermore, we illustrate how to find this transformation from additional metric information about the scene, namely some 3-D points with known coordinates in some world frame. A special emphasis is put on the fact that explicit camera calibration is avoided.

The field of projective stereo vision deals with what can be done with completely uncalibrated cameras. The importance of this lies on the fact that, for some applications, full knowledge of the camera parameters is not required. As an example we can cite the work of Beardsley *et al.* [6], where robot navigation is accomplished without metric recovery of environment structure.

Image Model	Matrix form	p	Domain
Translation	$T_{2D} = \begin{bmatrix} t_1 & 0 & t_2 \\ 0 & t_1 & t_3 \\ 0 & 0 & t_1 \end{bmatrix}$	2	Image plane is parallel to the planar scene. No rotation.
Translation and zoom	$T_{2D} = \begin{bmatrix} t_1 & 0 & t_2 \\ 0 & t_1 & t_3 \\ 0 & 0 & t_4 \end{bmatrix}$	3	Same as above but with variable focal length.
"Semi-Rigid"	$T_{2D} = \begin{bmatrix} t_1 & t_2 & t_3 \\ -t_2 & t_1 & t_4 \\ 0 & 0 & t_5 \end{bmatrix}$	4	Same as above but with rotation and scaling along the image axes.
Affine Transformation	$T_{2D} = \begin{bmatrix} t_1 & t_2 & t_3 \\ t_4 & t_5 & t_6 \\ 0 & 0 & t_7 \end{bmatrix}$	6	Distant scene subtending a small field of view.
Projective Transformation	$T_{2D} = \begin{bmatrix} t_1 & t_2 & t_3 \\ t_4 & t_5 & t_6 \\ t_7 & t_8 & t_9 \end{bmatrix}$	8	Most general planar transformation.

Table 2.2: Description of the models used for image merging, ordered by the number of free parameters p.

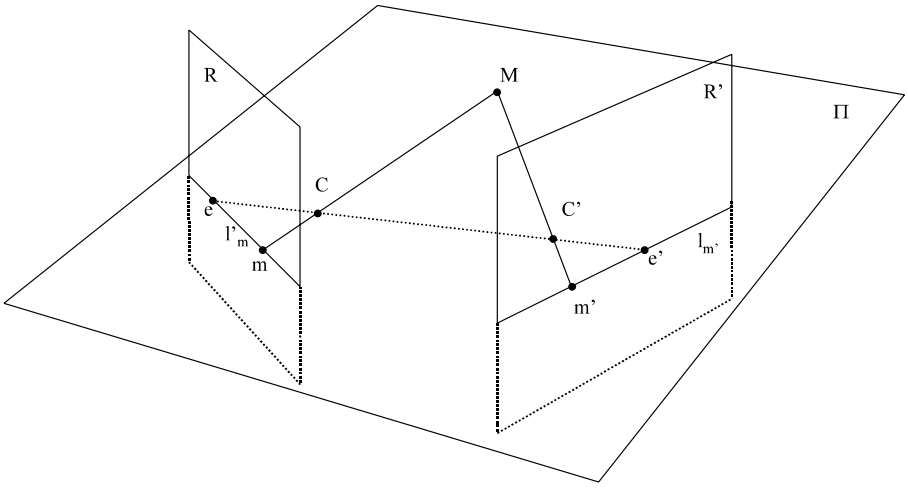


Figure 2.2: The epipolar geometry

2.4.1 Basic considerations on epipolar geometry

The epipolar geometry is the basic constraint which arises from the existence of two projective cameras. Consider the case of two images of a rigid scene, as depicted in Figure 2.2. Let M be a 3-D point projected on $\mathbf{m}$ and $\mathbf{m}'$ on the two retinal planes R and R', respectively. The optical centers for the cameras are the points C and C'. The line defined by two centers $\overline{CC'}$ intersects the retinal planes on $\mathbf{e}$ and $\mathbf{e}'$. These points are the projections of each of the optical centers on the other camera, and are called the *epipoles*. The *epipolar plane* Π is defined by the point M and the two optical centers. It intersects the image planes on the *epipolar lines* $l_{m'}$ and l'_m.

The epipolar geometry provides very useful information about a stereo rig: it relates the projections of the same 3-D point, since these projections are always bound to be on the corresponding epipolar lines. Therefore, an important application of epipolar geometry is in the search of point correspondences between images. For a given point projection on one image, one can reduce the search for its correspondence from 2-D to a 1-D. Instead of seeking on the other whole image, the search can be performed only on the corresponding epipolar line.

The epipolar geometry can be obtained by calibrating the two cameras. However, as we shall see on the next section, it can also be found just by the use of image information, provided that some correspondences are known.

2.4.2 The Fundamental Matrix

From now on let us consider image projective coordinates $(u, v, 1)$. As it is pointed out in [43], the relationship between a point $\mathbf{u}$ and its epipolar line $l'_{\mathbf{u}}$ is projective linear because

the relations between $\mathbf{u}$ and the 3-D line $\overline{UC}$, and the relation between $\overline{UC}$ and $l'_{\mathbf{u}}$ are both projective linear. In other words, the projective linearity comes from the fact that $l'_{\mathbf{u}}$ is the projection of the optical ray of $\mathbf{u}$ on the second image. This correspondence can be written has

$$l'_{\mathbf{u}} = F\mathbf{u} \tag{2.5}$$

where F is a (3×3) matrix called the *fundamental matrix*.

A great deal of research has been devoted to the study of fundamental matrix in the last few years[14, 43, 40, 10]. It can be shown[43, 49] that the matrix F is rank deficient and, for a non-degenerate case it has exactly rank 2. Since it is also defined up to scale, it has seven independent parameters. From Figure 2.2 one can check that all epipolar lines go through the epipoles. Therefore it is easy to see that the epipoles $\mathbf{e}$ and $\mathbf{e}'$ are the right and the left null spaces of F.

Let $\mathbf{u}'$ be the correspondence of $\mathbf{u}$. Since $\mathbf{u}'$ lies on $l'_{\mathbf{u}}$ we have $\mathbf{u}'^T\ l'_{\mathbf{u}} = 0$. This leads to the fundamental equation:

$$\mathbf{u}'^T F\mathbf{u} = 0 \tag{2.6}$$

In the case where the cameras intrinsic parameters are known the fundamental matrix reduces to the *essential matrix* E, given by the Longuet-Higgins equation[41]. This equation relates the normalized coordinates $\mathbf{m}$ and $\mathbf{m}'$ of the projections $\mathbf{u}$ and $\mathbf{u}'$ of a given 3-D point U, for some rotation R and translation $\mathbf{t}$ between the two camera frames, as

$$\mathbf{m}'(\mathbf{t}\times R\mathbf{m}) = \mathbf{m}'^T E\mathbf{m} = 0 \tag{2.7}$$

where $\times$ denotes the cross product of two vectors.

Let A_1 and A_2 be two non-singular matrices containing the intrinsic parameter matrices of two cameras, as described in 2.1. Then,

$$\begin{aligned} \mathbf{m} &= A_1^{-1}\mathbf{u} \\ \mathbf{m}' &= A_2^{-1}\mathbf{u}' \end{aligned}$$

The relation between the fundamental and the essential matrix can be easily derived:

$$\begin{aligned} \mathbf{m}'^T E\mathbf{m} = 0 \quad &\Leftrightarrow \quad \mathbf{u}'^T A_2^{-T} E\ A_1^{-1}\mathbf{u} = 0 \\ &\Rightarrow \quad F = A_2^{-T} E\ A_1^{-1} \end{aligned} \tag{2.8}$$

By the use of the first part of the Longuet-Higgins equation, F can also be written as

$$F = A_2^{-T}\,[\mathbf{t}]_\times\ R\ A_1^{-1} \tag{2.9}$$

where $[\mathbf{t}]_\times$ is a skew-symmetric matrix that performs a cross product of the vector $\mathbf{t}$ when left-multiplied by some vector $\mathbf{p}$, i.e., $[\mathbf{t}]_\times\,\mathbf{p} = \mathbf{t} \times \mathbf{p}$. This explicitly shows how F

is affected by camera translation and rotation. Therefore, it can be used for imposing additional constraints on the structure of F for particular camera setups, such as stereo rigs with known baseline. Examples can be found in [40, 11].

2.4.3 Estimation of the Fundamental Matrix

We shall now briefly address the problem of computing the fundamental matrix using image information, namely some point correspondences. A simple linear procedure, similar to the one presented above for the camera calibration will be outlined. A deeper explanation on this subject is given on chapter 5, where several methods for estimating F are presented and compared.

Given a set of image point correspondences, one can see from equation (2.6) that each pair of matched points between two images provides a singular linear constraint on the parameters of F. This allows linear estimation up to scale from 8 correspondences on a general configuration[2], by the use of the eight point algorithm introduced by Longuet-Higgins[41] and extensively studied in the literature[26, 43, 71, 6]. In practice using 8 correspondences proves to be extremely sensitive to noise. For more correspondences, a least-squares method follows. Each linear constraint on the parameters of F for a pair of correspondences, $\mathbf{u} = (u, v, 1)$ and $\mathbf{u}' = (u', v', 1)$, can be expressed as

$$\mathbf{h}^T \mathbf{f} = 0$$

where $\mathbf{h}^T = (uu', vu', u', uv', vv', v', u, v, 1)$ and $\mathbf{f}$ is a vector containing the elements of F such that $\mathbf{f} = (F_{11}, F_{12}, F_{13}, F_{21}, F_{22}, F_{23}, F_{31}, F_{32}, F_{33})$. We can now formulate the problem

$$\min_{\mathbf{f}} \|H\mathbf{f}\| \text{ constrained to } \|\mathbf{f}\| = 1$$

that can be solved using the SVD.

However, since F has 7 independent parameters, it can be estimated just from 7 pairs of corresponding points, by the use of a non-linear iterative method. Linear estimation has the major advantage of being non iterative, but it does not explicitly force the rank 2 condition of F. This condition has to be imposed afterwards. Otherwise, for experimental matches with localization inaccuracies, the epipolar lines will not all meet at a single point[43].

2.4.4 Reconstruction from calibrated cameras

We will now address the problem of recovering 3-D point locations. Consider the image points $\mathbf{u}$ and $\mathbf{u}'$ from the same object point U, obtained from cameras with projection

[2] In this thesis we will not investigate degenerate configurations for the F matrix. A discussion on this topic can be found on [77]. However it is worth stating that matches corresponding to 3-D points lying on the same epipolar plane impose the same redundant constraint on the elements of F.

matrices $P_1 = [M_1\ \mathbf{p}_1]$ and $P_2 = [M_2\ \mathbf{p}_2]$, respectively. The corresponding optical rays are given by equation 2.4:

$$\begin{aligned} \mathbf{x} &= \lambda M_1^{-1}\tilde{\mathbf{u}} - M_1^{-1}\mathbf{p}_1 \\ \mathbf{x}' &= \lambda' M_2^{-1}\tilde{\mathbf{u}}' - M_2^{-1}\mathbf{p}_2 \end{aligned} \tag{2.10}$$

The above equations impose 6 constraints on 5 unknowns: the 3 coordinates of U plus λ and λ'. The intended U point will be the intersection of the two rays. However, on practical situations, the rays may not intersect due to limited camera resolution and inaccuracies on the estimation of the camera matrices. Therefore, a vector $\widehat{U} = (\widehat{x}, \widehat{y}, \widehat{z})^T$ can be computed that minimizes the distance to the rays, corresponding to midway between the points of their closest approach.

2.4.5 Uncalibrated reconstruction

In many practical situations it is not possible to perform camera calibration, which enables straightforward reconstruction for given matched projections and camera matrices. However if some geometric information is available on the structure of the scene (such as knowledge of parallel lines, segment mid-points or 3-D point locations), then some *sort* of reconstruction can be accomplished. In order to clarify what is meant by sort of reconstruction we will briefly discuss an hierarchy of geometric groups for reconstruction and then describe a method for computing 3-D point locations from uncalibrated cameras.

We have seen in section 2.1 that there is a standard mapping from the usual Euclidean space to the projective space. The question now is, given only structure information on the projective space, what additional information is required in order to get back to Euclidean space. There are four main groups of transformations that define four different geometries. These are arranged hierarchically as

$$\text{Projective} \supset \text{Affine} \supset \text{Similarity} \supset \text{Euclidean}$$

in the sense that they can be considered strata overlaid one after another. The projective stratum is the most general and therefore the one with less geometric invariants[15]. In projective space there is no notion of rigidity, distance or points at infinity[3]. The affine space is obtained from the projective space by considering an arbitrary hyperplane as the hyperplane containing the points at infinity. The similarity space is invariant under the group of rigid motions (translation and rotation) and uniform scaling. It can be obtained from the affine space by establishing a conic in the hyperplane at infinity to be the *absolute conic*[16]. Under the similarity group there is no notion of scale or absolute distance. By fixing the scale, the usual Euclidean space is found.

[3] In the projective space, the usual Euclidean notion of a point at infinity corresponds to a point with the last coordinate equal to zero, and is treated in a similar way to any other point.

As it is shown in [48] the projective structure of a scene can be recovered just by establishing point correspondences over two images. As it would be expected, structure is recovered up to a projective transformation. In the work reported on this thesis, we recover the Euclidean structure by means of finding a perspective transformation G of $I\!P^3$, that linearly transforms the projective structure (obtained directly from the point correspondences) back into the Euclidean space.

We will now present a method for Euclidean reconstruction using point correspondences and known 3-D point locations *without explicit camera calibration*. The theory behind this method is supported by a lemma and a theorem by Hartley, whose proofs can be found in [23] and [24] respectively.

Lemma 1 *The fundamental matrix corresponding to the pair of camera matrices* $P_1 = [L_1 \quad -L_1\mathbf{o}_1]$ *and* $P_2 = [L_2 \quad -L_2\mathbf{o}_2]$ *is given by*

$$F \doteq L_2^* L_1^T \left[L_1 \left(\mathbf{o}_2 - \mathbf{o}_1\right)\right]_\times \tag{2.11}$$

where L_2^* *represents the adjoint of* L_2*. If* L_2 *is invertible then* $L_2^* \doteq L_2^{-T}$*.*

Theorem 1 *Let* $\{P_1, P_2\}$ *and* $\{P_1', P_2'\}$ *be two sets of camera projection matrices. Then* $\{P_1, P_2\}$ *and* $\{P_1', P_2'\}$ *correspond to the same fundamental matrix* F *if and only if there exists a* (4×4) *non-singular matrix* G*, such that* $P_1 G \doteq P_1'$ *and* $P_2 G \doteq P_2'$*.*

Let us consider now any two projection matrices P_1' and P_2' agreeing with an estimated F. Using P_1' and P_2', it is easy to compute the coordinates of 3D points x_i' from the image projections. In a general case these points differ from the original 3D points x_i, by the G transformation,

$$P_1' x_i' = P_1 G x_i' = P_1 x_i \quad \Rightarrow \quad x_i = G x_i' \tag{2.12}$$

The G matrix is a collineation in $I\!P^3$ performing a general perspective transformation [4]. Therefore it accounts for linear geometric rigid (rotation and translation) and non-rigid operations (scaling and skewing). According to proposition ??, it can be recovered from a set of 5 pairs of points $\{x_i, x_i'\}$ where x_i' are ground-truth points with known 3D coordinates. We can now present an Euclidean reconstruction procedure, based on the use of ground-truth points:

1. Estimate the fundamental matrix F from a set of matched points.
2. Determine some P_1' and P_2' agreeing with F.
3. Recover the projective 3D structure using P_1' and P_2'.
4. Estimate the G matrix by the use of ground points.

5. Apply G to the points determined in 3, to recover the Euclidean structure.

The problem remaining to be solved is to determine a pair of camera projection matrices agreeing with the estimated F. We will now present a lemma which can be used directly to obtain such pair. In [24, 25] a method is presented to recover two projection matrices in accordance to an estimated fundamental matrix, F. Although not stated clearly, there is the assumption that such cameras have normalized intrinsic parameters and thereby the matrix L, described in equation (2.11) is a rotation matrix. In this section we generalize these results for the case of arbitrary cameras.

Lemma 2 *Let F be a fundamental matrix with SVD decomposition $F = U diag(r, s, 0)V^T$, and $r \geq s > 0$. Let $P_1 = (L_1 \mid -L_1\mathbf{o}_1)$ be an arbitrary camera matrix with $\det(L_1) \neq 0$. Then the matrices,*

$$P_1 = (L_1 \mid -L_1\mathbf{o}_1) \qquad P_2 = (L_2 \mid -L_2\mathbf{o}_2)$$

correspond to the fundamental matrix F, where M_2 and T_2 are given by :

$$L_2 = [U diag(r, s, \gamma)EV^T]^{-T}L_1 \qquad \gamma \in \Re^+$$

$$\mathbf{o}_2 = L_1^{-1}V[0, 0, 1]^T + \mathbf{o}_1 \qquad E = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Proof.

It is known that the null space of the columns of F is a 3x1 vector corresponding to the projective coordinates of the right epipole. According to equation (2.11),

$$F \doteq L_2^{-T}L_1^T\,[L_1\,(\mathbf{o}_2 - \mathbf{o}_1)]_\times$$

$$\begin{aligned} F &\doteq U\mathrm{diag}(r, s, \gamma)EV^T L_1^{-T}L_1^T[L_1(L_1^{-1}V[0, 0, 1]^T + \mathbf{o}_1 - \mathbf{o}_1)]_\times \\ &\doteq U\mathrm{diag}(r, s, \gamma)EV^T[V[0, 0, 1]^T]_\times \end{aligned}$$

By construction of the SVD, the last column of V is a base vector for the null space of UDV^T, i.e., the null space of F, as required. The columns form a orthonormal basis spanning the 3D space. Bearing this is mind, it is easy to verify by inspection that

$$[V[0, 0, 1]^T]_\times = VZV^T \text{ with } Z = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Performing this replacement, we will have,

$$F \doteq U\mathrm{diag}(r, s, \gamma)EV^TVZV^T \doteq U\mathrm{diag}(r, s, 0)EV^T = SVD(F)$$

thus validating the lemma.

A simpler version of this lemma, on which this was based, is presented in [24] where L is a rotation matrix, hence simplifying the derivation process. Results on the implementation of the Euclidean reconstruction procedure are presented in chapter 5.

Chapter 3

Robust Motion Estimation

Robust parameter estimation is an essential part of many computer vision algorithms. The attempt of minimizing the effects of noise and uncertainty leads to the use of as much data as possible. Traditionally the combination of the data is accomplished under the least-squares framework. However, real world applications require the awareness to data in gross disagreement with the assumed model. The main drawback of the least-squares framework is that these *outliers* can make arbitrarily the result of the estimation process.

The organization of this chapter is as follows. Section 3.1 briefly presents the two main approaches used in the literature for the computation of motion from image sequences. These are the *feature-based* approach and the *optical flow* approach. The comparative advantages and disadvantages are outlined. Since this thesis evolves entirely around feature-based techniques, which can be used both for mosaicing and reconstruction under the same framework, the rest of the chapter will deal only with this class of methods. In section 3.2 we review some of the most commonly used model estimation techniques, including unconstrained optimization and robust estimation. An introductory explanation is given for orthogonal and re-weighted least squares, M-estimators, case deletion diagnostics, and random sampling techniques. This section ends with the proposal of a new algorithm based on the least median of squares which is extensively used in this thesis.

3.1 Motion Estimation Techniques

There are two main approaches to the problem of motion estimation from multiple images. The first *feature-based* involves the extraction of a set of features from the sequence, such as image corners, line segments or curves. These features are usually sparse, when compared with the extent of the underlying images. After the feature extraction process, this approach requires establishing correspondences between features over the images. One of the drawbacks of this approach, as pointed out by Ayer[3], lies on the difficulty of

the matching process which is prone to gross error. However, recent progress on robust methods applied to geometric constraints such as the fundamental matrix[66], has been able to lighten this difficulty.

The second method, commonly referred to as the *optical flow* approach [32, 30, 5], is based on the computation of the velocity field of brightness patterns in the image plane. As opposed to the feature-based approach, the optical flow does not require a matching process, but suffers from the *generalized aperture problem*[3, 8, 35]. According to Black and Anandan[8], most of the current techniques for the estimation of the optical flow are based on two image motion constraints: *data conservation* and *spatial coherence*. The first arises from the observation that the intensity patterns of the surfaces of the world objects remain constant over time, although their image position may change. The second assumes that the surfaces have spatial extent, thus making neighboring pixels likely to belong to the same surface. This is usually implemented in the form of a *smoothness constraint* on the motion of spatially close pixels.

The generalized aperture problem refers to the dilemma of choosing the appropriate size for the area of analysis (aperture) R. In order for the motion estimation to present some insensivity to noise and be constrained[8], a large R is desirable. However, the larger the aperture is, the less realistic the data conservation and the spatial coherence become.

One other problem with optical flow techniques lies on the fact that it is only possible to determine the flow in the direction of the image brightness gradient. The optical flow along this direction is therefore perpendicular to the image contour, hence called *normal flow*. The flow component along the contour cannot be established directly from the brightness patterns, without resorting to additional constraints such as smoothness or second order derivatives. This condition is referred to as the *aperture problem* [32, 30, 5].

For some applications, the choice of the approach, either feature-based or optical flow, is not trivial. This statement is supported by the large amount of research in the last few years using the two approaches as a starting point for higher level image interpretation. Optical flow has successfully been used on tasks such as egomotion estimation[58, 57, 59], motion segmentation[3] and image registration[54, 64], whereas feature-based approaches have proven adequate for 3-D reconstruction[17, 44] and image registration as well[76].

3.2 Robust Model Estimation

In this section we will present some established methods for model-based estimation with emphasis on robust techniques. The methods are drawn from the principal categories of estimators: M-estimators, case-deletion methods and random sampling. The application and comparison of these methods in the context of the fundamental matrix estimation has been investigated by Torr *et al.* in [66], where an in-depth analysis and discussion can be

found.

Model estimation (in the sense of model fitting to noisy data) is employed in computer vision on a large variety of tasks. The most commonly used method is the least-squares mainly due to the ease of implementation and fast computation. The least-squares is optimal when the underlying error distribution of the data is Gaussian[47]. However, in many applications the data are not only noisy, but it also contains *outliers*, *i.e.* data in gross disagreement with the assumed model. Under a least-square framework, outliers can distort the fitting process to the point of making the fitted parameter arbitrary. As pointed out in [66], this can be particularly severe if the non-outlying data are degenerate or near-degenerate with respect to the expected model. In such case outliers can mask the degeneracy, making it hard to evaluate the adequacy of the postulated model.

According to Meer *et al.*[47], there are three concepts usually employed in evaluation a robust regression method: the relative efficiency, the breakdown point and the time complexity. The *relative efficiency* is defined as the ratio between the lowest achievable variance for the estimated parameters[1] and the actual variance provided by given method. The *breakdown point* is the smallest proportion of outliers that can force the estimated parameters outside an arbitrary range. For the least-squares estimation, the breakdown point is 0 since just one outlier is required for corrupting the estimated values. The *time complexity* can be defined from the number of arithmetic operations performed by the algorithm.

3.2.1 Orthogonal and Iterative Re-weighted Least-Squares

In Section 2.4.3, a simple least-squares procedure was presented, for the estimation of the fundamental matrix. The singular value decomposition is used for solving a set of equations for the elements of the F matrix,

$$\min_{\mathbf{f}} \|H\mathbf{f}\| \text{ constrained to } \|\mathbf{f}\| = 1$$

One can consider this problem as fitting an hyperplane

$$\mathbf{f} = (F_{11}, F_{12}, F_{13}, F_{21}, F_{22}, F_{23}, F_{31}, F_{32}, F_{33})$$

to a set of n points $\mathbf{h}_i \in I\!R^9$, where $\mathbf{h}_i^T = (u_i u'_i, v_i u'_i, u'_i, u_i v'_i, v_i v'_i, v'_i, u_i, v_i, 1)$ is a vector containing the coordinates of a pair of matched points $\mathbf{u}_i=(u_i, v_i, 1)$ and $\mathbf{u}'_i = (u'_i, v'_i, 1)$. The hyperplane can be approximated by the minimum of the sum of the Euclidean distances of the points to the plane[52],

$$\min_{\mathbf{f}} \sum_{i=1}^{n} \left(\mathbf{h}_i^T \mathbf{f}\right)^2 = \min_{\mathbf{f}} \sum_{i=1}^{n} \mathbf{r}_i^2 \text{ constrained to } \|\mathbf{f}\| = 1 \tag{3.1}$$

[1] This can be given by the Cramer-Rao bound[39], although there are other lower bounds.

