Inner Camera Invariants and Their Applications

Michael Werman[1] Maolin Qiu[1] Subhashis Banerjee[2] Sumantra Dutta Roy[2]

[1]Institute of Computer Science
Hebrew University, 91904 Jerusalem, Israel
Email: {werman,qiuml}@cs.huji.ac.il

[2]Department of Computer Science and Engineering
Indian Institute of Technology, New Delhi 110016, India
Email: {suban,sumantra}@cse.iitd.ernet.in

Abstract

In this paper we present some results on carrying out geometric computer vision tasks ignoring the (possibly changing) internal parameters of the camera. We assume absence of camera skew and derive new invariants which are independent of the internal parameters of the camera. These invariants can be used to address several vision problems:

1) Unlike previous methods of camera pose determination, we localize the camera with respect to a landmark frame without the assumption that the internal parameters are known, and, consequently, camera calibration is not required. Indices computed only from images can be used for robot navigation.

2) Conversely, when the camera positions are known, we show that is is possible to compute the Euclidean structure from multiple views even with varying and unknown internal camera parameters. In general the solutions are non-linear, though in some special cases we propose linear methods.

3) For the problem of simultaneous computation of Euclidean pose and structure using uncalibrated cameras, few of the existing approaches to camera self-calibration and 3-D reconstruction - neither the stratified approach nor the method using Kruppa's equations, can be effectively used in the case of varying and unknown internal parameters. We show that Euclidean pose and 3-D structure can be simultaneously computed directly using these inner camera invariants.

For each of the cases mentioned above, we derive the necessary conditions under which a finite number of (or unique) solutions can be found. Our experiments demonstrate the applicability of the methods.

Index Terms: Image invariants, Euclidean 3-D reconstruction, camera pose/localization, uncalibrated images.

1 Introduction

A commonly used basic perspective projection model of a pin-hole camera [6], describing the relationship between a 3-D world point $\mathbf{M}$ and its corresponding image point $\mathbf{m}$ is given as follows:

$$\lambda \mathbf{m} = \mathbf{P}\mathbf{M} = \mathbf{A}\left[\mathbf{R} \mid \mathbf{t}\right]\mathbf{M} \tag{1}$$

in which $\mathbf{R}$ (3×3) and $\mathbf{t}$ (3×1) are the rotation and translation aligning the world coordinate system with the camera coordinate system (the *External Parameters* of the camera), and $\mathbf{A}$ is the matrix of the *Internal Parameters* of the camera, which takes the form of

$$\mathbf{A} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

where f_x and f_y are the effective focal lengths in the x and y directions and (u_0, v_0) is the principal point. (A more general model assumes the term A_{12} in $\mathbf{A}$ to represent a *camera skew* term. However, such a term may often be considered negligible [6, 21].) The external parameters of a camera have six degrees of freedom, whereas the internal parameters have four [6].

In this paper, we use this model to derive new constraints which are invariant to the internal parameters of the camera. We show that these new constraints can be used for several visually guided tasks such as pose estimation, robot navigation, and computation of Euclidean structure even when the internal parameters of the camera are allowed to vary, as may be the case with an auto-focus and auto-aperture camera. Our approach is closely related to conventional camera calibration [27, 24, 4, 6] and self-calibration [15, 1, 10, 31, 17, 2, 13, 3].

Given the correspondence between 3-D control points (landmarks) and the images of these control points, the task of camera localization is to determine the translation and orientation of the camera. This problem is important for visually-guided robot navigation and hand-eye calibration, by which the accumulative positioning error read from the odometer can be corrected. Camera localization, or pose determination, is closely related to conventional camera calibration.

In conventional camera calibration, given a standard regularly-patterned object, one computes both the internal and the external camera parameters from the corresponding constraints between the 3-D control points and their images. In the absence of skew, five control points are enough to solve for all the parameters non-linearly from the basic perspective projection Equation (Equation 1). In the case of redundant control points, for example $n \geq 6$, linear methods can be used to tackle this problem [4, 24].

The problem of pose determination can be viewed as a special case of conventional camera calibration, with the implicit assumption that the internals of the camera are known. Therefore, they share a common theoretical basis and basic constraints. The orientation and translation of the camera relative to the landmark have six degrees of freedom. Thus, the minimal number of control points for pose estimation is three [8]. In this case, the approach to the solution is non-linear. Quan and Lan[23] presented a linear method of pose determination using redundant control points. They take advantage of the special form of the 4-degree polynomial equation deduced from each triplet of the control points, so that the method is *quasi*-linear.

Most of the approaches to pose-determination assume that the internal parameters of the camera are fixed and known. In this paper we address the cases under which the internal parameters of the camera can be changed, whether accidentally or voluntarily. We show that it is theoretically possible to use the proposed invariants for robot navigation even when the 3-D coordinates of the control points on the landmarks are not explicitly given. In addition, we show that our new invariants can be used to compute the Euclidean structure of the landmark points with or without knowledge of the camera positions. In each of these cases we derive the necessary conditions regarding the number of views and points required.

We first briefly review the existing methods for Euclidean reconstruction using the techniques of camera (self) calibration and bring out the fundamental differences between these methods and our approach in Section 2. In Section 3, we show how to derive the constraints which are independent of the internal parameters. Two different methods are given. In Section 4, we demonstrate how to use these constraints for camera/robot localization and Section 5 describes robot navigation using arbitrary landmarks. Section 6 gives the applications of these invariants in pose and structure. We present experimental results both on synthetic and real image sequences. In Section 8, we discuss issues relating to the robustness and stability of inner camera invariants. The last section is our concluding remarks.

2 Pin-hole camera model and Euclidean reconstruction

Recently, camera (self) calibration has attracted a lot of attention as a popular method for upgrading a projective reconstruction to Euclidean. In this section, we show the common thread between all existing methods of camera self-calibration, by providing a unified description. While these methods attempt to actually find the internal parameters of the camera, we investigate cases where the internals of the camera are allowed to vary, either intentionally or accidentally.

The classical pin-hole camera model is often assumed when the camera is used as a measuring device in vision tasks. A 3-D point not on the focal plane is denoted as $\mathbf{M} = (X, Y, Z, W)^T$ in the 3-D world coordinate system. For points not on the plane at infinity we can set $W = 1$. If $\mathbf{m} = (x, y, 1)^T$ is the corresponding image point, then the imaging process can be described by Equation 1.

Conventional camera calibration methods assume that the correspondences between the world points $\mathbf{M}$ and the image points $\mathbf{m}$ are given via the imaging of a known calibration object and estimate the parameters of $\mathbf{A}$, $\mathbf{R}$ and $\mathbf{t}$ either linearly or non-linearly by considering a sufficient number of points. See, for example, [27, 24, 4, 6].

Many recent methods of carrying out Euclidean measurements using computer vision have attempted to get rid of the known calibration object required for obtaining the camera internals using camera self-calibration. Maybank and Faugeras [15] first introduced a method of self-calibration using Kruppa's equation based on the absolute conic on the plane at infinity [15]. Faugeras also proposed the projective reconstruction technique of two uncalibrated views [7] and the projective-affine-metric stratification paradigm

[5]. The problem of camera self-calibration is regarded as a problem of refinement from the projective stratum, or affine stratum to the metric or Euclidean stratum, by eliminating the ambiguity [1, 10, 31, 17, 2, 13, 3]. In order to make the refinement process work, different kinds of 3-D information have to be used, and some restrictions need to be imposed on the motion of cameras.

These approaches to self-calibration fall roughly into two categories: stratified approaches and direct approaches. In the stratified approaches, the camera is first calibrated in a projective sense, then the plane at infinity must be identified so that the induced homography H_∞ can be computed from which the internals of the camera are obtained using matrix manipulations such as the singular value decomposition [1, 10, 31, 17, 2, 13, 3, 18]. Although some authors addressed the case of varying internal parameters of the camera[19, 20, 12, 21], often only the focal length is allowed to vary. The key idea of these methods is to undo the changes caused by the changing internal parameters. The use of the *absolute quadric* for auto-calibration of the camera falls into this category[26, 21], because the invariance of the absolute quadric to similarity transformations gives the constraint of the projective matrices and the internals of the camera, and we can regard this as a refinement from projective to metric. Actually this constraint is obtained by elimination of the rotation matrix from the basic perspective projective equations.

The direct approaches are based on solving Kruppa's equations for the internal parameters of the camera, which also fail to handle the case of varying internal parameters.

