Applications of Non-Metric Vision to some Visually Guided Robotics Tasks

Martial Hebert, Cyril Zeller, Olivier Faugeras, Luc Robert

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Applications of non-metric vision to some visually guided robotics tasks

Luc Robert, Cyril Zeller, Olivier Faugeras

and

Martial Hébert

N° 2584

Juin 1995

PROGRAMME 4

ISSN 0249-6399

Applications of non-metric vision to some visually guided robotics tasks*

Luc Robert, Cyril Zeller, Olivier Faugeras**
and
Martial Hébert***

Programme 4 — Robotique, image et vision
Projet Robotvis

Rapport de recherche n°2584 — Juin 1995 — 54 pages

Abstract: We usually think of the physical space as being embedded in a three-dimensional Euclidean space where measurements of lengths and angles do make sense. It turns out that for artificial systems, such as robots, this is not a mandatory viewpoint and that it is sometimes sufficient to think of the physical space as being embedded in an affine or even projective space. The question then arises of how to relate these geometric models to image measurements and to geometric properties of sets of cameras.

We first consider that the world is modelled as a projective space and determine how projective invariant information can be recovered from the images and used in applications. Next we consider that the world is an affine space and determine how affine invariant information can be recovered from the images and used in applications. Finally, we do not move to the Euclidean layer because this is the layer where everybody else has been working with from the early days on, but rather to an intermediate level between the affine and Euclidean ones. For each of the three layers we explain various calibration procedures, from fully automatic ones to ones that use some a priori information. The calibration increases in difficulty from the projective to the Euclidean layer at the same time as the information that can be recovered from the images becomes more and more specific and detailed. The two main applications that we consider are the detection of obstacles and the navigation of a robot vehicle.

Key-words: projective, affine, Euclidean geometry, stereo, motion, self-calibration, robot navigation, obstacle avoidance

(Résumé : tsvp)

*This work was partially funded by the EEC under Esprit Project 6448 (*VIVA*) and 8878 (*REALISE*).
**Email : {lucr,faugeras,zeller}@sophia.inria.fr
***Email : hebert@cs.cmu.edu

Unité de recherche INRIA Sophia-Antipolis
2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex (France)
Téléphone : (33) 93 65 77 77 – Télécopie : (33) 93 65 77 65

Applications de la vision non-métrique à des tâches robotiques guidées par la vision

Résumé : Nous considérons souvent que l'epace physique peut être représenté par un espace tridimensionnel Euclidien où les notions de longueur et d'angle ont un sens. Il s'avère que pour des systèmes artificiels comme les robots, ceci n'est pas nécessaire, et qu'il est parfois suffisant de considérer l'espace physique comme représenté par un espace affine ou projectif. Le problème se pose alors de savoir comment relier ces modèles géométriques aux mesures effectuées dans les images, et aux propriétés géométriques des caméras.

Nous considérons tout d'abord que le monde est représenté par l'espace projectif, et nous déterminons comment de l'information projectivement invariante peut être extraite des images et utilisée dans un certain nombre d'applications. Ensuite, nous considérons que le monde est un espace affine, et nous déterminons comment des informations invariantes au sens affine peuvent être extraites des images et utilisées dans des applications robotiques. Enfin, nous n'allons pas jusqu'au niveau Euclidien, car c'est le niveau que tout le monde a considéré depuis le début, mais nous nous limitons à un niveau intermédiaire entre l'affine et l'Euclidien. Pour chacun de ces trois niveaux, nous décrivons un certain nombre de procédures de calibration, certaines étant complètement automatiques, d'autres utilisant des connaissances *a-priori*. lorsque l'on passe du niveau projectif au niveau Euclidien, la calibration devient de plus en plus difficile, mais l'information que l'on peut extraire des images est de plus en plus spécifique et détaillée. Les deux principales applications que nous considérons sont la détection d'obstacles et la robotique mobile.

Mots-clé : géométrie projective, affine, Euclidienne, stéréo, mouvement, auto-calibration, robotique mobile, évitement d'obstacles

1 Introduction

Many visual tasks require recovering 3-D information from sequences of images. This chapter takes the natural point of view that, depending on the task at hand, some geometric information is relevant and some is not. Therefore, the questions of exactly what kind of information is necessary for a given task, how it can be computed from the data, after which preprocessing steps, are central to our discussion. Since we are dealing with geometric information, a very natural question that arises from the previous ones is the question of the invariance of this information under various transformations. An obvious example is viewpoint invariance which is of course of direct concern to us.

It turns out that viewpoint invariance can be separated into three components: invariance to changes of internal parameters of the cameras i.e to some changes of coordinates in the images, invariance to some transformations of space, and invariance to perspective projection via the imaging process.

Thus, the question of viewpoint invariance is mainly concerned with the invariance of geometric information to certain two- and three-dimensional transformations. It turns out that a neat way to classify geometric transformations is by considering the projective, affine, and Euclidean groups of transformations. These three groups are subgroups of each other and each one can be thought of as determining an action on geometric configuration. For example, applying a rigid displacement to a camera does not change the distances between points in the scene but in general changes their distances in the images. These actions determine three natural layers, or strata in the processing of visual information. This has the advantages of 2) of identifying the 3-D information that can thereafter be recovered from those images and 1) clearly identifying the information that needs to be collected from the images in order to "calibrate" the vision system with respect to each the three strata.

Point 1) can be considered as the definition of the preprocessing which is necessary in order to be able to recover 3-D geometric information which is invariant to transformations of the given subgroup. Point 2) is the study of how such information can effectively be recovered from the images. This viewpoint has been adopted in [4]. In this chapter we follow the same track and enrich it considerably on two counts. From the theoretical viewpoint, the analysis is broadened to include a detailed study of the relations between the images and a number of 3-D planes which are then used in the development of the second viewpoint (absent in [4]) the viewpoint of the applications.

To summarize, we will first consider that the world is modeled as a projective space and determine how projective invariant information can be recovered from the images and used in applications. Next we will consider that the world is an affine space and determine how affine invariant information can be recovered from the images and used in applications. Finally, we will not move to the Euclidean layer because this is the layer where everybody else has been working with from the early days on, but rather to an intermediate level between the affine and Euclidean ones. For each of the three layers we explain various calibration procedures, from fully automatic ones to ones that use some a priori information. Clearly, the calibration increases in difficulty from the projective to the Euclidean layer at the same time as the information that can be recovered from the images becomes more and more specific and detailed. The two main applications that we consider are the detection of obstacles and the navigation of a robot vehicle.

Section (2) describes the model used for the camera and its relation to the three-dimensional scene. After deriving from this model a number of relations between two views, we analyze the links

between the partial knowledge of the model's parameters and the invariant properties of the reconstructed scene from those two views. Section (3) describes techniques to compute some of the model's parameters without assuming full calibration of the cameras. Section (4) describes the technique of the rectification with respect to a plane. This technique, which does not require full calibration of the cameras either, allows to compute information on the structure of the scene and is at the basis of all the remaining sections. Section (5) shows how to locate 3-D points with respect to a plane. Section (6) shows how to compute local surface orientations. Lastly, section (7) presents several obstacle avoidance and navigation applications based on a partially calibrated stereo rig.

2 Stratification of the reconstruction process

In this section we investigate the relations between the three-dimensional structure of the scene and its images taken by one or several cameras. We define three types of three-dimensional reconstructions that can be obtained from such views. These reconstructions are obtained modulo the action of one of the three groups, Euclidean, affine, and projective considered as acting on the scene. For example, to say that we have obtained a projective (resp. affine, Euclidean) reconstruction of the scene means that the *real* scene can be obtained from this reconstruction by applying to it an unknown projective (resp. affine, Euclidean) transformation. Therefore the only properties of the scene that can be recovered from this reconstruction are those which are *invariant* under the group of projective (resp. affine, Euclidean) transformations. A detailed analysis of this stratification can be found in [4].

We also relate the possibility of obtaining such reconstructions to the amount of information that needs to be known about the set of cameras in a quantitative manner, through a set of geometric parameters such as the fundamental matrix [5] of a pair of cameras, the collineation of the plane at infinity, and the intrinsic and extrinsic parameters of the cameras.

We first recall some properties of the classical *pinhole* camera model, which we will use in the remainder of the chapter. Then, we analyze the dissimilarity (disparity) between two pinhole images of a scene, and its relation to three-dimensional structure.

2.1 Notations

We assume that the reader has some familiarity with projective geometry at the level of [6, 23] and with some of its basic applications to computer vision such as the use of the fundamental matrix [5]. We will be using the following notations. Geometric entities such as points, lines, planes, etc…are represented by normal latin or greek letters; upper-case letters usually represent 3-D objects, lower-case letters 2-D (image based) objects. When these geometric entities are represented by vectors or matrixes, they appear in boldface. For example, m represents a pixel, $\mathbf{m}$ its coordinate vector, M represents a 3-D point, $\mathbf{M}$ its coordinate vector.

The line going through m and n is represented by $\langle m, n \rangle$. For a three-dimensional vector $\mathbf{x}$, we note $[\mathbf{x}]_\times$ the 3×3 antisymmetric matrix such that $\mathbf{x} \times \mathbf{y} = [\mathbf{x}]_\times \mathbf{y}$ for all vectors $\mathbf{y}$, where $\times$ indicates the cross-product. $\mathbf{I}_3$ represents the 3×3 identity matrix, $\mathbf{0}_3$ the 3×1 null vector.

We will also be using projective, affine and Euclidean coordinate frames. They are denoted by the letter $\mathcal{F}$, usually indexed by a point such as in $\mathcal{F}_C$. This notation means that the projective frame $\mathcal{F}_C$ is either an affine or a Euclidean frame of origin C. To indicate that the coordinates of a vector $\mathbf{M}$ are expressed in the frame $\mathcal{F}$, we write $\mathbf{M}_{/\mathcal{F}}$. Given two coordinate frames $\mathcal{F}_1$ and $\mathcal{F}_2$, we note $\mathbf{Q}_{\mathcal{F}_1}^{\mathcal{F}_2}$ the matrix of change of coordinates from frame $\mathcal{F}_1$ to frame $\mathcal{F}_2$, i.e. we have $\mathbf{M}_{/\mathcal{F}_2} = \mathbf{Q}_{\mathcal{F}_1}^{\mathcal{F}_2} \mathbf{M}_{/\mathcal{F}_1}$. Note that $\mathbf{Q}_{\mathcal{F}_1}^{\mathcal{F}_2} = (\mathbf{Q}_{\mathcal{F}_2}^{\mathcal{F}_1})^{-1}$.

2.2 The camera

The camera model that we use is the classical *pinhole model*. Widely used in computer vision, it captures quite accurately the actual geometry of many real imaging devices. It is also very general, and

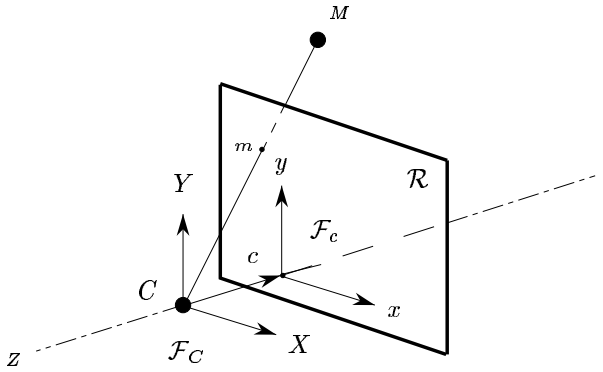

Figure 1: The pinhole model.

encompasses many camera models used in computer vision, such as perspective, weak-perspective, paraperspective, affine, parallel or orthographic projection. It can be described mathematically as follows: If the object space is considered to be the three-dimensional Euclidean space $\mathcal{R}^3$ embedded in the usual way in the three-dimensional projective space $\mathcal{P}^3$ and the image space to be the two-dimensional Euclidean space $\mathcal{R}^2$ embedded in the usual way in the two-dimensional projective space $\mathcal{P}^2$, the camera is then described as a *linear projective application* from $\mathcal{P}^3$ to $\mathcal{P}^2$ (see [6]). We can write the projection matrix in any object frame $\mathcal{F}_O$ of $\mathcal{P}^3$:

$$\underbrace{\begin{bmatrix} \alpha_u & \gamma & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\mathbf{K}} \mathbf{Q}_{\mathcal{F}_O}^{\mathcal{F}_C} \tag{1}$$

where $\mathbf{A}$ is the matrix of the *intrinsic parameters*, C the optical center (see figure 1). The special frame in which the projection matrix of the camera is equal to the matrix $\mathbf{K}$ is called the normalized camera frame.

In particular, the projection equation, relating a point not in the focal plane $\mathbf{M}_{/\mathcal{F}_C}^T = [X_C, Y_C, Z_C, T_C]^T$, expressed in the normalized camera frame, to its projection $\mathbf{m}_{/\mathcal{F}_c} = [x, y, 1]^T$, expressed in the image frame and written $\mathbf{m}$ for simplicity, is

$$Z_C \mathbf{m} = \mathbf{AKM}_{/\mathcal{F}_C} \tag{2}$$

2.3 Disparity between two views

We now consider two views of the scene, obtained from either two cameras or one camera in motion. If the two images have not been acquired simultaneously, we make the further assumption that no object of the scene has moved in the mean time.

The optical centers corresponding to the views are denoted by C for the first and C' for the second, the intrinsic parameters matrixes by $\mathbf{A}$ and $\mathbf{A}'$ respectively, the normalized camera frames respectively by $\mathcal{F}_C$ and $\mathcal{F}_{C'}$. The matrix of change of frame $\mathcal{F}_C$ to frame $\mathcal{F}_{C'}$ is a matrix of displacement defined by a rotation matrix $\mathbf{R}$ and a translation vector $\mathbf{t}$:

$$\mathbf{Q}_{\mathcal{F}_C}^{\mathcal{F}_{C'}} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_3^T & 1 \end{bmatrix} \tag{3}$$

More precisely, given a point M of an object o, we are interested in establishing the disparity equation of M for the two views, that is the equation relating the projection m' of M in the second view to the projection m of M in the first view.

2.3.1 The general case

Assuming that M is not in either one of the two focal planes corresponding to the first and second views, we have, from equations (2) and (3):

$$Z'_{C'}\mathbf{m}' = \mathbf{A}'\mathbf{K}\mathbf{M}_{/\mathcal{F}_{C'}} = \mathbf{A}' \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \mathbf{M}_{/\mathcal{F}_C} = Z_C\mathbf{A}'\mathbf{R}\mathbf{A}^{-1}\mathbf{m} + T_C\mathbf{A}'\mathbf{t}$$

This is the general *disparity equation* relating m' to m, which we rewrite as:

$$Z'_{C'}\mathbf{m}' = Z_C\mathbf{H}_\infty\mathbf{m} + T_C\mathbf{e}' \tag{4}$$

where we have introduced the following notations:

$$\mathbf{H}_\infty = \mathbf{A}'\mathbf{R}\mathbf{A}^{-1} \quad \text{and} \quad \mathbf{e}' = \mathbf{A}'\mathbf{t} \tag{5}$$

$\mathbf{H}_\infty$ represents the *collineation of the plane at infinity*, as it will become clear below in section (2.3.3). $\mathbf{e}'$ is a vector representing the *epipole* in the second view, that is, the image of C in the second view. Indeed, this image is

$$\mathbf{A}'\mathbf{K}\mathbf{C}_{/\mathcal{F}_{C'}} = \mathbf{A}' \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \mathbf{C}_{/\mathcal{F}_C} = \mathbf{A}'\mathbf{t}$$

since $\mathbf{C}_{/\mathcal{F}_C} = [0,\ 0,\ 0,\ 1]^T$. Similarly,

$$\mathbf{e} = \mathbf{A}\mathbf{R}^T\mathbf{t} \tag{6}$$

is a vector representing the epipole e in the first view.

Equation (4) means that m' lies on the line going through e' and the point represented by $\mathbf{H}_\infty\mathbf{m}$, which is the *epipolar line* of m. This line is represented by the vector

$$\mathbf{F}\mathbf{m} \tag{7}$$

where

$$\mathbf{F} = [\mathbf{e}']_\times\mathbf{H}_\infty \tag{8}$$

or equivalently [1]

$$\mathbf{F} = \mathbf{A}'^*[\mathbf{t}]_\times\mathbf{R}\mathbf{A}^{-1} \tag{9}$$

$\mathbf{F}$ is the *fundamental matrix* which describes the correspondence between an image point in the first view and its epipolar line in the second (see [5]).

[1] using the algebraic equation $[\mathbf{A}\mathbf{u}]_\times = \mathbf{A}^*[\mathbf{u}]_\times\mathbf{A}$ (valid if $\det(\mathbf{A}) \neq 0$), where $\mathbf{A}^* = \det(\mathbf{A})\mathbf{A}^{-1T}$ is the adjoint matrix of $\mathbf{A}$.

2.3.2 The case of coplanar points

Let us now consider the special case of points lying in a plane π. The plane is represented in $\mathcal{F}_C$ by the vector $\mathbf{\Pi}^T = \begin{bmatrix} \mathbf{n}^T & -d \end{bmatrix}$, where $\mathbf{n}$ is unit normal in $\mathcal{F}_C$ and d, the distance of C to the plane. Its equation is $\mathbf{\Pi}^T \mathbf{M}_{/\mathcal{F}_C} = 0$, which can be written, using equation (2),

$$\mathbf{n}^T \mathbf{K} \mathbf{M}_{/\mathcal{F}_C} - T_C d = Z_C \mathbf{n}^T \mathbf{A}^{-1} \mathbf{m} - T_C d = 0 \tag{10}$$

If we first assume that $d \neq 0$, that is the plane does not go through C, we obtain the new form of the disparity equation[2]:

$$Z'_{C'} \mathbf{m}' = Z_C \mathbf{H} \mathbf{m} \tag{11}$$

where

$$\mathbf{H} = \mathbf{H}_\infty + \mathbf{e}' \frac{\mathbf{n}^T}{d} \mathbf{A}^{-1} \tag{12}$$

This equation defines the projective linear mapping, represented by $\mathbf{H}$, the *H-matrix* of the plane, relating the images of the points of the plane in the first view to their images in the second. It is at the basis of the idea which consists of segmenting the scene in planar structures given by their respective H-matrices and, using this segmentation, to compute motion and structure (see [7] or [29]).