The distances are taken perpendicularly to the plane. Therefore, this method is called *orthogonal least squares*[52]. If the noise on all elements of $\mathbf{h}_i$ is Gaussian, independent and has equal variance, then the equation (3.1) represents a maximum likelihood estimator (MLE) for the hyperplane. Since this is not the case (due to the structure of $\mathbf{h}_i$) this estimator is an approximation of the MLE.

Bearing in mind the way the fundamental matrix relates the point correspondences, given by equation (2.6), it has been suggested in [66, 43] that we should consider its estimation as fitting the quadric surface represented by F to features on the 4-dimensional image coordinate space[2], rather than an 9-dimensional hyperplane fitting. In fact, the orthogonal least squares will produce a sub-optimal estimate of F, because the residuals $\mathbf{r}_i$ do not follow a Gaussian distribution, even if the image points do. These residuals are usually referred as the *algebraic* distances, as opposed to the *geometric* distances measured perpendicularly to the quadric surface in the image coordinate space.

As discussed in [36], it has been shown that the maximum likelihood quadratic curve is the one that minimizes the sum of the squares of the geometric distances. However there is no closed form solution for a general case, due to the fact that the perpendicular distance segments are not unique nor parallel. Weng *et al.*[73, 72] have adopt a first order approximation to the distance for the computation of the fundamental matrix, in the form

$$\mathbf{f}_{min} = \min_{\mathbf{f}} \sum_{i=1}^{n} \left(w_{S_i} \mathbf{h}_i^T \mathbf{f} \right)^2$$

This method requires the computation of the weights w_{S_i} corresponding to the inverse of the first order approximation of the standard deviation of the residual. An iterative procedure is thus required for the estimation of both $\mathbf{f}$ and w_{S_i}. Slightly modified versions of this method have been proposed by Kanatani[37] and Torr *et al.*[66].

A different iterative method was put forward by Luong *et al.*[42]. Instead of the weighted residuals, the error criterion to be minimized is the distance of each image point to the corresponding epipolar line. Unlike the previous method, the points on the two images do not play a symmetric role under this criterion. Therefore the minimization must account for the distances on both images for each pair of correspondences. The criterion is

$$\min_{F} \sum_{i=1}^{n} \left(d^2 \left(\mathbf{u}_i', F\mathbf{u}_i \right) + d^2 \left(\mathbf{u}_i, F^T \mathbf{u}_i' \right) \right) \tag{3.2}$$

where $d(\cdot, \cdot)$ represents the point to line orthogonal distance. It is assumed that point localization errors on the image have a Gaussian distribution. As noted in [66], this

[2] also called the *cyclopean retina*[42].

minimization is equivalent to the minimization of the algebraic residuals weighted by

$$w_{E_i} = \left(\frac{1}{r_{u_i}^2 + r_{v_i}^2} + \frac{1}{r_{u_i'}^2 + r_{v_i'}^2} \right)^2$$

being $r_{u_i}, r_{v_i}, r_{u_i'}, r_{v_i'}$ the corresponding partial derivatives of r_i.

This epipolar distance method will be used on chapter 5, blended with a parameterization for F using 7 independent parameters, and yielding good results. A comparison with the orthogonal least squares is also presented. A more extensive comparison covering the three methods, is given in [65] where it is shown that the two iterative methods have similar performance, and are superior to the orthogonal, unweighted. It is worth noticing that these methods make assumptions on the error distributions and perform poorly in the presence of outliers.

3.2.2 M-Estimators and Case Deletion

The M-estimators are a class of robust methods with generalized use. In computer vision, M-estimators and have been used by Olsen[50] and Deriche *et al.*[11] for epipolar geometry estimation, and by Bober and Kittler[9] and Black and Anandan[8] for robust estimation of the optical flow. These estimators minimize

$$\sum_{i=1}^{n} \rho\left(\mathbf{r}_i\right) \tag{3.3}$$

where $\rho\left(\mathbf{r}_i\right)$ is a positive-definite function of the residuals [47] with a single minimum at $\mathbf{r}_i = 0$. Several of these functions have been proposed in the literature in order to reduce the influence of large residual in the estimated fit, such as square error for small residual and linear for large residuals. Equation (3.3) reduces to the usual least-squares if we consider $\rho\left(\mathbf{r}_i\right) = \mathbf{r}_i^2$. Although the M-estimators are robust for various error distributions, these methods present a breakdown point of less than $1/\left(p-1\right)$ where p is the number of parameters to be estimated[47]. The M-estimators perform poorly in the presence of outliers, when compared with the sampling techniques presented on the next subsection which attain a breakdown point of 0.5.

Case deletion methods are based on the analysis of the effects of removing data in the estimation process [65, 66, 47]. More specifically, these methods aim at identifying and removing the data points whose influence on the estimation suggest them to be outliers. We will not give a comprehensive explanation on this category of methods, which can be found in [66], but just present some important topics for the sake of completeness of this chapter. By extending the work of Cook and Weisberg, Torr[66] has derived a formula for the *influence* of a data point in the case of orthogonal regression and used it for the F matrix estimation. This influence is a scalar measure $T_i(L)$ in the form

$$T_i(L) = \left(\mathbf{f}_i - \mathbf{f}\right)^T L \left(\mathbf{f}_i - \mathbf{f}\right) \tag{3.4}$$

where L is the positive definite, symmetric $(p \times p)$ *moment matrix* $L = H^T H$, and $\mathbf{f}$, $\mathbf{f}_i$ are the least-squares estimates of the parameter vector, taken with all data points and with all data points except the ith element, respectively. It is shown that T_i can be written as the product of the squared residual of the ith data point by a *leverage factor* l_i, scaled by the variance σ^2 of the data noise, assumed Gaussian.

$$T_i(L) = \frac{\mathbf{r}_i^2}{\sigma^2}\, l_i \tag{3.5}$$

As reported by Torr, the leverage factor is a measure of the effect of each data point, being large for outliers *even if the outliers residuals are small*. The high leverage points should be removed and the parameters re-estimated iteratively, until the data falls bellow a χ^2 threshold determined by σ. Experimental results[66] indicate a superior convergence and accuracy of case deletion diagnostics when compared to the M-estimators, due to the fact that the latter use all the data points in the estimation whereas the former completely discard some outliers. The main disadvantage of case deletion methods is the sensitivity to the variance σ^2 assumed, as they required a good estimation of σ^2.

3.2.3 Random Sampling Algorithms

A non-linear minimization method was proposed by Rousseeuw, called the *least-median-of-squares* [52] (LMedS). The parameters are estimated by solving

$$\min_i \operatorname{med} \mathbf{r}_i^2$$

As pointed out in [47], this minimization problem cannot be reduced to a least-squares based solution, unlike the M-estimators. The minimization on the space of all possible solutions is usually impracticable. As an example, if the LMedS is used for the estimation of the fundamental matrix, and 8 correspondence sets are used for model instantiantion from a population of 100, the number of combinations is

$$\frac{100!}{92!\,8!} \approx 1.861 \times 10^{11}$$

which is too large. Therefore it is common practice to use a Monte Carlo technique and analyze only randomly sampled subsets of points. The number of samples to be performed may be chosen as to insure a high probability[3] of selecting an outlier-free subset. The expression for this probability P_f is

$$P_f = 1 - (1 - (1 - \varepsilon)^p)^m \tag{3.6}$$

for m samples of size p, taken from a data set where the fraction of outliers is ε. From this expression, it can easily be seen that the number of samples is not directly linked to

[3] tipically 95%

the absolute number of outliers, but just with its proportion. Clearly, this is also true for the time complexity of the sampling algorithm. The expression also implies that the less data point are used for instantiating the model, the less samples will be required for the same P_f. The random sampling greatly reduces the time complexity of the basic LMedS, from $O(n^{p+1} \log n)$ to $O(nm \log n)$[47], while keeping the breakdown point of 0.5. In spite of the high breakdown point, the relative efficiency of the LMedS method is low when Gaussian noise is present in addition to outliers. Therefore an association of LMedS with weighted least-squares which has high Gaussian efficiency, can be used, as proposed by Rousseeuw[52].

Another robust estimator, based on random sampling, is the *Random Sampling Consensus* (RANSAC). It was proposed by Fishler and Bolles[18] in 1981, and originally used in the context of computer vision, for automated cartography. The RANSAC is based on the following paradigm. The estimation is performed on a subset of data points sampled from all the available points, such that the subset has the minimum number of elements required for instantiating the model. All the data is then evaluated according to the instantiated model. For a given error threshold, the points are classified as being part of the consensus group of the model if they are within the threshold. This process is repeated until a sufficiently large consensus group is found (or eventually a maximum number of iterations is reached). The final estimation is performed on the largest consensus group found. The RANSAC requires therefore, the specification of three algorithm parameters: the error threshold for evaluating the compatibility with the model, an estimate of the cardinality consensus set for checking if a sufficiently supported model has been found, and a maximum number of samples to try.

Although developed independently, LMedS and RANSAC are based on similar concepts. According to Meer *et al.*[47], the main difference lies on the fact that the LMedS generates the error measure during estimation, while RANSAC requires it beforehand. An in depth comparison of LMedS and RANSAC can be found in [47]. The two methods are also compared in [66], for the estimation of the fundamental matrix.

3.2.4 A Two-Step Variant of LMedS

Several variants to these random sampling algorithms have been proposed in the literature, in the context of Computer Vision. As an example we can point out an "empirically optimal algorithm" presented in [66], which combines LMedS and M-estimators.

In the work presented on this thesis, we have used a two-step variant of LMedS, which we will refer to as MEDSERE[4]. It exhibits a similar breakdown point but requires less random sampling in order to achieve the same degree of outlier rejection. Up to this point, no extensive testing has been performed for evaluating its relative efficiency or

[4]MEDSERE stands for MEDian SEt REduction.

precise breakdown point. However, the results presented on chapter 5 testify its good performance, when compared to LMedS.

The MEDSERE algorithm comprises two phases of random sampling LMedS. After the first phase, the data set is reduced by selecting the best data points in the sense of the chosen cost function. Next, the reduced data undergoes another random sampling LMedS phase. For the computation of the fundamental matrix the algorithm is illustrated by the following operations :

1. Randomly sample the complete set of matched points S_{total} for a set of p pairs.
2. Estimate the F matrix and compute the median of the point-to-epipolar line distance for S_{total},
$$\underset{i}{\text{med}} \left(d^2 \left(\mathbf{u}_i', F\,\mathbf{u}_i \right) + d^2 \left(\mathbf{u}_i, F^T \mathbf{u}_i' \right) \right)$$
where $d(\cdot,\cdot)$ is the orthogonal distance. If the median is below a given threshold d_T, return F and exit.
3. Repeat 1. and 2. for a specified number of samples m_1.
4. Select the F matrix for which the minimal median was found, and sort the matched points by their point- to-epipolar line distance, using F.
5. Create the set S_{best} with the elements of S_{total} whose distance is below the median.
6. Repeat 1. and 2. on S_{best} for a m_2 number of samples.
7. Return the minimal median matrix found.

The required parameters are the number of samplings on each part m_1 and m_2, and the median threshold. Since the first two directly determine the number of operations, they can be defined by processing time constraints.

Chapter 4

Application to Video Mosaics

In this chapter we deal with the problem of creating mosaics from a sequence of video images. The creation of video mosaics is accomplished in two stages: registration and rendering. On the registration stage we estimate the parameters of point correspondence between frames, then fit individual frames to a global model of the sequence. The rendering stage deals with the creation of a single mosaic, by applying a temporal operator over the registered and aligned images.

This chapter is organized as follows. Section 4.1 describes the selection of features, namely image points with high intensity variation in several directions. These points usually correspond to object corners or highly textured surfaces, and allow efficient correspondence finding. The matching procedure is presented on section 4.2. Section 4.3 describes the estimation of the image registration parameters. The following section deals with image merging, namely creating a single mosaic from a collection of images. Finally, section 4.5 presents several different examples of video mosaics along with a discussion on their applications.

4.1 Feature selection

The work presented on this thesis evolves around the analysis of point projections and their correspondence between image frames. In order to improve the correspondence finding, a number of points are selected corresponding to image corners or highly textured patches.

Image edges are defined in the literature as sets of contiguous points of high intensity variation. Corner pixels can be defined as the junction points of two or more edge lines. Corners are second order entities of a surface, and can be detected by using second order derivatives. However, differentiation is a very noise sensitive operation. A simple example illustrating noise amplification is given in [15], which points the fact that differentiation is an *ill-posed problem*. An ill-posed problem [7] refers to the condition of non existence

of solution, or non-uniqueness, or the solution not depending continuously on the data. This last condition affects the robustness to noise and numerical stability of the solution. The methods for converting ill-posed problems into well-posed ones, usually resort to regularization, imposing additional constraints, namely on the smoothness of the data.

The selection of image points is based on a simplified version of the well-known corner detector proposed by Harris and Stephens[22]. This detector finds corners in step edges by using only first order image derivative approximations. Regularization is performed by smoothing the image using a 2-D Gaussian filter, before edge extraction. A trade off condition is therefore created by the filter mask size. Large masks promote better noise exclusion but reduce the accuracy of the corner localization. Conversely, small masks achieve higher accuracy at the cost of higher noise sensitivity.

Let $I(u, v)$ be the image intensity level at point $\mathbf{u} = (u, v)$. The autocorrelation function may be defined as

$$R_I(\delta_u, \delta_v) = \sum_{u,v} \left(I\left(u + \delta_u, v + \delta_v\right) - I\left(u, v\right)\right)^2$$

The first order Taylor series expansion will be

$$R_I(\delta_u, \delta_v) = \left[\delta_u, \delta_v\right] G\left(I_u, I_v\right) \left[\delta_u, \delta_v\right]^T$$

where I_u and I_v represent the first order derivatives of the intensity function and

$$G\left(I_u, I_v\right) = \begin{bmatrix} I_u^2 & I_u I_v \\ I_u I_v & I_v^2 \end{bmatrix}$$

The eigenvalues of matrix G are useful in corner detection. If an image point has high intensity variation on adjacent pixels in all directions, then the eigenvalues of matrix G will be both large. For this detector, the smallest eigenvalue is usually used as an indicator of a point "cornerness".

The difference between the Harris corner detector and the one implemented, lies on the fact that no Gaussian filtering is performed on the implemented one. Instead, the regularization is achieved by using the Sobel operator[15] when computing the first order derivative approximations, and by computing the G matrix over an image area W such that,

$$G\left(I_u, I_v\right) = \begin{bmatrix} \sum\limits_{(u,v)\in W} I_u^2 & \sum\limits_{(u,v)\in W} I_u I_v \\ \sum\limits_{(u,v)\in W} I_u I_v & \sum\limits_{(u,v)\in W} I_v^2 \end{bmatrix}$$

The Sobel operator is implemented by convolving the image with two (3×3) masks, one for each image axis direction. The derivative estimation is performed taking into account the intensity values of the central point plus its 8-neighborhood. Figure 4.1 illustrates the

Figure 4.1: Feature Selection: original image (left), sum of the squared derivatives (center), smallest eigenvalue of G with features marked (right)

feature extraction process. The middle image was obtained from the original on the left, by summing the squared derivatives for each point. On the right, one can see the image of the smallest eigenvalues of G, for an area patch W of (3×3). The intensity peaks of this 'texture' image correspond to image corners or small highly textured surfaces, and are selected as point features.

The extracted features will be matched over two images, and used for motion estimation. Since motion estimation is more noise sensitive to location errors when the features are close to each other, it is convenient to select features not just on the basis of the smallest eigenvalues of G, but also using some inter-feature distance criterion. Bearing this in mind, the implemented algorithm selects the features by finding the peaks of the 'texture' image and excluding the subsequent selection on a circular neighborhood. This process is repeated iteratively, up to the point where no peaks above a defined threshold can be found.

A comparative study on some of the most used grey level corner detectors can be found in [13].

4.2 Matching

The first step towards the estimation of the image registration parameters, consists of finding point correspondences between images. This is referred to as the matching problem, which is considered a challenging task due to its difficulty. Contributing factors to this difficulty include the lack of image texture, object occlusion and acquisition noise, which are frequent in real imaging applications. Several matching algorithms have been proposed over the last two decades, usually based on correlation techniques or dynamic programming. For a comparative analysis of stereo matching algorithms dealing with pair of images, refer to [2].

In this work, a correlation-based matching procedure was implemented. It takes a list of features selected from the first image I_1, and tries to find the best match for each, over a second image I_2. The cost criterium, that drives the search on the second image, is known in the literature as the *sum of squared differences* (SSD) [1]. For a given feature $\mathbf{f}_i = (u_i, v_i)$, it is defined as

$$SSD(x, y) = \sum_{(u,v) \in W_i} \left[I_1(u, v) - I_2(u - x, v - y) \right]^2$$

where W_i is an image patch around $\mathbf{f}_i$.

Underlying the use of the SSD is the hypothesis of image brightness constancy, where one assumes that the brightness patterns around the features do not change significantly between images. This assumption is plausible if there is no significant object pose or light condition changes, which is often the case for time consecutive frames on sequences acquired with short time intervals.

The SSD defines a mismatch energy, whose minimum has a high probability of being the true match. To achieve the minimum, it is sufficient to maximize the cross-correlation,

$$c(x, y) = \sum_{(u,v) \in W_i} I_1(u, v)\, I_2(u - x, v - y) \tag{4.1}$$

where W_i is a rectangular area around each feature $\mathbf{f}_i = (u_i, v_i)$.

Using the Cauchy-Schwartz inequality, it can easily be shown [34] that $c(x, y)$ attains the maximum when the feature patch coincides with some area on the second image. The implemented matching procedure computes $c(x, y)$ directly from the previous equation, and searches for its peak. The peak coordinates are taken to be the *displacement vector*, which relates the coordinates of the same feature on both images. An alternative to the direct implementation of equation (4.1) is to compute the cross-correlation by means of the Fourier transform. However this second method is only computationally less expensive if the areas being correlated are both large, as pointed out in [34].

The assumption of large overlap of image contents between the two frames can be used to significantly reduce the computational burden of the matching. This is achieved by limiting the search area in I_2. In order to compute the appropriate limits, the two images are cross-correlated and a global displacement vector $\mathbf{d}_G$ is obtained. By applying a threshold to the cross-correlation image, we can estimate a bounding box around $\mathbf{d}_G$, that can be loosely interpreted as a confidence area for the global displacement. Then, for a given feature $\mathbf{f}_i$ the search area on I_2 is constrained to the rectangular area with the size of the bounding box and centered on $\mathbf{f}_i + \mathbf{d}_G$. Figure 4.2 illustrates the procedure.

It is worth noting that image motions that cannot be described by simple translations cause the features to be warped. This warping can severely degrade the performance of the correlation matching if large rotation, zooming or perspective effects are present. A method for dealing with this limitation is now presented.

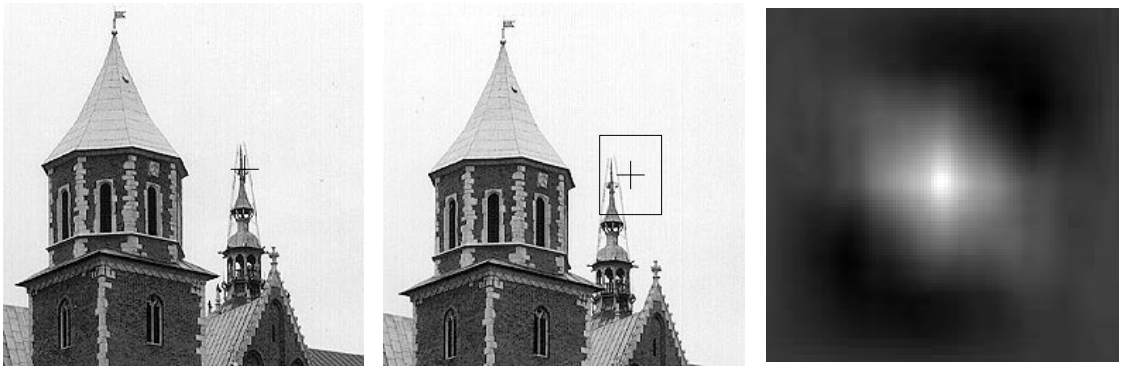


Figure 4.2: Search area selection: image I_1(left) with selected feature, search area on I_2(center) and cross-correlation image(right)

4.2.1 Sub-pixel accuracy

In order to improve the accuracy of the matching procedure, an additional refinement method was implemented. The localization of the feature correspondences can be computed with subpixel resolution by one of the following two methods. Let $\mathbf{c}_{max}$ be the (integer) coordinates of the maximum of $c(x, y)$. Firstly, the peak location of the cross-correlation can be re-estimated by fitting a parametric surface, usually a paraboloid or a cubic curve, to the neighborhood of $\mathbf{c}_{max}$. The improved peak location is obtained directly from the surface parameters. Secondly, one can use an optical flow technique, as discussed in section 3.1, applied the patches around each feature. We have opted for the use of optical flow estimation, for reasons that will become clear after the following explanation.

We will use a notation close to the one presented in [9]. Let us define the transformed pixel difference as

$$\epsilon\left(\overrightarrow{a}, \mathbf{u}\right) = I_1\left(\mathbf{u}\right) - I_2\left(\mathbf{u} + \mathbf{d}_u\left(\overrightarrow{a}\right)\right)$$

where $I_1\left(\mathbf{u}\right)$ and $I_2\left(\mathbf{u} + \mathbf{d}_u\left(\overrightarrow{a}\right)\right)$ are the image intensities at pixel locations $\mathbf{u}$ and $\mathbf{u} + \mathbf{d}_u\left(\overrightarrow{a}\right)$, and $\mathbf{d}_u\left(\overrightarrow{a}\right)$ is the displacement vector defined by the motion parameter vector $\overrightarrow{a}$. Several motion models can be considered for the displacement, ranging from a simple translation (two parameters) to a general projective transformation (eight parameters). We have implemented a four parameter model that accounts for translation, rotation and independent scaling along the image axes, defined by

$$\mathbf{d}_u\left(\overrightarrow{a}\right) = (a_1 u + a_2 v + a_3, -a_2 u + a_1 v + a_4)$$

This model is also used in [9], and was found to be a good compromise between matching accuracy and convergence speed. The cost functional to be minimized is the sum of the

squared pixel difference over the feature patch W,

$$H\left(W, \overrightarrow{a}\right) = \sum_{\mathbf{u} \in W} \left(\epsilon\left(\overrightarrow{a}, \mathbf{u}\right)\right)^2$$

This function is well behaved on the vicinity of the minimum [9], and therefore a steepest descent minimization method can be applied. The starting value for $\overrightarrow{a}$ is set to correspond to pure translation given by the result of the correlation based matching procedure described above, *i.e.*,

$$\overrightarrow{a_0} = (a_1, a_2, a_3, a_4) = (1, 0, x, y)$$

As a stopping condition, the implemented method uses a criterion based on the norm of the gradient of H and a maximum number of iterations. The resulting values for translation components are taken to be the location of the feature correspondence. The main advantage of using this method lies on the fact that feature rotation is accounted, thus enabling a more accurate matching, at the expense of the use of an iterative method.

4.3 Motion Parameter Estimation

In this section we will describe a procedure for the estimation of the motion parameters for a sequence of images. The images are processed as shown on the diagram of Figure 4.3. For each image I_k, a set of features is extracted and matched directly on the following image I_{k+1}, as described in the previous sections. The result of the matching process are two lists of coordinates of corresponding points. Due to the error prone nature of the matching process, it is likely that a number of point correspondences will not relate to the same 3-D point. For this reason, the next subsection is devoted to the robust estimation of the motion parameters taking into account the existence of mismatches.

4.3.1 Frame-to-frame motion estimation

The first step in image registration is to find the motion parameters for the image motion, between consecutive frames. On the work reported on this thesis, no automatic selection for the motion model is performed. The most appropriate model is assumed to be known. On the following subsections, the most general planar tranformation model (performing a collineation between planes), will be considered. For the restricted motion models presented on Table 2.2 the following considerations and equations are also valid.