Theoretically, different approaches to camera self-calibration share common basic constraints. We first briefly describe the different methods in a unified framework. Both the stratified approach and the direct approach can be deduced from the basic constraint of Equation 1 by eliminating different entities. We then point out the fundamental difference between our method and the other methods.

Since the choice of the world coordinate system is arbitrary, we can express the imaging process for the first frame as:

$$\lambda \mathbf{m} = \mathbf{P}\mathbf{M} = \mathbf{A}\left[\mathbf{I} \mid \mathbf{0}\right]\mathbf{M} \tag{3}$$

in which $\mathbf{I}$ is the 3×3 identity matrix. This puts the origin of the world coordinate system on the *unknown* but *fixed* projection center of the first camera. Here, we represent a 3-D world point as $\mathbf{M} = (X, Y, Z, W)^T$ and a 2-D image point as $\mathbf{m} = (x, y, 1)^T$. For the second view, we write:

$$\lambda' \mathbf{m}' = \mathbf{P}'\mathbf{M} = \mathbf{A}'\left[\mathbf{R} \mid \mathbf{t}\right]\mathbf{M}, \tag{4}$$

where $\mathbf{m}' = (x', y', 1)^T$, $\mathbf{A}'$ the internal parameter matrix for the second view, and $\mathbf{R}$ and $\mathbf{t}$ the rotation and translation between the two camera frames. We use the notation $[\mathbf{t}]_\times$ to denote the antisymmetric matrix of $\mathbf{t}$. $[\mathbf{t}]_\times$ has two independent parameters, if $\mathbf{t}$ is defined up to a scale.

Now consider the relationship of the two views. The image of a 3-D point in the second view is related to the image of the same 3-D point in the first view by the following relationship:

$$\lambda' \mathbf{m}' = \lambda \mathbf{A}'\mathbf{R}\mathbf{A}^{-1}\mathbf{m} + W\mathbf{A}'\mathbf{t}. \tag{5}$$

In 3-D space, if the points at infinity are considered, i.e., $W = 0$, Equation 5 can be rewritten as:

$$\lambda' \mathbf{m}' = \lambda \mathbf{A}'\mathbf{R}\mathbf{A}^{-1}\mathbf{m} = \lambda \mathbf{H}_\infty \mathbf{m}, \tag{6}$$

where $\mathbf{H}_\infty = \mathbf{A}'\mathbf{R}\mathbf{A}^{-1}$ is the homography induced by *the plane at infinity*, which depends only on the rotation between the two camera frames and the internal parameters. The homography induced by the plane at infinity $\mathbf{H}_\infty$ describes the corresponding relationship of two images of a point at infinity and is independent of the translation of camera frames. There are methods of self-calibration for computing the internal parameters based on finding $\mathbf{H}_\infty$ and then deriving the matrix of internal parameters from it [1, 10, 31, 17, 2, 13, 3], which is also regarded as an important stage in the stratification paradigm of camera calibration and 3-D reconstruction[5]. In order to identify $\mathbf{H}_\infty$, either constrained motions must be imposed, or *a priori* information must be given.

Consider the image of the optical center of the first frame on the second image plane,

$$\lambda' \mathbf{m}' = W\mathbf{A}'\mathbf{t}, \tag{7}$$

$\mathbf{e}' \equiv \mathbf{A}'\mathbf{t}$ is the *epipole* on the second image plane. It is easy to see that the epipole on the first image plane is $\mathbf{e} \equiv -\mathbf{A}\mathbf{R}^{-\mathbf{1}}\mathbf{t}$. We rewrite Equation 5 as follows

$$\lambda' \mathbf{m}' = \lambda \mathbf{H}_\infty \mathbf{m} + W\mathbf{e}', \tag{8}$$

which means that in the second image $\mathbf{m}'$ is on a line passing through $\mathbf{H}_\infty \mathbf{m}$ and $\mathbf{e}'$. This relationship can be formalized as

$$\mathbf{m}'\cdot(\mathbf{e}'\times(\mathbf{H}_\infty\mathbf{m})) = \mathbf{m}'^T[\mathbf{e}']_\times\mathbf{H}_\infty\mathbf{m} = 0. \qquad (9)$$

The Fundamental Matrix [6] between the two views is given as

$$\mathbf{F} = [\mathbf{e}']_\times\mathbf{H}_\infty = [\mathbf{e}']_\times\mathbf{A}'\mathbf{R}\mathbf{A}^{-1} \qquad (10)$$

and is equivalent to the following equation by some algebraic manipulations [30]

$$\mathbf{F} = det(\mathbf{A}')\mathbf{A}'^{-1^T}[\mathbf{t}]_\times\mathbf{R}\mathbf{A}^{-1}. \qquad (11)$$

For a moving camera, in the case of $\mathbf{A} = \mathbf{A}'$, we can eliminate the rotation matrix $\mathbf{R}$ from Equation 10, which means that the following relationship is independent of the rotation between the two camera frames

$$\mathbf{F}\mathbf{A}\mathbf{A}^T\mathbf{F}^T = [\mathbf{e}']_\times\mathbf{A}\mathbf{A}^T[\mathbf{e}']_\times^T. \qquad (12)$$

Equation 12 is another form of Kruppa's equations[30], which is equivalent to the original form that was first introduced by Maybank and Faugeras[15] by using the absolute conic. Two important results are:

1. If the camera internals remain fixed then at least three views are needed for eliminating the projective ambiguity and finding a finite number of Euclidean solutions.

2. If the internal parameters of the camera change across views, in general at least 8 views are necessary in the absence of camera skew [12],[21],[11].

The auto-calibration method using the absolute quadric can also be considered as a method of eliminating the rotation matrix $\mathbf{R}$ directly from the perspective projective equation (Equation 1).

Thus we see that the different methods of self-calibration are all based on elimination of different entities from the basic equations. The constraint obtained by elimination is independent of what has been eliminated, therefore if we eliminate the internal parameters of the camera from the basic constraint, the resulting constraint is independent of the camera internals. In such a case the change of the internal parameters of the camera does not affect the constraint. This is the key point of the proposed method in this paper. Most approaches have attempted to explicitly find out the internals of the camera, and used them for various vision tasks.

In this paper, we attempt to investigate the process of working with functions which are invariant to the internal parameters of a camera, and using them for various applications.

3 Basic Constraints Independent of the Camera Internals

In order to make the constraints independent of the internal parameters of the camera, we eliminate the internal parameters from the basic constraint between the 3-D control points and the images [28]. In the following sections, we present two different methods to derive the above constraints.

3.1 Constraint Derivation: Method 1

Consider the case when the image projection $\mathbf{m} = (u, v, 1)^T$ of a world point $\mathbf{M} = (X, Y, Z, 1)^T$ is described by the perspective projection, which we write as

$$\lambda\mathbf{m} = \mathbf{A}[\mathbf{R} \mid \mathbf{t}]\mathbf{M}. \qquad (13)$$

Here $\mathbf{A}$ is the matrix of camera internals and $\mathbf{R}$ and $\mathbf{t}$ are the rotation and the translation between the world and the camera frames, and $[\mathbf{R} \mid \mathbf{t}]$ is the matrix of camera externals given by

$$[\mathbf{R} \mid \mathbf{t}] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} = \begin{bmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \\ \mathbf{r}_3 \end{bmatrix} \qquad (14)$$

Here, $r_{ij},\ i,j \in \{1,2,3\}$ are functions of R_x, R_y and R_z – the rotations about the X-, Y- and Z- axes, respectively.