If the plane does not go either through C', its H-matrix represents a collineation ($\det(\mathbf{H}) \neq 0$) and its inverse is given by

$$\mathbf{H}^{-1} = \mathbf{H}' = \mathbf{H}_\infty^{-1} + \mathbf{e} \frac{\mathbf{n}'^T}{d'} \mathbf{A}'^{-1} \tag{13}$$

where $\mathbf{n}'$ is the unit normal in $\mathcal{F}_{C'}$ and d', the distance from the plane to C'. If the plane goes through only one of the two points C or C', its H-matrix is still defined by the one of the two equations (12) or (13) which remains valid, but is no longer a collineation; equation (10) shows that the plane then projects in one of the two views in a line represented by the vector

$$\mathbf{A}^* \mathbf{n} \quad \text{or} \quad \mathbf{A}'^* \mathbf{n}' \tag{14}$$

If the plane is an epipolar plane, i.e. goes through both C and C', its H-matrix is undefined.

Finally, equations (5) and (6) show that e' and e always verify equation (11), as expected, since e' and e are the images of the intersection of the line $< CC' >$ with the plane.

2.3.3 The case of points at infinity

For the points of the plane at infinity, represented by $[0, 0, 0, 1]^T$, thus of equation $T_C = 0$, the disparity equation becomes

$$Z'_{C'} \mathbf{m}' = Z_C \mathbf{H}_\infty \mathbf{m} \tag{15}$$

Thus, $\mathbf{H}_\infty$ is indeed the H-matrix of the plane at infinity. Equation (15) is also the limit of equation (11), when $d \to \infty$, which is compatible with the fact that the points at infinity correspond to the remote points of the scene.

[2] using the algebraic equation $(\mathbf{u}^T \mathbf{M} \mathbf{v}) \mathbf{w} = (\mathbf{w} \mathbf{u}^T \mathbf{M}) \mathbf{v}$.

2.4 Reconstruction

Reconstruction is the process of computing three-dimensional structure from two-dimensional image measurements. The three-dimensional structure of the scene can be captured only up to a group of transformations in space, related to the degree of knowledge of the imaging parameters. For instance, with a calibrated stereo rig (i.e., for which intrinsic and extrinsic parameters are known), it is well known that structure can be captured up to a rigid displacement in space. This has been used for a long time in photogrammetry. It has been shown more recently [14] that with non-calibrated affine cameras (i.e. that perform orthographic projection), structure can be recovered only up to an affine transformation. Then, the case of uncalibrated projective cameras has been addressed [2, 11, 28] and it has been shown that in this case, three-dimensional structure can be recovered only up to a projective transformation.

We will now use the formalism introduced above to describe these three cases in more detail.

2.4.1 Euclidean reconstruction

Here we suppose that we know the intrinsic parameters of the cameras $\mathbf{A}$, $\mathbf{A}'$, and the extrinsic parameters of the rig, $\mathbf{R}$ and $\mathbf{t}$. This is the case when cameras have been calibrated. For clarity we call it the *strong calibration* case. Through equation (5) we can compute $\mathbf{H}_\infty$ and $\mathbf{e}'$. Equation (4) gives us

$$\frac{Z_C}{T_C} = -\frac{(\mathbf{m}' \times \mathbf{e}') \cdot (\mathbf{m}' \times \mathbf{H}_\infty \mathbf{m})}{\|\mathbf{m}' \times \mathbf{H}_\infty \mathbf{m}\|^2}$$

and equation (2)

$$\begin{bmatrix} \frac{X_C}{T_C} \\ \frac{Y_C}{T_C} \end{bmatrix} = \frac{Z_C}{T_C} \mathbf{A}^{-1} \begin{bmatrix} x \\ y \end{bmatrix}$$

Thus, we have computed the coordinates of M with respect to $\mathcal{F}_C$.

The projection matrices for the first and second views, expressed in their respective image frames and in $\mathcal{F}_C$, are then written

$$\mathbf{A} \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 \end{bmatrix} \quad \text{and} \quad \mathbf{A}' \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix}$$

These matrices and the coordinates of M are thus known up to an unknown displacement $\mathbf{P}_{\mathcal{F}_O}^{\mathcal{F}_C}$ corresponding to an arbitrary change of the Euclidean reference frame.

2.4.2 Affine reconstruction

We now assume that both the fundamental matrix and the homography of the plane at infinity are known, but the intrinsic parameters of the cameras are unknown.

We show below that by applying an affine transformation of space, i.e., a transformation of $\mathcal{P}^3$ which leaves invariant the plane at infinity, we can compensate for the unknown parameters of the camera system. The guiding idea is to choose the affine transformation in such a way that the projection matrix of the first camera is equal to $\mathbf{K}$ as in [21]. We can then use the same reconstruction equations as in the Euclidean case (strong calibration). Since structure is known up to this unknown

affine transformation, we call this case the *affine calibration* case. Let us now describe the operations in detail.

Suppose then that we have estimated H_∞ (see section 3.4), thus we know $\mathbf{H}_\infty$ up to an unknown scale factor. Let us denote by $\tilde{\mathbf{H}}_\infty$ one of the possible representations of H_∞:

$$\tilde{\mathbf{H}}_\infty = \lambda \mathbf{H}_\infty$$

where λ is an unknown nonzero scalar. Suppose also that we have estimated the fundamental matrix $\mathbf{F}$ (see section 3.2) which is of rank 2, i.e its null-space is of dimension 1. Equation (8) shows that $\mathbf{e}'$ is in the null-space of $\mathbf{F}^T$, hence $\mathbf{e}'$ is known up to a nonzero scalar μ and we write in analogy with the previous case:

$$\tilde{\mathbf{e}}' = \mu \mathbf{e}' \tag{16}$$

Neither equation (2) nor equation (4) is usable since $\mathbf{A}$, $\mathbf{H}_\infty$ and $\mathbf{e}'$ are unknown. Both equations can be rewritten in another frame $\mathcal{F}_A$ defined by the matrix of change of frame $\mathbf{Q}_{\mathcal{F}_C}^{\mathcal{F}_A}$:

$$\mathbf{Q}_{\mathcal{F}_C}^{\mathcal{F}_A} = \begin{bmatrix} \frac{1}{\lambda}\mathbf{A} & \mathbf{0}_3 \\ \mathbf{0}_3^T & \frac{1}{\mu} \end{bmatrix}$$

Hence

$$\mathbf{Q}_{\mathcal{F}_A}^{\mathcal{F}_C} = \begin{bmatrix} \lambda \mathbf{A}^{-1} & \mathbf{0}_3 \\ \mathbf{0}_3^T & \mu \end{bmatrix}$$

Since we have

$$\mathbf{M}_{/\mathcal{F}_{C'}} = \mathbf{Q}_{\mathcal{F}_C}^{\mathcal{F}_{C'}} \mathbf{M}_{/\mathcal{F}_C} = \mathbf{Q}_{\mathcal{F}_C}^{\mathcal{F}_{C'}} \mathbf{Q}_{\mathcal{F}_A}^{\mathcal{F}_C} \mathbf{M}_{/\mathcal{F}_A}$$

this implies

$$\mathbf{M}_{/\mathcal{F}_{C'}} = \mathbf{Q}_{\mathcal{F}_C}^{\mathcal{F}_{C'}} (\mathbf{Q}_{\mathcal{F}_C}^{\mathcal{F}_A})^{-1} \mathbf{M}_{/\mathcal{F}_A} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_3^T & 1 \end{bmatrix} \begin{bmatrix} \lambda \mathbf{A}^{-1} & \mathbf{0}_3 \\ \mathbf{0}_3^T & \mu \end{bmatrix} \mathbf{M}_{/\mathcal{F}_A}$$

If $\mathbf{M}_{/\mathcal{F}_A}^T = [X_A, Y_A, Z_A, T_A]^T$ is a vector representing M in $\mathcal{F}_A$, equation (4), written in frame $\mathcal{F}_A$, becomes

$$Z'_{C'} \mathbf{m}' = Z_A \tilde{\mathbf{H}}_\infty \mathbf{m} + T_A \tilde{\mathbf{e}}' \tag{17}$$

and equation (2),

$$Z_A \mathbf{m} = \mathbf{K} \mathbf{M}_{/\mathcal{F}_A} \tag{18}$$

Equation (17) yields

$$\frac{Z_A}{T_A} = -\frac{(\mathbf{m}' \times \tilde{\mathbf{e}}') \cdot (\mathbf{m}' \times \tilde{\mathbf{H}}_\infty \mathbf{m})}{\|\mathbf{m}' \times \tilde{\mathbf{H}}_\infty \mathbf{m}\|^2}$$

and equation (18),

$$\begin{bmatrix} \frac{X_A}{T_A} \\ \frac{Y_A}{T_A} \end{bmatrix} = \frac{Z_A}{T_A} \begin{bmatrix} x \\ y \end{bmatrix}$$

Thus, we have computed the coordinates of M with respect to $\mathcal{F}_A$.

It is easy to verify that the projection matrices for the first and second views, expressed in their respective image frames and in $\mathcal{F}_A$, are then written

$$\left[\begin{array}{cc} \mathbf{I}_3 & \mathbf{0}_3 \end{array}\right] = \mathbf{K} \quad \text{and} \quad \left[\begin{array}{cc} \tilde{\mathbf{H}}_\infty & \tilde{\mathbf{e}}' \end{array}\right]$$

They are thus known up to the unknown affine transformation $\mathbf{P}_{\mathcal{F}_O}^{\mathcal{F}_A}$, corresponding to an arbitrary change of the affine reference frame[3].

2.4.3 Projective reconstruction

We now address the case when only the fundamental matrix $\mathbf{F}$ is known. This is known as the *weak calibration* case. The representation of the epipole $\mathbf{e}'$ is also known up to a nonzero scalar factor, as belonging to the null-space of $\mathbf{F}^T$. Neither equation (2) nor equation (4) is usable since $\mathbf{A}$, $\mathbf{H}_\infty$ and $\mathbf{e}'$ are unknown. As in the previous paragraph, we eliminate the unknown parameters by applying a projective transformation of space. Here, the plane at infinity is not (necessarily) left invariant by the transformation: It is mapped to an arbitrary plane. Let us now go into more details:

Let us first assume that we know, up to a nonzero scalar factor λ, the H-matrix of a plane not going through the optical center C of the first camera, as defined in section (2.3.2):

$$\mathbf{H} = \lambda(\mathbf{H}_\infty + \mathbf{e}'\frac{\mathbf{n}^T}{d}\mathbf{A}^{-1}) \tag{19}$$

where $\mathbf{n}$ is the unit normal expressed in $\mathcal{F}_C$ of the plane and d, with $d \neq 0$, the distance of C to the plane. We define a frame $\mathcal{F}_P$ by the matrix of change of frame from $\mathcal{F}_C$

$$\mathbf{Q}_{\mathcal{F}_C}^{\mathcal{F}_P} = \left[\begin{array}{cc} \frac{1}{\lambda}\mathbf{A} & \mathbf{0}_3 \\ -\frac{1}{\mu}\frac{\mathbf{n}^T}{d} & \frac{1}{\mu} \end{array}\right] \tag{20}$$

hence

$$\mathbf{Q}_{\mathcal{F}_P}^{\mathcal{F}_C} = \left[\begin{array}{cc} \lambda\mathbf{A}^{-1} & \mathbf{0}_3 \\ \lambda\frac{\mathbf{n}^T\mathbf{A}^{-1}}{d} & \mu \end{array}\right]$$

so that $\left[\begin{array}{cc} \frac{\lambda}{\mu}\frac{\mathbf{n}^T}{d}\mathbf{A}^{-1} & 1 \end{array}\right]^T$ is the vector representing the plane at infinity in $\mathcal{F}_P$. If $\mathbf{M}_{/\mathcal{F}_P}^T = [X_P, Y_P, Z_P, T_P]^T$ is the vector representing M in $\mathcal{F}_P$, we have then, using equation (2),

$$T_P = -\frac{1}{\mu}\frac{\mathbf{n}^T}{d}\mathbf{K}\mathbf{M}_{/\mathcal{F}_C} + \frac{1}{\mu}T_C = -Z_C\frac{1}{\mu}\frac{\mathbf{n}^T}{d}\mathbf{A}^{-1}\mathbf{m} + \frac{1}{\mu}T_C \tag{21}$$

and, eliminating T_C from equation (4),

$$Z'_{C'}\mathbf{m}' = Z_C(\mathbf{H}_\infty + \mathbf{e}'\frac{\mathbf{n}^T}{d}\mathbf{A}^{-1})\mathbf{m} + T_P\tilde{\mathbf{e}}' \tag{22}$$

[3] It is affine because it does not change the plane at infinity

Equation (4) is thus written in $\mathcal{F}_P$

$$Z'_{C'}\mathbf{m}' = Z_P\mathbf{H}\mathbf{m} + T_P\tilde{\mathbf{e}}' \tag{23}$$

As for equation (2), it is written in $\mathcal{F}_P$

$$Z_P\mathbf{m} = \mathbf{K}\mathbf{M}_{/\mathcal{F}_P} \tag{24}$$

Equation (23) then gives us

$$\frac{Z_P}{T_P} = -\frac{(\mathbf{m}' \times \tilde{\mathbf{e}}') \cdot (\mathbf{m}' \times \mathbf{H}\mathbf{m})}{\|\mathbf{m}' \times \mathbf{H}\mathbf{m}\|^2}$$

and equation (24),

$$\begin{bmatrix} \frac{X_P}{T_P} \\ \frac{Y_P}{T_P} \end{bmatrix} = \frac{Z_P}{T_P} \begin{bmatrix} x \\ y \end{bmatrix}$$

Thus, we have computed the coordinates of M with respect to frame $\mathcal{F}_P$.

The projection matrices for the first and second views, expressed in their respective image frames and in $\mathcal{F}_P$, are then written

$$\begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \mathbf{H} & \tilde{\mathbf{e}}' \end{bmatrix}$$

Indeed, the projection matrix for the second view is

$$\mathbf{A}'\mathbf{K}\mathbf{P}^{\mathcal{F}_{C'}}_{\mathcal{F}_P} = \begin{bmatrix} \mathbf{A}' & \mathbf{0}_3 \end{bmatrix} \mathbf{P}^{\mathcal{F}_{C'}}_{\mathcal{F}_C} \mathbf{P}^{\mathcal{F}_C}_{\mathcal{F}_P} = \begin{bmatrix} \mathbf{A}'\mathbf{R} & \mathbf{e}' \end{bmatrix} \begin{bmatrix} \lambda\mathbf{A}^{-1} & \mathbf{0}_3 \\ \lambda\frac{\mathbf{n}^T}{d}\mathbf{A}^{-1} & \mu \end{bmatrix}$$

and is actually of rank 3 as product of a 3×4-matrix of rank 3 and a 4×4-matrix of rank 4.

Both projection matrices and the coordinates of M are thus known up to the unknown collineation $\mathbf{Q}^{\mathcal{F}_P}_{\mathcal{F}_O}$, corresponding to an arbitrary change of the projective reference frame. This result had already been found in a quite different manner in [2, 12].

The reconstruction described above is possible as soon as the H-matrix of a plane which does not go through C is known. In particular, when $\mathbf{F}$ is known, one is always available as suggested by equations (8) and (22). It is defined by

$$\frac{\mathbf{n}^T}{d} = -\frac{\mathbf{e}'^T}{\|\mathbf{e}'\|^2}\mathbf{H}_\infty\mathbf{A} = -\frac{\mathbf{t}^T\mathbf{A}'^T\mathbf{A}'\mathbf{R}}{\|\mathbf{A}'\mathbf{t}\|^2} \tag{25}$$

which gives, using equation (8),[4]

$$\mathbf{H} = [\frac{\mathbf{e}'}{\|\mathbf{e}'\|}]_\times\mathbf{F}$$

The equation, expressed in $\mathcal{F}_{C'}$, of the corresponding plane is $\begin{bmatrix} \mathbf{n}^T & -d \end{bmatrix} \mathbf{P}^{\mathcal{F}_C}_{\mathcal{F}_{C'}}\mathbf{M}_{/\mathcal{F}_{C'}} = 0$, thus, using equation (25),

$$\mathbf{e}'^T\mathbf{A}'\mathbf{K}\mathbf{M}_{/\mathcal{F}_{C'}} = 0$$

which shows, using equation (2), that this plane is the plane going through C' which projects, in the second view, to the line representing by $\mathbf{e}'$, as already noticed in [21].

[4] using the algebraic equation $\mathbf{u}\mathbf{u}^T = \|\mathbf{u}\|^2\mathbf{I}_3 + [\mathbf{u}]^2_\times$.

3 Computing the geometric parameters

Now that we have established which parameters are necessary to deduce information on the structure of the scene, we describe methods to compute these parameters, from real images.

If no a priori knowledge is assumed, the only source of information is the images themselves and the correspondences established between them.

After showing how accurate and reliable point correspondences can be obtained in general from the images, we describe how they can be used for estimating the fundamental matrix on the one hand, plane collineations on the other hand.

3.1 Finding correspondences

Matching is done using the image intensity function $I(x, y)$. A criterion, usually depending on the local value of $I(x, y)$ in both images, is chosen to decide whether a point m_1 of the first image and a point m_2 of the second are the images of the same point of the scene. It is generally based on on a physical model of the scene. A classical measure for similarity between the two images within a given area is the cross-correlation coefficient

$$C(m_1, m_2) = \cos(\mathbf{i}_1 - \overline{\mathbf{i}_1}, \mathbf{i}_2 - \overline{\mathbf{i}_2}) = \frac{(\mathbf{i}_1 - \overline{\mathbf{i}_1}).(\mathbf{i}_2 - \overline{\mathbf{i}_2})}{\|\mathbf{i}_1 - \overline{\mathbf{i}_1}\|\|\mathbf{i}_2 - \overline{\mathbf{i}_2}\|}$$

where $\mathbf{i}_j$ is the vector of the image intensity values in a neighborhood around the point m_j and $\overline{\mathbf{i}_j}$ its mean in this neighborhood.