Let ${}^{(k)}\mathbf{u}$ be a point on frame k, and ${}^{(k+1)}\mathbf{u}$ be its correspondence on frame $k+1$. If $T_{k,k+1}$ is the planar transformation matrix relating the frames k and $k+1$, then

$${}^{(k)}\tilde{\mathbf{u}} = T_{k,k+1} \; {}^{(k+1)}\tilde{\mathbf{u}}$$

and

$${}^{(k+1)}\tilde{\mathbf{u}} = T_{k,k+1}^{-1} \; {}^{(k)}\tilde{\mathbf{u}}$$

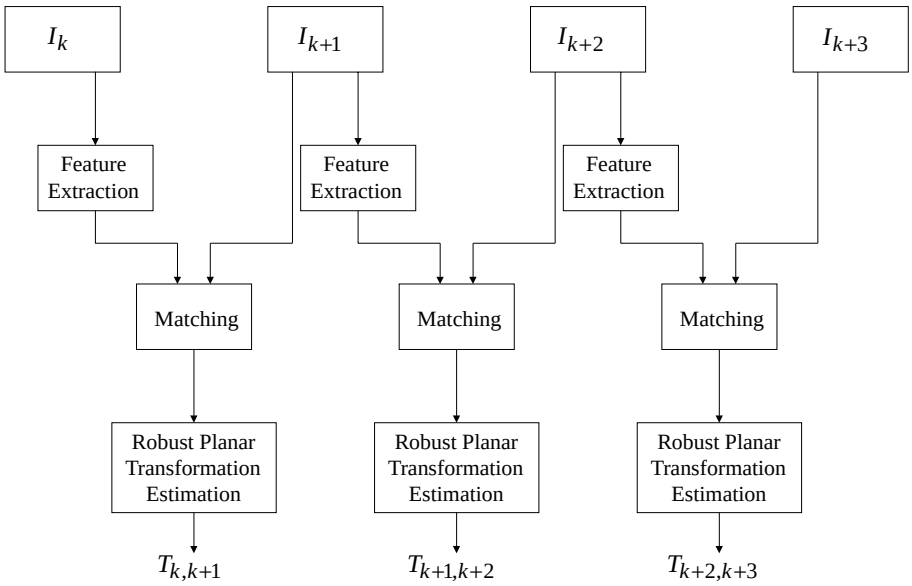


Figure 4.3: Block diagram of the sequence of operations on the images I_k for the motion parameter estimation. The output is the set of planar transformation matrices $T_{k,k+1}$.

A robust estimation method is required for the estimation of $T_{k,k+1}$. For this, the MEDSERE algorithm was used, but with a different criterion from the one presented in section 3.2.4. Let ${}^{(k)}\mathbf{u}_i$ be the location of the i^{th} feature extracted from image I_k, and matched with ${}^{(k+1)}\mathbf{u}$ on image I_{k+1}. The criterion to be minimized is the median of sum of the square distances,

$$\underset{i}{\text{med}} \left(d^2 \left({}^{(k)}\mathbf{u}_i, T_{k,k+1} \; {}^{(k+1)}\mathbf{u}_i \right) + d^2 \left({}^{(k+1)}\mathbf{u}_i, T_{k,k+1}^{-1} \; {}^{(k)}\mathbf{u} \right) \right) \tag{4.2}$$

where $d(\cdot, \cdot)$ stands for the point-to-point Euclidean distance.

Relating the frames of the image sequence

The transformations between non-contiguous frames can be computed by sequentially multiplying the transformation matrices of the in-between frames. The planar transformation relating the frame k and l, such that $k < l$ is

$$T_{k,l} = \prod_{i=k}^{l-1} T_{i,i+1}$$

4.3.2 Global registration

After estimating the frame-to-frame motion parameters, these parameters are cascaded together to form a global model. The global model takes the form of a global registration, where all frames are mapped into a common, arbitrarily chosen, reference frame. With

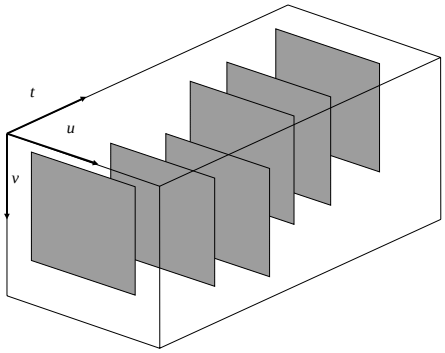


Figure 4.4: Three-dimensional space-time volume formed by the globally aligned frames.

it, each individual image frame is mapped to the same frame as any of the others. Let $T_{Ref,1}$ be the transformation matrix relating the frames of the chosen reference and the first image frame. The global registration is defined by the set of transformation matrices $\{T_{Ref,k} : k = 1 \ldots N\}$, where for $2 \leq k \leq N$,

$$T_{Ref,k} = T_{Ref,1} \prod_{i=1}^{k-1} T_{i,i+1}$$

The globally aligned frames can be considered to form a three-dimensional space-time continuum, as depicted on Figure 4.4. The spacial dimensions are the image axes u and v, and the time evolves along the t axis. The main particularity of this volume is that if the images are captured so that there is no parallax, then a vector perpendicular to the image planes will correspond to the same world point on each image.

4.4 Mosaic Rendering

After global registration, the following step consists in merging the images. As pointed in [46], there are several issues to consider in the mosaic rendering such as the choice of the reference frame, which frames to be used, the temporal operator to be applied and how moving objects on the scene will be handled. We will now briefly discuss these issues and present examples illustrating the effects of some of the choices.

As referred above, the global registration establishes the mappings between each frame and an arbitrary frame. For the creation of the mosaic an absolute frame has to be chosen, to which the images will be mapped. The choice of the reference frame can drastically affect the appearance of the resulting mosaic. In most of the mosaics presented on this thesis, the reference frame is the one of the first image. However, on some applications a useful reference frame can be set, which may not correspond to any of the images. Figure 4.5 depicts two mosaics created from a soccer game sequence of 43 images, using the temporal median filtering, that will be explained further on. The top image was rendered

using the first frame as the reference. On the lower, a reference frame was selected which allows a better perception of the playing field. This frame was set by computing the planar transformation that relates the four corners of the goal area to a vertically aligned rectangle whose sides are proportional to the ones of a real playing field.

Some applications such as real-time mosaicing impose important constraints on the computational cost of mosaic creation. Therefore, some of the acquired images may be discarded before the mosaic rendering process. The discarding criteria is clearly application-dependent. For instances, if the mosaic being created depicts an aerial scene captured by a moving plane, then the frames can be selected such that the amount of image overlap is kept roughly constant. Furthermore, if the frame-to-frame motion estimation is performed while the images are being captured, then the selection process can be done *on the fly*, thus resulting in memory saving.

It is to expect that some of the selected frames will overlap. On ovelapping regions there are more multiple contributions for a single point on the output image, and some method has to be established in order to determine the unique intensity value that will be used. As we have seen, the contributions for the same output point can be thought of as lying on a line which is parallel to the time axis, in the space-time continuum of Figure 4.4. Therefore, the referred method operates on the time domain, and is thus called a *temporal operator*. Some of the commonly used methods are the use-first, use-last, mean and median. The first two use only a single value from the contributions vector, respectively the first and the last entries of the timely ordered vector. Intuitively, the rendering of a mosaic using the use-last method can thought of as placing the frames on top of each other in the order that the images were captured. Each point of the final mosaic contains the pixel value of the last image that contributed to that point.

The mean operator takes the average over all the point contributions. It is therefore effective in removing temporal noise inherent in video. If the sequence contains moving objects on a static background, they will appear motion blurred, in a similar way to a long exposure photography. The median operator also removes temporal noise but is particularly effective in removing transient data, such as fast moving objects whose intensity patterns are stationary for less than half the frames. In a informal way, one can consider that moving objects are treated as outliers, because their intensity values are forced out of the central region of the intensity-ordered vector of contributions, which corresponds to the prevailing background.

The effects of the four operators described can be seen on Figure 4.6. The top row contains two of the original frames of an aerial sequence taken by a high altitude plane flying over the Arlington district, in Washington. The frames have a superimposed time-stamp and a horizontally shifted lower scan-line. Their intensity patterns do not follow the dominant upwards image motion, therefore allowing for the differences on the operators

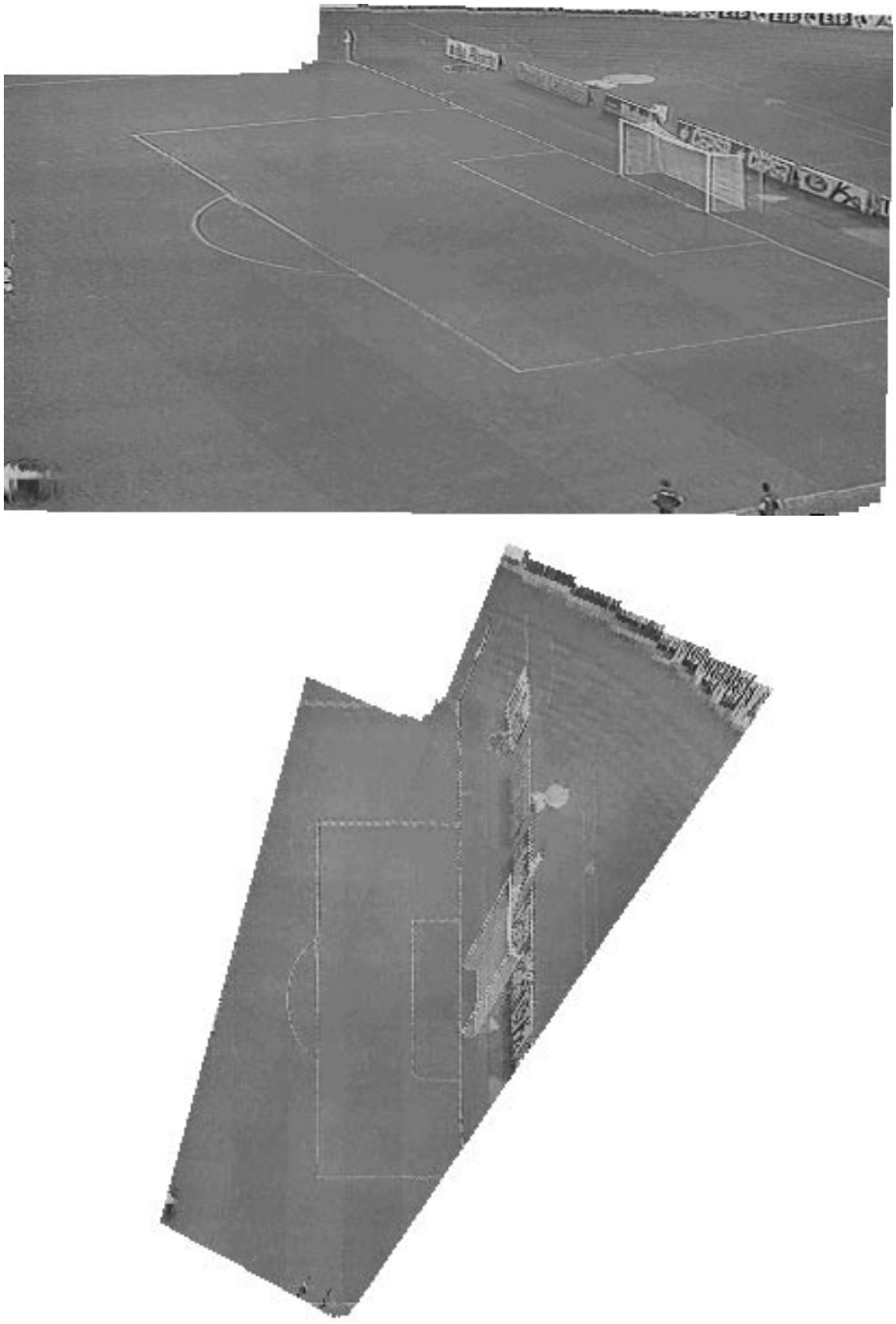


Figure 4.5: Soccer field mosaic constructed from a TV sequence, using the median temporal filtering. The rendering of the top image was performed using the first image as the reference frame. For the lower image, an usefull reference frame was chosen, corresponding to none of the image frames.

results to became noticeable. The middle row contains the use-first (left) and use-last (right) methods. On the lower row are the mean (left) and median (right). Note the time-stamp which is faded but visible on the central region of the left mosaic, as opposed to the median mosaic where it completely disappears. The maximum number of superimposed frames is 9, in the central region.

Other temporal operator have been used on the literature, such as the mode [46] and the weighted median [64, 46]. The weighted median is a refinement over the simple median, in order to account with the contributions from frames taken with different zoom settings. The higher the zoom of a frame, the higher is the resolution of its contribution. Therefore, in order to make full use of the available resolution, larger weights are associated with high zoom frames.

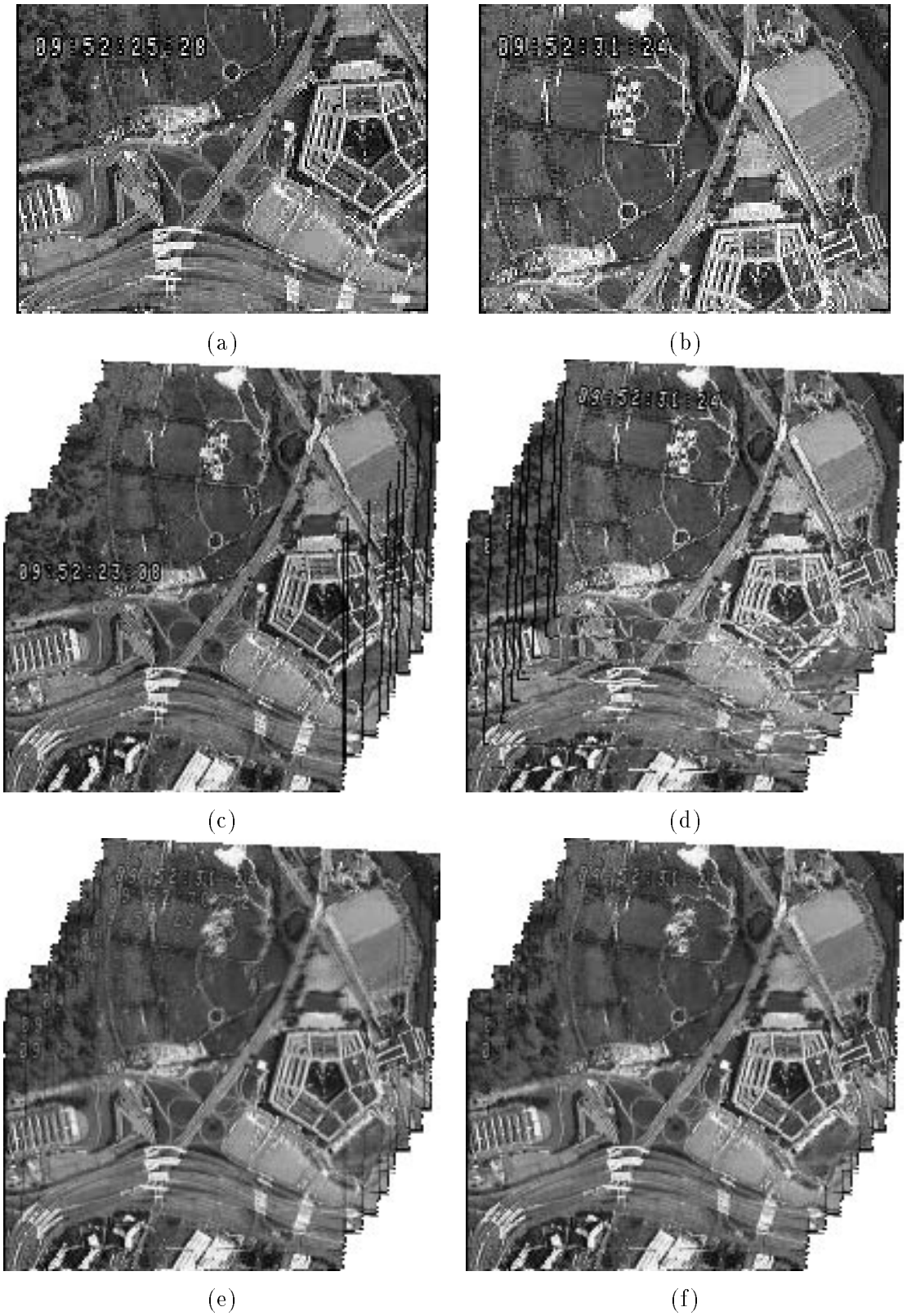
4.4.1 Direct mosaic registration

So far, we have described the mosaic creation process divided into two major sequential steps: modelling and rendering. Now we will present an alternative method in which modelling and mosaic rendering are accomplished simultaneously, therefore better suited for real-time mosaic creation. However, our main motivation for the alternative scheme is not real-time applications, but quality enhancement in motion estimation.

The frame-to-frame motion estimation procedure allows the construction of mosaics by the analysis of consecutive pairs of frames. In the global registration step, the frame-to-mosaic transformation for the last frame is computed by sequentially cascading all the previous inter-frame transformations. It is easy to realize that, small errors on the motion estimation due to the limited matching accuracy and image resolution, will accumulate and produce noticeable misalignment from the first to the last frame. This is notorious on cycled sequences, where the camera returns to previously captured parts of the scene.

Let us now consider an image sequence with a large number of overlapping frames. An explicit way to exploit this condition is to match and estimate the transformation parameters of each new image with the mosaic constructed with the all the previous ones. Figure 4.7 illustrates the required operations. A major drawback of this approach lies on the fact that great care has to be taken in estimating the correct motion parameters. Failling to do so will result in creating a mosaic with flaws, in which the features are spacially inconsistent with the next frame being registered. Since the robust matching selection assumes feature location consistency under the chosen motion model, this condition will lead to poor results.

In order to test the effectiveness of the scheme, a test sequence of 67 images of a road map was used. It was captured at close range by a translating and rotating camera, thus inducing noticeable perspective distortion. The map was scanned following an inverted *S*-shape, starting from the map upper left corner and finishing on the diagonally opposite



(a) (b) (c) (d) (e) (f)

Figure 4.6: Temporal operators: First (a) and last (b) frames of the original aerial sequence, use-first method (c), use-last (d), mean (e) and median (f).

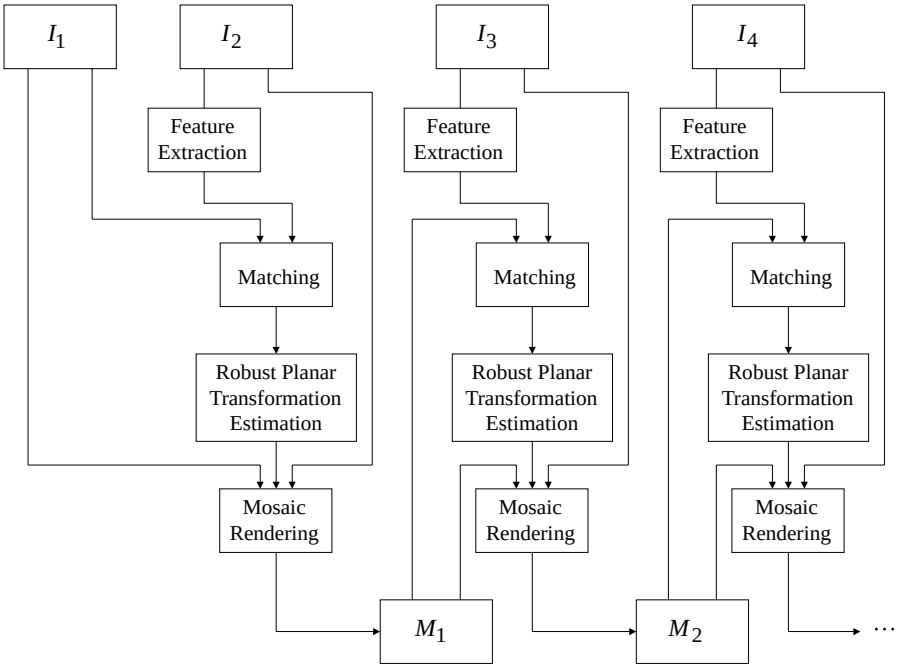


Figure 4.7: Block diagram of the sequence of operations for the direct mosaic registration procedure.

corner. Figure 4.8 shows two mosaics of the same sequence, constructed using the two alternative methods for motion estimation of the planar transformation model. The top image was created using the frame-to-frame motion estimation. The accumulation of small errors on the transformations has visible effects, which can be seen on the left edge and on the lower right corner of the map. Conversely, direct mosaic registration was used for the bottom image. In this case, no error accumulation effects are noticeable. However, the discontinuities on the lower edge of the map indicate some image misalignement. This is probably due to inconsistencies on the center area of the map, where the bottom row of images were registered.

4.5 Results and Applications

This section is devoted to the presentation of some results in video mosaicing obtained with the techniques described above. Several test sequences were used, containing various type of scenes, ranging from simple static planar scenes to sequences containing moving objects and noticeable depth variations. Sample frames from the original sequences are given in appendix C.

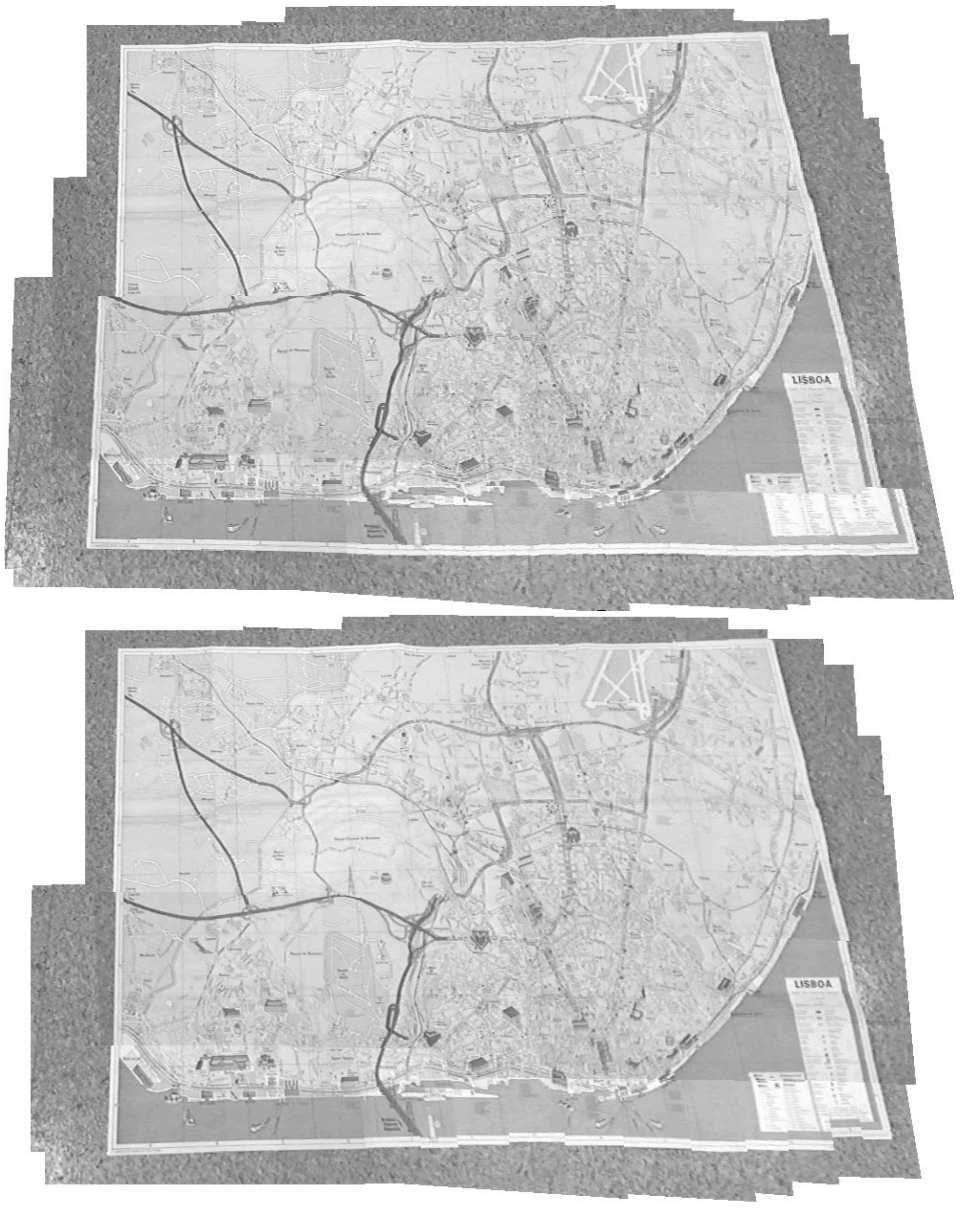


Figure 4.8: Example of direct mosaic registration. The top mosaic was created using the frame-to-frame motion estimation whereas direct mosaic registration was used for the lower one.

Aerial image composition

Examples of early use of image mosaicing are on aerial and satellite imaging applications [38]. The composition of images taken on a plane by a camera facing down is alleviated because of the minimal perspective distortion. The motion model can be as simple as the two parameter translation model, for an aeroplane with constant heading. However a more adequate motion model for aerial sequences should also account for rotation. The mosaic on Figures 4.9 and 4.10 were created from a sequence of aerial pictures taken by a US plane on high altitude surveillance operations. The original frames are 160×120 pixels and exhibit moderate image quality due to the fact that they where extracted from a highly compressed MPEG stream.