Rewriting Equation 13, we have:

$$\left\{ \begin{array}{l} u = f_x\frac{\mathbf{r}_1\mathbf{M}}{\mathbf{r}_3\mathbf{M}} + u_0 \\ v = f_y\frac{\mathbf{r}_2\mathbf{M}}{\mathbf{r}_3\mathbf{M}} + v_0 \end{array} \right. \qquad (15)$$

Suppose we know three 3-D points, $\mathbf{M}_p = (X_p, Y_p, Z_p, 1)^T$, $p \in \{i, j, k\}$, and their images on the image plane, $\mathbf{m}_p = (u_p, v_p, 1)^T$, $p \in \{i, j, k\}$, then by eliminating the internals of the camera, we obtain

$$\left\{ \begin{array}{l} J_{ijk} = \frac{u_i - u_j}{u_i - u_k} = \frac{\frac{\mathbf{r}_1\mathbf{M}_i}{\mathbf{r}_3\mathbf{M}_i} - \frac{\mathbf{r}_1\mathbf{M}_j}{\mathbf{r}_3\mathbf{M}_j}}{\frac{\mathbf{r}_1\mathbf{M}_i}{\mathbf{r}_3\mathbf{M}_i} - \frac{\mathbf{r}_1\mathbf{M}_k}{\mathbf{r}_3\mathbf{M}_k}} \\ K_{ijk} = \frac{v_i - v_j}{v_i - v_k} = \frac{\frac{\mathbf{r}_2\mathbf{M}_i}{\mathbf{r}_3\mathbf{M}_i} - \frac{\mathbf{r}_2\mathbf{M}_j}{\mathbf{r}_3\mathbf{M}_j}}{\frac{\mathbf{r}_2\mathbf{M}_i}{\mathbf{r}_3\mathbf{M}_i} - \frac{\mathbf{r}_2\mathbf{M}_k}{\mathbf{r}_3\mathbf{M}_k}} \end{array} \right. , \qquad (16)$$

in which J_{ijk} and K_{ijk} are image measurements that are functions of $\mathbf{R}$, $\mathbf{t}$ and $\mathbf{M}_p$, $p \in \{i, j, k\}$, and are independent of the internals of the camera. Thus the above equations can be re-written as the following constraints

$$\left\{ \begin{array}{l} J_{ijk} = f_{ijk}(\mathbf{R}, \mathbf{t}, \mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_k) \\ K_{ijk} = g_{ijk}(\mathbf{R}, \mathbf{t}, \mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_k) \end{array} \right. \qquad (17)$$

The left hand sides of **constraint** Equation 17 represent image measurements based on three points which are invariant to the camera internals. The right hand sides are non-linear trigonometric expressions which are functions of only the camera externals and the structure (Euclidean coordinates) of the three points. We refer to J_{ijk} and K_{ijk} as **Inner Camera Invariants**. These parameters are indeed the invariants of the homography $\mathbf{A}$ which represents the change of the projective coordinate system between the camera and image.

3.2 Constraint Derivation: Method 2

Another method of eliminating the internals of the camera is as follows.

First let us rewrite Equation 13 as

$$\left\{ \begin{array}{l} \lambda u = \mathbf{r}_1\mathbf{M} f_x + \mathbf{r}_3\mathbf{M} u_0 \\ \lambda v = \mathbf{r}_2\mathbf{M} f_y + \mathbf{r}_3\mathbf{M} v_0 \\ \lambda = \mathbf{r}_3\mathbf{M} \end{array} \right. \tag{18}$$

By eliminating the scaling factor λ, we have

$$\left\{ \begin{array}{l} \left[\begin{array}{ccc} \mathbf{r}_1\mathbf{M} & \mathbf{r}_3\mathbf{M} & -\mathbf{r}_3\mathbf{M}u \end{array} \right] \left[\begin{array}{c} f_x \\ u_0 \\ 1 \end{array} \right] = 0 \\ \\ \left[\begin{array}{ccc} \mathbf{r}_2\mathbf{M} & \mathbf{r}_3\mathbf{M} & -\mathbf{r}_3\mathbf{M}v \end{array} \right] \left[\begin{array}{c} f_y \\ v_0 \\ 1 \end{array} \right] = 0 \end{array} \right. \tag{19}$$

Given three 3-D points, say $\mathbf{M}_p = (X_p, Y_p, Z_p, 1)^T$, $p \in \{i, j, k\}$, and their images on the image plane, $\mathbf{m}_p = (u_p, v_p, 1)^T$, $p \in \{i, j, k\}$, we have the following relationship

$$\left\{ \begin{array}{l} \mathbf{P}_{ijk} \left[\begin{array}{c} f_x \\ u_0 \\ 1 \end{array} \right] = \left[\begin{array}{ccc} \mathbf{r}_1\mathbf{M}_i & \mathbf{r}_3\mathbf{M}_i & -\mathbf{r}_3\mathbf{M}_i u_i \\ \mathbf{r}_1\mathbf{M}_j & \mathbf{r}_3\mathbf{M}_j & -\mathbf{r}_3\mathbf{M}_j u_j \\ \mathbf{r}_1\mathbf{M}_k & \mathbf{r}_3\mathbf{M}_k & -\mathbf{r}_3\mathbf{M}_k u_k \end{array} \right] \left[\begin{array}{c} f_x \\ u_0 \\ 1 \end{array} \right] = \mathbf{0} \\ \mathbf{Q}_{ijk} \left[\begin{array}{c} f_y \\ v_0 \\ 1 \end{array} \right] = \left[\begin{array}{ccc} \mathbf{r}_2\mathbf{M}_i & \mathbf{r}_3\mathbf{M}_i & -\mathbf{r}_3\mathbf{M}_i v_i \\ \mathbf{r}_2\mathbf{M}_j & \mathbf{r}_3\mathbf{M}_j & -\mathbf{r}_3\mathbf{M}_j v_j \\ \mathbf{r}_2\mathbf{M}_k & \mathbf{r}_3\mathbf{M}_k & -\mathbf{r}_3\mathbf{M}_k v_k \end{array} \right] \left[\begin{array}{c} f_y \\ v_0 \\ 1 \end{array} \right] = \mathbf{0} \end{array} \right. \tag{20}$$

Since they have non-trivial solutions, the determinant of 3×3 matrices $\mathbf{P}_{ijk}$ and $\mathbf{Q}_{ijk}$ must be 0, which gives 2 constraints independent of the internal parameters of the camera. It is easy to show that the constraints of Equation 20 are equivalent to those of Equation 17.

Given $n \geq 3$ control points in the 3-D world, we can get $2(n-2)$ *independent* constraints (of the total possible $2 \binom{n}{3}$) from one view of the 3-D scene. Suppose the number of views is N, then the total number of the *independent* constraints is $2N(n-2)$.

4 Camera/Robot Localization

In this section we address the problem of pose estimation from known landmarks using the invariants described above.

4.1 Pose estimation using Euclidean landmarks: general case

Suppose that we know the Euclidean coordinates $(X_i, Y_i, Z_i, 1)^T$ of 5 points in the world coordinate system. We wish to compute the the pose of the camera. Six independent invariant measurements give us six equations in terms of the six unknowns in $(\mathbf{R}, \mathbf{t})$. The six equations can be solved numerically (using constrained nonlinear optimization routines for systems of nonlinear equations, for example) for pose estimation using an uncalibrated camera and known landmarks. We use a method similar to bundle adjustment, using a Levenberg-Marquardt optimization scheme [22].

In the case of a constrained planar motion, $\mathbf{R}$ has only one degree of freedom and $\mathbf{t}$ has two degrees of freedom. The total number of unknowns thus is three, and four control points are sufficient for pose estimation.

It turns out that in some special cases it is possible to obtain closed-form or linear solutions to the pose estimation problem using the image invariants.

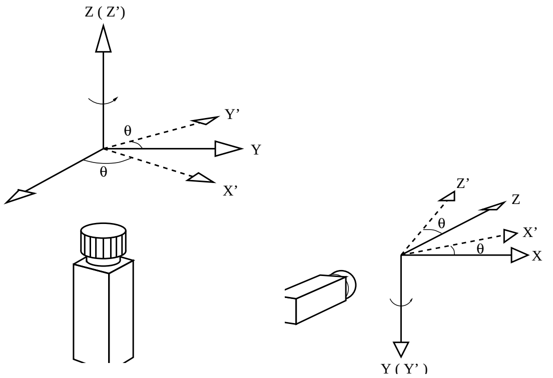
4.2 Special Case: Rotation only about Z-axis

Let $\mathbf{X}_w\mathbf{Y}_w\mathbf{Z}_w$ be the world coordinate system. Consider restricted motion with rotation θ only about the $\mathbf{Z}_w$-axis, shown in Figure 1(a). This would be the case of a robot moving on the ground with a camera looking vertically up at the landmarks on the ceiling to facilitate localization. The robot would like to know its pose with respect to the world coordinate system.