The context in which the views have been taken plays a significant role. Two main cases have to be considered: the case where the views are very similar and the opposite case. The first case usually corresponds to consecutive views of a sequence taken by one camera, the second, to views taken by a stereo rig with a large baseline. In the first case, the distance between the images of a point in two consecutive frames is small. This allows to limit the search space when trying to find point correspondences. Below, we briefly describe a simple point tracker which, relying on this property, provides robust correspondences at a relatively low computational cost. In the second case, corresponding points may have quite different positions in the two images. Thus, point matching requires more sophisticated techniques. This is the price to pay if we want to manipulate pairs of images taken simultaneously from different viewpoints, which allow general reconstruction of the scene without worrying about the motion of the observed objects, as mentioned in section (2.3).

In both cases, the criterion that we use for estimating the similarity between image points is not computed for all of them, but only for points of interest. These points are usually intensity corners in the image, obtained as the maxima of some operators applied to $I(x, y)$. Indeed, they are the most likely to be invariant to view changes for these operators since they usually correspond to object markings.

The corner detector. The operator that we use is the one presented in [10], which is a slightly modified version of the Plessey corner detector:

$$\det(\hat{\mathbf{C}}) - k(\text{trace}(\hat{\mathbf{C}}))^2$$

where

$$\hat{\mathbf{C}} = \begin{bmatrix} \hat{I}_x^2 & \hat{I_x I_y} \\ \hat{I_x I_y} & \hat{I}_y^2 \end{bmatrix}$$

and $\hat{I}$ denotes a smoothed version of I. Based on experiments, Harris suggests to set $k = 0.04$ for best results. $\hat{\mathbf{C}}$ is computed at each point of the image, and points for which it is larger than a given threshold are retained as corners.

The points tracker. The implementation has been strongly influenced by the corner tracker described in [18].

It works as follows: First, corners are extracted in both images. Then, for a given corner of the first image, the following operation is performed: its neighborhood is searched for corners of the second image; the criterion C is computed for each pair of the corner of the first image and one of the possible matches in the second image; The pair with the best score is retained as a correspondence if the score is above a fixed threshold.

Then, for each corner of the second image for which a corresponding point in the first image has been found, the preceding operation is applied from the second image to the first. If the corresponding point found by this operation is the same as the previous one, it is then definitely taken as valid.

The stereo points matcher. The method described in the previous section no longer works as soon as the views are quite different. More precisely, the correlation criterion is not selective enough: there are, for a given point of an image, several points of the other image that lead to a good correlation score, without the best of them being the real correspondent point searched. To achieve correspondence matching, the process must then keep all those potentially good but conflicting correspondences and invokes global techniques to decide between them: a classical relaxation technique is used to converge towards a globally coherent system of point correspondences, given some constraints of uniqueness and continuity (see [30]).

3.2 The fundamental matrix

Once some image point correspondences, represented in the image frame by $(\mathbf{m}'_i, \mathbf{m}_i)$, have been found, the fundamental matrix $\mathbf{F}$ is computed, up to a nonzero scalar factor, as the unique solution of the system of equations, derived from the disparity equations,

$$\mathbf{m}_i'^T \mathbf{F} \mathbf{m}_i = 0 \tag{26}$$

This system can be solved as soon as seven such correspondences are available: only eight coefficients of $\mathbf{F}$ need to be computed, since $\mathbf{F}$ is defined up to a nonzero scalar factor, while equation (26) supplies one scalar equation per correspondence and $\det(\mathbf{F}) = 0$, the eighth. If there are more correspondences available, which are not exact, as it is the case in practice, the goal of the computation is to find the matrix which best approximates the solution of this system according to a given least squares criterion.

A study of the computation of the fundamental matrix from image point correspondences can be found in [20]. Here, we just mention our particular implementation, which consists, on the one hand, of a direct computation considering that all the correspondences are valid and in the other hand, of a method for rejecting some possible outliers among the correspondences.

The direct computation computes $\mathbf{F}$ which minimizes the following criterion:

$$\sum_i \left(\frac{1}{[\mathbf{F}\mathbf{m}_i]_x^2 + [\mathbf{F}\mathbf{m}_i]_y^2} + \frac{1}{[\mathbf{F}^T\mathbf{m}'_i]_x^2 + [\mathbf{F}^T\mathbf{m}'_i]_y^2} \right) (\mathbf{m}_i'^T \mathbf{F} \mathbf{m}_i)^2$$

which is the sum of the squares of the distance of m_i to the epipolar line of m'_i and the distance of m'_i to the epipolar line of m_i. Minimization is performed with the classical Levenberg-Marquardt method (see [26]). In order to take in account both its definition up to a scale factor and the fact that it is of rank 2, a parametrization of $\mathbf{F}$ with seven parameters is used, which parametrizes all the 3×3-matrices of rank strictly less than 3. These parameters are computed from $\mathbf{F}$ the following way: a line l (respectively, a column c) of $\mathbf{F}$ is chosen and written as a linear combination of the other two lines (respectively, columns); the four entries of $\mathbf{F}$ of these two combinations are taken as parameters; among the four coefficients not belonging to l and c, the three smallest, in absolute value, are divided by the biggest and taken as the last three parameters. l and c are chosen in order to maximize the rank of the derivative of $\mathbf{F}$ with respect to the parameters. Denoting the parameters by p_1, p_2, p_3, p_4, p_5, p_6 and p_7 and assuming, for instance, l and c equals to 1 and the bottom right coefficient being the normalized coefficient, leads to the following matrix:

$$\begin{bmatrix} p_6(p_4p_1 + p_5p_3) + p_7(p_4p_2 + p_5) & p_4p_1 + p_5p_3 & p_4p_2 + p_5 \\ p_6p_1 + p_7p_2 & p_1 & p_2 \\ p_6p_3 + p_7 & p_3 & 1 \end{bmatrix}$$

During the minimization process, the parametrization of $\mathbf{F}$ can change: the parametrization chosen for the matrix at the beginning of the process is not necessarily the most suitable for the final matrix.

The outliers rejection method used is a classical least median of squares method. It is described in detail in [30].

3.3 The H-matrix of a plane

If we have at our disposal correspondences, represented in the image frames by $(\mathbf{m}'_i, \mathbf{m}_i)$, of points belonging to a plane, the H-matrix $\mathbf{H}$ of this plane is computed, up to a nonzero scalar factor, as the unique solution of the system of equations (11),

$$Z'_{C'}\mathbf{m}'_i = Z_C \mathbf{H} \mathbf{m}_i$$

This system can be solved as soon as four such correspondences are available: only eight coefficients of $\mathbf{H}$ need to be computed, since $\mathbf{H}$ is defined up to a nonzero scalar factor, while equation (11) supplies two scalar equation for each correspondence. If there are more correspondences available, which are not exact, as it is the case in practice, the goal of the computation is to find the matrix

which approximates at best the solution of this system according to a given criterion: a study of the computation of plane H-matrices from image point correspondences can be found in [6].

Since e' and e verify equation (11), three point correspondences are in general sufficient for defining H. In fact, this is true as long as the homography is defined, i.e., when three points are not aligned in either image (a proof can be found in [3]). If the plane is defined by one point and a line L, given by its projections (l, l'), so that e does not belong to l and e' does not belong to l', its H-matrix is computable the same way, as soon as we know the fundamental matrix. Indeed, the projections of two other points M and N of the plane are given by choosing two points m and n on l, which amounts to choosing M and N on L: the corresponding points m' and n' are then given by intersecting l' with the epipolar line of m and the epipolar line of n, given by the fundamental matrix.

As an application of this idea, we have a purely image-based way of solving the following problem: given a point correspondence $(m,\, m')$ defining a 3-D point M, and a line correspondence $(l,\, l')$ defining a 3-D line L, find the H-matrix of the plane going through M and L. In particular, if L is at infinity, it defines a direction of plane (all planes going through L are parallel) and we can find the H-matrix of the plane going through M and parallel to that direction. This will be used in section 6.2.

Given the H-matrix $\mathbf{H}$ of a plane Π and the correspondences (m, m') and (n, n') of two points M and N, it is possible to directly compute in the images the correspondences (i, i') of the intersection I of the line $\langle M,\, N\rangle$ with Π. Indeed, i' belongs both to $\langle m',\, n'\rangle$ and the image of $\langle m,\, n\rangle$ by $\mathbf{H}$, so:

$$\mathbf{i}' = (\mathbf{m}' \times \mathbf{n}') \times (\mathbf{H}\mathbf{m} \times \mathbf{H}\mathbf{n})$$

(see [27])

Similarly, given two planes Π_1 and Π_2 represented by their H-matrices $\mathbf{H}_1$ and $\mathbf{H}_2$, it is possible to directly compute in the images the correspondences of the intersection L of Π_1 with Π_2. Indeed, the correspondences of two points of L are computed, for example, as intersections of two lines L_1 and L_2 of Π_1 with Π_2; the correspondences of such lines are obtained by choosing two lines in the first image representing by the vectors $\mathbf{l}_1$ and $\mathbf{l}_2$, their corresponding lines in the second image being given by $\mathbf{H}_1^{-1T}\mathbf{l}_1$ and $\mathbf{H}_1^{-1T}\mathbf{l}_2$.

3.4 The homography of the plane at infinity

To compute the homography of the plane at infinity $\mathbf{H}_\infty$, we can no longer use the disparity equation (4) with correspondences of points not at infinity, even if we know the fundamental matrix, since $Z'_{C'}$, Z_C and T_C are not known. We must, thus, know correspondences of points at infinity $(\mathbf{m}'_i, \mathbf{m}_i)$ and compute $\mathbf{H}_\infty$ like any other plane H-matrices, as described in section 3.3.

The only way to obtain correspondences of points at infinity is to assume some additional knowledge.

First, we can assume that we have some additional knowledge of the observed scene that allows to identify, in the images, some projections of points at infinity, like, for instance, the vanishing points of parallel lines of the scene, or the images of some points on the horizon, which provide sufficiently good approximations to points at infinity.

Another way to proceed is to assume that we have an additional pair of views. More precisely, if this second pair differs from the first only by a translation of the rig, any pair (M, N) of stationary

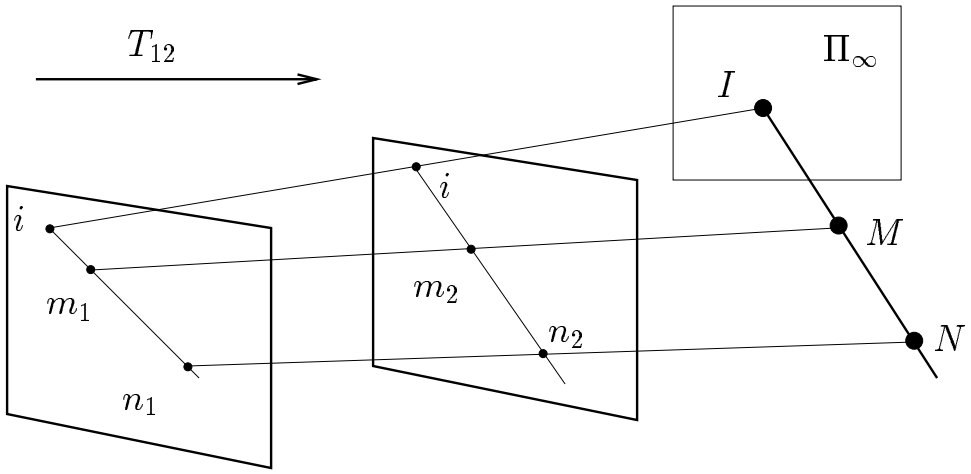

Figure 2: Determining the projections of points at infinity (see section 3.4).

object points (see figure 2), seen in the first views as $(\mathbf{m}_1, \mathbf{m}'_1)$ and $(\mathbf{n}_1, \mathbf{n}'_1)$, and in the second as $(\mathbf{m}_2, \mathbf{m}'_2)$ and $(\mathbf{n}_2, \mathbf{n}'_2)$, gives us the images $(\mathbf{i}_1, \mathbf{i}'_1)$ and $(\mathbf{i}_2, \mathbf{i}'_2)$ in the four images of the intersection I of the line $\langle M,\, N\rangle$ with the plane at infinity. Indeed, on one hand, since I is at infinity and the stationarity of M and N implies the stationarity of I, we have, from equations (15) and (4),

$$Z_{C_2}\mathbf{i}_2 = Z_{C_1}\mathbf{H}_{\infty 12}\mathbf{i}_1 \quad \text{and} \quad Z'_{C'_2}\mathbf{i}'_2 = Z'_{C'_1}\mathbf{H}'_{\infty 12}\mathbf{i}'_1$$

where $\mathbf{H}_{\infty 12}$ (respectively, $\mathbf{H}'_{\infty 12}$) is the homography of the plane at infinity between the first (respectively, second) view of the first pair and its corresponding view in the second pair. In the case where the two pairs of views differ only by a translation, $\mathbf{A}_1 = \mathbf{A}_2$, $\mathbf{R}_{12} = \mathbf{I}_3$, $\mathbf{A}'_1 = \mathbf{A}'_2$, $\mathbf{R}'_{12} = \mathbf{I}_3$ and we have, by equation (5),

$$\mathbf{H}_{\infty 12} = \mathbf{I}_3 \quad \text{and} \quad \mathbf{H}'_{\infty 12} = \mathbf{I}_3$$

which implies that $i_1 = i_2 = i$ and $i'_1 = i'_2 = i'$. On the other hand, as I lies on $\langle M,\, N\rangle$, i_1 lies on $\langle m_1,\, n_1\rangle$, i_2, on $\langle m_2,\, n_2\rangle$, i'_1, on $\langle m'_1,\, n'_1\rangle$ and i'_2, on $\langle m'_2,\, n'_2\rangle$. Consequently, i and i' are obtained as the intersections of $\langle m_1,\, n_1\rangle$ with $\langle m_2,\, n_2\rangle$ and of $\langle m'_1,\, n'_1\rangle$ with $\langle m'_2,\, n'_2\rangle$, respectively:

$$\mathbf{i} = (\mathbf{m}_1 \times \mathbf{n}_1) \times (\mathbf{m}_2 \times \mathbf{n}_2) \quad \text{and} \quad \mathbf{i}' = (\mathbf{m}'_1 \times \mathbf{n}'_1) \times (\mathbf{m}'_2 \times \mathbf{n}'_2)$$

Once $\mathbf{H}_\infty$ has been obtained, the ratio of the lengths of any two aligned segments of the scene can be computed directly in the images. Indeed, given three points M_1, M_2 and M_3 on a line, as in figure 3, from their images (m_1, m'_1), (m_2, m'_2) and (m_3, m'_3), we can compute the images (m, m') of the intersection of this line with the plane at infinity, using $\mathbf{H}_\infty$, as explained in section 3.3. We can compute in each image the cross-ratio of those four points. As a projective invariant, this cross-ratio

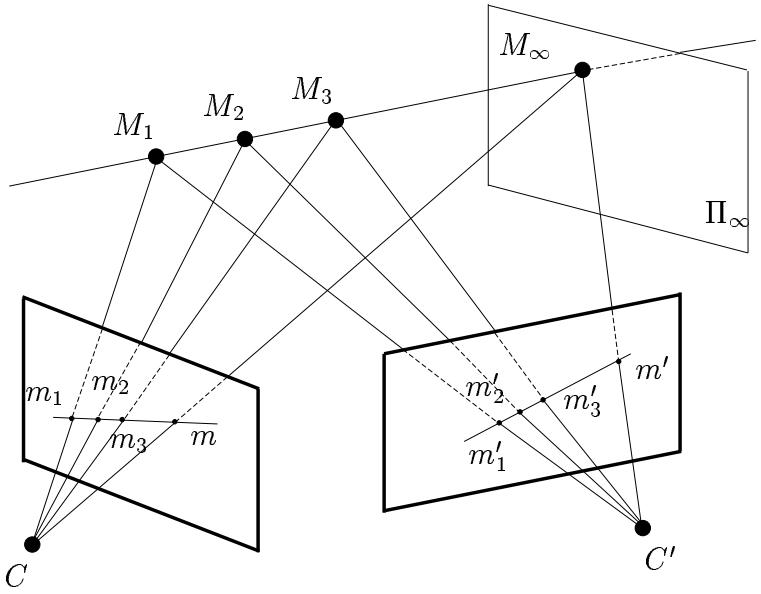

Figure 3: Determining ratios of lengths in affine calibration (see text).

is then exactly equal to the ratio of M_1, M_2 and M_3. More precisely:

$$\frac{\overline{M_1M_3}}{\overline{M_2M_3}} = \frac{\overline{M_1M_3}}{\overline{M_1M_\infty}} : \frac{\overline{M_2M_3}}{\overline{M_2M_\infty}} = \frac{\overline{m_1m_3}}{\overline{m_1m}} : \frac{\overline{m_2m_3}}{\overline{m_2m}}$$

4 The rectification with respect to a plane

In this section, we assume that we know the epipolar geometry. This allows us to *rectify the images with respect to a plane* of the scene. This process explained below, allows not only to compute a map of image point correspondences, but also to assign to each of them a scalar that represents a measure of the disparity between the two projections of the correspondence.