No information is available on the most adequate motion model. For this reason four different motion models were used, namely translation plus zoom, semi-rigid, affine and full planar. However, considering the fact that the camera appears to be facing down (thus making the image plane parallel to the ground plane), and that there is a slight rotation near the middle of the sequence, it can be argued that the semi-rigid is the most appropriate. This model is the simplest that accounts for rotation. The importance of the choice of the most appropriate model is apparent on Figure 4.10. The use of a too general model (right) allows for small registration errors to produce poor results.The semi-rigid model was used for the creation of the mosaic on Figure 4.11. Again, the test images where captured by a high altitude plane, flying over an urban scenario.

Ocean Exploration

Another important area for video mosaicing is ocean floor exploration. Here, mosaics can be useful for many applications such as site exploration, navigation and wreckage visualization [45]. Furthermore, due to the underwater limited visual range, registration of close range images is often the only solution for obtaining large visual areas of the floor.

Research on automatic mosaic creation for underwater applications has been conducted in the last few years. In [27] a setup is proposed for creating mosaics by taking images at locations whose coordinates are known with high precision. Image merging can thus be performed without image analysis, because the frame-to-frame motion parameters can be computed directly from the camera positions. Marks *et al.* [45] have developed a system for ocean floor mosaic creation in real-time. In this work, the authors use the four-parameter semi-rigid motion model, and assume rotation and zooming on the image frames to be small. This allows fast processing algorithms, but restricts the scope of applications to the case of a images taken by a camera whose retinal plane is parallel to the ocean floor.

An example of a sea bed mosaic is given in Figure 4.12. It was composed with 101

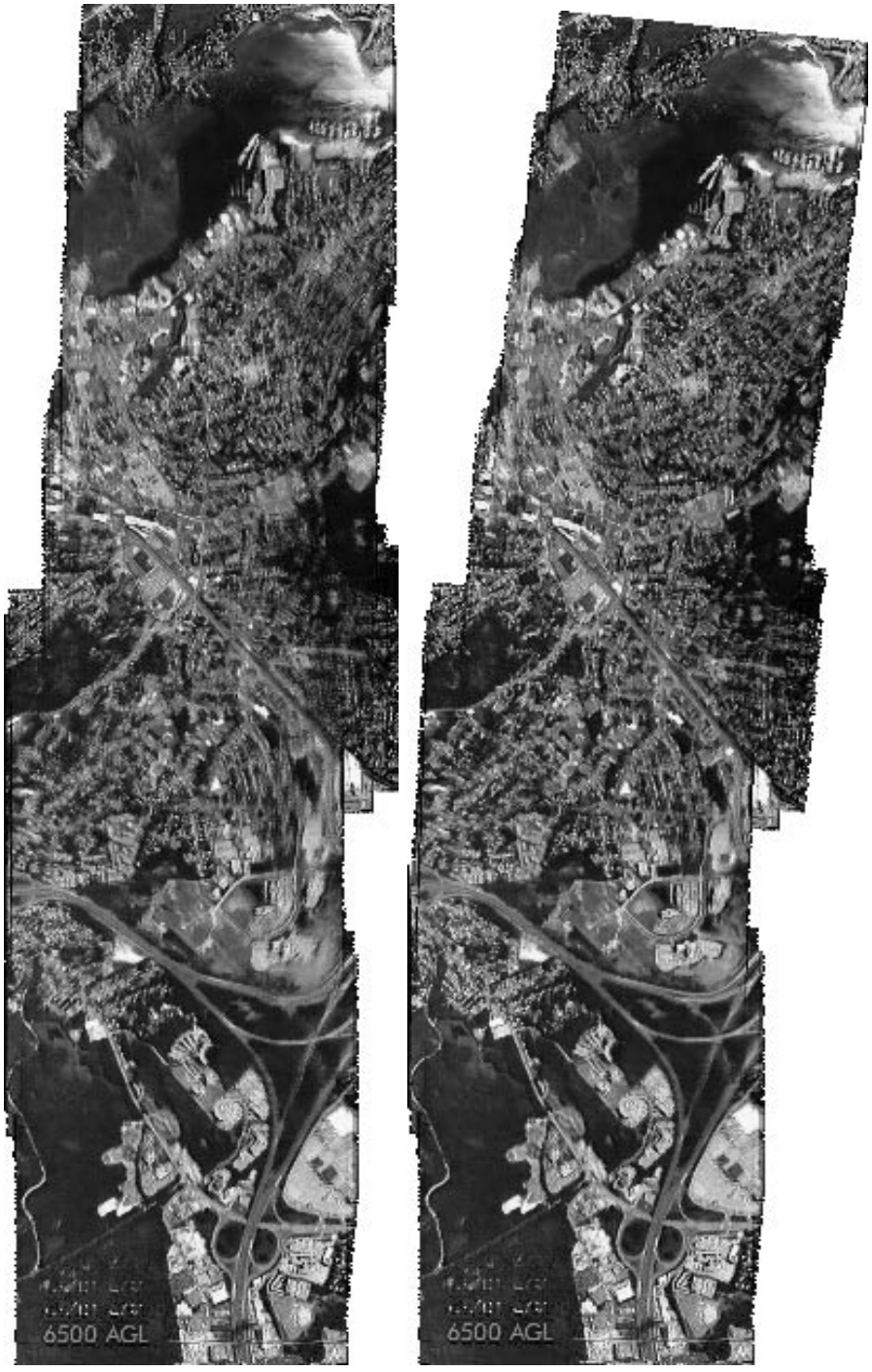


Figure 4.9: Aerial sequence example: translation and zoom motion model(left), and semi-rigid motion model (right).

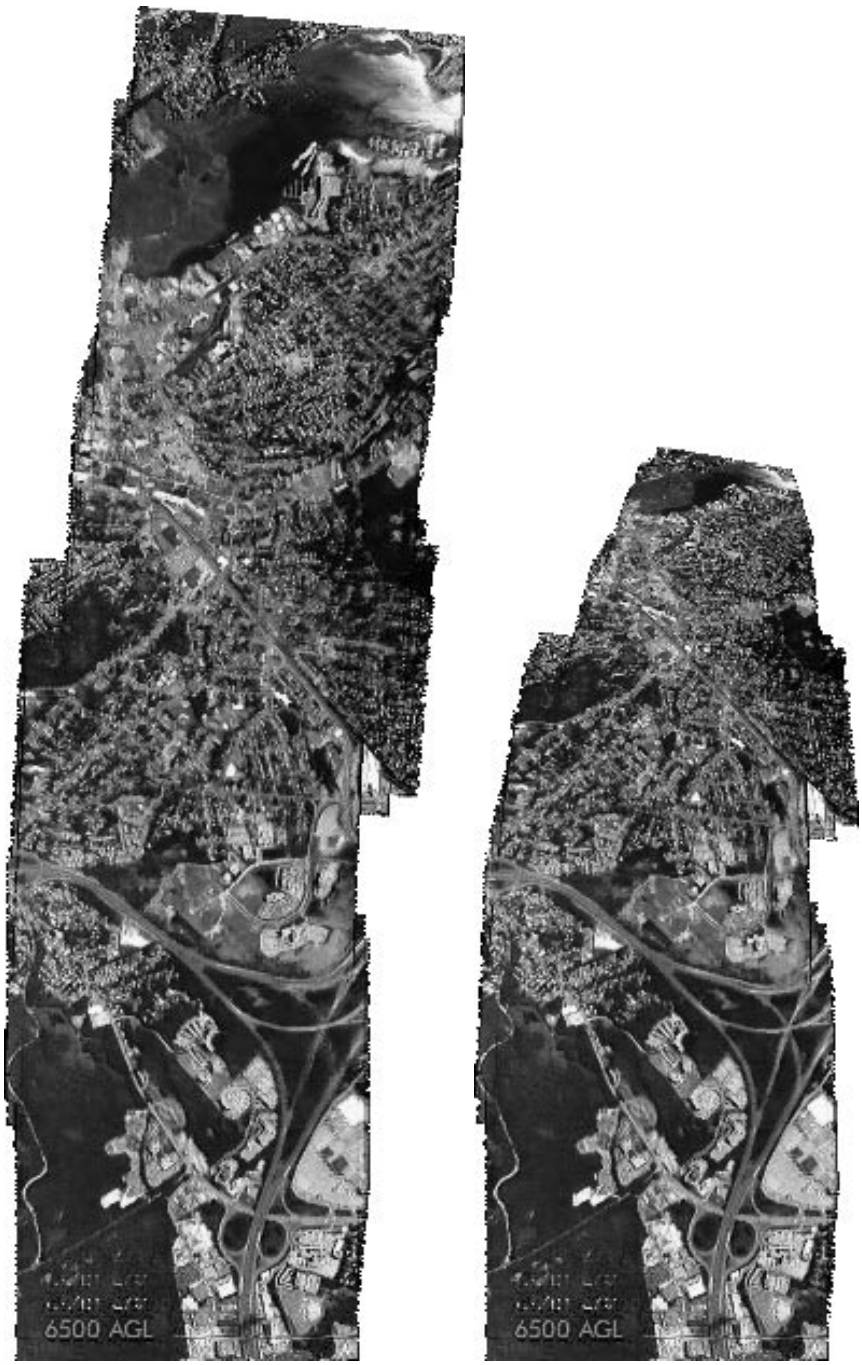


Figure 4.10: Aerial sequence example: affine motion model(left), and full planar motion model (right).

Figure 4.11: Aerial sequence example using the semi-rigid motion model.

frames, registered under the semi-rigid model and rendered with the median operator. The original sequence was obtained by a manually controlled underwater vehicle, and depicts a man-made construction. This scene is not planar nor static. The camera is moving along a fracture inside which some rocks can be seen. In the fracture there are noticeable depth variations as opposed to the almost planar surrounding sea bed. Even so, the sea bed is mostly covered with algae and weeds, which provide good features for the matching process, but violate the underlying planar scene assumption. Another assumption violation is due to some moving fish. Figures 4.13 (a) and (b) show two sub-mosaics in which the motion of the fish can be clearly noticed. Although constructed from the same sequence, these sub-mosaics were rendered using the use-last temporal operator.

The mosaic in Figure 4.12 is a good example of the performance of the implemented matching and registration methods. Even with notorious violations of the assumed model, the algorithm can still find the motion parameters as to create a mosaic with small misalignments to the human eye.

Video coding and enhancement

Video mosaics can be very useful in the visualization of video sequences, but can also be used as an efficient representation, on applications such as video compression, enhancement and search. Recent work [33, 62] has addressed the idea of using mosaics for complete scene representation as to fully recover the video sequence from a *dynamic mosaic*. This dynamic mosaic is an extension to the usual static mosaic, comprising three elements:

- a (static) background mosaic, just like the ones presented in this section. Static mosaics have also been called *salient stills* [64, 46].
- the set of frame transformations relating each frame to the mosaic frame.
- the image residuals containing the brightness differences of each frame to background mosaic.

Further details on mosaic classification can be found in [33], where a detailed taxonomy is proposed in the context of video applications.

High compression video coding can be attained by creating and coding the dynamic mosaic. Most video sequences tend to have a large amount of image overlap, and much of the image redundancy is due to a static background. In such cases, the dynamic mosaic residuals are small, when compared to the residuals between consecutive frames, even if motion compensation is performed[33]. Mosaic-based video coding requires, however, the whole sequence to be available for the background estimation. For this reason it is suited for off-line coding and storage.

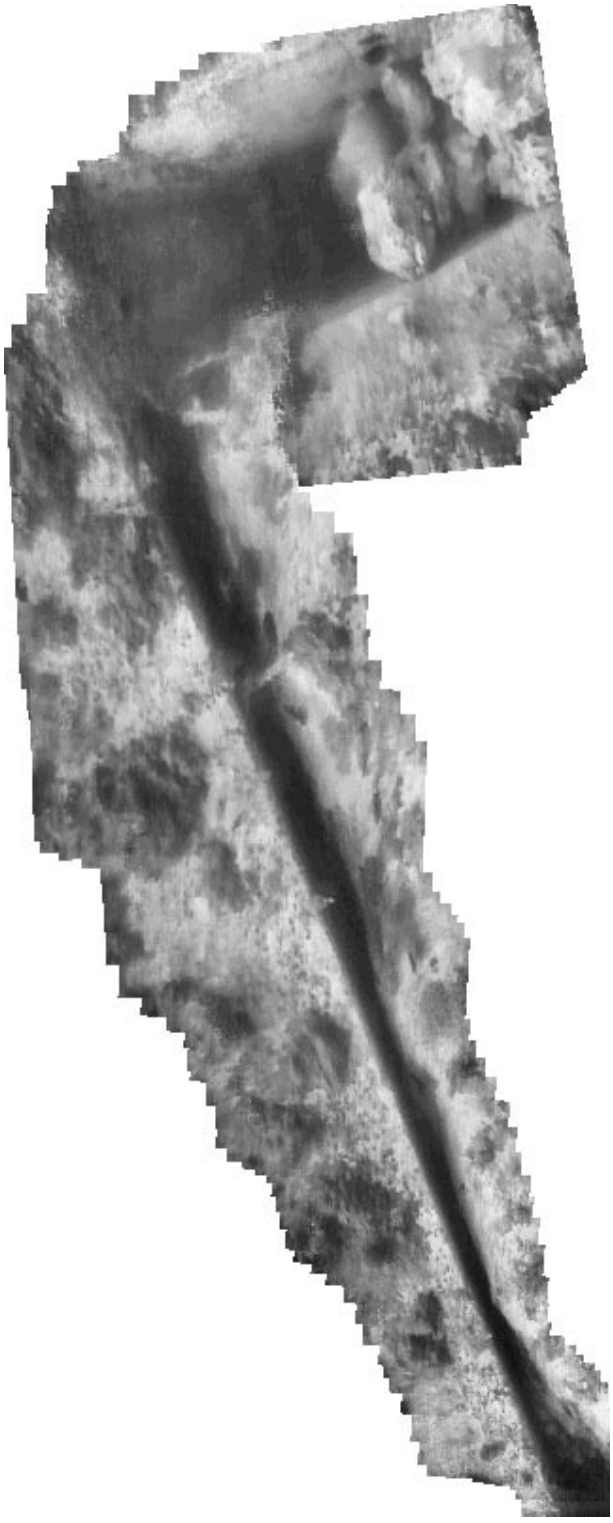


Figure 4.12: Sea bed mosaic example. The images were registered using the semi-rigid motion model and rendered using the median operator.

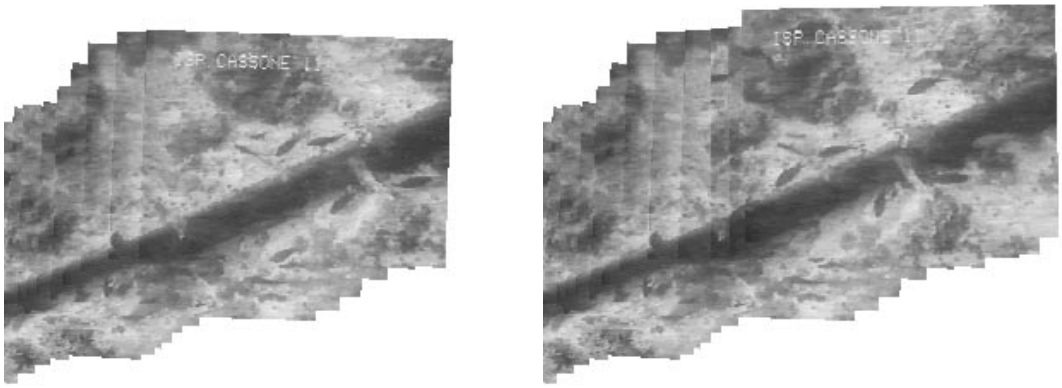


Figure 4.13: Example of mosaic creation where the static scene assumption is violated by the presence of moving fish.

Although no residual estimation has been implemented in this thesis, we have made used of the knowledge of the global registration model of section 4.3.2. The frame registration parameters together with the background mosaic allow the video sequence to be resynthesize using different camera motion, or no motion at all. An example, with no camera motion, is given on Figure 4.14. The background, rendered using the median operator, is shown in Figure 4.5. A new sequence was created with the original frames warped and superimposed on the background. The resulting effect is a better understanding of the player locations as compared to the original sequence, since the captured ground markings are visible from the first to the last image. Another use of this technique is in image stabilization for video sequences captured with a *shaky* camera. In this case the image motion can be low-pass filtered, removing the unwanted trembling. An interesting visual effect can also be obtained for sequences with moving objects. Selected frames from the sequence can be emphasized in the rendering stage of the mosaic creation in order to make the position of the moving objects notorious. Therefore, the resulting mosaic may allow a better perception of the object movement in the sequence. A similar effect can be acheived in traditional photography by making several exposures on the same photographic plate. This is commonly referred to as a *cronophotography* [46]. Figure 4.15 presents an example of this. The original sequence depicts a stunt motobike driver jumping from a platform and getting closer to the camera. The camera follows the driver, presumably from the same point of view, without inducing noticeable image rotation. The background for the sequence of 56 images was hence estimated using the image translation and zoom motion model (top). The cronophotography (bottom) was created by computing the intensity differences of three individual registered frames. The three difference images are then added to the background.

Figure 4.14: Static camera example: Frames 1 and 32 of a new sequence created with the original frames of the soccer sequence warped and superimposed on the median background.

Figure 4.15: Cronophotography example: for a sequence of 56 images, the median background was estimated (top). Three selected frames are used for the creation of the cronophotography (bottom), allowing the perception of the bike's motion.

Panoramic views

An intuititive use of video mosaics is in the creation of panoramic images. Here we refer to panoramic images or *panoramas* as stills of wide field-of-view, which surpass the human eyesight beyond the boundary of peripheral vision. These images allow a rich perception of the surrounding space because they depict the environment in several directions. A simple setup for panoramic acquisition using conventional photography consists in a photo camera rotation around the vertical axis. This rotation is synchronized with the film movement across a narrow vertical slit creating a scanning effect. Commercial cameras are available, such as the *Globuscope*, which allows a horizontal field-of-view of more than 360 degrees. Video mosaicing can easily be applied to the creation of panoramic imaging, provided that a set of overlapping images is captured from the same view-point.

As applications for panoramas, we can point out scenic representations for outdoors environments (such as landscapes) or the recreation of indoor environments for virtual reality modelling. We will now provide examples for both. The panorama of Figure 4.16 was created from a sequence of 90 images captured by a hand held camcorder. For the image registration a simple motion model accounting just for translation and zoom was used. It depicts a landscape of *Serra da Peneda* in the north part of Portugal. The camera followed the hill tops inducing vertical motion, while panning for more than 360 degrees. In fact, the regions on the sides of the mosaic overlap by approximately 65 degrees. Conversely, an indoor scene is presented in Figure 4.17, describing the interior of the Vision Laboratory at the Institute for Systems and Robotics in Lisbon. In this case the camera was set atop a tripod and rotated around the vertical axis. The residual vertical motion was removed by zeroing the corresponding component on the motion parameters before the mosaic rendering. By combining selected portions of the mosaic with a virtual reality model of the laboratory[1] comprising six vertical walls, new views of the room can be created. Three of such views are shown in Figure 4.18. Although not entirely realistic, this example illustrates the application of mosaics as a way of providing real texture to virtual environments.

Panoramic representations for robot navigation

Mosaicing techniques can be valuable for robot navigation, as a way of providing visual maps for robot navigation. We will finish this chapter by presenting an example where mosaicing and image registration can be used by a robot in order to locate and orientate itself.

The mosaic of Figure 4.19 was created from a sequence of 45 images captured by a cam-

[1] This model was created using the VRML 2.0 modelling language. The author kindly thanks Etienne Grossman for the programming.

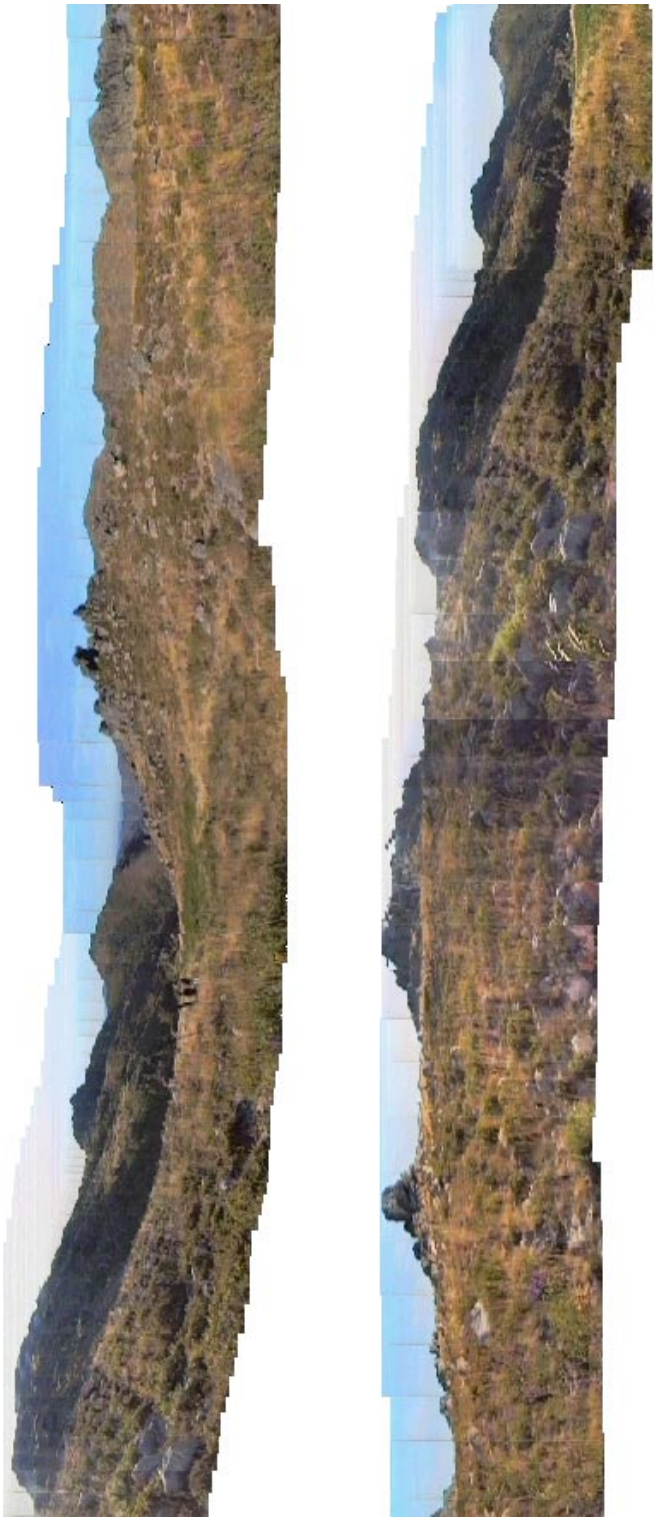


Figure 4.16: Panoramic mosaic example: Outdoor scene of a landscape in Serra da Peneda. The mosaic was created using 90 images.

Figure 4.17: Panoramic mosaic example: Indoor scene of the computer vision lab created using 83 images.

Figure 4.18: Three synthetic views generated with a virtual reality model.

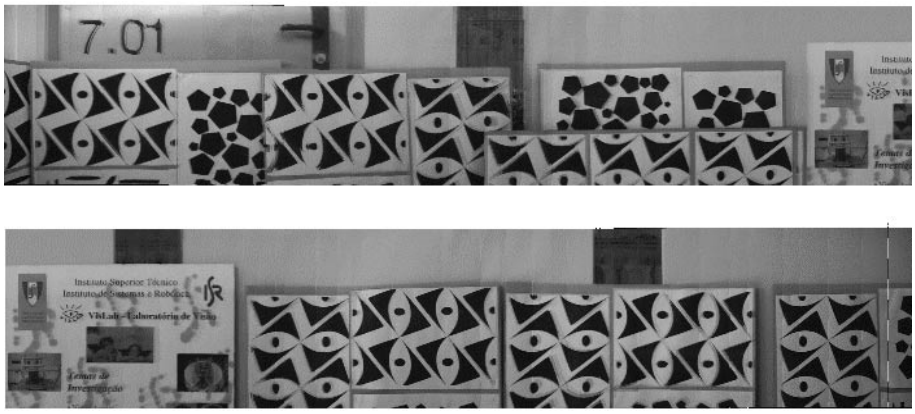


Figure 4.19: Indoor wall mosaic, used for robot localization during navigation.

era on a mobile platform, while moving along a corridor wall. During image acquisition, the platform kept approximately the same heading and distance to the wall. Therefore, it allowed for the simplest motion model - image translation - to be used during the image registration step. A second set of images, depicting a smaller area of the same corridor, was captured afterwards by the same setup. In this second travelling, the platform was allowed to change both its heading and its distance to the wall. Since the image plane is no longer parallel to the corridor wall, some perspective distortion is thus induced. Therefore, an appropriate motion model for registering these images on the mosaic is the full planar transformation model. Figure 4.21 shows the central part of the corridor mosaic, with the superimposed frame outlines of 5 images.