Suppose that there are three 3-D control points $\mathbf{M}_p$, $p \in \{i, j, k\}$ lying on the $\mathbf{X}_w\mathbf{Y}_w$ plane (ceiling) of the 3-D world coordinate system, where $\mathbf{M}_p$ $= (X_p, Y_p, 0, 1)^T$, and the corresponding image coordinates are $\mathbf{m}_p = (u_p, v_p, 1)^T$. The image projection is

$$\begin{array}{l} u_p = f_x \frac{X_p cos\theta - Y_p sin\theta + t_x}{t_z} + u_0 \\ v_p = f_y \frac{X_p sin\theta + Y_p cos\theta + t_y}{t_z} + v_0 \end{array} , p \in \{i, j, k\}. \tag{21}$$

By eliminating the internals of the camera from Equation 21, we obtain



(a) (b)

Figure 1: Illustrations of special cases: (a) constrained camera rotation only about the **Z**-axis; and (b) constrained camera rotation only about the **Y**-axis. The axes are transformed from OX, OY and OZ to OX', OY' and OZ'.

$$\begin{array}{l} \frac{u_i-u_j}{u_i-u_k} = \frac{(X_i-X_j)cos\theta-(Y_i-Y_j)sin\theta}{(X_i-X_k)cos\theta-(Y_i-Y_k)sin\theta} \\ \frac{v_i-v_j}{v_i-v_k} = \frac{(X_i-X_j)sin\theta+(Y_i-Y_j)cos\theta}{(X_i-X_k)sin\theta+(Y_i-Y_k)cos\theta} \end{array} . \qquad (22)$$

Let $J_{ijk} = (u_i - u_j)/(u_i - u_k)$ and $K_{ijk} = (v_i - v_j)/(v_i - v_k)$. Let $\mathcal{P}_{pq}$ stand for $\mathcal{P}_p - \mathcal{P}_q$. We obtain

$$tan\theta = \frac{X_{ij} - J_{ijk}X_{ik}}{Y_{ij} - J_{ijk}Y_{ik}} = \frac{Y_{ij} - K_{ijk}Y_{ik}}{K_{ijk}X_{ik} - X_{ij}}. \qquad (23)$$

We can easily compute θ (the rotation about **Z**) from Equation 23. Once θ is obtained, the translation vector **t** can be computed as follows. Suppose we have another 3-D control point $\mathbf{M}_l = (X_l, Y_l, Z_l, 1)^T$ which is not on the plane $\mathbf{X}_w\mathbf{Y}_w$(let us not stop at just painting the ceiling, but hang a few sticks as well). We compute J_{ijl} and K_{ijl} in the same way as in Equation 22. Then,

$$\begin{array}{l} \frac{t_z}{Z_l} = \frac{J_{ijl}(a_i+t_x)-(a_i-a_j)}{(a_i-a_j)-J_{ijl}(a_i-a_l)} \\ \frac{t_z}{Z_l} = \frac{K_{ijl}(b_i+t_y)-(b_i-b_j)}{(b_i-b_j)-K_{ijl}(b_i-b_l)} \end{array} \qquad (24)$$

where

$a_p = X_p cos\theta - Y_p sin\theta$, and
$b_p = X_p sin\theta + Y_p cos\theta$, for $p \in \{i, j, l\}$

By equating the two equations in Equation 24, and rearranging the terms,

$$A_{ijl}\ t_x + B_{ijl}\ t_y = C_{ijl} - D_{ijl} \qquad (25)$$

where

$$\begin{array}{l} A_{ijl} = \frac{J_{ijl}}{(a_i-a_j)-J_{ijl}(a_i-a_l)}; \\ B_{ijl} = \frac{-K_{ijl}}{(b_i-b_j)-K_{ijl}(b_i-b_l)}; \\ C_{ijl} = \frac{b_iK_{ijl}-(b_i-b_j)}{(b_i-b_j)-K_{ijl}(b_i-b_l)}; \\ D_{ijl} = \frac{a_iJ_{ijl}-(a_i-a_j)}{(a_i-a_j)-J_{ijl}(a_i-a_l)}. \end{array}$$

If we know more 3-D control points which are not on the $\mathbf{X}_w\mathbf{Y}_w$ plane, we can get more equations in the form of Equation 25. In such a case, a linear least squares technique can be used to solve for t_x and t_y. These values can be substituted in Equations 24 to compute t_z:

$$t_z = Z_l\ (D_{ijl} + A_{ijl}\ t_x) = Z_l\ (C_{ijl} - B_{ijl}\ t_y) \qquad (26)$$

Thus, in this special case, we obtain a closed-form solution to the localization problem, given three Euclidean landmarks on the $\mathbf{X}_w\mathbf{Y}_w$ plane and at least two landmarks off the $\mathbf{X}_w\mathbf{Y}_w$ plane. If we have more control points, we have a linear method to do the same, as shown above.

4.3 Special Case: Planar motion and rotation about Y-axis

Let $\mathbf{X}_w\mathbf{Y}_w\mathbf{Z}_w$ be the world coordinate system. Consider a restricted motion with rotation θ only about the $\mathbf{Y}_w$-axis, shown in Figure 1(b). This would be the case of a robot moving on the ground with a camera mounted horizontally (we look normally, at last). Consider three points in the world coordinate system $\mathbf{M}_p$, $p \in \{i, j, k\}$, where $\mathbf{M}_p = (X_p, Y_p, Z_p, 1)^T$ and the corresponding image coordinates are $\mathbf{m}_p = (u_p, v_p, 1)^T$.

By projecting the 3-D control points on to the image plane,

$$\begin{array}{l} u_p = f_x \frac{X_p cos\theta + Z_p sin\theta + t_x}{-X_p sin\theta + Z_p cos\theta + t_z} + u_0 \\ v_p = f_y \frac{Y_p + t_y}{-X_p sin\theta + Z_p cos\theta + t_z} + v_0 \end{array} , p \in \{i, j, k\}. \qquad (27)$$

Now, let the first two points i and j be on the $\mathbf{Y}_w$-axis and the third point k be on the $\mathbf{X}_w\mathbf{Y}_w$ plane. By eliminating the internal parameters of the camera from the above equation we obtain

$$\begin{array}{l} J_{ijk} = \frac{u_i-u_j}{u_i-u_k} = 0 \\ K_{ijk} = \frac{v_i-v_j}{v_i-v_k} = \frac{Y_i-Y_j}{(Y_i+t_y)-\frac{Y_k+t_y}{1-X_k\frac{sin\theta}{t_z}}} \end{array} . \qquad (28)$$

If we consider another point l on the $\mathbf{X}_w\mathbf{Y}_w$ plane, we obtain, similarly,

$$\begin{array}{l} J_{ijl} = \frac{u_i-u_j}{u_i-u_l} = 0 \\ K_{ijl} = \frac{v_i-v_j}{v_i-v_l} = \frac{Y_i-Y_j}{(Y_i+t_y)-\frac{Y_l+t_y}{1-X_l\frac{sin\theta}{t_z}}} \end{array} . \qquad (29)$$

Because J_{ijk} and J_{ijl} vanish, we can separate the terms of $sin\theta$ and t_z to obtain

$$\frac{t_z}{sin\theta} = \frac{X_k((1-K_{ijk})Y_i - Y_j - K_{ijk}t_y)}{Y_i - Y_j + K_{ijk}(Y_k - Y_i)}$$
$$= \frac{X_l((1-K_{ijl})Y_i - Y_j - K_{ijl}t_y)}{Y_i - Y_j + K_{ijl}(Y_l - Y_i)} = T_s. \qquad (30)$$

Therefore t_y can be easily found from Equation 30. In order to find θ, we have to know another control point lying on the plane $\mathbf{Y}_w\mathbf{Z}_w$, say m. Following exactly the derivation process of Equation 30, we can obtain a similar relationship as follows

$$\frac{t_z}{cos\theta} = -\frac{Z_m((1-K_{ijm})Y_i - Y_j - K_{ijm}t_y)}{Y_i - Y_j + K_{ijm}(Y_m - Y_i)} = T_c. \qquad (31)$$

Since we know t_y, T_s and T_c could be computed, then $tan\theta = T_c/T_s$ and $t_z = T_s sin\theta = T_c cos\theta$.