4.1 The process of rectification

Like in section 2.4.3, we assume that we know, up to nonzero scale factors, $\mathbf{F}$, thus $\tilde{\mathbf{e}}'$, given by equation (16), and the H-matrix $\mathbf{H}$ of a plane Π given by equation (19). Let us then choose two homographies, represented by the matrices $\hat{\mathbf{H}}'$ and $\hat{\mathbf{H}}$, such that

$$\hat{\mathbf{H}}'\tilde{\mathbf{e}}' = \alpha[1,0,0]^T \tag{27}$$

$$\hat{\mathbf{H}} = \hat{\mathbf{H}}'\mathbf{H} \tag{28}$$

where α is any nonzero scalar. Equation (23) can then be rewritten

$$\hat{z}' Z'_{C'} \hat{\mathbf{m}}' = \hat{z} Z_P \hat{\mathbf{m}} + T_P \alpha [1,0,0]^T \tag{29}$$

where $\hat{\mathbf{m}} = [\hat{x}, \hat{y}, 1]^T$, $\hat{\mathbf{m}}' = [\hat{x}', \hat{y}', 1]^T$ and

$$\hat{z}'\hat{\mathbf{m}}' = \hat{\mathbf{H}}'\mathbf{m}' \quad \text{and} \quad \hat{z}\hat{\mathbf{m}} = \hat{\mathbf{H}}\mathbf{m} \tag{30}$$

The rectification with respect to a plane consists of applying such matrices, called the *rectification matrices*, $\hat{\mathbf{H}}'$ to the second image and $\hat{\mathbf{H}}$ to the first.

Equation (29) shows that the corresponding point $\hat{m}'$ in the second rectified image of a point $\hat{m}$ of the first rectified image lies on the line parallel to the x-axis and going through $\hat{m}$. Applying a correlation criterion to $\hat{m}$ and each point of this line thus allows to determine $\hat{m}'$, if the image is not too distorted through the process of rectification. Equations (27) and (28) do not completely determine $\hat{\mathbf{H}}$ and $\hat{\mathbf{H}}'$: This indetermination is used to minimize the distortion of the images, as explained in section 4.3.

Once $\hat{m}'$ has been determined, a measure of the disparity between $\hat{m}'$ and $\hat{m}$ with respect to this plane is given by $\hat{x}' - \hat{x}$. If M belongs to Π, it is equal to zero since T_P then vanishes as shown by equations (10) and (21); otherwise, its interpretation depends on the information available for the model, as explained in section 5.

4.2 Geometric interpretation of the rectification

From the QR-decomposition [9] any nonsingular matrix $\mathbf{H}$ decomposed as:

$$\mathbf{H} = \mathbf{R}\mathbf{U} \tag{31}$$

where $\mathbf{R}$ is a rotation matrix and $\mathbf{U}$, a nonsingular upper triangular matrix. Decomposing $\mathbf{H}^{-1}$ like in equation (31), inverting it, and noticing that the inverse of an upper triangular matrix is also an upper triangular matrix, we see that $\mathbf{H}$ can also be decomposed as:

$$\mathbf{H} = \mathbf{U}'\mathbf{R}' \tag{32}$$

where $\mathbf{R}'$ is a rotation matrix and $\mathbf{U}'$, a nonsingular upper triangular matrix.

To give a geometric interpretation of the rectification, we decompose $\hat{\mathbf{H}}$ and $\hat{\mathbf{H}}'$ the following way: By applying equation (32) to the non-singular matrices $\hat{\mathbf{H}}\mathbf{A}$ and $\hat{\mathbf{H}}'\mathbf{A}'$, there exist two scalars $\hat{\lambda}$ and $\hat{\lambda}'$, two rotation matrices $\hat{\mathbf{R}}$ and $\hat{\mathbf{R}}'$ and two upper triangular matrices $\hat{\mathbf{A}}$ and $\hat{\mathbf{A}}'$ of the same form as $\mathbf{A}$ in equation (1), such that

$$\hat{\mathbf{H}} = \hat{\lambda}\hat{\mathbf{A}}\hat{\mathbf{R}}\mathbf{A}^{-1} \quad \text{and} \quad \hat{\mathbf{H}}' = \hat{\lambda}'\hat{\mathbf{A}}'\hat{\mathbf{R}}'\mathbf{A}'^{-1} \tag{33}$$

Then, we study how the constraints on $\hat{\mathbf{H}}'$ and $\hat{\mathbf{H}}$ given by equations (27) and (28) propagate to $\hat{\lambda}$, $\hat{\lambda}'$, $\hat{\mathbf{R}}$ and $\hat{\mathbf{R}}'$. On one hand, from equations (27), (33), (5) and (16), we have

$$\hat{\mathbf{R}}'\mathbf{t} = \frac{\alpha}{\mu\hat{\lambda}'}\hat{\mathbf{A}}'^{-1}[1,0,0]^T$$

and we define $\hat{t}_x$ such that

$$\hat{\mathbf{R}}'\mathbf{t} = [\hat{t}_x,0,0]^T \tag{34}$$

On the other hand, from equations (28), (33), (19), (5) and (34), we have then

$$\begin{gathered}
\mathbf{I}_3 = \hat{\mathbf{H}}'\mathbf{H}\hat{\mathbf{H}}^{-1} \\
\Longleftrightarrow \\
\mathbf{I}_3 = \lambda\hat{\lambda}'\hat{\mathbf{A}}'\hat{\mathbf{R}}'\mathbf{A}'^{-1}(\mathbf{A}'\mathbf{R}\mathbf{A}^{-1} + \mathbf{A}'\mathbf{t}\tfrac{\mathbf{n}^T}{d}\mathbf{A}^{-1})\tfrac{1}{\hat{\lambda}}\mathbf{A}\hat{\mathbf{R}}^T\hat{\mathbf{A}}^{-1} \\
\Longleftrightarrow \\
\hat{\mathbf{R}}'\mathbf{R}\hat{\mathbf{R}}^T = \tfrac{\hat{\lambda}}{\lambda\hat{\lambda}'}\hat{\mathbf{A}}'^{-1}\hat{\mathbf{A}} - [\hat{t}_x,0,0]^T\tfrac{\mathbf{n}^T\hat{\mathbf{R}}^T}{d}
\end{gathered}$$

from which we deduce that $\hat{\mathbf{R}}'\mathbf{R}\hat{\mathbf{R}}^T$ is an upper-triangular matrix. Since it is a rotation matrix, this means that

$$\hat{\mathbf{R}}'\mathbf{R}\hat{\mathbf{R}}^T = \mathbf{I}_3 \tag{35}$$

We then also deduce that

$$\lambda\hat{\lambda}' = \hat{\lambda} \tag{36}$$

and

$$\mathbf{I}_3 = \hat{\mathbf{A}}'\hat{\mathbf{A}}^{-1} + \hat{\mathbf{A}}'[\hat{t}_x,0,0]^T\frac{\mathbf{n}^T\hat{\mathbf{R}}^T}{d}\hat{\mathbf{A}}^{-1} \tag{37}$$

We are now able to interpret the equations (30). From equation (27) and (20), we have $\hat{z}'Z'_{C'} = \hat{z}Z_P = \frac{\hat{z}Z_C}{\lambda}$. Using equation (36), we can then define Z_R by

$$Z_R = \frac{\hat{z}'Z'_{C'}}{\hat{\lambda}'} = \frac{\hat{z}Z_C}{\hat{\lambda}} \tag{38}$$

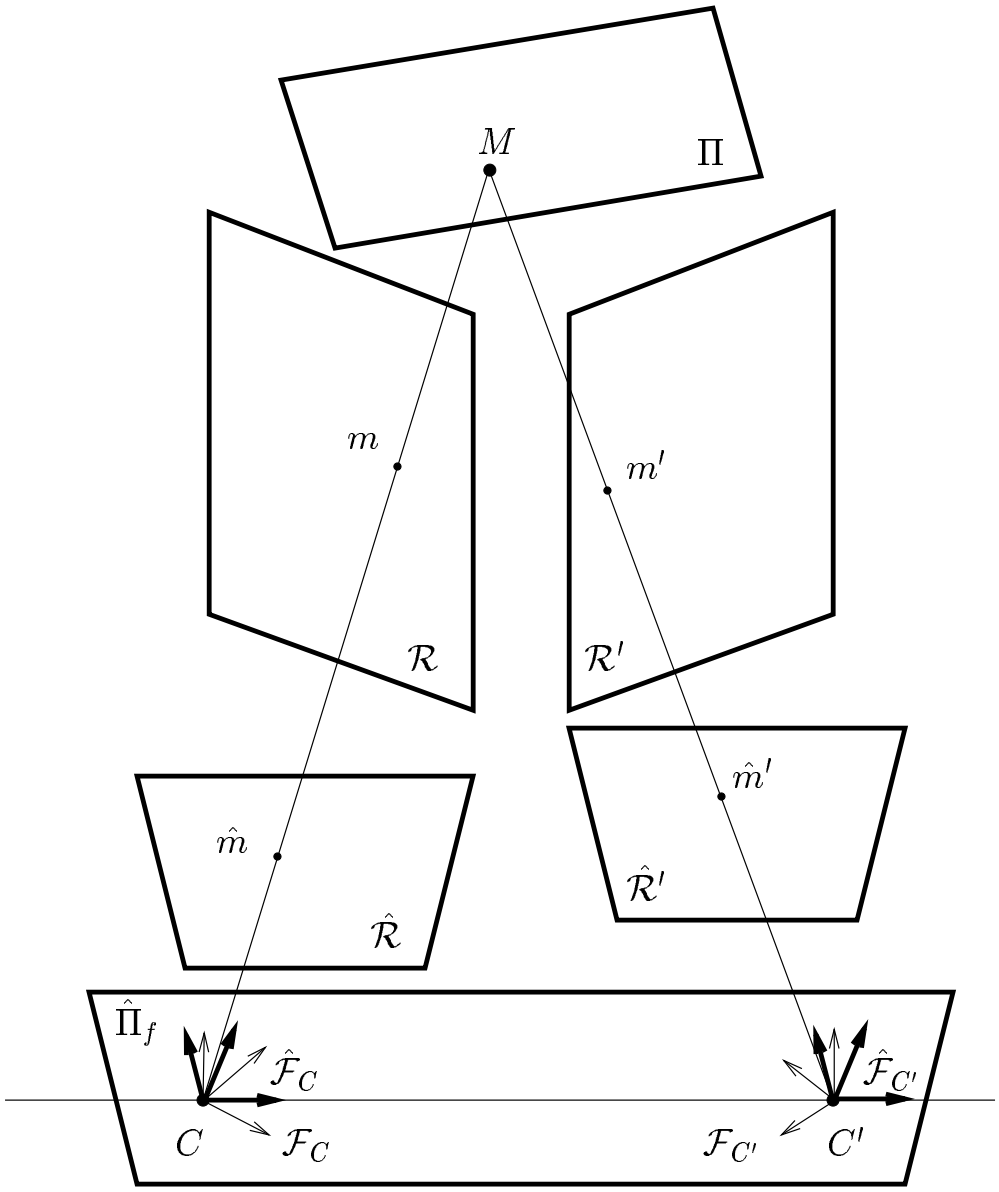

Figure 4: The rectification with respect to a plane Π.

so that the equations (30) are written

$$Z_R\hat{\mathbf{m}} = Z_C\hat{\mathbf{A}}\hat{\mathbf{R}}\hat{\mathbf{A}}^{-1}\mathbf{m} \quad \text{and} \quad Z_R\hat{\mathbf{m}}' = Z'_{C'}\hat{\mathbf{A}}'\hat{\mathbf{R}}'\hat{\mathbf{A}}'^{-1}\mathbf{m}'$$

They are interpreted as the disparity equations of two pairs of views (see figure 4): The first pair is composed of the view of optical center C, camera frame $\mathcal{F}_C$, retinal plane $\mathcal{R}$ and intrinsic parameters matrix $\mathbf{A}$ and its rectified view of optical center C, camera frame $\hat{\mathcal{F}}_C$, retinal plane $\hat{\mathcal{R}}$ and intrinsic parameters matrix $\hat{\mathbf{A}}$; similarly, the second pair is composed of the view of optical center C', camera frame $\mathcal{F}_{C'}$, retinal plane $\mathcal{R}'$ and intrinsic parameters matrix $\mathbf{A}'$ and its rectified view of optical center C', camera frame $\hat{\mathcal{F}}_{C'}$, retinal plane $\hat{\mathcal{R}}'$ and intrinsic parameters matrix $\hat{\mathbf{A}}'$. The basis of $\hat{\mathcal{F}}_C$ is the image of the basis of $\mathcal{F}_C$ by the rotation of matrix $\hat{\mathbf{R}}$. Similarly, the basis of $\hat{\mathcal{F}}_{C'}$ is the image of the basis of $\mathcal{F}_{C'}$ by the rotation of matrix $\hat{\mathbf{R}}'$. Furthermore, according to equations (35) and (34), we have

$$\begin{aligned}
\mathbf{Q}_{\hat{\mathcal{F}}_C}^{\hat{\mathcal{F}}_{C'}} &= \mathbf{Q}_{\mathcal{F}_{C'}}^{\hat{\mathcal{F}}_{C'}}\mathbf{Q}_{\mathcal{F}_C}^{\mathcal{F}_{C'}}\mathbf{Q}_{\hat{\mathcal{F}}_C}^{\mathcal{F}_C} \\
&= \begin{bmatrix} \hat{\mathbf{R}}' & \mathbf{0}_3 \\ \mathbf{0}_3^T & 1 \end{bmatrix}\begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_3^T & 1 \end{bmatrix}\begin{bmatrix} \hat{\mathbf{R}}^T & \mathbf{0}_3 \\ \mathbf{0}_3^T & 1 \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & 0 & \hat{t}_x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}$$

which shows that $\hat{\mathcal{F}}_C$ and $\hat{\mathcal{F}}_{C'}$ have the same basis $\mathcal{B}_R$ and that the x-axis of this basis is parallel to $\overrightarrow{CC'}$. Lastly, for the two rectified views, the homography of the plane at infinity is $\hat{\mathbf{H}}_\infty = \hat{\mathbf{A}}'\hat{\mathbf{A}}^{-1}$, the epipole of the second view is $\hat{\mathbf{e}}' = \hat{\mathbf{A}}'[\hat{t}_x, 0, 0]^T$, so that, according to equation (19), the homography of Π is $\mathbf{I}_3$.

In summary, the process of rectification consists of projecting the first image onto a retinal plane $\hat{\mathcal{R}}$ and the second image onto a retinal plane $\hat{\mathcal{R}}'$ such that $\hat{\mathcal{R}}$ and $\hat{\mathcal{R}}'$ are parallel and choosing the rectified image frames such that the x-axis of the two rectified images are parallel and the homography of Π for the two rectified images is the identity.

4.3 Minimizing image distortion

We now examine the distortion caused by $\hat{\mathbf{H}}$ and $\hat{\mathbf{H}}'$ to the images.

4.3.1 How many degrees of freedom are left ?

$\mathbf{H}$ being known, equation (28) shows that $\hat{\mathbf{H}}$ is completely determined as soon as $\hat{\mathbf{H}}'$ is. So, all the degrees of freedom left are concentrated in $\hat{\mathbf{H}}'$. Only eight coefficients of $\hat{\mathbf{H}}'$ need to be computed, since $\hat{\mathbf{H}}'$ is defined up to a nonzero scale factor, and equation (27) supplies two scalar equations: Six degrees of freedom remain, but how many of them are really involved in the distortion ?

To answer this question, we propose two approaches: The first one decomposes $\hat{\mathbf{H}}'$ and the second one, $\hat{\mathbf{H}}'^{-1}$. In each case, we propose a method for computing the values of the parameters which minimize the image distorsion.

4.3.2 The decomposition of $\hat{\mathbf{H}}'$.

According to equation (32), there exist two matrices, $\mathbf{U}$ and $\mathbf{R}$ such that

$$\hat{\mathbf{H}}' = \mathbf{U}\mathbf{R}$$

$\mathbf{U}$ is an upper triangular matrix and $\mathbf{R}$, a rotation matrix. If we decompose $\mathbf{R}$ as a product of three rotations around the x- y- and z-axis, we can write

$$\hat{\mathbf{H}}' = \underbrace{\begin{bmatrix} \mathbf{U}_2 & \mathbf{v} \\ \mathbf{0}_3^T & \lambda \end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix} \mathbf{R}_2 & \mathbf{0}_2 \\ \mathbf{0}_3^T & 1 \end{bmatrix}}_{\mathbf{R}_z} \mathbf{R}_y\mathbf{R}_x = \begin{bmatrix} \mathbf{U}_2\mathbf{R}_2 & \mathbf{v} \\ \mathbf{0}_3^T & \lambda \end{bmatrix} \mathbf{R}_y\mathbf{R}_x$$

where $\mathbf{U}_2$ is a 2×2 upper triangular matrix, $\mathbf{R}_2$, a 2×2 rotation matrix, $\mathbf{v}$ a vector and λ, a scalar. Now, according to equation (31), $\mathbf{U}_2\mathbf{R}_2$ can be rewritten as $\mathbf{R}_2'\mathbf{U}_2'$ where $\mathbf{R}_2'$ is a rotation matrix and $\mathbf{U}_2'$, an upper triangular matrix and we can write

$$\hat{\mathbf{H}}' = \underbrace{\begin{bmatrix} \mathbf{R}_2' & \mathbf{0}_2 \\ \mathbf{0}_3^T & 1 \end{bmatrix}}_{\mathbf{R}_z'} \underbrace{\begin{bmatrix} \mathbf{U}_2' & \mathbf{R}_2'^T\mathbf{v} \\ \mathbf{0}_3^T & \lambda \end{bmatrix}}_{\mathbf{U}'} \mathbf{R}_y\mathbf{R}_x$$

where $\mathbf{R}_z'$ is a rotation around the z-axis and $\mathbf{U}'$, an upper triangular matrix. Lastly, if we extract from $\mathbf{U}'$ the translation and scaling components, we have

$$\hat{\mathbf{H}}' = \lambda\mathbf{R}_z' \begin{bmatrix} s_x & 0 & u_0 \\ 0 & s_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & s_{xy} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{R}_y\mathbf{R}_x \tag{39}$$

Based on equation (39), $\mathbf{R}_y$ is chosen such as to cancel out the third coordinate of $\hat{\mathbf{H}}'\tilde{\mathbf{e}}'$, involved in equation (27), (making the epipolar lines parallel) and $\mathbf{R}_z'$, such as to cancel out its second coordinate (making the epipolar lines parallel to the x-axis). The translation terms, u_0 and v_0, are not involved in the distortion, four degrees of freedom are left, given by the two scaling factors, s_x and s_y, the skew s_{xy} and the rotation angle in $\mathbf{R}_x$.