A more suitable motion model can be devised by taking into account some geometric constraints arising from the camera setup. Two of these constraints are the constant height to the floor and the single degree of freedom for the platform rotation. Furthermore, this constrained model can be put in the form of a function of some useful navigation parameters, such as the distance of the optical center to the corridor wall. In the example presented here, the following parameterization for the planar transformation matrix was used:

$$T_{image,mosaic} = \begin{bmatrix} fk_u c_\theta - u_0\, s_\theta & 0 & fk_u\,(s_\theta d - u_{e0} c_\theta) + u_0\,(u_{e0} s_\theta + s_\theta d) \\ -v_0 s_\theta & fk_v & -fk_v v_{e0} + v_0\,(u_{e0} s_\theta + s_\theta d) \\ -s_\theta & 0 & u_{e0} s_\theta + s_\theta d \end{bmatrix}$$

$$c_\theta = \cos(\theta)$$
$$s_\theta = \sin(\theta)$$

where d is the perpendicular distance of the optical center to the corridor wall, θ is the angle between the camera optical axis and the normal to the wall, and (u_{e0}, v_{e0}) are the

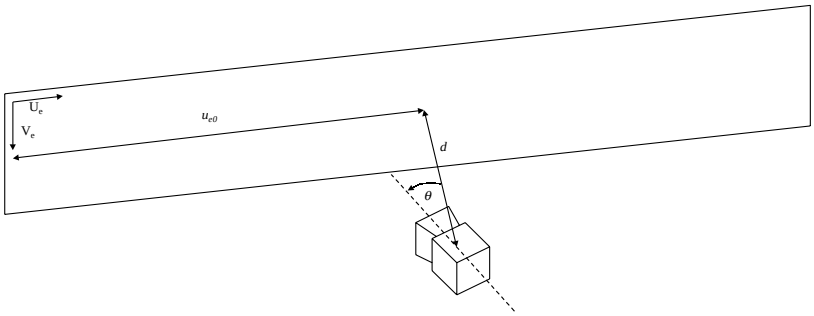


Figure 4.20: Useful parameters for navigation: perpendicular distance to the wall d, angle between the camera optical and the normal to the wall, δ, and the distance to the left side of the mosaic, $u_e 0$.

Frame	d $(meters)$	θ (deg)	$u_e 0$ $(meters)$
1	0.9931	-3.26	2.6182
2	0.9370	-6.16	2.2819
3	0.8874	-7.14	1.9744
4	0.8136	-11.73	1.6743
5	0.7092	-17.56	1.3031

Table 4.1: Estimated parameters for navigation.

coordinates of the perpendicular projection of the optical center on the wall (Figure 4.20). The remaining parameters are the camera intrinsic parameters, as described in section 2.2. Due to the non-linearity of this parameterization, the linear computation of the planar transformation using the SVD is no longer possible. Therefore an iterative minimization procedure, the Downhill Simplex method [51], was used to estimate d, θ, and (u_{e0}, v_{e0}).

The results for a 5 image sequence are presented on table 4.1. Figure 4.22 shows the frame outlines for the constrained model. The platform trajectory can be recovered in straightfoward manner and is shown in Figure 4.23.

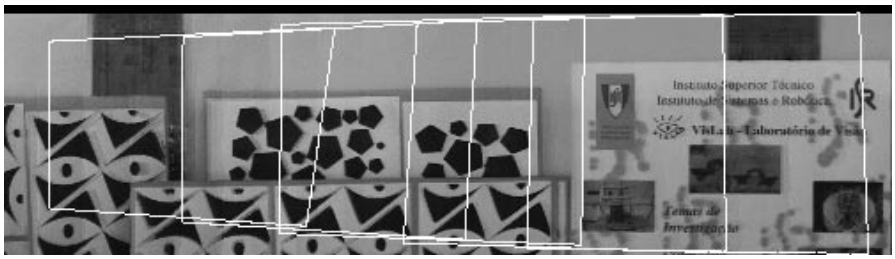


Figure 4.21: Corridor mosaic with superimposed frame outlines for 5 images, registered using the full planar motion model.

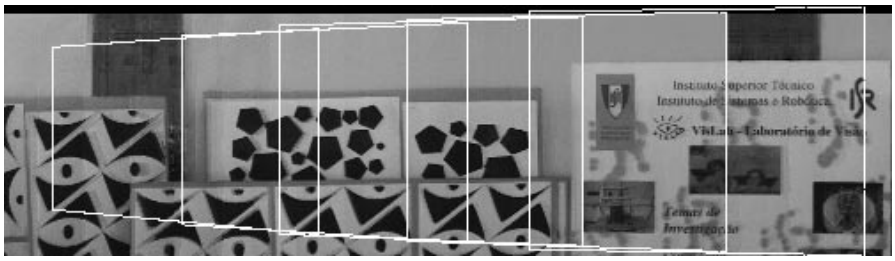


Figure 4.22: Corridor mosaic with superimposed frame outlines for 5 images, registered using the constrained motion model.

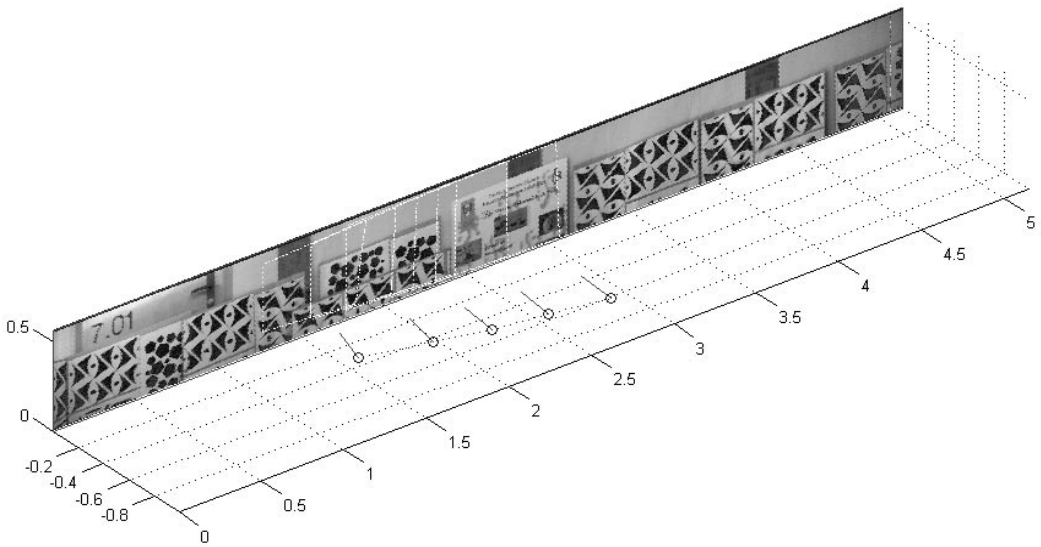


Figure 4.23: Camera trajectory reconstruction. The camera focal point and optical axis are indicated by the circles and the small lines, respectively.

Chapter 5

Application to 3-D Reconstruction

This chapter presents an application of robust techniques to the estimation of the epipolar geometry, and to 3-D structure recovery.

It is organized as follows. Section 5.1 describes two distinct minimization criteria for the computation of the fundamental matrix, namely linear least-squares, and a non-linear criterion based on the distance of each point to the corresponding epipolar line. In section 5.2, two different parameterizations, suited for the non-linear criterion, are discussed. An experimental comparison is undertaken in section 5.3 dealing separately with localization errors and mismatches. Some robust techniques discussed in chapter 3 have been implemented and compared. These are RANSAC, LMedS, MEDSERE and two methods based on case deletion diagnostics. The last section of the chapter presents examples of Euclidean reconstruction with both synthetic and real images. The effect of image correction is also addressed.

5.1 Minimization Criteria for the Fundamental Matrix

5.1.1 Linear Criterion

In section 2.4.3 we have presented an estimation criterion for the fundamental matrix, which is a direct use of the eight point algorithm introduced by Longuet-Higgins[41]. For a set of point correspondences $\mathbf{u}_i = (u_i, v_i, 1)$ and $\mathbf{u}'_i = (u'_i, v'_i, 1)$, the epipolar geometry imposes a constraint in the form of equation (2.6). An intuitive minimization criterion is therefore

$$\min_F \sum_i \left(\mathbf{u}_i'^T F \mathbf{u}_i \right)^2 \tag{5.1}$$

which is equivalent as to minimize

$$\min_{\mathbf{f}} \| H \ \mathbf{f} \| \tag{5.2}$$

where H and $\mathbf{f}$ are the correspondences observation matrix and the vector form of the elements of F, as defined in section 2.4.3. In order to avoid the trivial solution $\mathbf{f} = 0$, an additional constraint has to be imposed, usually $\|\mathbf{f}\| = 1$. Alternatively, one can set one of the elements of F to 1, and reformulate the H matrix in the expression (5.2) accordingly. However, this second approach has the disadvantage of not allowing all elements of F to play the same role, as noted in [43]. A convenient way to solve (5.2) is to used the SVD.

This criterion does not explicitly enforce the F matrix to be of rank 2. On practical applications, the F matrix will not be rank deficient, even if large numbers of point correspondences are use. Moreover, this condition is severally aggravated if it is used in the presence of mismatches.

Since most applications of the fundamental matrix require it to be rank 2, this condition can be enforced after the linear estimation. A convenient way to correct F, is to replace it [26] by the rank 2 matrix F' that minimizes the Frobenius norm $\|F - F'\|$. A suitable algorithm for the computation of F' is again the SVD, in the way described in appendix A.

From what has been said, one can summarize the estimation of the fundamental matrix by this criterion, as consisting of two steps[26] :

- **Linear solution**, by minimizing the expression (5.2). The solution in the least-square sense is the eigenvector of $H^T H$ corresponding to the smallest singular value of H.
- **Constraint enforcement**, by approximating the solution by a rank 2 matrix.

5.1.2 A criterion based on the distance to the epipolar lines

We have also used a non-linear approach which is based on a geometric interpretation of the criterion (5.1). It can be easily shown that the Euclidean distance on the image plane of a point $\mathbf{u} = (u, v, 1)$ to a line $\mathbf{l} = (l_1, l_2, l_3)$ is given by

$$d(\mathbf{u}, \mathbf{l}) = \frac{|\mathbf{u} \cdot \mathbf{l}|}{\sqrt{l_1^2 + l_2^2}} \tag{5.3}$$

Therefore an intuitive criterion can be put forward, by minimizing the distances of the points to the corresponding epipolar lines,

$$\min_F \sum_i d^2(\mathbf{u}'_i, F\mathbf{u}_i)$$

As noted in [43], the two images do not play a symmetric role, as it did in the case of the linear criterion. This is apparent from the fact that in this last expression we are only minimizing the distances taken on the second image. A way to solve the problem is to

incorporate the distances taken on the first image, and to note that by exchanging the coordinates of the correspondences, the fundamental matrix is changed to its transpose, *i.e.*,

$$\mathbf{u}'^T F \mathbf{u} = \mathbf{u}^T F^T \mathbf{u}'$$

The extended criterion is therefore

$$\min_F \sum_i d^2\left(\mathbf{u}'_i, F\mathbf{u}_i\right) + d^2\left(\mathbf{u}_i, F^T\mathbf{u}'_i\right) \tag{5.4}$$

and can be written using equation (5.3) as

$$\min_F \sum_i \left(\frac{(\mathbf{u}'_i, F\mathbf{u}_i)^2}{(F\mathbf{u}_i)_1^2 + (F\mathbf{u}_i)_2^2} + \frac{(\mathbf{u}'_i, F\mathbf{u}_i)^2}{(F^T\mathbf{u}'_i)_1^2 + (F^T\mathbf{u}'_i)_2^2} \right) \tag{5.5}$$

One can easily see from (5.3) that the minimization is independent of the scale factor of F, thus not requiring the additional constraint on the elements of F used in the linear criterion. Even so, the rank 2 condition still has to be imposed afterwards.

The main drawback of the criterion is that it is non-linear on the elements of F. Therefore, a non-linear minimization technique has to be used, which is much more computationally expensive than the least-squares solution presented earlier.

5.2 Fundamental Matrix Parameterizations

When using criterion (5.4) the need for the subsequent rank 2 enforcement can be avoided if we use a structure for the fundamental matrix in which the rank condition is implicit. Following a proposal by Luong [43], we have implemented the following parameterization:

$$F = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7a_1 + a_8a_4 & a_7a_2 + a_8a_5 & a_7a_3 + a_8a_6 \end{bmatrix}$$

It is easy to check that a matrix defined by this equation is rank 2, at the most, for a general configuration of the parameters. Due to the non-linearity of this parameterization, it is not suited to be used with the criterion (5.2) as the solution would no longer be linear.

An alternative parameterization has been used in [43, 75], expliciting the dependency of F some epipolar geometry entities. The structure of F is

$$F = \begin{bmatrix} b & a & -ay - bx \\ -d & -c & cy + dx \\ dy' - bx' & cy' - ax' & -cyy' - dy'x + ayx' + bxx' \end{bmatrix}$$

where (x, y) and (x', y') are the coordinates of epipoles on the image plane and a, b, c, d are coefficients of the homography between the sets of epipolar lines [43]. This parameterization always implies finite epipoles, and for this reason it was not used in the final implementation.

5.3 Experimental Comparison For Robust Matching Selection

This section describes the implemented algorithms and tests for the estimation of the fundamental matrix. A comparison on the robustness under the presence of feature localization errors and gross mismatches is presented. Different strategies are used to deal with the two types of errors. For this reason we will treat it separately.

5.3.1 Feature localization errors

This class of errors refers to the image point location inaccuracy, due to the limited resolution of the feature extractors and matching procedures.

We assume that these errors exhibit Gaussian behavior. This assumption is reasonable since the errors are small on practical situations, typically within two or three pixels.

On the experiments described in this subsection no mismatch errors are considered. Each pair of point correspondences is correct in the sense that they are projections of the same 3D point.

Normalization of the Input

The error sensitivity of the linear criteria can be reduced by applying an input coordinate transformation, prior to the computation of F. Traditionally, on image analysis applications, the coordinate origin of the image frame is often placed on the top- left corner of the image. This does not promote coordinate homogeneity, and frequently leads to large magnitude differences on the elements of matrix H in expression (5.2), thus creating numerical instability.

A simple procedure of coordinate transformation is presented in [25] which performs coordinate translation and scaling. By moving the origin to the point mass centre on each image, and by scaling the axes so to have unitary standard deviation (figure 5.1), one can improve the problem condition and stability. Moreover, the transformation has the advantage of better balancing the elements of F. The singularity enforcement is therefore less disruptive.

Experimental evidence reported in [25] indicates that the normalized version of the least-squares criterion can sometimes outperform iterative techniques, while maintaining a much lower level of algorithmic complexity. All the results based on the linear criterion were obtained using input normalization, unless stated otherwise.

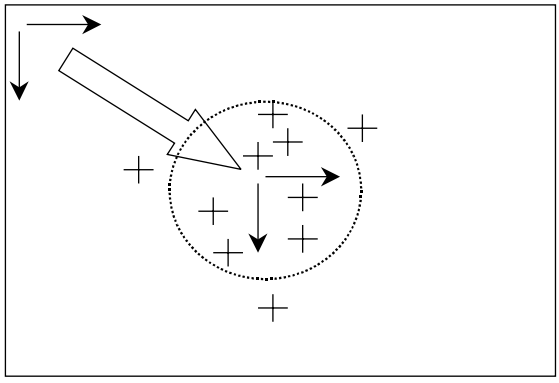


Figure 5.1: Coordinate normalization: The origin of the image plane frame is moved to the points centroid, and the axis are scaled.

Results on synthetic data

In order to evaluate the performance of the estimation algorithms we have used both synthetic and real images. The experimental method, using synthetic images for testing feature localization errors, is now described.

A set of 50 randomly scattered 3-D points are projected using two camera matrices P_1 and P_2. The cameras have the same intrinsic parameters and are positioned so that both optical axes meet at the points centroid. The synthetic images are 256×256 pixels, and the projected points are spread over a large part of the image area. For a set of 100 trials, the following operations are carried:

1. Add Gaussian noise to the point projections,
2. Estimate the fundamental matrix F,
3. Compute the mean distance of each noise-free point to the corresponding epipolar line.

Results for the linear and the distance criteria are shown in Figures 5.2 and 5.3 . The evolution of the mean distance to epipolar line is plotted against the standard variation of the localization noise, ranging from 0 to 3 pixels.

The minimization procedure for the linear criterion was the SVD, whereas for the distance criterion the Downhill Simplex method [51] was used. Although not particularly fast in terms of convergence speed, this method does not require cost function derivatives and is usually very robust.

Figure 5.2 illustrates the effect of coordinate normalization on the linear criterion. In fact, without normalization the noise sensitivity is very high and the performance degrades much faster when compared to the normalized version or to the non-linear criterion. A

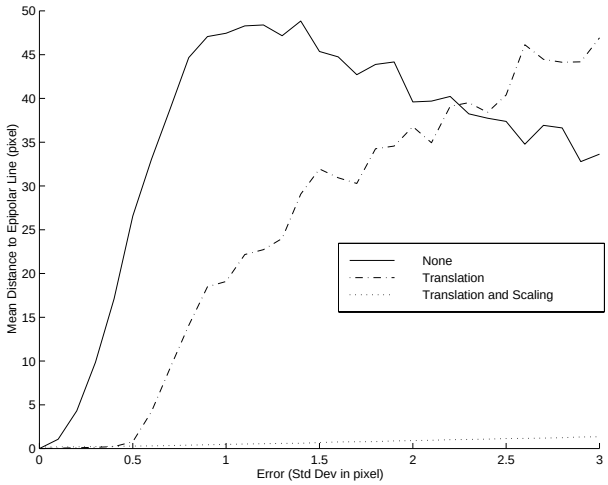


Figure 5.2: Sensitivity to feature localization errors, for the linear criterion, without data normalization (None), translating the data centroid to the origin (Translation), and translating and scaling the point coordinates (Translation and Scaling)

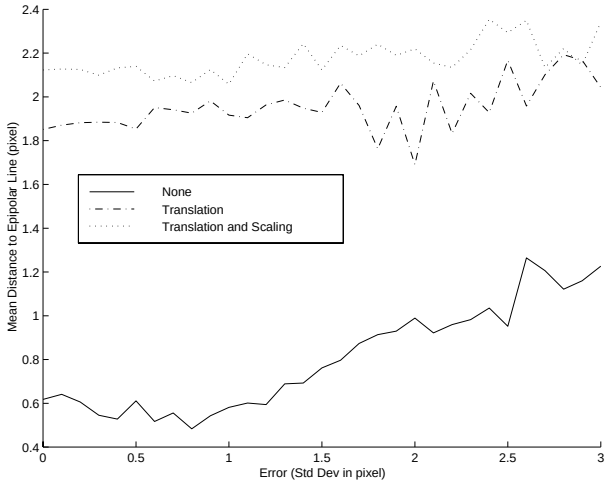


Figure 5.3: Sensitivity to feature localization errors, for the non-linear distance criterion, without normalization (None), translating the data centroid to the origin (Translation), and translating and scaling (Translation and Scaling)

somehow unexpected result is found in 5.3. Here the normalization degrades the performance of the distance criterion. Although not thoroughly investigated, a plausible explanation for this condition is the fact that normalization is very important in making the rank deficient matrix approximation less disruptive. Since no singular condition has to be enforced for the distance criterion, the normalization does not pay for the associated round-off errors due to the increase of numerical operations.

5.3.2 Mismatch errors

We will now consider the effect of mismatched points on the estimation of the fundamental matrix. This type of error happens frequently when real data is used, and is usually due to occluding objects or severe changes on the light conditions. For the applications dealt with in this thesis, methods robust to mismatches are essential. Therefore we have extensively used a random sampling technique, namely the two step variant of the LMedS described in section 3.2.4, for both mosaicing and 3-D reconstruction. However, for performance comparison purposes, two other methods for motion model estimation were implemented which do not use random sampling. These methods are based on simple and intuitive case deletion diagnostics. In the context of the fundamental matrix estimation, these are:

LSRes: F is estimated using the linear criteria on all point correspondences; then we select the 8 pairs of points that correspond to the smallest elements of the residuals vector $\mathbf{r} = H\ \mathbf{f}$ of Equation (5.2) and re-estimate the fundamental matrix.

CDDist: For a set of N matched pairs, this method consists of $N-8$ iterations where we estimate F using the linear criteria. In each iteration we discard the correspondence pair that correspond to the largest distance to the epipolar line.

As described in chapter 3 we have also used an algorithm based on the Random Sampling Consensus (hereafter referred as RANSAC), implemented by the following steps:

1. Sample all the matched points for a set of the minimum number of pairs required for the model instantiation (thus 8 for a linear computation), and compute the matrix F.

2. For a given distance threshold d_T, select all the pairs whose distance to the respective epipolar lines is smaller than d_T. This is the consensus group for this F.

3. If the number of selected pairs is greater than an initial estimate of the number of correct data points, then compute the F matrix based on all the selected pairs, using the linear criterion.

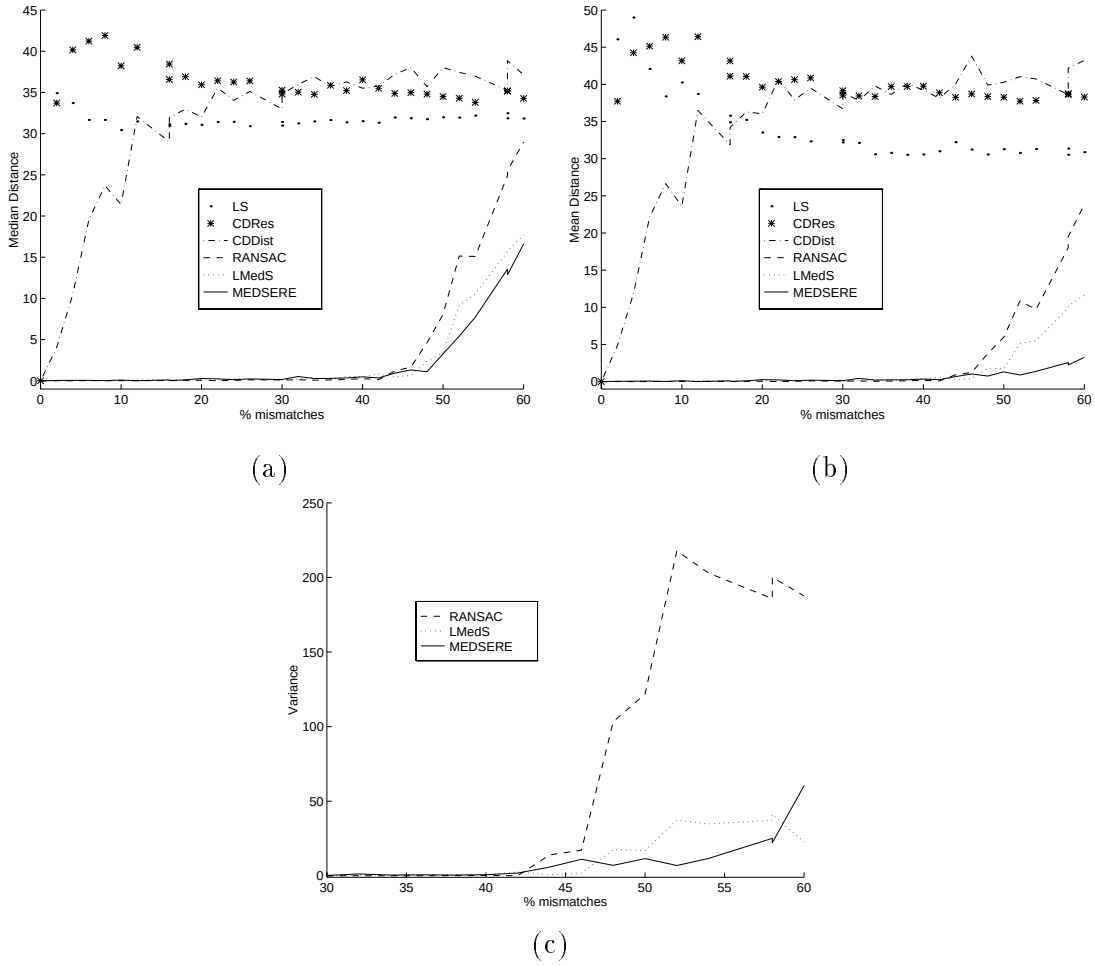


Figure 5.4: Performance of the algorithms under mismatch error conditions

4. Otherwise, repeat steps 1 to 3 up to a specified number of iterations, and return F computed with the largest consensus group found.