In Equation 27, J_{ijk} vanishes because both 3-D points i and j are on the $\mathbf{Y}_w$-axis. J_{ijk} would not be zero if we choose 3 points in general position. Let us select a point $\mathbf{M}_n = (X_n, Y_n, Z_n)^T$ in general position. Now that we know θ, t_y and t_z, solving for t_x is trivial. We use the two points $\mathbf{M}_l$ and $\mathbf{M}_k$ on the $\mathbf{X}_w\mathbf{Y}_w$ plane we had selected above.

$$t_x = \frac{b_n\ N_{kl} - J_{kln}\ a_l\ D_{kn}}{J_{kln}\ a_l\ (b_n - a_k) - b_n(a_l - a_k)}, \qquad (32)$$

where
$a_p = -X_p sin\theta + t_z$, $p \in \{k, l\}$;
$b_n = -X_n sin\theta + Z_n cos\theta + t_z$;
$N_{kl} = (a_l X_k - a_k X_l)cos\theta$;
$D_{nk} = (b_n X_k - a_k X_n)cos\theta - a_k Z_n sin\theta$.

Thus, in this special case, we obtain a closed form solution to the localization problems with specially located landmarks - two on the $\mathbf{Y}_w$-axis, two on the $\mathbf{X}_w\mathbf{Y}_w$ plane, one on the $\mathbf{Z}_w\mathbf{Y}_w$ plane, and one in general configuration.

5 Inner Camera Invariants and Robot Navigation

In this section, we discuss the use of arbitrary landmarks for robot navigation. By the term *arbitrary landmark*, we mean that the coordinates of the control points on the landmark are unknown or not initially given. We consider a robot equipped with a camera, whose internal parameters are not known. Further, the internals may vary intentionally (the camera may have the ability to zoom into an object of interest, for example), or unintentionally.

We first show that if the pose of the camera is known then it is possible to compute the Euclidean structure of the landmark points using the invariants.

5.1 3-D Euclidean reconstruction from known ego-motions (known pose)

Suppose we know the $\mathbf{R}$ and $\mathbf{t}$ (pose) at N locations of the camera. The task at hand is to find out the 3-D Euclidean structure of *unknown* landmark points. Such pose information can be obtained, for example, by solving the pose estimation problem for the new view using previously known landmarks. As before, we assume that the internals of the camera are unknown, and may vary. For Euclidean structure estimation of n 3-D control (landmark) points we have $3n$ unknowns. With N views we have $2N(n-2)$ independent constraint equations and the relationship $2N(n-2) \geq 3n$ must be satisfied. Thus with 3 views ($N = 3$) of at least 4 points ($n \geq 4$) we can compute the 3-D Euclidean structure provided the pose of each of the camera stations are known [1].

If the relationship between the image invariants of arbitrary landmarks and the pose of the camera/robot is established, image invariants from five generically-positioned points (with coordinates not given) are necessary for general robot navigation, and this relationship needs only to established once. In the special cases, the number of points is less. In what follows we show how a new landmark set can be established in the environment and used for subsequent navigation.

We consider two cases.

1. First, we consider the case of a robot not knowing the Euclidean structure of points of interest on a landmark. Additionally, it does not know its pose at a particular position (it knows its pose at all other positions). It wishes to interpolate the camera's $\mathbf{R}$ and $\mathbf{t}$ corresponding to this position.

2. Next, we consider the case when the robot knows its pose at different positions (from accurate odometric information, for example), However, it wishes to interpolate image measurement at a particular location.

These two cases may be applied to situations of *virtual navigation*, or *walkthroughs* - A remote user may wish

[1] Taking $N = 2$ yields $n \geq 8$. For two views and varying camera internal parameters, we have 14 parameters. $\mathbf{R}$ and $\mathbf{t}$ describe the rigid body transformaiton between the two camera stations – this accounts for 6 parameters. 4 parameters each come from the each of the camera internal parameter matrices corresponding to these two positions. Thus, 7 points completely describe the geometric relationship between the two cameras, and the 8-th point does not give any extra information.

to know his pose with respect to a particular view. For the second case, a virtual navigation system may not store image information for a large number of viewpoints - it can interpolate information from different viewpoints to generate a view.

5.2 Interpolation of camera pose: 3-D Interpolation

Suppose we know $\mathbf{R}$ and $\mathbf{t}$ of $N-1$ locations of the camera, with the pose of the N-th view unknown. We wish to interpolate the pose of the camera corresponding the N-th view. We also assume that the $3n$ structure parameters of the n landmark points are initially unknown. Then we have a total of $3n+6$ unknown parameters in terms of the Euclidean structure of the n points and the $\mathbf{R}$ and $\mathbf{t}$ of the N-th station. The relationship $2N(n-2) \geq (3n+6)$ gives us that with 4 views of at least 5 points we can compute all the unknowns using a two stage process. First we can compute the Euclidean structure of the landmark points using the first 3 views (see Section 5.1) and then use the structure information to solve for the pose estimation (localization) problem for the last view. The newly-selected 5 points could be used as the new landmark. If the 3-D structure is not of interest then the coordinates of the 3-D control points can be eliminated from the constraints and we arrive at the equations for interpolation of camera pose. N and n must be chosen to satisfy $N \geq 4$ and $n \geq 5$.

5.3 Interpolation of image measurements: 2-D Interpolation

Suppose we know the $\mathbf{R}$ and $\mathbf{t}$ for all stations but do not know the image measurements J_{ijk}'s and K_{ijk}'s corresponding to one of the stations. Thus, we want to interpolate image measurements at one of the camera positions. Now we have the relationship $2N(n-2) \geq 3n+2(n-2)$. Thus we can interpolate the image measurements in any of the following situations - i) $N=4$ and $n \geq 4$ ii) $N=6$ and $n \geq 3$. As in the previous case we can eliminate the coordinates of the 3-D control points if the Euclidean structure is not of interest.

6 Simultaneous computation of Pose and 3-D Euclidean Structure

The proposed image invariants can also be used for simultaneous computation of pose and structure even when the internal of the camera are varying and unknown.

6.1 Motion and Structure from planar motion and rotation about Z

Consider planar motion on a plane parallel to the $\mathbf{X}_w\mathbf{Y}_w$ plane and rotation only about $\mathbf{Z}_w$, shown in Figure 1(a). For this special case, we wish to determine both the motion (rotation about the $\mathbf{Z}$-axis), as well as the 3-D Euclidean structure of the points of interest. We denote the coordinates of 4 points on the $\mathbf{X}_w\mathbf{Y}_w$ plane (the ceiling) as $\mathbf{M}_q$ $(q \in \{i,j,k,l\})$. By rearranging the terms in Equation 22 and letting $\mathcal{P}_{pq}$ stand for $\mathcal{P}_p - \mathcal{P}_q$, we obtain

$$\begin{cases} X_{ij} - Y_{ij} tan\theta - J_{ijp} X_{ip} + J_{ijp} tan\theta Y_{ip} = 0 \\ X_{ij} tan\theta + Y_{ij} - K_{ijp} tan\theta X_{ip} - K_{ijp} Y_{ip} = 0 \end{cases}, \tag{33}$$

where $p \in \{k,l\}$. We then have $2(4-2)=4$ equations as follows

$$\mathbf{M}(\theta)\mathbf{X} = \mathbf{0} \tag{34}$$

where $\mathbf{M}$ is a 4×6 matrix whose entries depend on $\tan(\theta)$ and the image invariants,

$$\mathbf{X} = \begin{bmatrix} X_{ij} & Y_{ij} & X_{ik} & Y_{ik} & X_{il} & Y_{il} \end{bmatrix}^T. \tag{35}$$

If the camera undergoes rotation with another angle θ', and we know the matching points of the first view, then we have a relationship similar to Equation 34

$$\mathbf{N}(\theta')\mathbf{X} = \mathbf{0} \tag{36}$$

Putting Equation 34 and Equation 36 together we obtain

$$\mathbf{C}(\theta,\theta')\mathbf{X} = \mathbf{0}, \mathbf{C}(\theta,\theta') = \begin{bmatrix} \mathbf{M}(\theta) \\ \mathbf{N}(\theta') \end{bmatrix} \tag{37}$$

Equation 37 is a set of 8 non-linear equations in $\mathbf{X}$, $tan(\theta)$ and $tan(\theta')$. Because $\mathbf{X}$ is a nonzero vector, we have $det(\mathbf{C}^T\mathbf{C}) = 0$, which is a polynomial equation in $tan(\theta)$ and $tan(\theta')$. Given a reference view with $\theta' = \theta_0$, e.g., $\theta' = 0$, which we can take as the initial orientation of the 3-D world coordinate system, we can easily get any other rotation angle θ by solving the polynomial equation. If the initial orientation is not set, the rotation angles can only be determined up to a shift.