Minimizing distorsion using a criterion based on areas. The criterion to be minimized is the ratio of the area of the rectangle with sides parallel to the x- and y-axes circumscribing the rectified image to the area of the rectified image (see figure 5).

This criterion is valid as soon as these areas are not infinite, that is, as soon as the line l (resp. l'), which is mapped by $\hat{\mathbf{H}}$ (resp. $\hat{\mathbf{H}}'$), to the line at infinity, does not go through any point of the first (resp. second) image. If e (resp. e') does not lie in the first (resp. second) image, $\hat{\mathbf{H}}$ (resp. $\hat{\mathbf{H}}'$) can

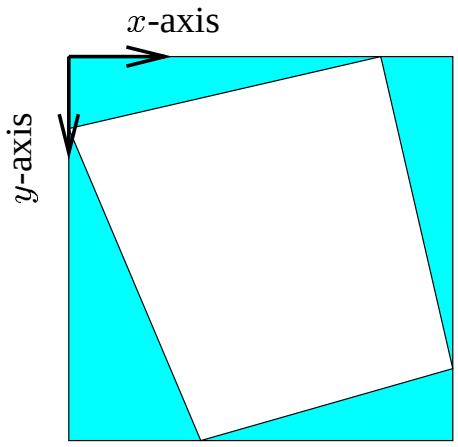

Figure 5: The area-based criterion: Minimizing the relative area of the filled region.

be chosen to verify this constraint, since equation (27) (resp. (28)) show that l (resp. l'), which is represented by the last row of $\hat{\mathbf{H}}$ (resp. $\hat{\mathbf{H}}'$), is only constrained to go through e (resp. e').

$\hat{\mathbf{H}}'$ is decomposed as explained in the paragraph above so that the criterion is a scalar function of s_x, s_y, s_{xy} and the angle θ_x of $\mathbf{R}_x$. Since the criterion is non-linear and its derivatives are not easily computable, a direction-set method is used, namely, the Powell's method. s_x and s_y are initialized to 1 and s_{xy} and θ_x, to 0. At the end of the minimization, s_x, s_y, u_0 and v_0 are adjusted so that the rectified image is of the same size and at the same position in the plane as the initial image.

4.3.3 The decomposition of $\hat{\mathbf{H}}^{-1}$.

Here we present another approach in which a particular parametrization of $\hat{\mathbf{H}}^{-1}$ allows us to isolate the parameters responsible for image distorsion, and estimate their values so as to minimize distorsion.

For simplicity, we express image points coordinates with respect to a normalized coordinate system, in which the image occupies the unit square. Using homogeneous coordinates, we denote by $[e_x, e_y, e_z]$ the coordinates of the epipole e. We now describe a parametrization of $\hat{\mathbf{H}}^{-1}$ that explicitly introduces two free rectification parameters. The other parameters correspond to two scaling factors (one horizontal and one vertical), and one horizontal translation which can be applied to both rectified images. These parameters can be set arbitrarily, and represent the magnification and clipping of the rectified images.

Let us now see how, using the mapping of four particular points, we define a parameterization for $\hat{\mathbf{H}}^{-1}$.

1. $\hat{\mathbf{H}}^{-1}$ maps point $[1, 0, 0]^{\mathrm{T}}$ onto the epipole. This is the condition for the epipolar lines to be horizontal in the rectified images.

2. We impose that the origin of the rectified image be mapped onto the origin of the image. This sets two translation parameters in the rectified image plane). In other words, $\hat{\mathbf{H}}^{-1}[0,0,1]^{\mathrm{T}} = \lambda[0,0,1]^{\mathrm{T}}$.

3. Since $\hat{\mathbf{H}}^{-1}$ maps horizontal lines onto epipolar lines, we impose that the top-right corner of the image be mapped onto point $t_r = [e_x, e_y, e_x]$ of the image, intersection of the epipolar line of the left corner with the right edge of the image[5] (Figure 6). This sets the horizontal scale factor on the rectified image coordinates.

4. Fourth, we impose that the low-lefthand corner of the rectified image be mapped onto the epipolar line of the low-lefthand corner of the image. This sets the vertical scale factor of the rectified image coordinates.

From the first three points, we infer that matrix $\hat{\mathbf{H}}^{-1}$ is of the form:

$$\hat{\mathbf{H}}^{-1} = \begin{bmatrix} e_x & ? & 0 \\ e_y & ? & 0 \\ e_z & ? & e_x - e_z \end{bmatrix}$$

From the fourth point, we have $(\hat{\mathbf{H}}^{-1}[0,1,1]^{\mathrm{T}})^{\mathrm{T}}(\mathbf{e} \times [0,1,1]^{\mathrm{T}}) = 0$. In other words, $\hat{\mathbf{H}}^{-1}[0,1,1]^{\mathrm{T}}$ is a linear combination of $\mathbf{e}$ and $[0,1,1]^{\mathrm{T}}$, so there exist α, β such that

$$\hat{\mathbf{H}}^{-1} = \begin{bmatrix} e_x & \alpha e_x & 0 \\ e_y & \alpha e_y + \beta & 0 \\ e_z & \alpha e_z + \beta + e_z - e_x & e_x - e_z \end{bmatrix}$$

Assuming that the rectification plane is known (homography H), any choice of α, β defines a rectification matrix for the two images.

Minimizing distorsion using orthogonality. We choose α, β so as to introduce as little image distortion as possible. Since there is no absolute measure of global distortion for images, the criterion that we use is based on the following remark: In the rectified images, epipolar lines are orthogonal to pixel columns. If the rectification transformation induced no deformation, it would preserve orthogonality, so the image by $\hat{\mathbf{H}}^{-1}$ of lines along pixel columns would be orthogonal to epipolar lines.

Let us now consider one scanline $\langle d_r \rangle$ of the rectified image, and two points t, b which are respectively the top, bottom points of a vertical line $\langle l \rangle$ of the rectified image. The epipolar line $\langle d_r \rangle$ corresponds to epipolar lines $\langle d_i \rangle, \langle d'_i \rangle$ in the initial images. The two lines d_r and $\langle l \rangle$ are orthogonal. Assuming that the rectification transformation preserves orthogonality, lines $\langle \hat{\mathbf{H}}^{-1}(t), \hat{\mathbf{H}}^{-1}(b) \rangle$ and $\langle d_i \rangle$ should be orthogonal (see Figure 7), as well as lines $\langle \hat{\mathbf{H}}'^{-1}(t), \hat{\mathbf{H}}'^{-1}(b) \rangle$ and $\langle d'_i \rangle$.

For a given value of parameters α, β, we define the residual

$$R(\alpha, \beta) = ((\mathbf{V}\mathbf{d}_i)^T \mathbf{V}(\hat{\mathbf{H}}^{-1}\mathbf{t}) \times (\hat{\mathbf{H}}^{-1}\mathbf{b}))^2$$

[5] Since cameras are in a horizontal configuration, this intersection point exists. However, the same method can be easily adapted to other cases.

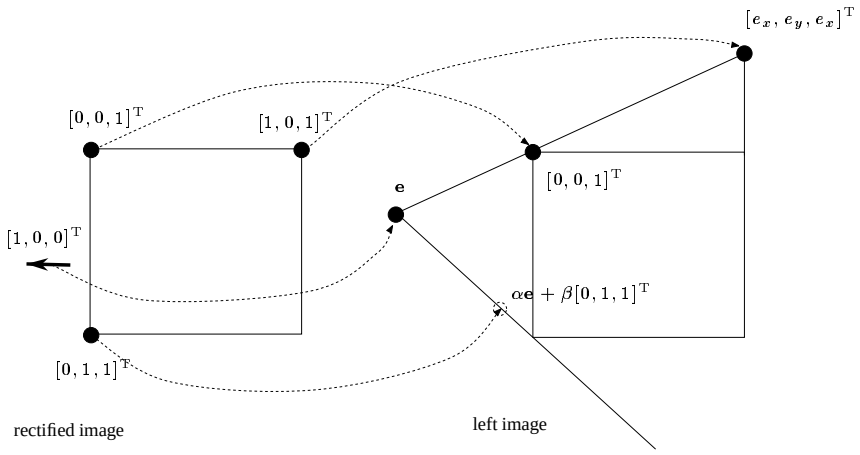

Figure 6: $\hat{\mathbf{H}}^{-1}$ maps three corners of the rectified image onto particular points of the image, and the point at infinity $[1,0,0]^{\mathbf{T}}$ onto the epipole $\mathbf{e}$.

with

$$\mathbf{V} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

This term is the dot-product of the directions of the two lines $\langle \hat{\mathbf{H}}^{-1}(t), \hat{\mathbf{H}}^{-1}(b) \rangle$ and $\langle d_i \rangle$. An analogous term $R'(\alpha, \beta)$ can be defined in the right image, with the rectification transformation $\hat{\mathbf{H}}'^{-1} = \mathbf{H}^{-1}\hat{\mathbf{H}}^{-1}$.

To determine rectification transformations, we compute α, β which minimize the sum $(R(\alpha, \beta) + R'(\alpha, \beta))$ computed for both the left and the right columns of the rectified image, i.e. $\mathbf{t}$ and $\mathbf{b}$ having respective values $[0,0,1]^T$ and $[0,1,1]^T$ on the one hand, $[1,0,1]^T$ and $[1,1,1]^T$ on the other hand. One can see easily that the resulting expression is the square of a linear expression in α, β. It can be minimized very efficiently using standard linear least-squares techniques.

The two epipolar lines $\langle d_i \rangle, \langle d'_i \rangle$ are chosen arbitrarily, so as to represent an "average direction" of the epipolar lines in the images. In practice, the pair of epipolar lines defined by the center of the left image provides satisfactory results.

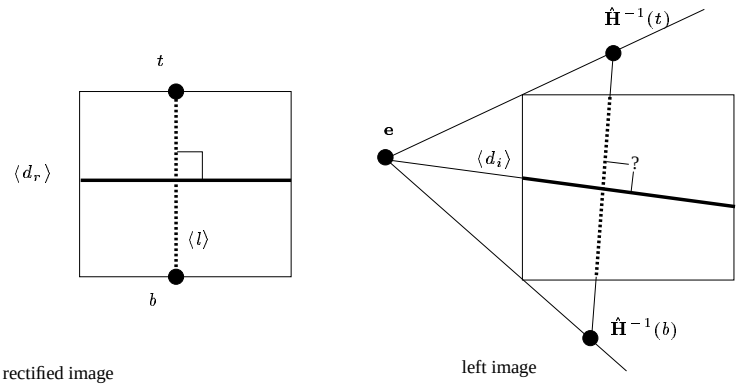

Figure 7: Lines involved in the determination of the rectification transformation (see text).

5 Positioning points with respect to a plane

Measuring the positions of points with respect to a reference plane is essential for robot navigation. We will show in sections 6 and 7 several applications which are based on this measurement. In this section we study how to compare distances of points to a reference plane under minimal calibration assumptions.

5.1 Comparing point distances to a reference plane

For convenience we will adopt the terminology which corresponds to the particular application of section 7.3 where the reference plane is the *ground* plane, and the robot needs to estimate the relative *heights* of visible points, i.e., their relative distances to the ground plane.

The distance of a point M to the reference plane is related to the location of M along the direction orthogonal to the plane. This notion can clearly not be captured at the projective level. Let us now see under which calibration assumptions we will be able to compare point heights.

Let us introduce an (arbitrary) reference point O which does not belong to the ground plane. In practice this point is defined by its two image projections o, o', chosen arbitrarily so as to satisfy the above constraints:

- o, o' satisfy the epipolar constraint,
- both o and o' lie outside of the images,
- o and o' do not satisfy the homographic relation of the reference plane

This guaranties that point O does not lie on the plane, and is different from any observable point.

We now consider the line $\langle D \rangle$ orthogonal to the reference plane and passing through O. Denoting by Q_D the intersection between $\langle D \rangle$ and the ground plane, the height of M is in fact equal to the signed distance $Q_D M_D$, where M_D is obtained by projecting M on $\langle D \rangle$ parallel to the reference plane.

From simple affine geometric properties, the ratios of signed distances $Q_D M_D / O Q_D$ and QM/OQ are equal. Thus, if we consider an arbitrary point M_r which we declare to be at height one from the reference plane, the height of M in terms of this unit can be expressed as (see Figure 8):

$$h = \frac{QM/OQ}{Q_\mathrm{r} M_\mathrm{r}/OQ_\mathrm{r}} \tag{40}$$

Affine projection: In practice, we cannot directly compute distances between 3D points. However, we can compute their projections on the image planes. Ratios of distances are affine invariants, so if we assume that, say, the right camera performs affine projection, we can write

$$h = \frac{q'm'/o'q'}{q'_\mathrm{r} m'_\mathrm{r}/o'q'_\mathrm{r}}$$

Under affine viewing, this definition of height is exact in the sense that h is proportional to the distance between M and the reference plane. Otherwise, this formula is only an approximation. At

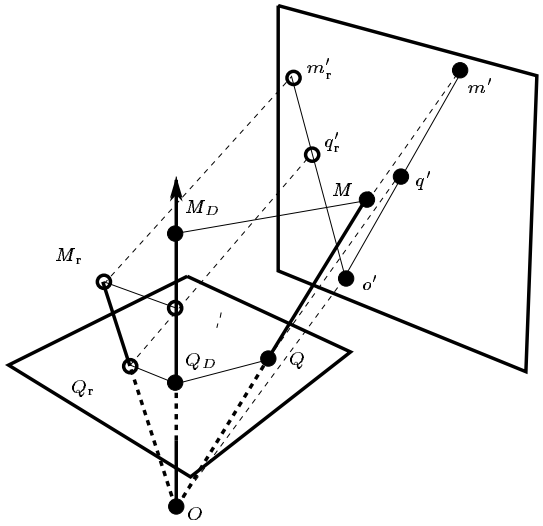

Figure 8: Computation of relative heights with respect to unit point M_r under affine projection (see text).

any rate, it turns out to be accurate enough for some navigation applications, in which points for which heights have to be compared are at relatively long range from the camera, and within a relatively shallow depth of field (cf Section 7.3).

Perspective projection: If the affine approximation is not valid, we need to relate relative heights to projective invariants. Instead of considering ratios of distances, we consider cross-ratios, which are invariant by projection onto the images.

Let us assume that we know the homography of a plane $\Pi_{//}$ parallel to the ground plane (This is in fact equivalent to knowing the line at infinity of the ground plane). Intersecting line $\langle OM \rangle$ (resp. $\langle OM_r \rangle$) with plane $\Pi_{//}$ defines a point N (resp. N_r) aligned with O, Q, M (resp. O, Q_r, M_r).

The cross-ratio $\{Q, N; M, O\}$ is by definition equal to

$$\frac{QM/OQ}{NM/ON}$$

Based on simple affine properties, the denominator of the above fraction is also equal to

$$N_D M_D / O N_D$$

where N_D is the projection of N on $\langle D \rangle$ parallel to plane $\Pi_{//}$, i.e., the intersection of $\Pi_{//}$ and $\langle D \rangle$.

Similarly, we have

$$\{O, Q_{\mathrm{r}}; M_{\mathrm{r}}, N_{\mathrm{r}}\} = \frac{Q_{\mathrm{r}}M_{\mathrm{r}}/OQ_{\mathrm{r}}}{N_D M_D/ON_D}$$

As a consequence, the ratio of cross-ratios $\{Q, N; M, O\}/\{Q_{\mathrm{r}}, N_{\mathrm{r}}; M_{\mathrm{r}}, O\}$ is equal to h (as defined in Equation 40). Using projective invariance, we can then express the height of M with respect to M_{r} as

$$h = \frac{\{q, n; m, o\}}{\{q_{\mathrm{r}}, n_{\mathrm{r}}; m_{\mathrm{r}}, o\}}$$

We remark that the ratio of heights with respect to a plane can be captured at a calibration level which is intermediate between projective and affine: knowing the plane at infinity is not necessary, one only needs to know the line at infinity of the reference plane.

5.2 Interpreting disparities

In this section we assume that images have been rectified with respect to the reference plane, and relate positions ot points relative to the plane to image disparities.

The measure $\mathcal{D}$ of the disparity assigned to a point correspondence after the rectification with respect to a plane and the correlation along the epipolar lines, described in section 4, is in turn related to the position of the corresponding point of the scene with respect to the plane.

Indeed, with the notations of section 4,

$$\mathcal{D} = \hat{x}' - \hat{x}$$

So, according to equations (29) and (38), we have

$$\mathcal{D} = \alpha \frac{T_P}{\hat{z}' Z'_{C'}} = \alpha \hat{\lambda}' \frac{T_P}{Z_R} \tag{41}$$

In order to interpret $\mathcal{D}$, we introduce the signed distance $d(M, \Pi)$ of a point $M = [X, Y, Z, T]^T$ to a plane Π defined by its unit normal $\mathbf{n}$ and its distance d to the origin:

$$d(M, \Pi) = d - \mathbf{n}^T [\frac{X}{T}, \frac{Y}{T}, \frac{Z}{T}]^T$$

The sign of $d(M, \Pi)$ is the same for all the points M located at the same side of Π and $|d(M, \Pi)|$ is equal to the distance of M to Π. Similarly, we introduce the signed distance $d(m, l)$ of a point $m = [x, y, z]^T$ to a line l defined by its unit normal $\mathbf{n}$ and its distance d to the origin:

$$d(m, l) = d - \mathbf{n}^T [\frac{x}{z}, \frac{y}{z}]^T$$

The sign of $d(m, l)$ is the same for all the points m located at the same side of l and $|d(m, l)|$ is equal to the distance of m to l. We have then, according to equations (21) and (3),

$$T_P = \frac{T_C}{d\mu} d(M, \Pi)$$

$$\begin{aligned} Z'_{C'} &= T_C d(M, \Pi'_f) \\ \hat{z}' &= \frac{1}{k} d(m', l'_\infty) \quad \text{with} \quad k = \sqrt{h'^2_{31} + h'^2_{32}} \\ Z_R &= T_C d(M, \hat{\Pi}_f) \end{aligned}$$

where Π'_f is the focal plane of the second view, $\hat{\Pi}_f$ the focal plane of the rectified views (see figure (4)) and l'_∞ the line of $\mathcal{R}'$ whose image by $\hat{\mathbf{H}}' = [h'_{ij}]$ is the line at infinity.