Figure 5.4 shows the performance of the described algorithms on the presence of mismatch errors, ranging from 0 to 60% of the total number of pairs. The images used are projections of a synthetic shape with 100 3D points. Averages of 30 runs for each algorithm were taken.

The evaluation criteria on the first plot is the median distance of each pair of points to the corresponding epipolar lines, for the whole set including mismatched data. To better assess the performance, the second and third plots show the mean (b) and the variance (c) of the point-to-epipolar line distance, for the original error-free data set.

One can see that the algorithms with best performance are MEDSERE and LMedS. MEDSERE is capable of good results even under conditions as severe as 60% of mismatches, slightly outperforming LMedS. The RANSAC also presents good results up to

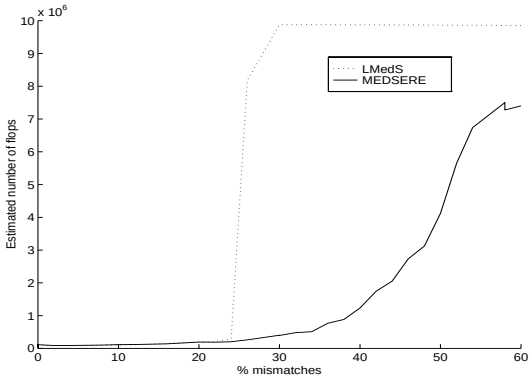


Figure 5.5: Evolution of the computational cost for the LMedS and MEDSERE algorithms

50% mismatches, but fails shortly afterwards. The other methods perform poorly.

Another relevant aspect when comparing the performance of different algorithms is to characterize the required computational effort. Figure 5.5 show the approximate number of floating point operations of LMedS and MEDSERE in the presence of a variable number of mismatched points. The maximum number of samples and the median threshold are the same for the two methods. In this aspect, MEDSERE compares favorably with LMedS.

5.4 Uncalibrated Reconstruction

In this section we will report some results on the recovering of scene structure from images captured by uncalibrated cameras. From what has been presented in chapters 2 and 5, we are now able to outline a projective reconstruction procedure which uses just a pair of images as the input. If we know the real world coordinates of at least five of the reconstructed points, then we are able to accomplish Euclidean reconstruction for all the 3-D points. In the scope of this thesis, we are interested in Euclidean reconstruction for the purposes of visualization and to access the accuracy of the reconstruction implementations.

5.4.1 Euclidean reconstruction

The Euclidean reconstruction comprises the following steps:

1. Robustly estimate the fundamental matrix F from a set of matched points.

2. Determine some P_1' and P_2' agreeing with F. For this, the formulas of lemma 2 are used.

3. Recover the projective 3D structure using P_1' and P_2'. For each matched pair, the intersection of the optical rays is computed using equations (2.10).

4. Estimate the G matrix by the use of ground points. This matrix embodies the collineation in $I\!P^3$ which relates the projective frame with the Euclidean one.

5. Apply G to the points recovered in 3, to recover the Euclidean structure.

The estimation of the collineation G from a set of n corresponding 3-D points can be performed using a simple linear procedure. As a collineation, it is completely defined by five pairs of points. For more than five correspondences a least-squares method is used, in a similar way to the linear estimation of planar transformations described in section 2.3.1. For each pair of 3-D points $\mathbf{x}_i = (x, y, z, 1)$ and $\mathbf{x}'_i = (x', y', z', 1)$, the equation

$$\mathbf{x}_i = G\mathbf{x}'_i$$

imposes three independent constraints on the elements of G. We can now assemble an homogeneous system of equations in the form $H.\mathbf{g}_l = 0$, where $\mathbf{g}_l$ is the column vector containing the elements of G in a row-wise fashion, and H is a $(3n \times 16)$ matrix

$$H = \begin{bmatrix} x'_1 & y'_1 & z'_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -x_1x'_1 & -x_1y'_1 & -x_1z'_1 & -x_1 \\ 0 & 0 & 0 & 0 & x'_1 & y'_1 & z'_1 & 1 & 0 & 0 & 0 & 0 & -y_1x'_1 & -y_1y'_1 & -y_1z'_1 & -y_1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x'_1 & y'_1 & z'_1 & 1 & -z_1x'_1 & -z_1y'_1 & -z_1z'_1 & -z_1 \\ & & & & & & & & & & \vdots & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x'_n & y'_n & z'_n & 1 & -z_nx'_n & -z_ny'_n & -z_nz'_n & -z_n \end{bmatrix}$$

The system is solved using the SVD, after imposing the additional constraint of unit norm, $\|\mathbf{g}_l\| = 1$. In order to avoid numerical problems due to large magnitude differences on the elements of H, the data is normalized prior to the estimation of G.

5.4.2 Experimental results

A number of tests were conducted, using the reconstruction procedure on synthetic and real data. The synthetic data consists of a set of 3-D points that are projected on two images given a pair of camera projection matrices. Gaussian noise is then added to the image coordinates, as localization noise. Then the fundamental matrix is estimated as described in the previous sections; P_1 and P_2 are computed and the 3D points reconstructed.

Figures 5.6(a) and (b) show the 3D shape used for tests, with superimposed reconstructed points. The left image was obtained with no noise, and the right one with added zero-mean, 1.2 pixel standard deviation Gaussian noise. The effects of increasing amounts of point location noise are shown in Figure 5.6(c), again using zero mean Gaussian noise.

Part of the real data experiments were conducted on the images of Figure 5.7. Theses images where taken with the robotic stereo head *Medusa* [69, 53], whose cameras have common elevation and are approximately $160mm$ apart. The toy house in the center of the

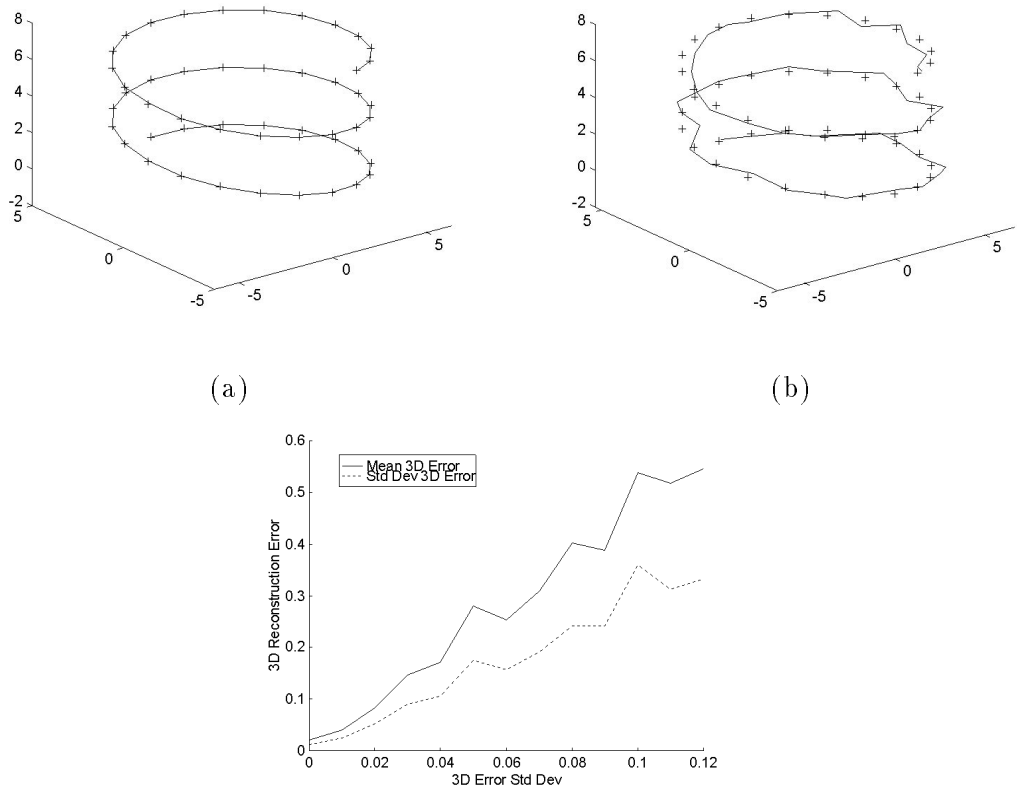
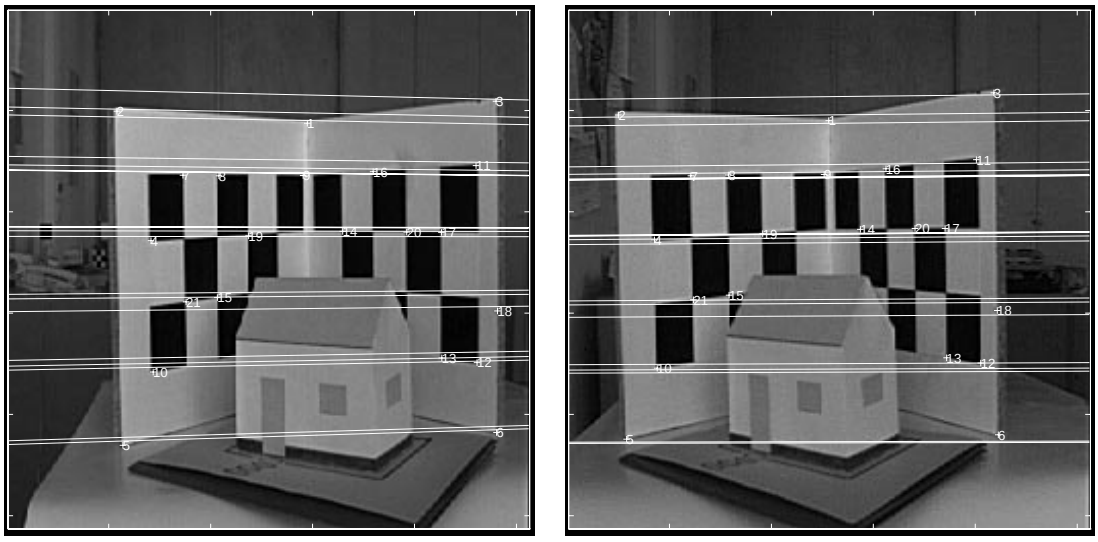


Figure 5.6: Euclidean reconstruction examples, with no added noise(a) and with additive Gaussian noise(b). The + marks indicate the original shape points. In (c) we show the effect of the noise in the reconstructed points.



(a) (b)

Figure 5.7: Toy house setup used for reconstruction with real images. The matched points are shown with the corresponding epipolar lines.

images measures $160 \times 80 \times 120mm$ and is approximately 1 meter away from each camera. A calibration grid was used to provide 21 ground truth points with known locations, which are used for the computation of the H matrix, and for computing the reconstruction error. In the results given below the error measure is the Euclidean distance of the recovered ground points to the correct locations.

Accounting for radial distortion

In order to gain insight on the effect of the inherent radial distortion of the images, some tests have been conducted after image correction.

For the toy house image set, the correction was performed using the calibration procedure described in [28]. This procedure makes use of the Levenberg-Marquardt minimization method [51] for estimating both the projection matrix and the radial distortion parameters. Alternatively, by assotiating a 2-D referential to the points in one of the facets of the calibration grid, one can also use the method described in section 2.3.1. The advantage of the former is that it does not require the calibration points to be on a plane. Since all the grid points are used, the accuracy of the distortion parameters is increased. Although the camera projection matrix is computed, only the distortion parameters are used for radially correcting the image points.

A different set of images was also used, picturing a model castle. The depth in this scene ranges between 40 and $400mm$. The images were obtained from the CMU image archive, and are part of a dataset that includes accurate information about object 3-D

Estimation Method	Average Error (Max) in mm
Standard Calibration	4.3 (9.5)
Linear	18.4 (42.9)
Non–linear	6.2 (14.7)
Standard Calib. with Radial Corr.	4.0 (9.0)
Linear with Radial Corr.	17.0 (34.5)
Non–linear with Radial Corr.	5.2 (12.83)

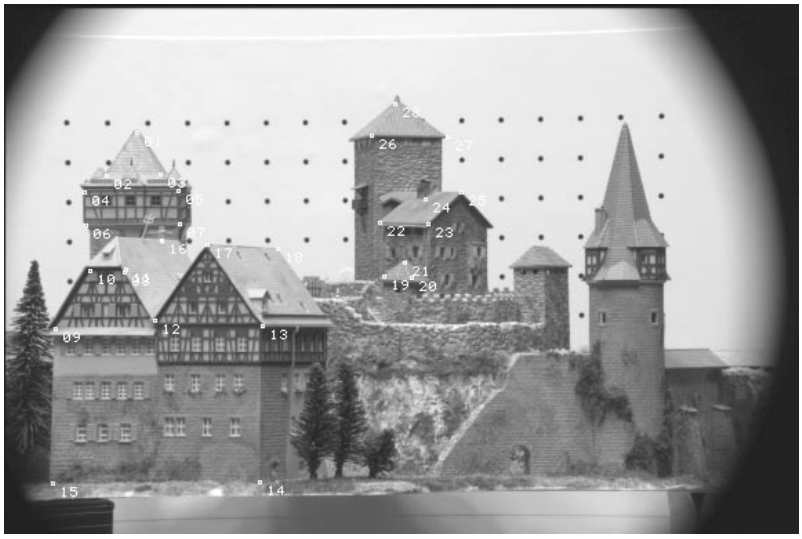
Table 5.1: Average and maximum errors on the reconstruction of the toy house scene for the two criteria, before and after image correction.

locations, thus providing ground-truth. Figure 5.8(a) shows one of the images of the used stereo pair, with marked ground-truth points. Although the images are not corrected for radial distortion, the parameters for the correction are provided with the dataset.

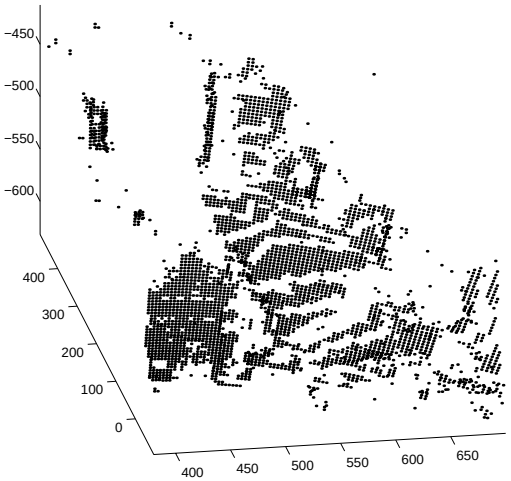
The Euclidean reconstruction using the linear criterion for the toy house scene is presented on Figure 5.9(b), where some of the corner points have superimposed lines for visualization purposes. In order to better evaluate the quality of the two criteria, Figure 5.10 presents top views for the linear criterion before (b) and after radial correction (d), and for the non-linear for the original uncorrected images (c)[1]. When compared with the standard calibration (a) for which the minimum reconstruction error is attained, one can see that the non-linear criterion compares favorably to the linear. Quantively results for the reconstruction error on the ground-truth points are given in table 5.1. It can be seen that image correction improves the results for both criteria.

Similar conclusions can be drawn from the castle scene. Figure 5.11 presents top views with the same configuration as Figure 5.10. The non-linear criterion performs superiorly when compared to the linear, for the uncorrected images. From table 5.2, one can notice that the reconstruction using the linear criterion improves considerably with image correction, when compared with the toy house example. In fact, the errors for the two criteria after image correction are very close and quite low. This can be explained by the fact that the ground-truth data is taken very accurately, and the main source of location uncertainty is lens distortion. Therefore, after image correction, similarly accurate results are attained. This example also shows the higher noise sensitivity of the linear criterion that was discussed in subsection 5.3.1.

[1] The top views of the use of image correction for the standard calibration and non-linear criterion are visually undistinguishable from (a) and (d) respectively, therefore not depicted.

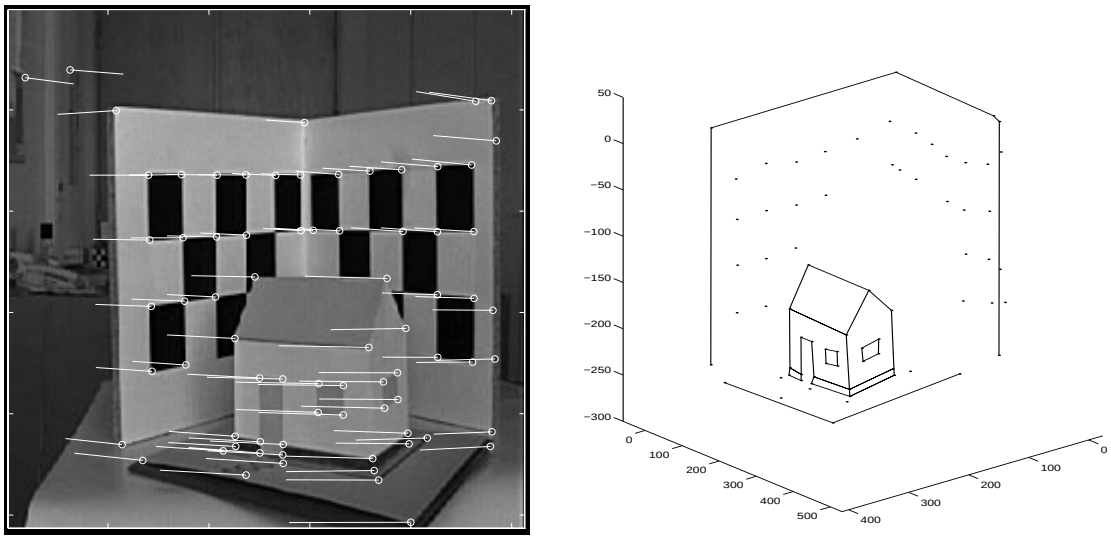


(a)



(b)

Figure 5.8: Castle scene test images: original image with marked ground-truth points(a) and reconstruction using the linear criterion(b).



(a) (b)

Figure 5.9: Euclidean reconstruction with real images: disparity(a) and reconstruction(b) with superimposed lines.

Estimation Method	Average Error (Max) in mm
Standard Calibration	2.8 (8.1)
Linear	17.3 (43.2)
Non–linear	4.4 (13.4)
Standard Calib. with Radial Corr.	0.05 (0.15)
Linear with Radial Corr.	0.05 (0.16)
Non–linear with Radial Corr.	0.06 (0.19)

Table 5.2: Average and maximum errors on the reconstruction of the castle scene for the two criteria, before and after image correction.

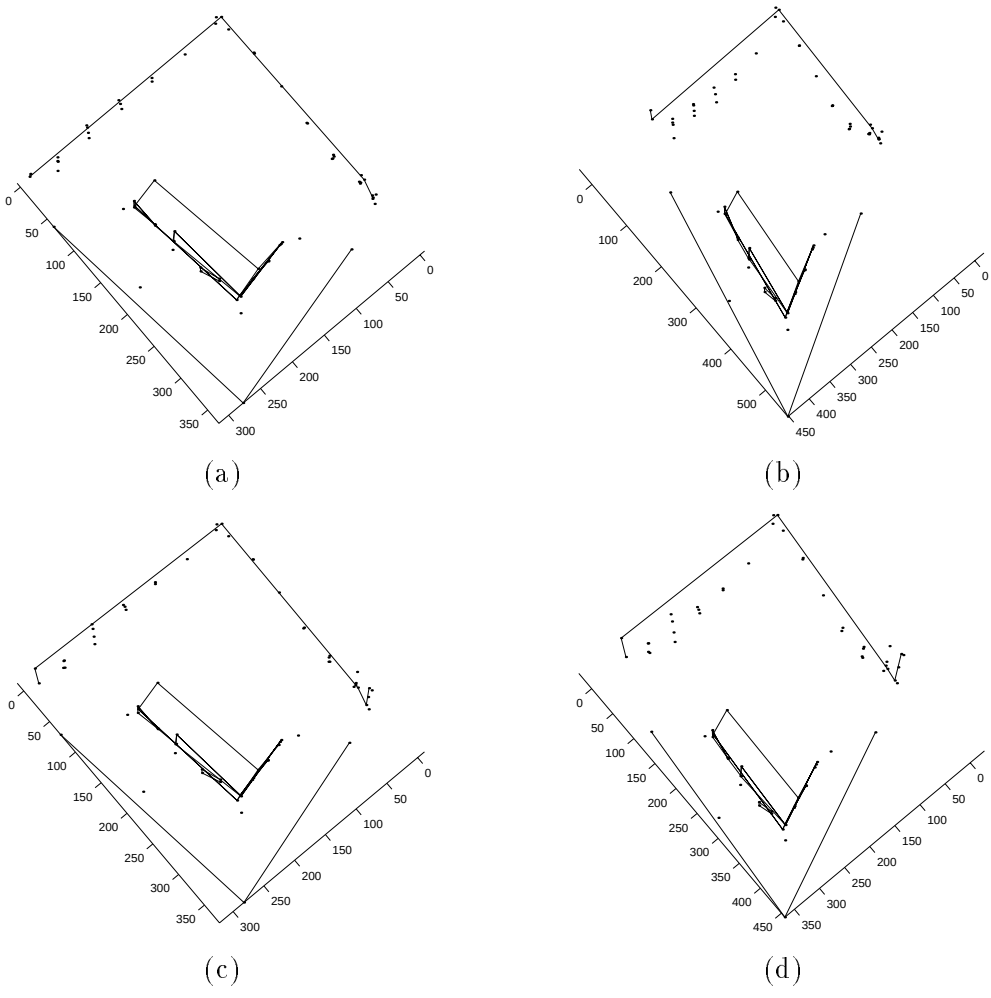


Figure 5.10: Top view of the toy house with superimposed lines: reconstruction from standard calibration(a), linear criterion(b), non-linear criterion(c) and linear criterion after radial correction(d). The scale is in mm.

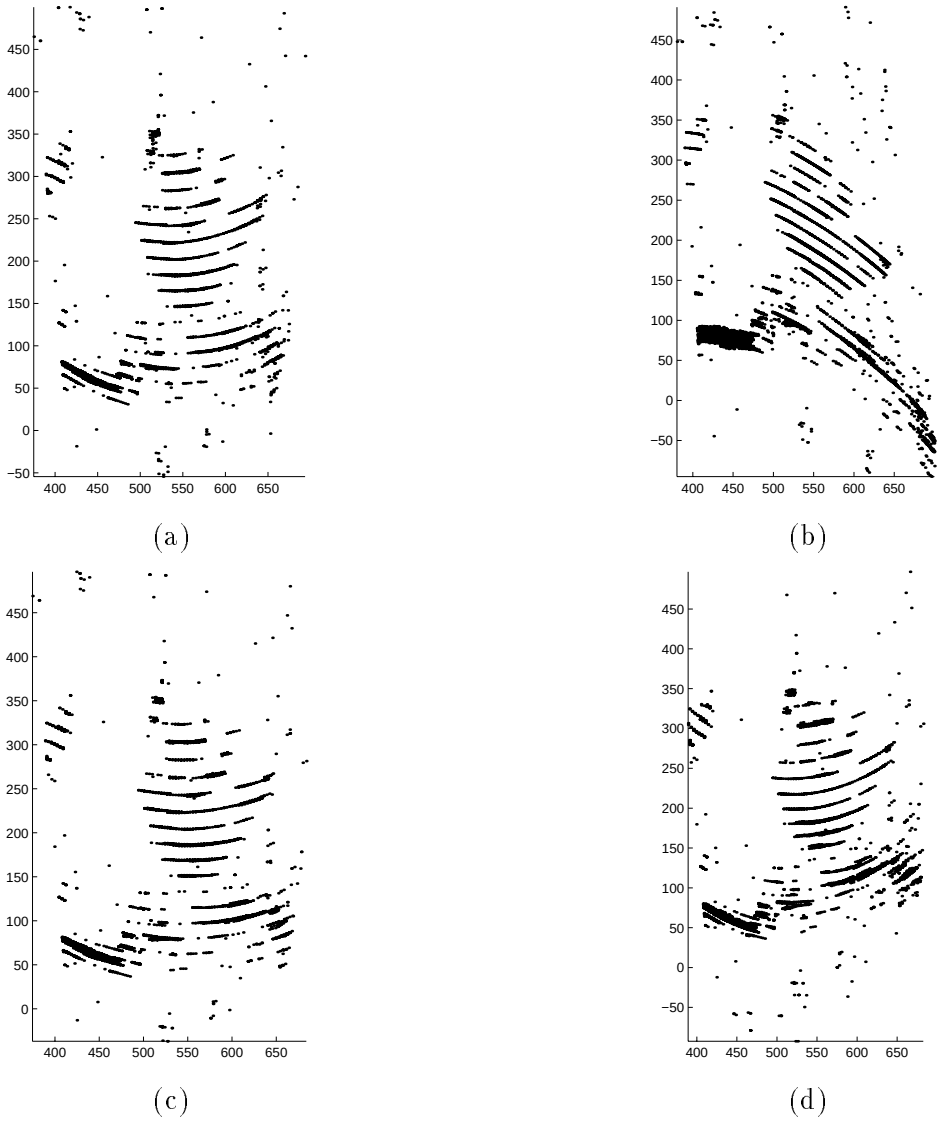


Figure 5.11: Top view of the castle scene: reconstruction from standard calibration(a), linear criterion(b), non-linear criterion(c) and linear criterion after radial correction(d). The scale is in mm.