Moreover, $\mathbf{X}$ is the eigenvector of the smallest eigenvalue of $\mathbf{C}^T\mathbf{C}$ and the Euclidean coordinates can be determined up to a scale assuming the first point to be the origin. The scale can be set by knowing any single length.

6.2 Simultaneous computation of pose and structure: general case

Suppose that the 3-D world coordinate system coincides with one of the camera frames, and we do not know the position of the 3-D control points. The problem of localization then becomes the problem of structure and motion without calibration. In the most general case, the relationship of constraints is $2N(n-2) \geq (3n + 6(N-1))$. The term $N-1$ in the relationship means that the $\mathbf{R}$ and $\mathbf{t}$ are unknown for $N-1$ frames of the camera. The first can be fixed arbitrarily to $\mathbf{A}\ [\mathbf{I} \mid \mathbf{0}]$. This amounts to setting the world origin to the unknown but fixed camera centre of the first camera. When the number of views and the number of matching points satisfy the above relationship, the problem of pose and structure can be solved up to scale. The condition for the general case is $N \geq 8$ and $n \geq 6$.

In each of the above cases it is possible to use a larger number of control points to robustly estimate the parameters using a suitable non-linear optimization technique.

Note that 8 views are necessary when camera internal parameters are unknown and may vary [12], [21, 17, 29, 1, 10]. These computation may not be feasible for certain *critical motions* [14, 25] and *critical configurations* [9, 16], though this has been proved typically for the case of internal camera parameters which do not vary.

7 Experiments

We carried out several experiments on both synthetic and real image sequences and compared the results obtained using our methods with standard calibration. The advantage of having calibration as a benchmark is two-fold:

1. Calibration provides the best bundle-adjustment over both internal and external parameters (unlike self-calibration, which does not consider external parameters at all).
2. Calibration methods such as that of Tsai [27] for example, consider radial distortion as well – going beyond the conventional camera pin-hole model.

We assume that all correspondence information is known (hand picked).

We have experimented with a calibration grid (Figure 3), and a model house. Figure 2 shows four views of a model house, which we used for our experiments on the general cases of pose estimation, structure estimation, and 3-D and 2-D interpolation. Across our experiments, the calibration and 3-D reconstruction results from classical camera calibration are also listed for comparison, all distances are in *mm*, all angles are in degrees.

For all our experiments we used the constrained non-linear optimization routine of MATLAB(`constr/fmincon`, using a Levenberg-Marquardt optimization scheme). Inner camera invariants are computed from 2-D image data:

$$J_{ijk} = \frac{u_i - u_j}{u_i - u_k};\ K_{ijk} = \frac{v_i - v_j}{v_i - v_k}$$

We compute the error between the left- and right-hand sides of Equation 17 as below:

$$\begin{cases} E_J = J_{ijk} - f_{ijk}(\mathbf{R}, \mathbf{t}, \mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_k) \\ E_k = K_{ijk} - g_{ijk}(\mathbf{R}, \mathbf{t}, \mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_k) \end{cases} \tag{38}$$

We consider the function F to optimize, as the sum of squares of E_J and E_K for each of the $n-2$ set of points:

$$F = \sum_{\mathit{All\ sets\ of\ n-2\ points}} ({E_J}^2 + {E_K}^2) \tag{39}$$

Such a function is optimized with respect to suitable constraints e.g., the allowable error (using Equation 38: $|E_J| \leq lim_J$, $|E_K| \leq lim_K$), solution search neighborhoods, etc.

7.1 General Pose Estimation

We carried out experiments on general pose estimation (Section 4.1) on two landmarks, which we call Landmark I (a calibration grid, shown in Figure 3) and Landmark II (a model house, shown in Figure 2). In Table 1 we present some sample results for two positions of the camera, using Landmark I (the calibration grid) as the object, using 5 points.

In Table 2, we show some sample results for Landmark II (the model house) using 20 points at two camera stations. As seen in Tables 1 and 2, the pose estimation results using inner camera invariants are quite close to the corresponding calibration values in most cases, the maximum discrepancy being less than 2 degrees for angles, and 50 *mm* for distances.

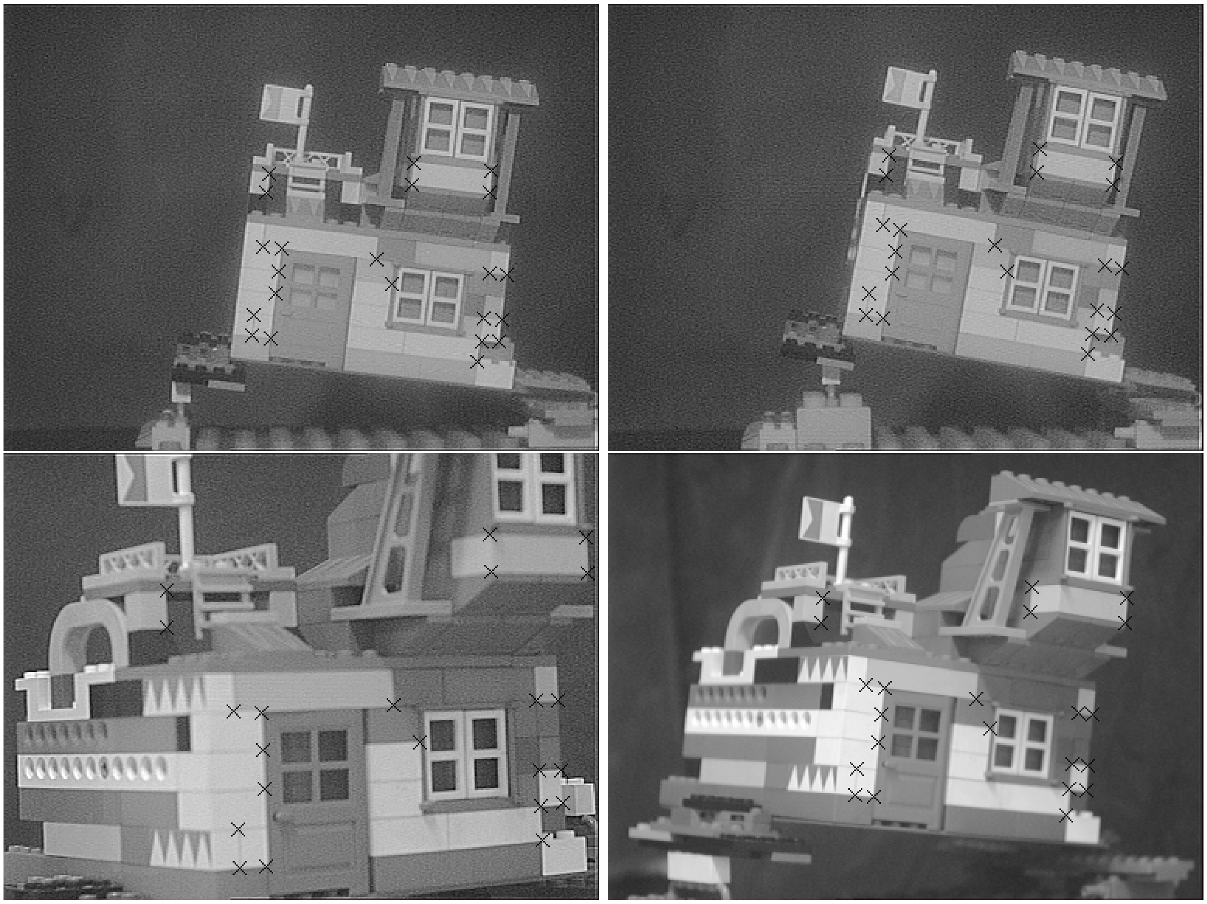


Figure 2: Images of Landmark II (a model house) used for pose estimation (general case), structure estimation (general case), and 3-D and 2-D interpolation. Image points used in our experiments are marked with crosses.

7.2 Pose Estimation: Special Case 1

For the first special case (Section 4.2), we used two camera stations, with two sets of 5 points each for each camera station.

Figure 3 shows images from three such camera stations (constrained rotation about the **Z**-axis only: We kept our camera horizontal and rotated our calibration object).

Table 3 summarizes the results. The pose estimated using inner camera invariants is off the calibration value by a maximum of about 3 degrees and 60 *mm*, for the angles and distances, respectively.