Now, using these signed distances, we write $\mathcal{D}$ in three different ways:

$$\hat{z}'\mathcal{D} = \frac{\alpha}{d\mu} \frac{d(M, \Pi)}{d(M, \Pi'_f)} \tag{42}$$

$$\mathcal{D} = \frac{k\alpha}{d\mu} \frac{d(M, \Pi)}{d(m', l'_\infty) d(M, \Pi'_f)} \tag{43}$$

$$\mathcal{D} = \frac{\alpha \hat{\lambda}'}{d\mu} \frac{d(M, \Pi)}{d(M, \hat{\Pi}_f)} \tag{44}$$

Since α, μ, d, $\hat{\lambda}'$ and k do not depend on M and m, we deduce from these equations the following three interpretations:

- From equation (42), we deduce that, if M is a visible point, which implies that $d(M, \Pi'_f) > 0$, the sign of $\hat{z}'\mathcal{D}$ gives its position with respect to Π. Furthermore, $|\hat{z}'\mathcal{D}|$ is proportional to the ratio of the distance of M to Π to the distance of M to Π'_f.
- From equation (43), we deduce that, if M is a visible point, the sign of $\mathcal{D}$ usually gives its position with respect to Π. Indeed, l'_∞ usually does not go through any point of the image so that the sign of $d(m', l'_\infty)$ is usually the same for all the points considered. In fact, l'_∞ is usually far away from the image, so that $d(m', l'_\infty)$ does not really depend on m for the points considered and $|\mathcal{D}|$ is approximately proportional to the ratio of the distance of M to Π to the distance of M to Π'_f.
- From equation (44), we deduce that the sign of $\mathcal{D}$ gives the position of M with respect to Π and $\hat{\Pi}_f$ and $|\mathcal{D}|$ is proportional to the ratio of the distance of M to Π to the distance of M to $\hat{\Pi}_f$. According to equation (27), $e' \in l'_\infty$ so that l'_∞ is an epipolar line. l'_∞ is thus the image in the second view of an epipolar plane Π_e. Now, the image in $\hat{\mathcal{R}}'$ of Π_e is the line at infinity, so Π_e is parallel to $\hat{\Pi}_f$. Since $\hat{\Pi}_f$ is an epipolar plane, $\hat{\Pi}_f = \Pi_e$ and l'_∞ is, indeed, the intersection of $\hat{\Pi}_f$ and $\mathcal{R}'$. Consequently, since l'_∞ is usually far away from the image, $\hat{\Pi}_f$ and $\mathcal{R}'$, thus $\hat{\Pi}_f$ and Π'_f, are approximately parallel around the image, so that Π'_f may be approximated by $\hat{\Pi}_f$ for the points considered and we turn again to the preceding interpretation.

When Π is the plane at infinity, according to equation (21), we have $T_P = T_C$ and, so,

$$\hat{z}'\mathcal{D} = \frac{\alpha}{d(M, \Pi'_f)}$$

$$\mathcal{D} = \frac{k\alpha}{d(m', l'_\infty) d(M, \Pi'_f)}$$

$$\mathcal{D} = \frac{\alpha \hat{\lambda}'}{d(M, \hat{\Pi}_f)}$$

Thus, in that case, $\hat{z}'\mathcal{D}$ is inversely proportional to the distance of M to Π'_f, $\mathcal{D}$ is approximately inversely proportional to the distance of M to Π'_f and $\mathcal{D}$ is inversely proportional to the distance of M to $\hat{\Pi}_f$.

6 Computing local terrain orientations using collineations

Once a point correspondence is obtained through the process of rectification and correlation along the epipolar lines described in section 4, it is possible to estimate, in addition to a measure of the disparity, a measure of local surface orientations, by using the image intensity function.

The traditional approach to computing such surface properties is to first build a metric model of the observed surfaces from the stereo matches, and then to compute local surface properties using standard tools from Euclidean geometry. This approach has two major drawbacks. First, reconstructing the geometry of the observed surfaces can be expensive because it requires not only applying geometric transformations to the image pixels and their disparity in order to recover three-dimensional coordinates, but also interpolating a sparse 3D-map in space to get dense three-dimensional information. Second, reconstructing the metric surface requires having full knowledge of the geometry of the camera system through exact calibration. In addition, surface properties such as slope are particularly sensitive to the calibration parameters, thus putting more demand on the quality of the calibration.

Here, we investigate algorithms for evaluating terrain orientations from pairs of stereo images using limited calibration information. More precisely, we want to obtain an image in which the value of each pixel is a measure of the difference in orientation relative to some reference orientation, e.g. the orientation of the ground plane, assuming that the only accurate calibration information is the epipolar geometry of the cameras.

We investigate two approaches based on these geometrical tools. In the first approach (Section 6.2), we compute the Sum of Squared Differences (SSD) at a pixel for all the possible skewing configurations of the windows. The skewing parameters of the window which produce the minimum of SSD corresponds to the most likely orientation at that point. This approach uses only knowledge of the epipolar geometry but does not allow the full recovery of the slopes. Rather, it permits the comparison of the slope at every pixel with a reference slope, e.g. the orientation of the ground plane for a mobile robot.

The second approach (Section 6.3) involves relating the actual orientations in space with window skewing parameters. Specifically, we parameterize the space of all possible windows at a given pixel by the corresponding directions on the unit sphere. This provides more information than in the previous case but requires additional calibration information, i.e, the knowledge of the approximate intrinsic parameters of one of the cameras and of point correspondences in the plane at infinity.

6.1 The principle

The guiding principle of this section is the following: the collineation that represents a planar surface is the one that best warps the first image onto the second one.

This principle is used implicitly in all area-based stereo techniques in which the images are rectified and the scene is supposed to be locally fronto-parallel (i.e., parallel to the cameras) at each point [8, 25, 22]. In this case, homographies are simple translations. A rectangular window in image 1 maps onto a similar window in image two, whose horizontal offset (disparity) characterizes the position of the plane in space. The pixel-to-pixel mapping defined by the homography allows to compute a similarity measure on the pixel intensities inside the windows, based on a cross-correlation of the intensity vectors or a SSD of the pixel intensities.

Another example of this concept is the use of windows of varying shapes in area-based stereo by compensating for the effects of foreshortening due to the orientation of the plane with respect to the camera. In the *TELEOS* system [24], for example, several window shapes are used in the computation of the disparity.

We use this principle for choosing, among all homographies that represent planes of various orientations passing through a surface point M, the one that best represents the surface at M. We use a standard window-based correlation algorithm to establish the correspondences between the images (m, m') of M. Since the methods presented above are sensitive to the disparity estimates, we also use a simple sub-pixel disparity estimator.

6.2 Window-based representation

When applying an homography to a rectangular window in image 1, one obtains in general a skewed window in image 2. Since the homographies that we study map m on m', two other pairs of points are sufficient for describing them (see section 3.3). This allows us to introduce a description of the plane orientation by two parameters measured directly in the images.

The standard configuration In order to simplify the presentation, we first describe the relations in the case of cameras in *standard configuration*, i.e.whose optical axes are aligned and such that the axes of the image planes are also aligned. We also assume that the camera intrinsic parameters are known, and considering metric coordinates in the retinal plane instead of pixel coordinates we end up with $\mathbf{A} = \mathbf{I}$ and $\mathbf{H}_\infty = \mathbf{I}$ (using the same notations as in Section 2). Choosing the frame attached to the first camera as the reference frame, the translation between the cameras is assumed to be $\mathbf{t} = [t_x, 0, 0]^t$. Although we describe the approach in this simplified case, the principle remains the same in the general case, though interpreting the equations is more complicated.

In the current case, Equation (12) gives us:

$$\mathbf{H} = \mathbf{I} + \frac{\mathbf{t}\mathbf{n}^t}{d} = \begin{bmatrix} 1 + \frac{n_x t_x}{d} & \frac{n_y t_x}{d} & \frac{n_z t_x}{d} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{45}$$

The distance parameter d can be obtained from the image points as follows: the images m, m' of the three-dimensional point M are known, and related by a horizontal disparity $\mathcal{D}$. From the projection geometry, we have:

$$\mathbf{CM} = \frac{t_x f}{d}\mathbf{Cm} \tag{46}$$

In this equation, $\mathbf{C}$ the origin of the image coordinate system (a 3D-point) and $\mathbf{m}$ is a 2-D projective point of the form $[u, v, 1]^t$. Therefore, we have a simple expression of d as a function of the known variables, $\mathbf{m}$ and $\mathcal{D}$:

$$d = \mathbf{n}^t\mathbf{CM} = \frac{t_x f}{\mathcal{D}}\mathbf{n}^t\mathbf{Cm} \tag{47}$$

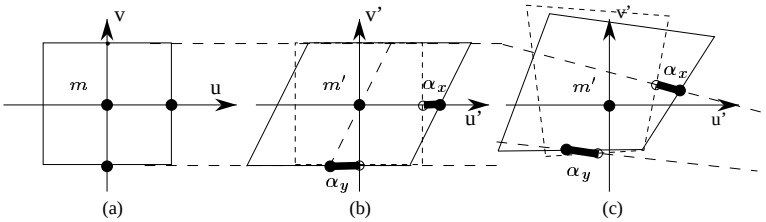

Figure 9: Parametrization of window deformation; (a): left image. (b): right image if the cameras are aligned. (c): right window in general camera configuration.

Substituting (47) in (45), $\mathbf{H}$ can be expressed as function of $\mathbf{n}$, $\mathcal{D}$, and $\mathbf{m}$:

$$\mathbf{H} = \begin{bmatrix} 1+\alpha_x & \alpha_y & \alpha_z \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \alpha_x = \frac{\mathcal{D} n_x}{f \mathbf{n}^t \mathbf{C} \mathbf{m}} \quad \alpha_y = \frac{\mathcal{D} n_y}{f \mathbf{n}^t \mathbf{C} \mathbf{m}} \tag{48}$$

α_z only has a translational effect through $\mathbf{H}$, so it has no influence on the resulting window shape. The two parameters α_x and α_y fully characterize the effect of $\mathbf{H}$ on the window shape. Geometrically, α_x is the horizontal displacement of the center points of the left and right edges of the correlation window and α_y is the displacement of the centers of the top and bottom edges of the window (See figure 9 (b)).

The general case To simplify the equations, this discussion of window skewing was presented in the case in which the cameras are aligned. The property remains essentially the same when the cameras are in a general configuration. In that case, the window shape can still be described by $\boldsymbol{\alpha} = [\alpha_x, \alpha_y]^t$ that represents the displacements of the centers of the window edges along the epipolar lines given by $\mathbf{F}$ instead of along the lines of the image in the case of aligned cameras (See figure 9 (c)). Also, we have not included the intrinsic parameter matrices in the computations. It can be easily seen that including these matrices does not change the general form of equation (48); it changes only the relation between $\boldsymbol{\alpha}$ and the orientation of the plane.

We have shown how to parametrize window shape from the corresponding homography. It is important to note that the reasoning can be reversed in that a given arbitrary value of $\boldsymbol{\alpha}$ corresponds to a unique homography at $\mathbf{m}$, which itself corresponds to a unique plane at $\mathbf{M}$.

Slope Computation Using Window Shape: The parameterization of window shape as a function of planar orientations suggests a simple algorithm for finding the slope at M. First, choose a set of values for α_x and α_y. Then, compute a measure (correlation or SSD) for each possible $\boldsymbol{\alpha}$, and find the best slope $\boldsymbol{\alpha}^{min}$ as corresponding to the minimum of the measure. This is very similar to the approach investigated in [1].

If all the parameters of the cameras were known, $\boldsymbol{\alpha}^{min}$ could be converted to the Euclidean description of the corresponding plane in space.

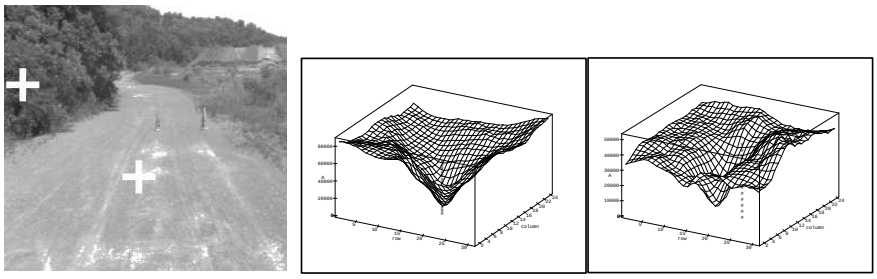

Figure 10: Left: Selected pixels in image 1. Center (resp. right): SSD error as function of $[\alpha_x, \alpha_y]$ at the road (resp. tree) pixel. The reference orientation (α_x^o, α_y^o) is approximately the orientation of the road, and is represented by the vertical dotted lines. Both surfaces have a sharp minimum, but only one (the road point) is located at (α_x^o, α_y^o).

Otherwise, it is still possible to compare the computed orientation with the orientation of a reference plane Π^o whose line at infinity is known. Indeed, the homography of the plane passing through M and parallel to Π^o can be computed (Section 3.3), and its parameters $\boldsymbol{\alpha}^o$ derived; the distance $D = \|\boldsymbol{\alpha}^{min} - \boldsymbol{\alpha}^o\|$ is a measure of the difference between the slope of the terrain at M and the reference slope. Figure 10 shows an example in the case of two single pixel correspondences selected in a pair of images. The SSD is computed from the Laplacian of the images rather than from the images themselves in order to eliminate the offset between the intensities of the two images.

In practice, Π^o can be defined by three point correspondences in a pair of training images, two of which lying far enough from the camera to be considered as lying at infinity, thus defining the line at infinity of the plane. Another way to proceed would be to estimate Π^o from any three point correspondences, and to compute its intersection with the plane at infinity Π_∞ estimated with three non-aligned correspondences corresponding to remote points.

Limitations: There are two problems with the $[\alpha_x, \alpha_y]$ representation of plane directions. First, depending on the position of the point in space, the discretization of the parameters may lead to very different results due to the non-uniform distribution of the plane directions in $\boldsymbol{\alpha}$-space. In particular, the discrimination between different plane directions becomes poorer as the range to the surface increases. This problem becomes particularly severe when the surfaces are at a relatively long range from the camera and when the variation of range across the image is significant.

The second problem is that it is difficult to interpret consistently the distance D between $\boldsymbol{\alpha}^{min}$ and $\boldsymbol{\alpha}^o$ across the image. Specifically, a particular value of D corresponds to different angular distances between planes depending on the disparity, but also on the position of the point in the image.

6.3 Normal-based representation

Based on the limitations identified above, we now develop an alternate parameterization, that consists of discretizing the set of all possible orientations in space, and then of evaluating the corresponding

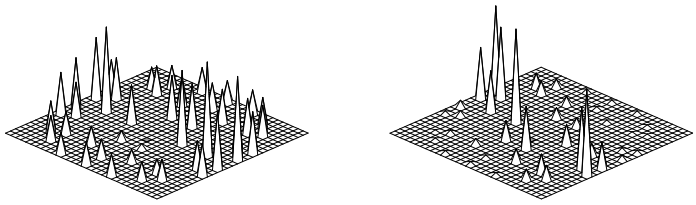

Figure 11: Distribution of SSD at the two selected pixels of figure (Left: road; Right: trees). The SSD distribution is plotted with respect to (n_x, n_y), i.e., two coordinates of the 3D normal. The reference orientation (close to the road) is represented as a plain surface patch. On the left diagram, the computed SSD values are low in the neighborhood of the reference orientation. On the other one, the low SSD values are further from the reference orientation.

homographies. This assumes that the intrinsic parameters of the first camera are known as well as the collineation of the plane at infinity.

Slope computation using normal representation: Assuming that we know $\mathbf{H}_\infty$ and $\mathbf{A}$, we can compute the homography $\mathbf{H}$ of a plane Π defined by a given pair of points (m, m') and a normal vector $\mathbf{n}$ in space. Indeed, the plane Π_C that is orthogonal to $\mathbf{n}$ and contains the optical center C projects in the first image onto a line l^∞ and Equation (14) tells us that

$$\mathbf{l}_1^\infty = \mathbf{A}^{-T}\mathbf{n} \tag{49}$$

This line is the projection in the first image of any line of the plane that does not contain C, so unless C is at infinity, it is the image of the line at infinity of Π_C and thus of Π since both planes are parallel. Given the homography $\mathbf{H}_\infty$ of the plane at infinity, we can then compute the corresponding line $\mathbf{l}_2^\infty = {\mathbf{H}_\infty}^{-1T}\mathbf{l}_1^\infty$ in the second image. Finally, according to section 3.3, (m, m'), $\mathbf{l}_1^\infty$, $\mathbf{l}_2^\infty$ and the knowledge of the epipolar geometry allow to compute $\mathbf{H}$.

By sampling the set of possible orientations in a uniform manner, we generate a set of homographies that represent planes of well-distributed orientations at a given point M. Then the algorithm of the previous section is used directly to evaluate each orientation $\mathbf{n}_i$. In the current implementation, we sample the sphere of unit normals into 40 different orientations using a regular subdivision of the icosahedron.

Figure 11 shows the SSD distributions in the case of the two pixels studied in Figure 10. Though correct, the results point out one remaining problem of this approach: the SSD distribution may be flat because of the lack of signal variation in the window. This is a problem with any area-based technique. In the section 6.4, we present a probabilistic framework which enables us to address this problem.

It is important to note that the orientation of the cameras does not need to be known and that the coordinate system in which the orientations are expressed is unimportant. In fact, we express all the orientations in a reference frame attached to the first camera, which is sufficient since all we need is to *compare* orientations, which does not require the use of a specific reference frame. Consequently, it is not necessary to use the complete metric calibration of the cameras.