Chapter 6

Conclusions

This chapter concludes this dissertation with a summary and a discussion of the work presented. In section 6.3 some directions for future developments are given.

6.1 Summary

Chapter 1 presented the goal of this work, the study and implementation of motion estimation algorithms and their application to video mosaicing and 3-D reconstruction.

Chapter 2 was devoted to the presentation of some concepts, definition and models used throughout the thesis. Many of the key concepts can be concisely and elegantly described under the framework of projective geometry. For this reasons the basic properties of projective spaces were explained, preceding a detailed description of the projective camera. This camera model is widely used in computer vision applications not requiring high accuracy modelling of real cameras. A simple method for finding the camera projection matrix was outlined, based on least-squares and solvable using the SVD. Next, it was shown that, for two views of the same planar scene, there exists a simple one-to-one relation between the locations of the corresponding image points. This relation has the form of a collineation and is also referred to as a planar transformation. It defines a model for image motion. Furthermore, it can be computed just from a set of 4 pairs of corresponding image points, again using least-squares. The importance of planar transformations comes from the fact that it can be used to map several images of the same planar scene into a common reference frame. This allows the creation of enhanced panoramic views. A different use of planar transformation is also illustrated, in the correction of the camera lens radial distortion. Radial distortion is not modelled by the projective camera model and constitutes the main source of geometric errors due to the shortcomings of this model. Next, a class of restricted planar transformations was presented. If the image motion is constrained, then a restricted model may better suited in the sense that it explains the

motion equally well, but requiring fewer parameters.

An important section on projective stereo vision is included in this chapter. It presents some concepts and methods useful for the analysis of a static scene using a pair of cameras. The fundamental matrix is introduced in the context of the epipolar geometry. It is shown how it can be computed using least-squares on a set of point correspondences. The topic of uncalibrated reconstruction is discussed, followed by the description of a procedure for Euclidean structure recovery. The presented method uses, as input, a set of matched points for which at least five of the corresponding 3-D points have known coordinates.

Chapter 3 dealt with the issue of robust motion estimation. The purpose of this chapter was twofold. Firstly, it described the two main approaches for motion estimation, namely feature matching and optical-flow. Secondly, it reviewed some of the mostly used robust estimation techniques. Due to the practical orientation of this thesis a strong emphasis was put on the use of robust methods. We are specifically interested in using motion estimates obtained from matched features. Since the matching process is quite error prone on real images, robust matching selection becomes essential. The described methods include iterative re-weighted least squares, M-estimators, case deletion diagnostics and random sampling algorithms. A variant of the Least Median Squares, MEDSERE, was proposed. Experimental testing on both synthetic images under controlled conditions and real images was conducted and was presented on chapter 5. It shown that this algorithm performs favorably when compared to the LMedS. For this reason it was the method of choice for the experimental work of the thesis.

Chapter 4 was devoted to the presentation of results in video mosaicing. It started by a description of the implemented method for corner features detection and area matching. The matching is performed in two steps. A correlation based technique is used for initial location finding. Then, this location is further refined by means of an optical flow technique capable of dealing with feature warping, to attain sub-pixel accuracy. The creation of video mosaics was described in two separate stages. On the registration stage, the MEDSERE algorithm was used for estimating the motion parameters, in the form of a planar transformation matrix. On the rendering stage the effects of the choice of the reference frame and temporal operator were shown.

In order to obtain high-quality, seamless mosaics, an alternative sequence of operations for mosaic creation was also illustrated. Instead of dealing with registration and rendering separately for the entire sequence, the mosaic can be created iteratively, by registering and rendering each individual image. It was found that this procedure reduced the effects of the accumulation of small motion estimation errors, but was very sensitive to the spatial coherence of the mosaic being created.

This chapter finished with several results on video mosaicing and a discussion on their applications.

Chapter 5 reported the application of robust techniques to the estimation of the epipolar geometry, and to 3-D structure recovery. Linear and non-linear minimization criteria were compared using two different parameterizations for the fundamental matrix. The influence of small localization errors and gross mismatches was treated separately. For small localization errors, the non-linear criterion based on the Euclidean distance of each point to the corresponding epipolar line performed better that the linear criterion. The issue of data normalization was addressed, and found to be effective on the linear criterion. As for mismatch errors, several robust algorithms were compared. The proposed MEDSERE algorithm showed very promising results on both outlier rejection and computational effort.

The procedure for Euclidean structure recovery, described on chapter 2, was implemented and tested on both synthetic and real images.

6.2 Discussion

In this thesis we dealt with a number of issues related with image motion estimation using point features. Two main issues are:

- **Robustness**. The effect of noise and outliers on robust and non-robust estimators was studied. It was shown that non-robust techniques have poor results and are completely inadequate for dealing with real image applications. Robust methods were implemented and compared. In the presence of outliers, the random sampling algorithms presented the best results in terms of high breakdown point.
- **Model-based estimation**. Image motion was estimated using motion models derived from geometric considerations of both scene structure and camera locations. The models can be divided into two main classes according to the 3-D information content of the images.
  - If there is no parallax then a one-to-one relation can be established between matched point locations. The most general model is the 8-parameter planar transformation. Image registration can be accomplished.
  - If parallax is present then a relation can be set between points on one of the images and corresponding epipolar lines on the other. The most general model is the fundamental matrix. The recovery of the projective structure of the scene can be performed, just by the analysis of a set of point correspondences.

Based on these two classes, two distinct application areas were considered, namely the creation of video mosaics and the recovery of 3-D structure.

For the creation of video mosaics a method was described which only requires the selection of the most adequate motion model. Feature point selection, matching and

global registration are performed automatically and present robustness to violations of the underlying assumptions of scene planarity and static content. The usefulness of video mosaicing was illustrated with selected mosaic examples for the applications such as aerial imaging, ocean exploration, video coding and enhancement, and panoramic views for virtual reality. Mosaics tend no longer to be considered just as simple visualization tools. By relating the image frames with a common referential, both global spatial and temporal information become easy available. A direct exploitation of this is in video compression and enhancement.

The estimation of the fundamental matrix is currently an area of intensive research. By conveniently encapsulating all the available information on the camera geometry that can be extracted from two views, it is an essential tool in the analysis of uncalibrated images, and the first step towards uncalibrated reconstruction. The impact of the choice of minimization criterion, parametrization, data normalization and radial distortion were issues addressed and studied in this thesis.

6.3 Future Work

Directions for improvement can be pointed out for many of the topics addressed in this thesis.

In the motion analysis presented in this thesis it was assumed that the most suitable motion model was known. Naturally, an important improvement would be the automatic model selection just from image motion analysis. This problem has been addressed in the literature [67, 65] in the context of the fundamental matrix estimation under degeneracy. One of the main problems arises from the fact that the presence of outliers can make degenerate cases appear non-degenerate, which difficults the model selection process.

The proposed MEDSERE algorithm, although it presented good results, was not fully evaluated. A more theoretical analysis of the assessment of the breakpoint is still required. Also, following a rule of thumb [19] on the time relation for new algorithm evaluation, *theory : implementation : testing = 1 : 10 : 100*, further testing is due.

A multitude of improvements can be considered for the mosaic creation:

- In this thesis, only planar retinas have been considered. It can be easily seen that, for image registration for sequences comprising a very wide field of view, cylindrical or spherical retinas are better suited.

- The direct mosaic registration method was found to be very sensitive to spatially incoherent features. A paradigm for reducing the sensitivity may be the introduction of 'elastic' terms on the motion models. By the use of loaded spring models for the elastic terms, a potential energy can be associated with the set of planar transfor-

mations from the previously registered and rendered images. The minimization of the potential energy would change all transformations towards a better image fitting. The main drawback of this approach is the fact that it implies complete mosaic re-rendering for each new image.

- An alternate procedure for mosaic creation can be devised for exploiting the fact that frames with large amount of overlap allow more accurate registration. As an initial step, these frames can be grouped together forming sub-mosaics. A subsequent step is to register and merge the sub-mosaics.

The Euclidean reconstruction was illustrated by the use of ground-truth points. In practical situations this information might not be available. Therefore, it would be quite profitable to be able to incorporate other type of geometric information, thus extending the range of applications. Useful 3D information includes distances between points, parallel lines and angles between coplanar lines. For cases where there is enough information to provide redundancy, a relevant issue is the best choice of geometric restrictions so as to minimize reconstruction uncertainty.

Appendix A

Singular Value Decomposition

The Singular Value Decomposition is a powerful tool for dealing with sets of equations and matrices. It reveals valuable information about the structure of a matrix, and is extremely useful in the analysis of matrix conditioning, and round-off errors in linear equation systems. The SVD is cited[20] as a method with increasing use by many statisticians and control engineers who are reformulating established theoretical concepts under its "light".

Definition 4 (Singular Value Decomposition) *If A is a real $(m \times n)$ matrix then there exists orthogonal matrices U and V,*

$$U = [u_1, \ldots, u_m] \in \mathbb{R}^{m \times m}$$
$$V = [v_1, \ldots, v_n] \in \mathbb{R}^{n \times n}$$

such that

$$U^T A V = diag\,(\sigma_1, \ldots, \sigma_p) \in \mathbb{R}^{m \times n} \qquad p = \min\{m, n\}$$
$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p \geq 0$$

Proof. Can be found in [20].

The scalars σ_i are the *singular values* of matrix A. These are the lengths of the semi-axes of the hiperellipsoid Ax where x is a unitary norm vector. The SVD can, therefore, be used to find the directions mostly "amplified" or "shortened" by the multiplication of A. It is also used for obtaining orthonormal bases for the range and the null-space of A. If σ_r is the smallest non-zero singular value, then A has rank r, its range is spanned by $\{u_1, \ldots, u_r\}$ and the null-space by $\{v_{r+1}, \ldots, v_n\}$.

A.1 Lower rank matrix approximation

The SVD provides a covinient way of approximationg a given matrix, by one of lower rank. The *Frobenius norm* of a real $(m \times n)$ matrix A is defined as the square root of sum of all

squared components,

$$\|A\|_{Frob} = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$$

A simple expression for this norm can be found, using the singular values:

$$\|A\|_{Frob} = \sqrt{\sigma_1^2 + \ldots + \sigma_p^2} \qquad p = \min\{m, n\}$$

Let R be a square $(n \times n)$ non-singular matrix (thus having all singular values positive) with singular value decomposition:

$$R = U \ \mathrm{diag}\,(\sigma_1, \ldots, \sigma_n) \ V^T$$

It has been proven [26] that the $(n \times n)$ matrix R' of rank $n-1$ which minimizes the Frobenius distance $\|R - R'\|_{Frob}$ can be found by zeroing the smallest singular value of R. Thus R' is given by:

$$R' = U \ \mathrm{diag}\,(\sigma_1, \ldots, \sigma_{n-1}, 0) \ V^T$$

Appendix B

Radial Image Correction

The pinhole camera model described on section 2.2 is an approximation of the projection mapping for real cameras. This model presents the useful property of being a linear projective transformation from $\mathbb{P}^3$ into $\mathbb{P}^2$ thus allowing a simple mathematical formulation. However the pinhole model is not valid for applications requiring high accuracy, such as photogrammetry and accurate metrology, as it does not model systematic non-linear image distortion, which is present on most cameras. When performing lens modelling, there are two main kinds of distortion to be taken into account [70]: radial and tangential. For each kind, an infinite series of correction terms is theoretically required. However, it has been shown that, for most off-the-shelf cameras[1] and industrial applications, the non-linearity can be dealt with just by using a single term of radial distortion. A four-step camera calibration procedure allowing radial correction was presented by Tsai in [70]. However, radial distortion can be corrected without full camera calibration. Using Tsai's formulation, a non-linear mapping can be found which relates the observed distorted point projections with the ideal undistorted counterparts for which the pinhole model is valid. The undistorted coordinates for a given distorted image point $\left(u^d, v^d\right)^T$ are,

$$\begin{bmatrix} u^u \\ v^u \end{bmatrix} = \left(\begin{bmatrix} u^d \\ v^d \end{bmatrix} - \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \right) \cdot r^2 \cdot k_1 + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}$$
$$r = \sqrt{\frac{\left(u^d - u_0\right)^2}{s_u^2} + \left(v^d - v_0\right)^2} \tag{B.1}$$

where $(u_0, v_0)^T$ is the location of the principal point, k_1 is the first term of the radial correction series and s_u is a scale factor accounting for differences on the image axes scaling. If these parameters are known then image correction for radial distortion can be performed.

[1] By *off-the-shelf*, we consider the normally used general purpose cameras, as opposed to professional metric cameras used in photogrametry.

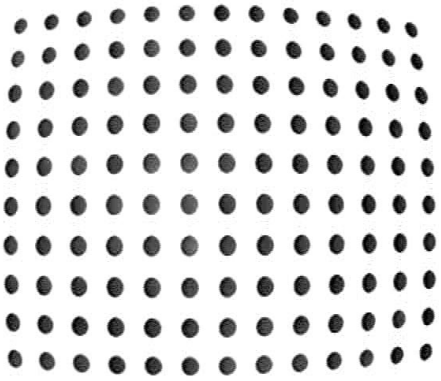


Figure B.1: Example of a planar calibration grid used for radial correction.

An algorithm was implemented for the estimation of the image correction parameters. It uses a planar calibration grid with a set of equally spaced black dots, such as the one depicted on Figure B.1. The centers of the dots are automatically extracted, and a list of the distorted grid point projections is formed. By associating these projections to corresponding undistorted points measured on the referential of the planar grid, the planar transformation between the image plane and the grid plane can be computed.

For a set of N grid points, let $\left(u_i^d, v_i^d\right)^T$ be the image projection of the i^{th} grid point $(U_i, V_i)^T$, and T be the planar transformation computed from all grid points and projections. Let $(u_i^t, v_i^t)^T$ be the mapping of $(U_i, V_i)^T$ on the image plane, using T. Due to the radial distortion, $(u_i^t, v_i^t)^T$ and $\left(u_i^d, v_i^d\right)^T$ will not, in general, be coincident. For a set of radial correction parameters (u_0, v_0, k_1, s_u), a corrected version $(u_i^u, v_i^u)^T$of $(u_i^t, v_i^t)^T$ can be computed, and a cost function can be devised, based on the distances measured on the image plane,

$$c\left(u_0, v_0, k_1, s_u\right) = \sum_i \sqrt{\left(u_i^u - u_i^t\right)^2 + \left(v_i^u - v_i^t\right)^2} \tag{B.2}$$

In order to find the appropriate correction parameters, a non-linear minimization technique is required for minimizing equation (B.2). On the implemented algorithm this is accomplished by means of the Simplex-Downhill method [51].

An example of an image where radial distortion is easily seen is given on Figure B.2 (top). This image was captured by a wide angle camera on top of a *Khepera* robot, and shows the regions near the frame corners to be bent inwards. The bottom images illustrates the result of the algorithm. The corrected image was created by linearly interpolating the intensity values at the points mapped onto fractional coordinates. After correction, the central region kept the original size, while the remaining area was gradually zoomed.

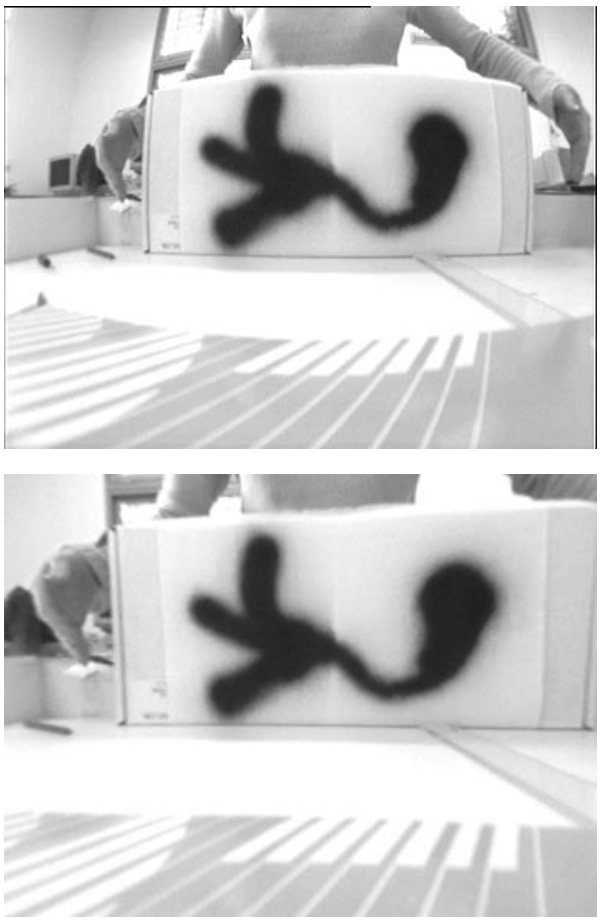


Figure B.2: Example of radial correction: original image (top) and corrected (bottom).

Appendix C

Original Sequences

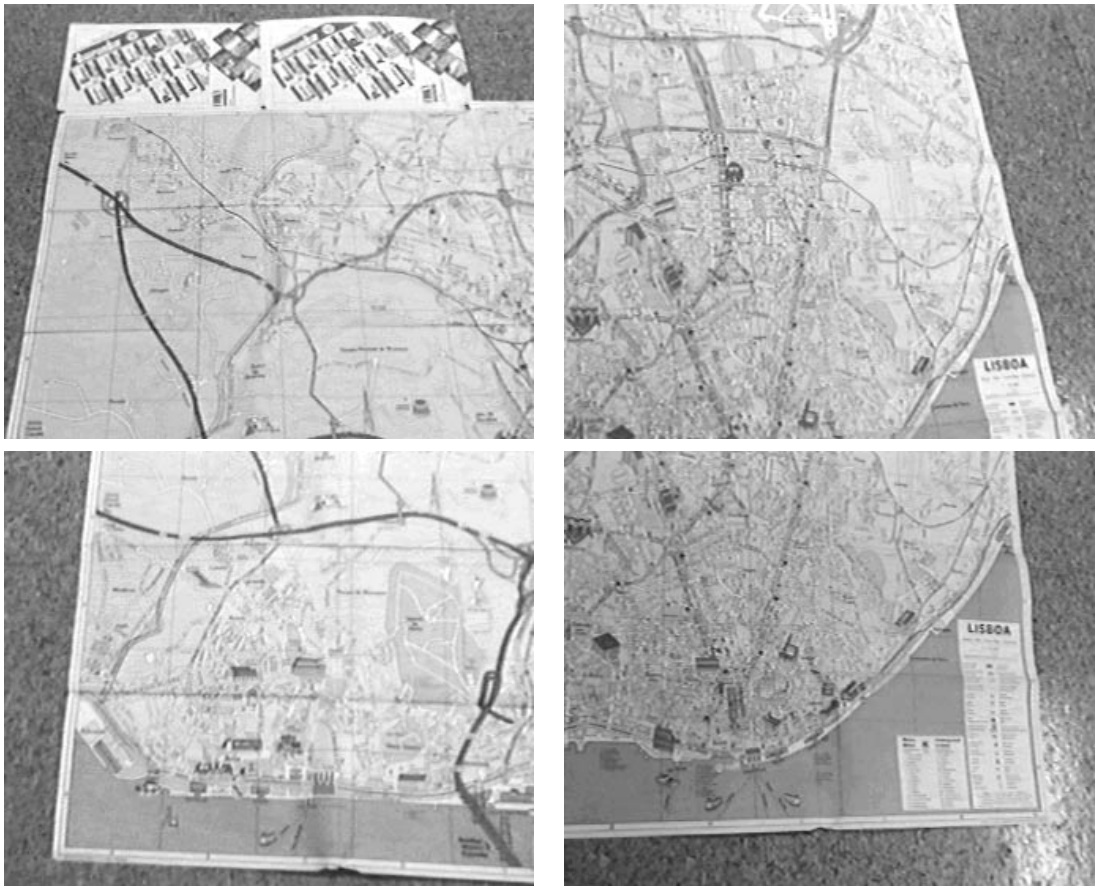


Figure C.1: The *map* sequence comprises 67 frames of a street map. It was captured at close range by a translating and rotating camera, thus inducing noticeable perspective distortion. The map was scanned following an inverted S-shape, starting from the map upper left corner and finishing on the diagonally opposite corner. The original images are 336×276.

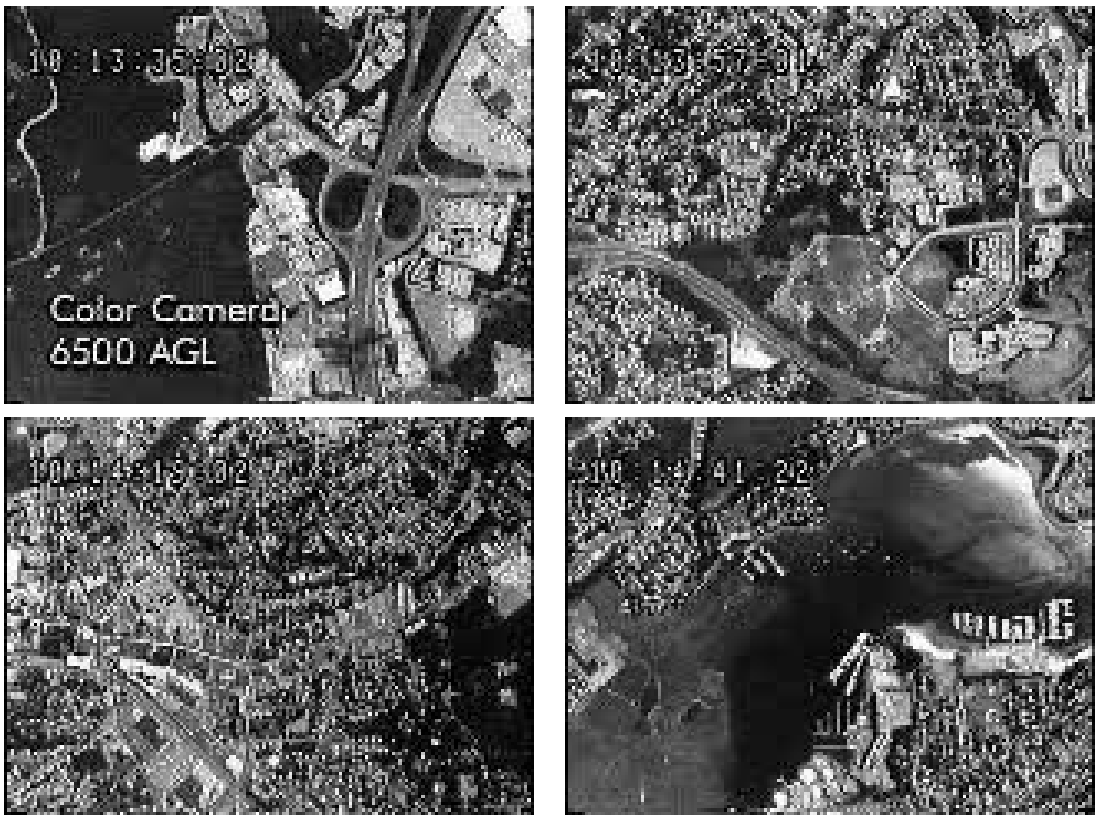


Figure C.2: The *Qn* aerial sequence was captured by a high altitude plane flying over an urban scenario. The 50 frames have a superimposed time-stamp and a horizontally shifted lower scan-line. The images are 160×120.

Figure C.3: The *Arli* aerial sequence was captured by a high altitude plane flying over the Arlington district, in Washington. The 59 frames have a superimposed time-stamp and a horizontally shifted lower scan-line. The images are 160×120.