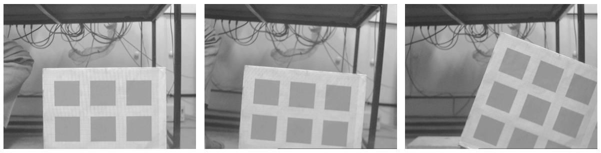


Figure 3: Images of Landmark I (the calibration grid) taken at three camera stations, which we have used for 1) pose estimation: special case 1 (constrained camera rotation about the **Z**-axis only: Section 4.2, results in Section 7.2); 2) computation of both pose and structure: planar camera motion (Section 6, results in Section 7.7)

7.3 Pose Estimation: Special Case 2

We took two sets of six points each, for two camera stations(shown in Figure 4).

In Table 4, we show results for special case 2 of pose estimation (Section 4.3). As is seen from the table, the results are off from the calibration values by a maximum of about 9 degrees and 225 *mm*.

	Pose 1		Pose 2	
	Calib.	*ICI*	*Calib.*	*ICI*
R_x	0.376	1.906	0.003	-0.252
R_y	-4.795	-6.518	-28.727	-28.449
R_z	0.649	0.509	0.316	0.454
t_x	-694.802	-728.196	-434.520	-434.542
t_y	24.189	17.798	18.000	17.971
t_z	1041.215	1087.956	822.988	822.989

Table 1: Pose estimation experiments with Landmark I (the calibration grid): Some sample results. We compare the poses (**R** and **t**) computed using inner camera invariants (*'ICI'*) and standard calibration (*'Calib.'*) at two camera stations denoted by Pose 1 and Pose 2, respectively. All angles are in degrees, and all distances in *mm*.

	Pose 1		Pose 2	
	Calib.	*ICI*	*Calib.*	*ICI*
R_x	-0.641	-0.052	0.311	-0.882
R_y	-2.787	-2.830	-32.467	-27.754
R_z	6.753	6.704	-1.797	-2.681
t_x	-519.353	-516.074	-461.840	-450.918
t_y	-62.001	-63.691	487.470	-491.092
t_z	855.277	833.717	183.907	168.206

Table 2: Pose estimation experiments (general case) with Landmark II (the model house): Some sample results at two camera stations (Pose 1 and Pose 2) using inner camera invariants (*'ICI'*), and comparison with calibration results (*'Calib.'*). All angles are in degrees, and all distances in *mm*.

7.4 Robot Navigation: 3-D Interpolation

Figure 5 shows the corresponding images taken from the camera stations around Landmark I (the calibration grid).

Images in Figure 5 were used for the 3-D and 2-D interpolation experiments with Landmark I (the calibration grid).

Table 5 shows some results for Section 5.2, for $N = 4$ and $n = 5$. We have some extremely accurate results here, with the maximum error being about a *mm* for the structure, a fraction of a degree for the angles, and about 60 *mm* for the distances.

Table 6 shows some results on Landmark II (the model house, Figure 2), for the same values of N and n. The estimated structure of the corresponding points are shown in the second part of Table 10. The discrepancy between the values estimated using inner camera invariants and calibration data, is a maximum of about 8 *mm* for the 3-D Euclidean structure, 6 degrees for the angles, and 20 *mm* for the distances.

<table>
<tr><td colspan="5"><b>Camera Station I: Calibration Results:</b>
t_x = -634.112, t_y = 16.029, t_z = 1057.223
R_x = 1.080, R_y = 0.837, R_z = -0.317</td></tr>
<tr><td colspan="5"><b>Results using Inner Camera Invariants:</b></td></tr>
<tr><td><i>Points</i></td><td>R_z</td><td>t_x</td><td>t_y</td><td>t_z</td></tr>
<tr><td>5,10,15,40,48</td><td>-0.578</td><td>-654.555</td><td>15.526</td><td>934.162</td></tr>
<tr><td>8,12,19,39,44</td><td>-0.524</td><td>-648.851</td><td>39.923</td><td>1063.521</td></tr>
<tr><td colspan="5"><b>Camera Station II: Calibration Results:</b>
t_x = -633.105, t_y = -35.049, t_z = 1053.015
R_x = 0.831, R_y = 0.672, R_z = 4.701</td></tr>
<tr><td colspan="5"><b>Results using Inner Camera Invariants:</b></td></tr>
<tr><td><i>Points</i></td><td>R_z</td><td>t_x</td><td>t_y</td><td>t_z</td></tr>
<tr><td>6,13,18,25,37</td><td>4.975</td><td>-634.835</td><td>-29.289</td><td>990.971</td></tr>
<tr><td>4,15,20,27,48</td><td>2.047</td><td>-575.648</td><td>-37.7681</td><td>1062.035</td></tr>
</table>

Table 3: Pose estimation results, Special Case 1 (Section 4.2: the case of constrained camera rotation about the **Z**-axis alone). We give the estimated values of R_z and **t** for two camera stations for different choices of 5 points. We also indicate the results from standard calibration for comparison. All angles are in degrees, and all distances are in *mm*.

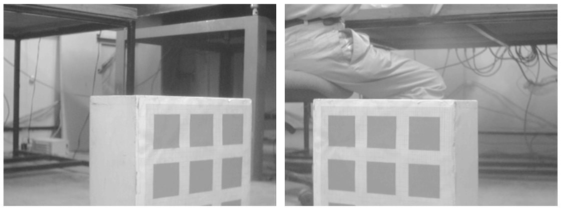


Figure 4: The sequence of images of Landmark I (the calibration grid) used for special case 2 of pose estimation (Section 4.3): constrained rotation about the **Y**-axis only (Results in Section 7.3)

7.5 Robot Navigation: 2-D Interpolation

In Table 7, we show results for 2-D interpolation (Section 5.3) for $N = 4$ and $n = 5$, for Landmark I (the calibration grid). In this case, the results are extremely accurate. The 3-D Euclidean structure is within a *mm* of calibration data, while the discrepancy values of the inner camera invariants are less than 1 in each case.

Table 8 shows some results for the same values of N and n for Landmark II (the model house). The re-

Camera Station I: Calibration Results: t_x = 21.948, t_y=31.988, t_z=1034.709 R_x = -0.931, R_y = -39.155, R_z = 1.314				
Results using Inner Camera Invariants:				
Points	R_y	t_x	t_y	t_z
7,19,6,21,31,42	-30.843	138.861	14.758	883.234
7,13,4,22,37,47	-33.573	133.137	20.163	957.999
Camera Station II: Calibration Results: t_x = -210.449, t_y = 18.132, t_z = 1195.758 R_x = 0.345, R_y = 23.262, R_z = 0.504				
Results using Inner Camera Invariants:				
Points	R_y	t_x	t_y	t_z
7,19,8,22,37,44	28.266	-99.024	47.190	1420.395
13,19,15,20,43,47	27.275	-153.946	45.356	1311.451

Table 4: Pose estimation results, Special Case 2(Section 4.3: the case of constrained camera rotation about the **Y**-axis alone). We give the estimated values of R_z and **t** for two camera stations for different choices of 6 points. We also indicate the results from standard calibration for comparison. All angles are in degrees, and all distances are in *mm*.

sults of the structure computation for the corresponding points are shown in the second part of Table 10. In this case again, the estimated 3-D Euclidean structure is off by at most about 4 *mm*. The maximum discrepancy between the estimated values of the inner camera invariants and the values from calibration data, is a small fraction.

7.6 Structure from known Ego-Motions (known pose)

Figure 5 shows images taken from the 4 viewpoints around Landmark I (the calibration grid), which are used for experiments in this case. In Table 9, we show some results for recovering the 3-D structure of points from known ego-motions (Section 5.1). We considered $N = 3$ and $n = 4$.

Table 10, we show some results for Landmark II (the model house). We considered two cases here, i) $N = 3$ and $n = 3$, and ii) $N = 4$, $n = 5$. Results for these are shown in the two parts of the table. In both these cases, we observe that the estimated values of the 3-D structure correspond very closely with the corresponding values using calibration data – the maximum discrepancy is about 7 *mm*.