A-priori geometric knowledge: In practice, $\mathbf{H}_\infty$ is estimated as described at the end of the previous section. Since this only gives an approximation, the lines $\mathbf{l}_1^\infty, \mathbf{l}_2^\infty$ that we compute from a given orientation do not really represent a line at infinity. Thus, the planes that correspond to this orientation rotate around a fixed line in space instead of being parallel. For practical purposes, the line is far enough so that this discrepancy does not introduce significant errors.

The matrix $\mathbf{A}$ represents the intrinsic parameters of the first image. Since we are interested in the slopes in the image relative to some reference plane, it is not necessary to know $\mathbf{A}$ precisely. Specifically, an error in $\mathbf{A}$ introduces a consistent error in the computation of the homographies which is the same for the reference plane and for an arbitrary plane, and does not affect the output of the algorithm dramatically.

We finally remark that if $\mathbf{A}$ is modified by changing the scale in the first image, the results remain unchanged. This geometric property, observed by Koenderink [16], implies that only the aspect ratio, the angle of the pixel axes and the principal point of the first camera need to be known.

6.4 Application to estimation of terrain traversability

Although the accuracy of the slopes computed using the algorithms of the previous section is not sufficient to, for example, reconstruct terrain shape, it provides a valuable indication of the traversability of the terrain. Specifically, we define the traversability at a point as the probability that the angle between a reference vertical direction and the normal to the terrain surface is lower than a given angular threshold. The term traversability comes from mobile robot navigation in which the angular threshold controls the range of slopes that the robot can navigate safely.

Estimating traversability involves converting the distribution of SSD values $S(\boldsymbol{\alpha})$ at a pixel m to a function $f(\boldsymbol{\alpha})$ which can be interpreted as the likelihood that $\boldsymbol{\alpha}$ corresponds to the terrain orientation at m. We then define the traversability measure $T(m)$ as the probability that this orientation is within a neighbourhood R around the direction of the reference plane $\boldsymbol{\alpha}^o$:

$$T(m) = \sum_{\boldsymbol{\alpha} \in R} f(\boldsymbol{\alpha})$$

We use a formalism similar to the one presented in [22] in order to define f. Assuming that the pixel values in both images are normally distributed with standard deviation σ^2, the distribution of $\boldsymbol{\alpha}$ is given by:

$$f(\boldsymbol{\alpha}) = \frac{1}{K} \exp - \frac{S(\boldsymbol{\alpha})}{\sigma^2} \tag{50}$$

where K is a normalizing factor.

This definition of T has two advantages. First, it integrates the confidence values computed for all the slope estimates (50) into one traversability measure. In particular, if the distribution of $f(\boldsymbol{\alpha})$ is relatively flat, $T(m)$ has a low value, reflecting the fact that the confidence in the position of the minimum of $S(\boldsymbol{\alpha})$ is low. This situation occurs when there is not enough signal variation in the images or when m is the projection of a scene point that is far from the cameras.

The second advantage of this definition of traversability is that the sensitivity of the algorithm can be adjusted in a natural way. For example, if R is defined as the set of plane orientations which are at an angle less than θ from Π^o, the sensitivity of $T(m)$ increases as θ decreases.

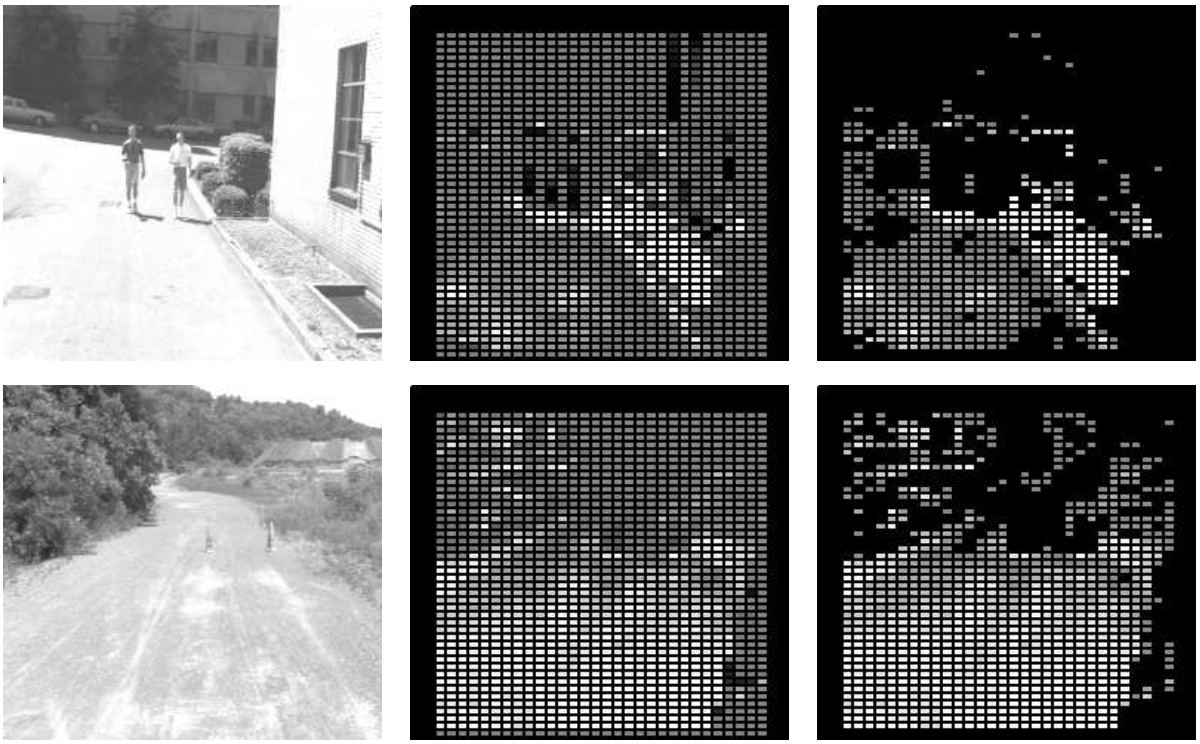

Figure 12: Examples of traversability maps computed on two pairs of images (see text).

Figure 12 shows the results on two pairs of images of outdoor scenes. The first image of each pair is displayed on the left. The center images show the complete traversability maps, Once again, the influence of the signal is noticeable. In particular, in the top example, a large part of the road has a rather low traversability, because there is little signal in the images. On the contrary, the values corresponding to the highly textured sidewalk are very high.

The right image shows the regions that have a probability greater than the value we would obtain if there were no signal in the images, i.e. the regions that could be considered as traversable. In both cases, the obstacles have low traversability values.

Figure 13 shows the result of evaluating $T(m)$ for three different values of θ. Only the traversable regions are shown. As θ increases, the influence of the signal becomes less noticeable, and the likelihood of a region to be traversable increases.

The measure of traversability can be easily integrated into navigation systems such as the one presented in section 7.

Figure 13: Traversability map from the distribution of slopes on real data. Left: small admissible region $\theta = 20^o$; Center: medium admissible region $\theta = 45^o$; Right: large admissible region $\theta = 75^o$.

7 Navigating

In this section we show three robotic applications of the geometric properties presented in the above sections. In the first one, stereo is used to detect close obstacles. In the second one, the robot uses affine geometry to follow the middle of a corridor. In the third one, relative heights and orientations with respect to the ground plane are used for trajectory planning.

7.1 Detecting obstacles

This section describes how to use the previous results to provide a robot with the ability to detect obstacles. The only requirement is that the robot is equipped with a stereo rig which can be very simply calibrated, as explained next.

Let us imagine the following calibration steps:

- as described in Section 3.1, some correspondences between two views taken by the cameras are found;
- these correspondences are used to compute the fundamental matrix, as described in Section 3.2;
- three particular correspondences are given to the system; they correspond to three object points defining a virtual plane Π in front of the robot;
- the H-matrix of Π is computed as described in Section 3.3;

The fundamental matrix, as well as the plane H-matrix, remain the same for any other pair of views taken by the system, as long as the intrinsic parameters of both cameras and the attitude of one camera with respect to the other do not change.

According to Section 5, by repeatedly performing rectifications with respect to Π, the robot knows whether there are points in front between itself and Π by looking at the sign of their disparity and can act in consequence. If the distance d_0 of the robot to Π is known, the robot may, for example, move forward from the distance d_0. Furthermore, if Π and Π'_f intersect sufficiently far away from the cameras, it can detect whether the points are moving away or towards itself. Indeed, Π and the focal plane Π'_f may then be considered as parallel around the images, so that, for the points considered, $d(M,\Pi)$ is proportional to $d(M,\Pi'_f) - d(\Pi,\Pi'_f)$, where $d(\Pi,\Pi'_f) = d(M_\Pi,\Pi'_f)$ for any point $M_\Pi \in \Pi$. According to equation (42), $\hat{z}'\mathcal{D}$ is then approximatively proportional to

$$1 - \frac{d(\Pi,\Pi'_f)}{d(M,\Pi'_f)}$$

thus, is a monotonic function of the distance of M to Π'_f.

At last, since we are only interested in the points near the plane, which have a disparity close to zero, we can limit the search along the epipolar line of the correspondent point $\hat{m}'$ of any point $\hat{m}$ to an interval around $\hat{m}$, which significantly reduces the computation time.

An example is given in Figures 14, 15, 16 and 17. Figure 14 shows as dark square boxes the points used to define a plane and the image of a fist taken by the left camera. Figure 15 shows the left

Figure 14: The paper used to define the plane and the left image of the fist taken as an example.

Figure 15: The left and right rectified images of the fist.

and right images once rectified with respect to this plane. Figure 16 shows the disparity map obtained by correlation. Figure 17 shows the segmentation of the disparity map in two parts. On the left side, points with negative disparities, that is points in front of the reference plane, are shown. The intensity encodes closeness to the camera. Similarly, the right side of the figure shows the points with positive disparities, that is the points which are beyond the reference plane.

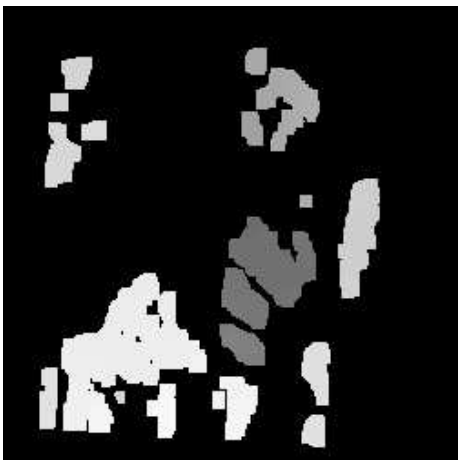

Figure 16: The disparity map obtained from the rectified images of Figure 15.

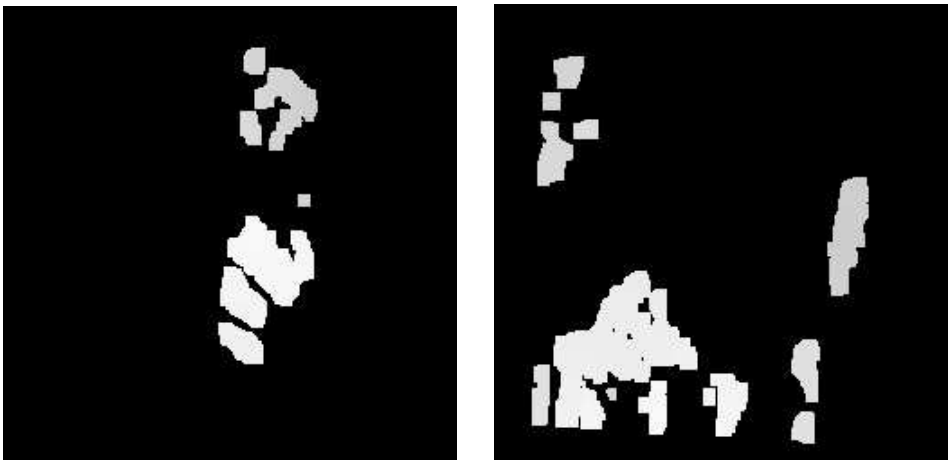

Figure 17: The absolute value of the negative disparities on the left, showing that the fist and a portion of the arm are between the robot and the plane of rectification, and the positive disparities on the right, corresponding to the points located beyond the plane.

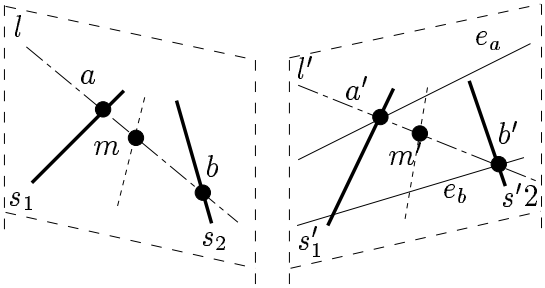

Figure 18: Determining a point at the middle of a corridor (see Section 7.2).

7.2 Navigating along the middle of a corridor

If we add to the computation of the fundamental matrix during the calibration stage, the computation of the homography of the plane at infinity, using the method described in Section 3.4, the robot becomes able to compute ratios of three aligned points, thus, for example, the middle of a corridor and do visual servoing.

Indeed, if we represent the projections of the sides S_1 and S_2 of the corridor by s_1 and s_2 in the first image and s'_1 and s'_2 in the second image (see Figure 18) and choose any point a of s_1 and any point b of s_2, projections in the first image of a point A of S_1 and a point B of S_2, then the corresponding points a' and b' in the second image are computed as the intersections, respectively, of the epipolar line e_a of a with s'_1 and the epipolar line e_b of b with s'_2. Having (a, a') and (b, b') allows to compute the projections (m, m') of the midpoint M of A and B. If we consider S_1 and S_2 as locally parallel, then M lies on the local middle line of the corridor and computing the projections of another point of this line the same way as M allows to have the projections of this line in the two images.

Figure 19 shows some real sequences used to perform the affine calibration of a stereoscopic system. Six strong correspondences between the four images have been extracted, from which fifteen correspondences of points at infinity have been computed to finally get the homography of the plane at infinity. Figure 20 shows some midpoints obtained once the system calibrated: the endpoints are represented as black squares and the midpoints as black crosses. Figure 21 shows the midline of a corridor obtained from another affinely calibrated system: the endpoints are represented as numbered oblique dark crosses, the midpoints as black crosses and the midline as a black line.

7.3 Trajectory evaluation using relative elevation

A limitation of the conventional approach to stereo driving is that it relies on precise metric calibration with respect to an external calibration target in order to convert matches to 3-D points. From a practical standpoint, this is a serious limitation in scenarios in which the sensing hardware cannot be physically accessed, such as in the case of planetary exploration. In particular, this limitation implies that the vision system must remain perfectly calibrated over the course of an entire mission. Ne-

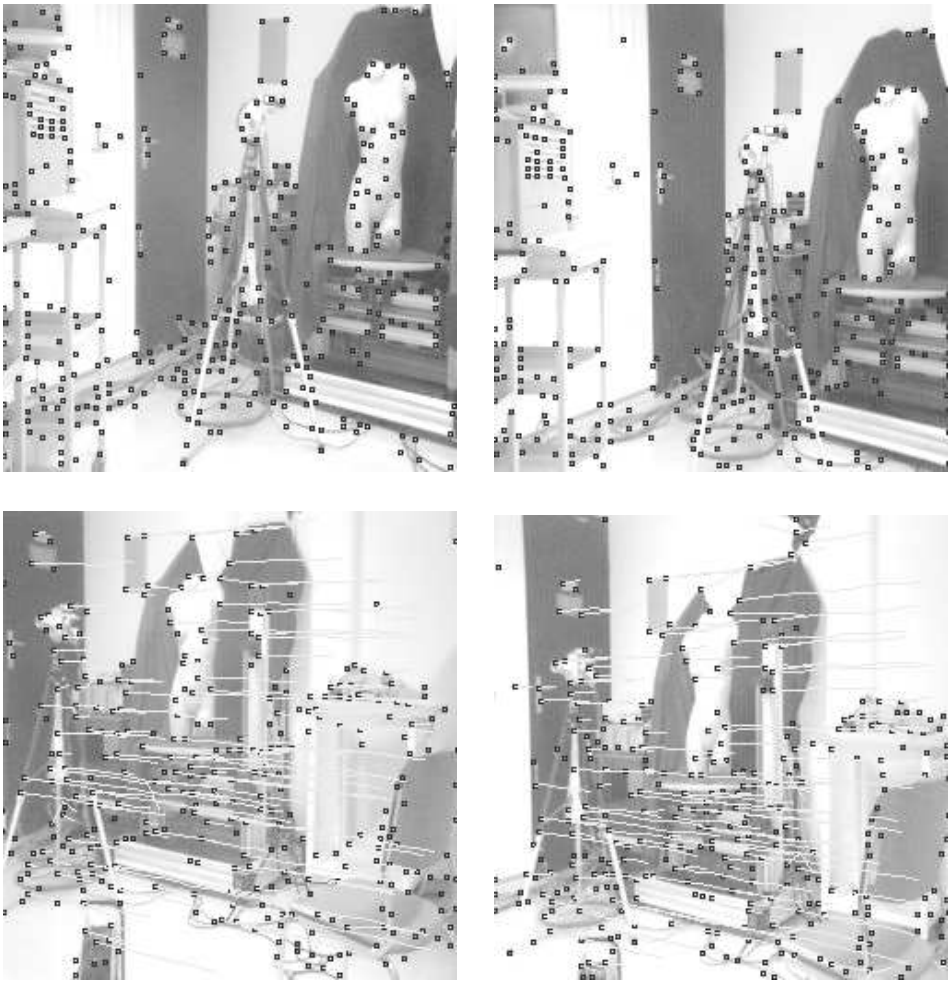

Figure 19: The top images correspond to a first pair of views taken by a stereoscopic system and the bottom images to a second pair taken by exactly the same system after a translation. Among the 297 detected corners of the top left image and the 276 of the top right image, 157 points correspondences have been found by stereo points matching (see Section 3.1), among which 7 outliers have been rejected when computing the fundamental matrix (see Section 3.2). The top to bottom correspondences matching has been obtained by tracking (see Section 3.1).