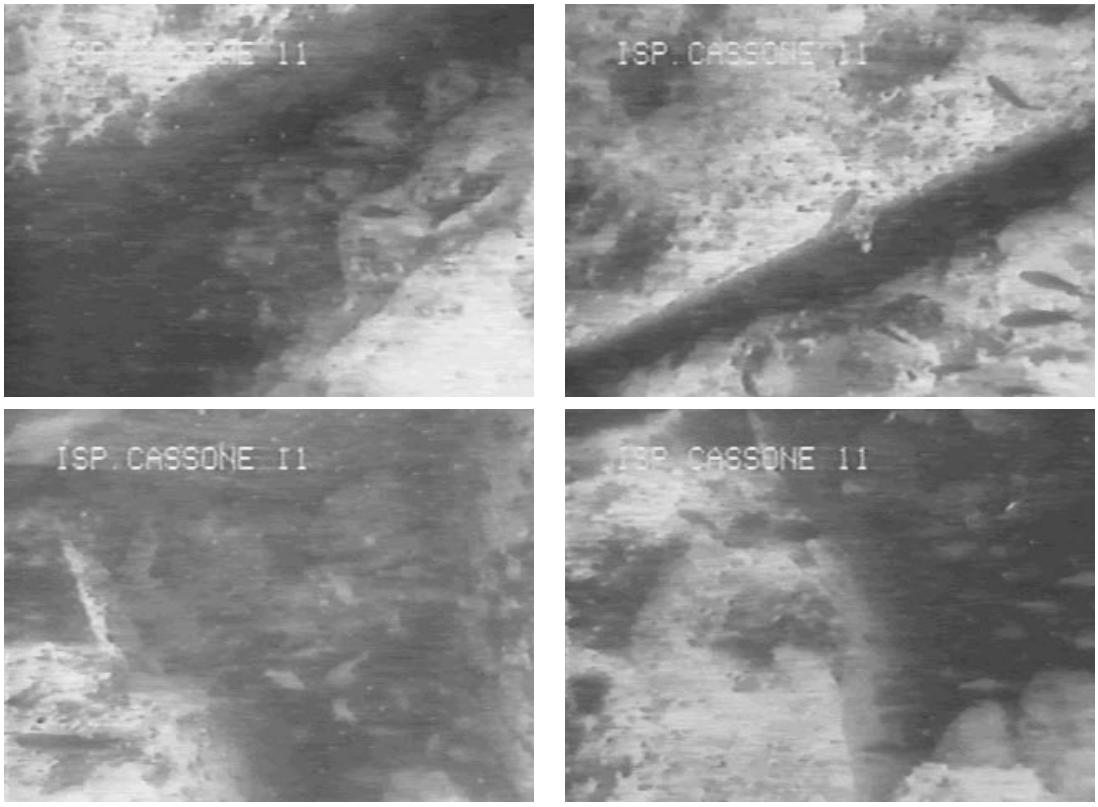


Figure C.4: The *draf1* underwater sequence was captured by manually controlled ROV, and depicts a man-made construction. The camera is moving along a fracture inside which some rocks can be seen. The fracture provides noticeable depth variations as opposed to the almost planar surrounding sea bed. Some moving fish can be seen. The sequence comprises 101 images of 320×240 pixels.

Figure C.5: The *football* sequence was captured from a TV broadcast, and shows 8 seconds of a football game during the goal. The camera is rotating and zoomming in towards the end of the sequence. It comprises 43 images of 320×240 pixels.

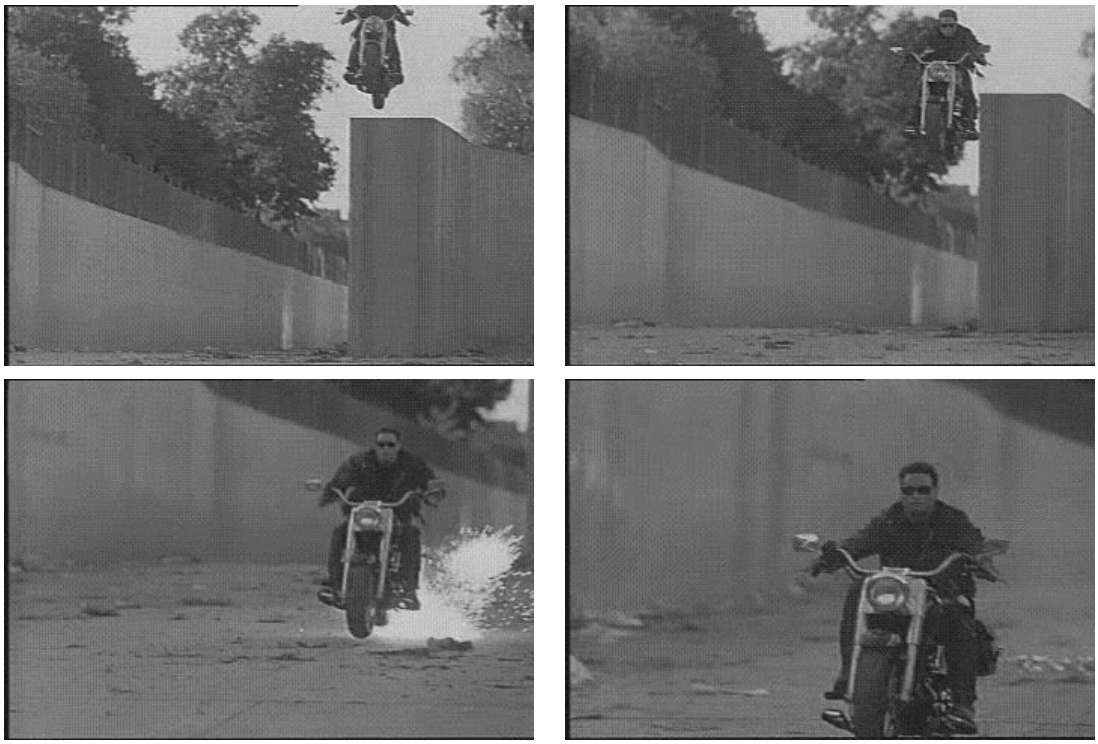


Figure C.6: The *bike* sequence shows a stunt bike driver moving closer to the camera. The sequence is part of the film *Terminator 2* and was obtained from a public domain archive. It comprises 56 images of 352×240 pixels.

Figure C.7: The *peneda* sequence was captured by a hand held camcorder following the hill tops, in *Serra da Peneda* in the north part of Portugal. It contains 90 images of 340×240 pixels.

Figure C.8: The *VisLab* sequence was recorded by a camera on top of a tripod and rotating around the vertical axis, thus inducing simple sideways image motion. It contains 84 images of 336×276 pixels.

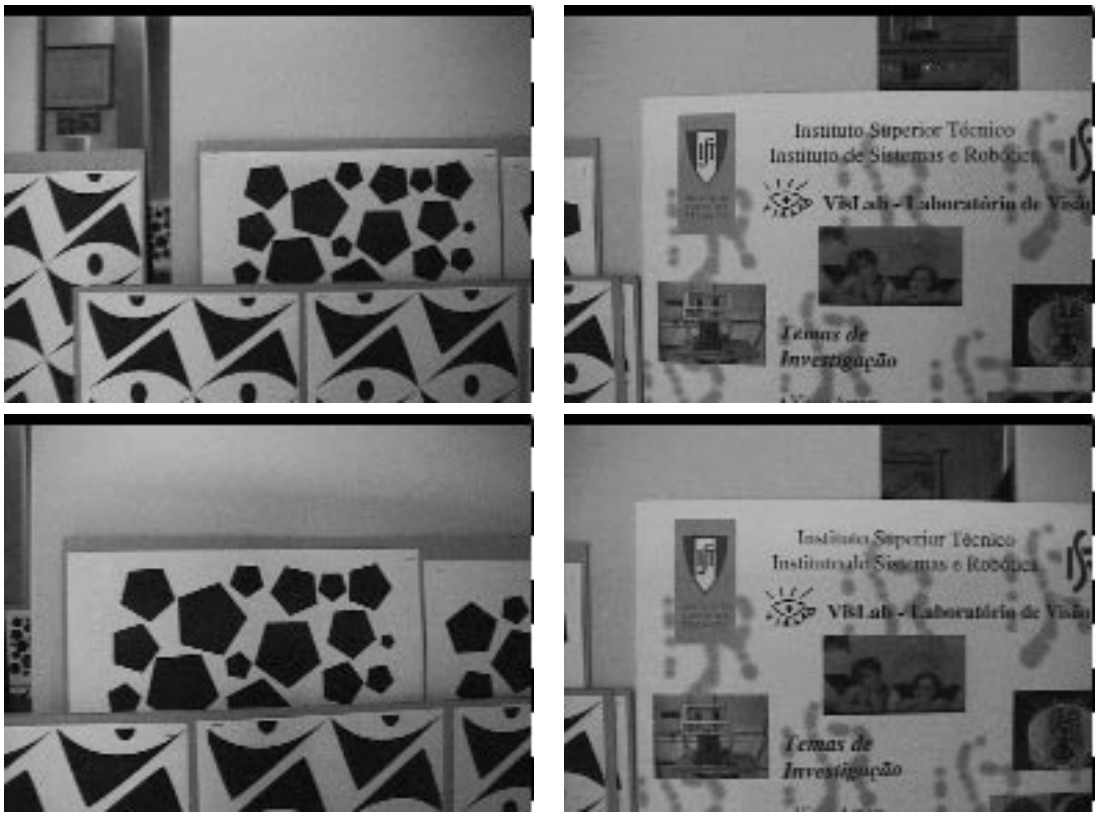


Figure C.9: The *LabMate* sequence contains two sets of images captured by a camera on top of a *TRC LabMate* mobile platform. During the acquisition of the first set (comprising 45 images) the platform moved along a corridor, keeping constant heading and distance to the wall (top row). On the second set of 5 images depicturing part of the same scene, the camera rotated and got closer to the wall (bottom row). All the images are 192×144 pixels.

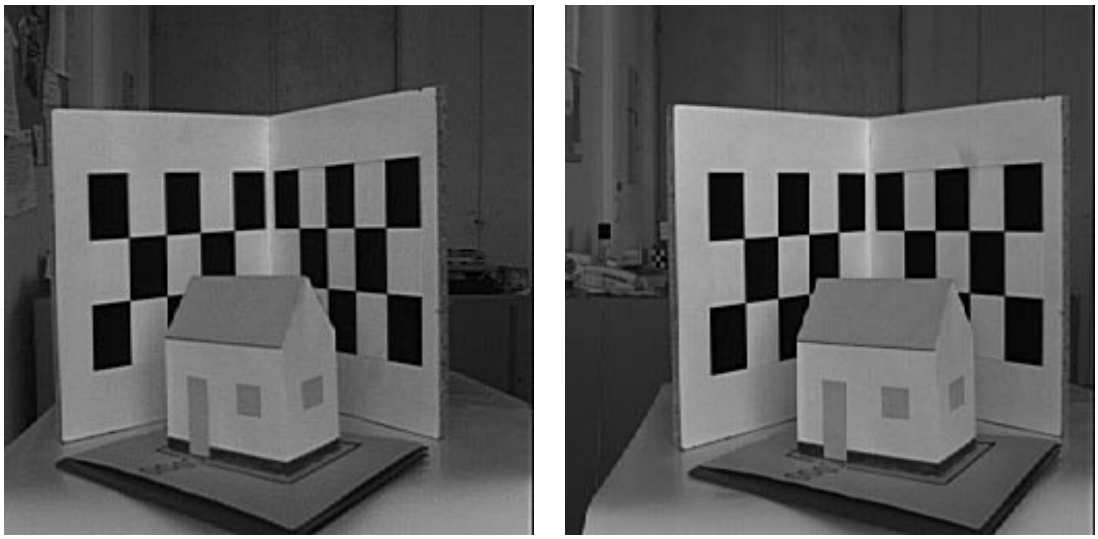


Figure C.10: The *toyhouse* stereo pair was captured by two cameras of a stereo head with horizontal baseline. The cameras are $160mm$ apart and the toy house is approximately 1 meter away from both. The images are displayed for cross-eye fusion.

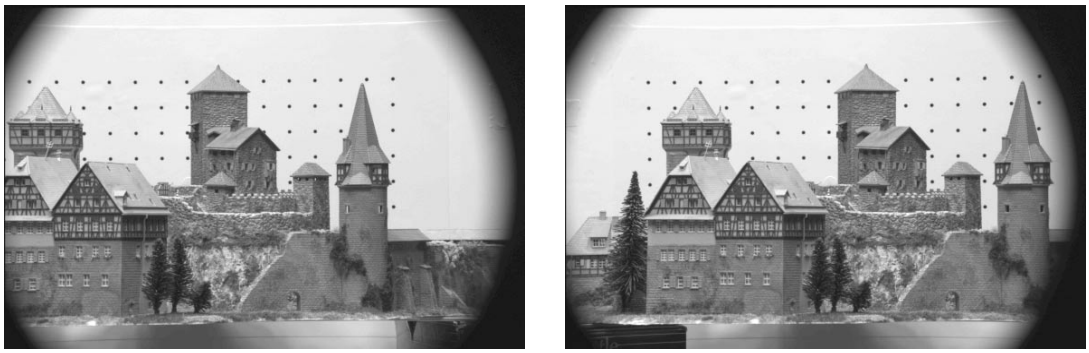


Figure C.11: The *castle* stereo pair is part of a sequence where a static model castle is captured by a moving camera. The depth in this scene ranges between 40 and $400mm$. The images were obtained from the CMU image archive, and are part of a dataset that includes accurate information about object 3-D locations. The images are displayed for cross-eye fusion.

Bibliography

[1] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2(4):283–310, 1989.

[2] A. Arsénio and J. Marques. Performance analysis and characterization of matching algorithms. In *Proc. of the International Symposium on Intelligent Robotic Systems*, Stockholm, Sweden, July 1997.

[3] S. Ayer. *Sequential and Competitive Methods for Estimation of Multiple Motions.* PhD thesis, École Polytechnique Fédérale de Lausanne, 1995.

[4] D. Ballard and C. Brown. *Computer vision.* Prentice-Hall, London, 1982.

[5] J.K. Barron, D.J. Fleet, and S.S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–78, 1994.

[6] P. Beardsley, A. Zisserman, and D.W. Murray. Navigation using affine structure from motion. In *Proc. of the 3rd. European Conference on Computer Vision*, volume II, pages 85–96, Stockholm, Sweeden, May 1994. Springer–Verlag.

[7] M. Bertero, T. Poggio, and V. Torre. Ill-posed problems in early vision. *Proceedings of the IEEE*, 76(8):869–889, 1988.

[8] M. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Proc. of the 4th. International Conference on Computer Vision*, Berlin, Germany, May 1993.

[9] M. Bober and J. Kittler. Robust motion analysis. In *Proc. of the IEEE Int. Conference on Computer Vision and Pattern Recognition*, Washington, USA, June 1994.

[10] B. Boufama and R. Mohr. Epipole and fundamental matrix estimation using virtual parallax. In *Proc. of IEEE Int. Conference on Computer Vision*, Cambridge MA, USA, June 1995.

[11] R. Deriche, Z. Zhang, Q.-T. Luong, and O.D. Faugeras. Robust recovery of the epipolar geometry for an uncalibrated stereo rig. In *Proc. of 3rd European Conference on Computer Vision*, volume I, pages 567–576, Stockholm, Sweden, May 1994.

[12] F. Devernay and O. Faugeras. From projective to euclidean reconstruction. In *Proc. of the International Conference on Computer Vision and Pattern Recognition*, pages 264–269, San Francisco, CA, June 1996.

[13] F. Eryurtlu and J. Kittler. A comparative study of grey level corner detectors. In J. Vanderwalle, R. Boite, M. Moonen, and A. Oosterlinck, editors, *Signal Processing VI: Theories and Applications*. Elsevier, 1992.

[14] O. Faugeras. What can we see in three dimensions with an uncalibrated stereo rig ? In *Proc. of the 2nd. European Conference on Computer Vision*, Santa Margherita, Italy, May 1992.

[15] O. Faugeras. *Three Dimensional Computer Vision*. MIT Press, 1993.

[16] O. Faugeras. Stratification of three-dimensional vision: projective, affine and metric representations. *Journal of the Optical Society of America A*, 12(3):465–484, March 1995.

[17] O. Faugeras, P. Fua, B. Hotz, R. Ma, L. Roberts, M. Thonnat, and Z. Zhang. Quantitative and qualitative comparision of some area and feature-based stereo algorithms. In W. Förstner and Ruwiedel, editors, *Robust Computer Vision*. Wichmann, Bonn, Germany, March 1992.

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

[19] W. Förstner. 10 pros and cons against performance characterisation of vision algorithms. In *Proc. of the Workshop on Performance Characteristics of Vision Algorithms*, Cambridge, UK, April 1996.

[20] G. Golub and C. van Loan. *Matrix Computations*. The John Hopkins University Press, 1989.

[21] N. Gracias and J. Santos-Victor. Robust estimation of the fundamental matrix and stereo correspondences. In *Proc. of the International Symposium on Intelligent Robotic Systems*, Stockholm, Sweden, July 1997.

[22] C. Harris. Determination of ego-motion from matched points. In *Proceedings Alvey Conference*, Cambridge, UK, 1987.

[23] R. Hartley. Estimation of relative camera positions for uncalibrated cameras. In *Proc. of the 2nd. European Conference on Computer Vision*, Santa Margherita, Italy, May 1992.

[24] R. Hartley. Stereo from uncalibrated cameras. In *Proc. of the IEEE International Conference on Computer Vision and Pattern Recognition CVPR*, Urbana Champaign, USA, June 1992.

[25] R. Hartley. In defence of the 8-point algorithm. In *Proc. of the 5th IEEE International Conference on Computer Vision*, Cambridge MA, USA, June 1995.

[26] R. Hartley. In defence of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, June 1997.

[27] R. Haywood. Acquisition of a micro scale photographic survey using an autonomous submersible. In *Proc. of the OCEANS 86 Conference*, New York NY, USA, 1986.

[28] J. Heikkilä and Olli Silvén. A four–step camera calibration procedure with implicit image correction. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, June 1997. IEEE Computer Society Press.

[29] A. Heyden and K. Åström. Euclidean reconstruction from image sequences with varying and unknown focal length and principal point. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, June 1997. IEEE Computer Society Press.

[30] B. Horn. *Robot vision*. MIT Press, 1986.

[31] B. Horn. Motion fields are hardly ever ambiguous. *International Journal of Computer Vision*, 1(3):259–274, October 1987.

[32] B. Horn and B. Shunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[33] M. Irani, P. Anandan, and S. Hsu. Mosaic based representations of video sequences and their applications. In *Proc. of the 5th IEEE International Conference on Computer Vision*, Cambridge, Massachusetts, June 1995. IEEE Computer Society Press.

[34] A. Jain, editor. *Fundamentals Digital Image Processing*. Prentice Hall, 1989.

[35] A. Jepson and M. Black. Mixture models for optical flow computation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 760–761, New York, May 1993.

[36] K. Kanatani. Statistical bias of conic fitting and renormalization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(3):320–326, March 1994.

[37] K. Kanatani, editor. *Statistical optimization for geometric computation : theory and practice*. Elsevier Science, 1996.

[38] S. Kang. A survey of image-based rendering techniques. Technical Report CRL 97/4, Digital Equipment Corporation, August 1997.

[39] S. M. Kay, editor. *Fundamentals of Statistical Signal Processing - Estimation Theory*. Prentice Hall, 1993.

[40] F. Li, M. Brady, and C. Wiles. Fast computation of the fundamental matrix for an active stereo vision system. In B. Buxton and R. Cipolla, editors, *Proc. of the 4th European Conference on Computer Vision*, volume I, pages 157–166, Cambridge, UK, April 1996.

[41] H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.

[42] Q. Luong, R. Deriche, O. Faugeras, and T. Papadopoulo. On determining the fundamental matrix: an analysis of different methods and experimental results. Technical Report 1894, INRIA, 1993.

[43] Q. Luong and O. Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *International Journal of Computer Vision*, 17(1):43–75, January 1996.

[44] S. Marapane and M. Trivedi. Multi–primitive hierarchical (MPH) stereo analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(3):227–240, March 1994.

[45] R. Marks, S. Rock, and M. Lee. Real-Time video mosaicking of the ocean floor. *IEEE Journal of Oceanic Engineering*, 20(3):229–241, July 1995.

[46] M. Massey and W. Bender. Salient stills: Process and practice. *IBM Systems Journal*, 35(3 and 4):557–573, 1996.

[47] P. Meer, D. Mintz, A. Rosenfeld, and D. Kim. Robust regression methods for computer vision: a review. *Int. Journal of Computer Vision*, 6(1):59–70, 1991.

[48] R. Mohr, B. Boufama, and P. Brand. Accurate projective reconstruction. In J. Mundy, A. Zisserman, and D. Forsyth, editors, *Proc. of the 2nd Joint European-US Workshop on Applications of Invariance in Computer Vision*, pages 257–276, Azores, Portugal, October 1993. Springer–Verlag.

[49] R. Mohr and B. Triggs. Projective geometry for image analysis. Tutorial given at ISPRS96, September 1996.

[50] S. Olsen. Epipolar line estimation. In *Proceedings of the 2nd. European Conference on Computer Vision*, Santa Margherita, Italy, May 1992.

[51] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1988.

[52] P. Rousseeuw and A. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, 1987.

[53] J. Santos-Victor. *Visual Perception for Mobile Robots: From Percepts to Behaviours*. PhD thesis, Universidade Técnica de Lisboa, Lisbon, Portugal, November 1994.

[54] H. Sawhney, S. Ayer, and M. Gorkani. Model-based 2D&3D dominant motion estimation for mosaicing and video representation. In *Proc. of the 5th IEEE International Conference on Computer Vision*, Cambridge, Massachusetts, June 1995. IEEE Computer Society Press.

[55] J. Semple and G. Kneebone. *Algebraic projective geometry*. Oxford University Press, 1952.

[56] L. Shapiro. *Affine Analysis of Image Sequences*. Oxford University Press, 1995.

[57] C. Silva and J. Santos-Victor. Direct egomotion estimation. In *Proc. of the 13th Int. Conference on Pattern Recognition*, Vienna, Austria, August 1996.

[58] C. Silva and J. Santos-Victor. Robust egomotion estimation from the normal flow using search subspaces. Submitted to IEEE Transactions on PAMI in 1996, 1996.

[59] D. Sinclair, A. Blake, and D. Murray. Robust estimation of egomotion from normal flow. *International Journal of Computer Vision*, 13(1):57–70, September 1994.

[60] M. Spetsakis and Y. Aloimonos. Optimal computing of structure from motion using point correspondences in two frames. In *Proc. of the Second International Conference on Computer Vision*, pages 449–453, Tampa, FL, USA, December 1988.

[61] M. Spetsakis and Y. Aloimonos. A multi-frame approach to visual motion perception. *International Journal of Computer Vision*, 6(3):245–255, August 1991.

[62] R. Szeliski. Image mosaicing for tele-reality applications. Technical Report CRL 94/2, Digital Equipment Corporation, May 1994.

[63] R. Szeliski and S. Kang. Direct methods for visual scene reconstruction. In *Proc. of the IEEE Workshop on Representations of Visual Scenes*, Cambridge, Massachusetts, June 1995.

[64] L. Teodosio and W. Bender. Salient video stills: content and context preserved. In *Proceedings of the ACM Multimedia Conference*, Anaheim, August 1993.

[65] P. Torr. *Outlier Detection and Motion Segmentation.* PhD thesis, Dept. of Engineering Science, University of Oxford, 1995.

[66] P. Torr and D. Murray. The development and comparision of robust methods for estimating the fundamental matrix. *International Journal of Computer Vision*, 24(3):271–300, September/October 1997.

[67] P. Torr, A. Zisserman, and S. Maybank. Robust detection of degenerate configurations for the fundamental matrix. In *Proc. of the 5th IEEE International Conference on Computer Vision*, Cambridge, Massachusetts, June 1995. IEEE Computer Society Press.

[68] B. Triggs. Autocalibration and the Absolute Quadric. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, June 1997. IEEE Computer Society Press.

[69] F.van Trigt, J. Santos-Victor, and J. Sentieiro. Medoesa : Design and construction of a stereo head for active vision. Technical Report rpt/07/93, VISLAB/ISR, Instituto Superior Técnico, 1993.

[70] R. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv camera and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4):323–344, 1987.

[71] R. Tsai and T. Huang. Uniqueness and estimation of three dimensional motion parameters of rigid objects with curved surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:13–27, 1984.

[72] J. Weng, N. Ahuja, and T. Huang. Optimal motion and structure estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):864–884, September 1993.

[73] J. Weng, T. Huang, and N. Ahuja. Motion and structure from two perspective views: Algorithms, error analysis, and error estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(5):451–476, May 1989.

[74] C. Wiles and M. Brady. On the appropriateness of camera models. In B. Buxton and R. Cipolla, editors, *Proc. of the 4th European Conference on Computer Vision*, volume II, pages 228–237, Cambridge, UK, April 1996. Springer–Verlag.

[75] O. Faugeras Z. Zhang, R. Deriche and Q. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. Technical Report RR 2273, INRIA, may 1994.

[76] Q. Zheng and R. Chellappa. A computational vision approach to image registration. *IEEE Transactions on Imaging Processing*, 2(3):311–326, July 1993.

[77] A. Zisserman and S. Maybank. A case against epipolar geometry. In *Proc. of the 2nd Joint European-US Workshop on Applications of Invariance in Computer Vision*, pages 69–88, Azores, Portugal, October 1993.