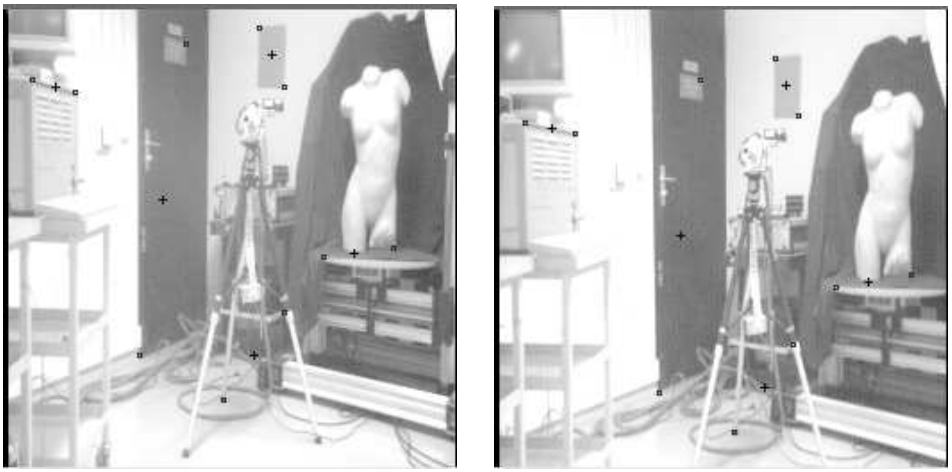

Figure 20: Midpoints obtained after affine calibration (see Section 3.4).

Figure 21: Midline of a corridor obtained after affine calibration (see Section 3.4).

vertheless, navigation should not require the precise knowledge of the 3-D position of points in the scene: What is important is how much a point deviates from the reference ground plane, not its exact position. Based on these observations, we have developed an approach which relies on the measure of relative height with respect to a ground plane (see Section 5.1).

7.3.1 The driving approach

We give only an overview of the approach since a detailed description of the driving system is outside the scope of this book. A detailed description of the stereo driving system can be found in [17]. The autonomous navigation architecture is described in [19] and [13].

In autonomous driving, the problem is to use the data in the stereo images for computing the best steering command for the vehicle, and to update the steering command every time a new image is taken. Our basic approach is to evaluate a set of possible steering directions based on the relative heights computed at a set of points which project onto a regular grid in the image. Specifically, a given steering radius corresponds to an arc which can be projected into a curve in the image. This curve traces the trajectory that the vehicle would follow in the image if it used this steering radius. Given the points of the measurement grid and the set of steering radii, we compute a vote for every arc and every point of the grid which reflects how drivable the arc is. The computed value lies between -1 (high obstacle) and +1 (no obstacle) (Figure 22).

For a given steering radius, the votes from all the grid points are aggregated into a single vote by taking the minimum of the votes computed from the individual grid points. The output is, therefore, a distribution of votes between -1 and +1, -1 being inhibitory, for the set of possible steering arcs. This distribution is then sent to an external module which combines the distribution of votes from stereo with distributions from other modules in order to generate the steering command corresponding to the highest combined vote. Figure 23 shows examples of vote distributions computed in front of visible obstacles.

Characterization of the reference ground plane: The homography of the reference ground plane is estimated from a number of point correspondences related to scene points on this plane. These point correspondences are obtained by selecting points in the first image. Their corresponding points in the second image are computed automatically through the process of rectification and correlation along the epipolar lines, described in Section 4.

Measuring obstacle heights Let us we consider one point of the grid in the first image, for which a corresponding point in the second image has been found by the stereo process. Based on the results of Section 5.1, we can compute its height with respect to the ground. The unit height is defined by a point of the scene, selected manually in one of the two images at the beginning of the experiment and matched automatically in the other image.

This measurement is not sufficient, since we aim at measuring heights along trajectories which are estimated in the ground plane. So, after determining the elevation of a point selected in one image, we determine the point of the ground plane to which this elevation has to be assigned, by projecting the measured 3D point on the (horizontal) ground plane, along the vertical direction. This means

computing the intersection between the ground plane and the vertical line passing through the 3D point. To apply the method of Section 3.3, we need to compute the images of the vertical the line passing through the observed point. For this, we compute the images of the point at infinity in the vertical direction (also called vertical vanishing point) in both images. First, we select manually four points representing two vertical lines in the left image. Matching two of these points we obtain one of the two corresponding lines in the right image. The left vertical vanishing point is obtained by intersecting the two lines in the left image. Computing the intersection of its epipolar line with the line in the right image, we obtain the right vertical vanishing point.

Computing images trajectories This approach assumes that a transformation is available for projecting the steering arcs onto the image plane. Such a transformation can be computed from features in sequences of images using an approach related to the estimation algorithm described above for computing the homography induced by the ground plane.

We first introduce a system of coordinates in the ground plane, attached to the rover, which we call "rover coordinates". At each time, we know in rover coordinates the trajectory which will be followed by the rover for each potential steering command. Further more, for a given motion/steering command sent to the robot, we know from the mechanical design the expected change of rover coordinates from the final position to the initial one. We can even estimate the actual motion using dead-reckoning. Since the transformation is a change of coordinates in the plane, it can be represented by a 3×3 matrix $\mathbf{T}^r$ operating on homogeneous coordinates.

The transformation which we compute is the homography $\mathbf{H}^{ir}$ which maps pixel coordinates in the left image onto rover coordinates. The inverse of this matrix then allows us to map potential rover trajectories onto the left image plane.

Computation of $\mathbf{H}^{ir}$ is done by tracking points across the left images taken at various rover positions. Let us consider two images acquired at positions 1 and 2, with a known rover motion $\mathbf{T}^r_{12}$. Given a point correspondence (p_1, p_2) we have the following equation (up to a scale factor):

$$\mathbf{H}^{ir}\mathbf{p}_1 = \mathbf{T}^r_{12}\mathbf{H}^{ir}\mathbf{p}_2$$

where the only unknown is the matrix $\mathbf{H}^{ir}$. This can be also written

$$\mathbf{H}^{ir}\mathbf{p}_1 \times (\mathbf{T}^r_{12}\mathbf{H}^{ir}\mathbf{p}_2) = 0$$

This yields a system of two independent quadratic equations in the coefficients of $\mathbf{H}^{ir}$. Given a set of displacements and point coordinates, we can write a large system of such equations, which we solve in the least-squares sense using the Levenberg-Marquardt technique.

Using heights to speed up stereo matching The relative height is also used for limiting the search in the stereo matching. More precisely, we define an interval $[h_{min}, h_{max}]$ of heights which we anticipate in a typical terrain. This interval is converted at each pixel to a disparity range $[d_{min}, d_{max}]$. This is an effective way of limiting the search by searching only for disparities that are physically meaningful at each pixel.

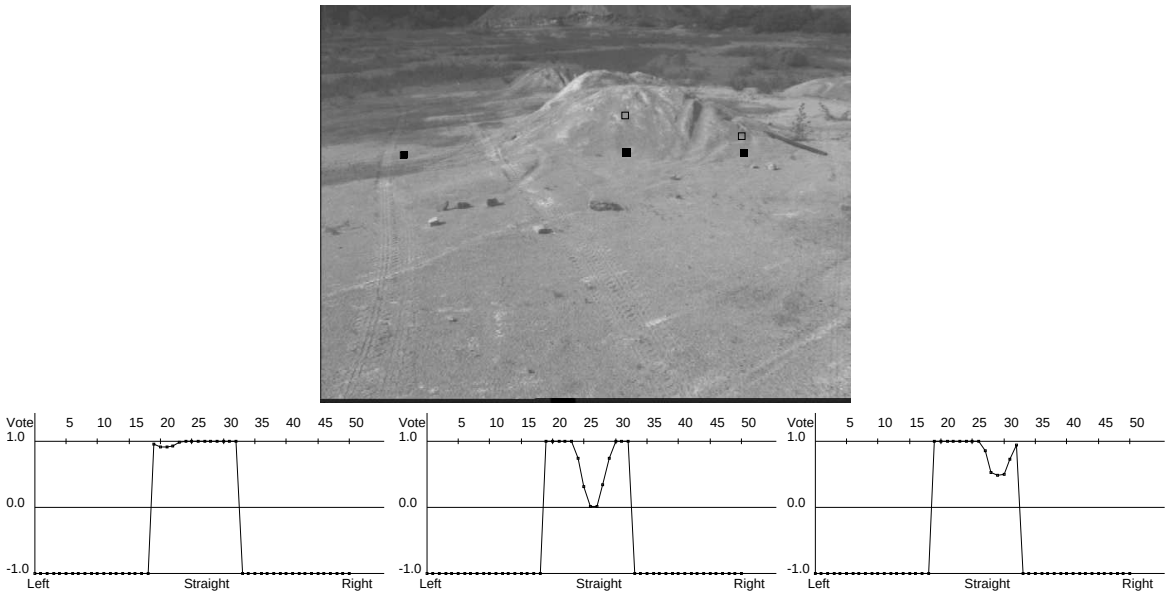

Figure 22: Evaluating steering at individual points; (top) Three points selected in a stereo pair and their projections; (bottom) Corresponding votes for all steering directions.

7.3.2 Experimental results

This algorithm has been successfully used for arc evaluation in initial experiments on the CMU HMMWV [13], a converted truck for autonomous navigation. In this case, a 400 points grid was used. The combination of stereo computation and arc evaluation was done at an average of 0.5s on a Sparc-10 workstations. New steering directions were issued to the vehicle at that rate. This update rate is comparable to what can be achieved using a laser range finder [19].

An important aspect of the system is that we are able to navigate even though a relatively small number of points is processed in the stereo images. This is in contrast with the more conventional approach in which a dense elevation map is necessary, thus dramatically increasing the computation time. Using such a small number of points is justified because it has been shown that the set of points needed for driving is a small fraction of the entire data set independent of the sensor used [15], and because we have designed our stereo matcher to compute matches at specific points.

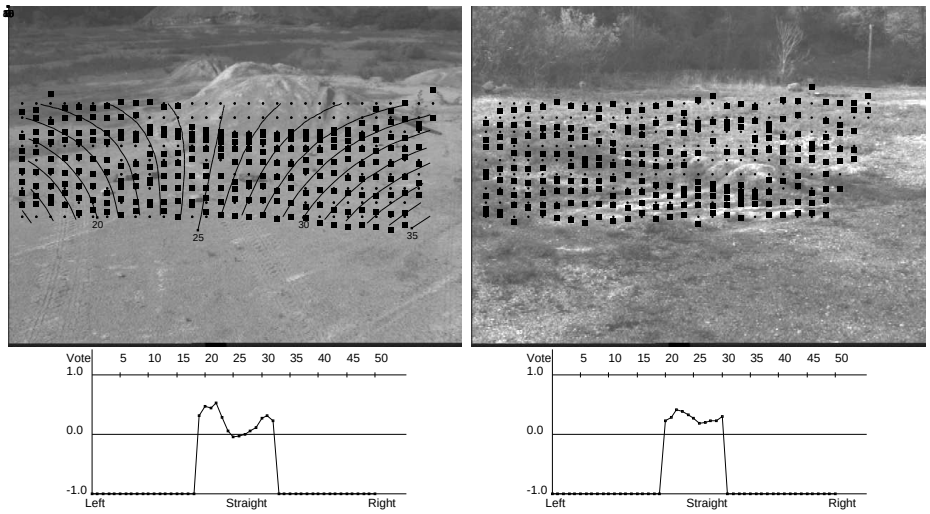

Figure 23: Evaluating steering directions in the case of a large obstacle (left) and a small obstacle (right). (Top): Regular grid of points (dots) and corresponding projections (squares); (Bottom): Distribution of votes (see text).

8 Conclusion

In this chapter we have pushed a little further the idea that only the information necessary to solve a given visual task needs to be recovered from the images and that this attitude pays off by considerably simplifying the complexity of the processing.

Our guiding light has been to exploit the natural mathematical idea of invariance under a group of transformations. This has led us to consider the three usual groups of transformations of the 3-D space, the projective, affine and Euclidean groups which determine a three-layer stratification of that space in which we found it convenient to think about and solve a number of vision problems related to robotics applications.

We believe that this path, even though it may look a bit arduous for a non mathematically enclined reader's point of view, offers enough practical advantages to make it worth investigating further. In particular we are convinced that, apart from the robotics applications that have been described in the paper and for which we believe our ideas have been successful, the approach can be used in other areas such as the representation and retrieval of images from digital libraries.

References

[1] Frédéric Devernay and Olivier Faugeras. Computing differential properties of 3-D shapes from stereoscopic images without 3-D models. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 208–213, Seattle, WA, June 1994. IEEE.

[2] O.D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig ? In G. Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision*, pages 563–578, Santa Margherita Ligure, Italy, May 1992. Springer-Verlag.

[3] O.D. Faugeras and L. Robert. What can two images tell us about a third one ? In J-O. Eklundh, editor, *Proceedings of the 3rd European Conference on Computer Vision*, pages 485–492, Stockholm, Sweden, May 1994. Springer-Verlag.

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

[5] Olivier Faugeras, Tuan Luong, and Steven Maybank. Camera self-calibration: theory and experiments. In G. Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision*, volume 588 of *Lecture Notes in Computer Science*, pages 321–334, Santa Margherita Ligure, Italy, May 1992. Springer-Verlag.

[6] Olivier D. Faugeras. *Three-Dimensional Computer Vision: a Geometric Viewpoint*. MIT Press, 1993.

[7] Olivier D. Faugeras and Francis Lustman. Let us suppose that the world is piecewise planar. In O. D. Faugeras and Georges Giralt, editors, *Robotics Research, The Third International Symposium*, pages 33–40. MIT Press, 1986.

[8] P. Fua. Combining stereo and monocular information to compute dense depth maps that preserve depth discontinuities. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, Sydney, Australia, August 1991.

[9] Gene H. Golub and Charles F. Van Loan. *Matrix computations*. The John Hopkins University Press, Baltimore, Maryland, 1983.

[10] C. Harris and M. Stephens. A Combined Corner and Edge Detector. In *Proceedings 4th Alvey Conference*, pages 147–151, Manchester, August 1988.

[11] R. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *Proceedings of the Conference on International Conference on Computer Vision and Pattern Recognition*, pages 761–764, Urbana Champaign, IL, June 1992. IEEE.

[12] Richard Hartley, Rajiv Gupta, and Tom Chang. Stereo from Uncalibrated Cameras. In *Proceedings of CVPR92, Champaign, Illinois*, pages 761–764, June 1992.

[13] M. Hebert, D. Pomerleau, A. Stentz, and C. Thorpe. A behavior-based approach to autonomous navigation systems: The cmu ugv project. In *To Appear in IEEE Expert, 1994*, 1994.

[14] Koenderink J. J. and A. J. Van Doorn. Affine structure from motion. *Journal Of The Optical Society Of America A*, 8(2):377–385, 1992.

[15] A. Kelly. A partial analysis of the high speed autonomous navigation problem. Technical Report CMU-RI-TR-94-16, The Robotics Institute, Carnegie Mellon, 1994.

[16] J.J Koenderink and A.J. Van Doorn. Goemetry of binocular vision and a model for stereopsis. *Journal Biol. Cybern.*, 21:29–35, 1976.

[17] E. Krotkov, M. Hebert, M. Buffa, F.G. Cozman, and L. Robert. Stereo driving and position estimation for autonomous planetary rovers. In *2nd International Workshop on Robotics in Space*, pages 320–328, Montreal, Quebec, July 1994.

[18] H. Wang L. S. Shapiro and J. M. Brady. A matching and tracking strategy for independently-moving, non-rigid objects. In *Proceedings of BMVC*, 1992.

[19] D. Langer, J. Rosenblatt, and M. Hebert. A reactive system for autonomous navigation in unstructured environments. In *Proc. International Conference on Robotics and Automation*, San Diego, 1994.

[20] Q.-T. Luong, R. Deriche, O.D. Faugeras, and T. Papadopoulo. On determining the Fundamental matrix: analysis of different methods and experimental results. Technical Report RR-1894, INRIA, 1993.

[21] Q.-T. Luong and T. Viéville. Canonic representations for the geometries of multiple projective views. Technical Report UCB/CSD-93-772, University of California at Berkeley, Sept 1993.

[22] L. Matthies. Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation. *The International Journal of Computer Vision*, 1(8), July 1992.

[23] J. L. Mundy and A. Zisserman, editors. *Geometric Invariance In Computer Vision*. MIT Press, 1992.

[24] H.K. Nishihara. Rtvs-3: Real-time binocular stereo and optical flow measurement system. system description manuscript. Technical report, Teleos, Palo Alto, CA, July 1990.

[25] M. Okutomi and T. Kanade. A multiple-baseline stereo. In *Proceedings of the Conference on International Conference on Computer Vision and Pattern Recognition*, pages 63–69, Lahaina, Hawai, June 1991. IEEE.

[26] W. H. Press, B. P. Flannery, S.A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C*. Cambridge University Press, 1988.

[27] L. Robert and O.D. Faugeras. Relative 3d positioning and 3d convex hull computation from a weakly calibrated stereo pair. *Image and Vision Computing*, 13(3):189–197, 1995.

[28] A. Shashua. Projective structure from two uncalibrated images: Structure from motion and recognition. Technical Report A.I. Memo No. 1363, MIT, September 1992.

[29] T. Viéville, C. Zeller, and L. Robert. Recovering motion and structure from a set of planar patches in an uncalibrated image sequence. In *Proceedings of ICPR94*, Jerusalem, Israel, Oct 1994.

[30] Zhengyou Zhang, Rachid Deriche, Olivier Faugeras, and Quang-Tuan Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Journal*, 1994. to appear. Also INRIA Research Report No.2273, May 1994.

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 46 avenue Félix Viallet, 38031 GRENOBLE Cedex 1
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399