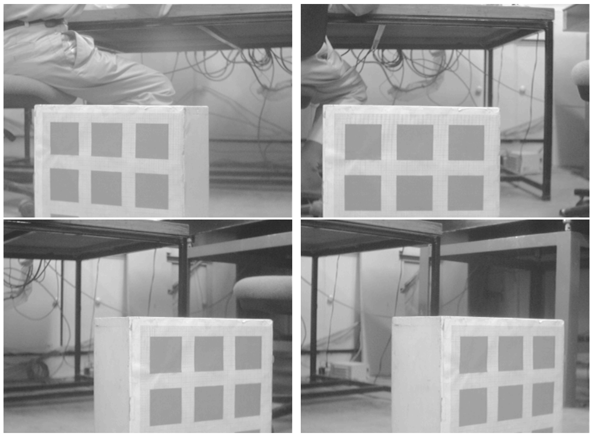


Figure 5: The 4 viewpoints used for the 3-D and 2-D interpolation experiments (Sections 7.4 and 7.5), respectively) with Landmark I (the calibration object).

7.7 Pose and Structure: Planar Camera Motion

The special case of planar camera motion, computation of pose and structure (Section 6.1) were tested on the image sequences shown in Figure 3, we show some results in Table 11. Again, there is a very good correspondence between parameters estimated using inner camera invariants, and those computed from calibration data. The angle is off by less than a degree for the angles, and a small fraction, for the 3-D structure ratio.

Here, the first image of Figure 3 was taken as the reference camera station, while the other two are the stations numbered I and II.

7.8 Simultaneous Pose and Structure

We obtained a sequence of synthetic images using a theoretical pin-hole camera model. The 3-D scene consists of a set of 3-D points randomly generated. We consider 8 images of the same 3-D scene from different points of view, with all the internal parameters of the camera varying across views. In this case, because we know the actual values of the 3-D points in the first camera frame, and the relative rotation and the translation of the camera with respect to the first frame, say $\mathbf{X}_{true}$, the initial guess $\mathbf{X}_0$ for optimization is given by $N(\mathbf{X}_{true}, \delta^2)$, $\delta^2 = 0.10|(\mathbf{X}_{true})|$. We got exact results, though each time the starting point was different.

Table 12, Table 13 and Table 14 show the results of recovering simultaneously both pose and structure (Section 6.2) from a synthetic sequence, for $N = 8$ and $n = 6$.

Calibration Data			*ICI Results*		
X	*Y*	*Z*	*X*	*Y*	*Z*
500	0	200	500.089	-0.012	200.039
700	70	0	701.062	70.362	-1.036
500	100	200	500.0873	99.849	200.039
670	0	200	670.978	0.0388	199.963
700	0	200	700.663	0.020	199.989

Calibration Results		*ICI Results*	
R	**T**	**R**	**T**
-0.083	676.421	-0.083	606.801
0.046	14.203	0.0172	3.559
0.079	-1076.164	0.072	-1061.471

Table 5: 3-D Interpolation results for Landmark I (the calibration grid): results with $N = 4$, $n = 5$. We show results for the structure computation using inner camera invariants (*'ICI Results'*), and well as the pose (**R** and **T** values) at the required camera station. All angles (for **R**) are in degrees, and all coordinates (X, Y and Z) and distances (for **T**) are in *mm*.

Calibration Results		*ICI Results*	
R	**T**	**R**	**T**
-4.829	-425.031	-8.864	-429.434
-35.014	-64.124	-29.633	-45.181
7.938	343.221	10.537	-337.075

Table 6: 3-D Interpolation results for the model house: we consider $N = 4$, $n = 5$. The results of 3-D structure computations using inner camera invariants (*'ICI Results'*) are shown in the second part of Table 10. All angles (for **R**) are in degrees, and all distances (for **T**) are in mm.

The real image sequence on which we tested our method of simultaneous pose and structure is given in Fig. 6.

Table 15, Table 16 and Table 17, show the results of recovering simultaneously both pose and structure (Section 6.2) on the real image sequence, $N = 8$ and $n = 6$. Even for the real image sequence, the estimated parameters are in agreement with those computed from calibration values, the values being less than a fraction, in each case. Here, we mention that if we star far away from the solution, the computations may converge to a non-optimal local minimum.

Calibration Data			*ICI Results*		
X	*Y*	*Z*	*X*	*Y*	*Z*
500	0	200	500.089	-0.012	200.039
700	70	0	701.062	70.362	-1.036
500	100	200	500.087	99.849	200.039
670	0	200	670.979	0.039	199.963
700	0	200	700.663	0.020	199.989

	Calibration Data	*ICI Results*
J_{123}	29.001	28.984
K_{123}	0.664	0.676
J_{124}	1.315	1.309
K_{124}	5.187	4.988
J_{125}	1.114	1.109
K_{125}	4.536	4.363

Table 7: 2-D Interpolation results for Landmark I (the calibration grid): We show a comparison of the results with inner camera invariants (*'ICI Results'*) with calibration results We have considered $N = 4$ and $n = 5$. All coordinates (X, Y and Z) are in *mm*.

	Calibration Data	*ICI Results*
J_{123}	-0.029	-0.023
K_{123}	2.000	2.051
J_{124}	-0.017	-0.014
K_{124}	-0.214	-0.210
J_{125}	0.051	0.041
K_{125}	-0.305	-0.300

Table 8: 2-D Interpolation results for Landmark II (the model house): We compare results with inner camera invariants (*'ICI Results'*) with calibration data. We consider $N = 4$, $n = 5$. The results of 3-D structure computations for the points used are shown in the second part of Table 10.

8 Robustness and Stability of Inner Camera Invariants

In this section, we present an experimental analysis of the effect of noise on the computations based on inner camera invariants.

Let us recall that the computation of inner camera invariants J_{ijk} and K_{ijk} involves the ratio of differences of pixel coordinates (details in Section 3, Equation 16):

$$J_{ijk} = \frac{u_i - u_j}{u_i - u_k};\ K_{ijk} = \frac{v_i - v_j}{v_i - v_k}$$

It is obvious that to reduce the effect of pixel noise on the computation of J_{ijk}'s and K_{ijk}'s, the triplets of points should be so chosen that the numerator and

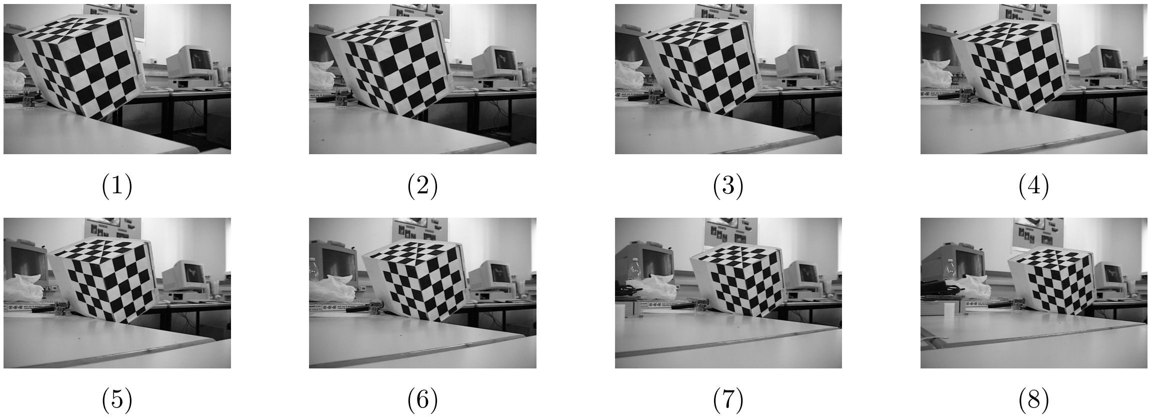

(1) (2) (3) (4)
(5) (6) (7) (8)

Figure 6: The sequence of 8 images for used in our experiments for simultaneous computation of pose and structure.

Calibration Data			ICI Results		
X	Y	Z	X	Y	Z
500	0	200	495.884	3.501	197.995
700	70	0	704.152	69.4679	-1.348
500	100	200	507.501	98.997	203.334
670	0	200	669.324	0.190	199.888

Table 9: Structure estimation from known ego-motions (Section 5.1): Some sample results for Landmark I (the calibration grid). We compare results using inner camera invariants (*'ICI Results'*) with calibration data. All coordinates (X, Y and Z) are in mm.

Calibration Data			ICI Results		
X	Y	Z	X	Y	Z
515.875	19.05	0.0	515.884	19.088	0.0532
515.875	28.575	0.0	515.871	28.584	-0.140
563.5	19.05	0.0	563.504	19.085	-0.005
563.5	-31.75	-31.75	563.489	-31.897	-31.662
515.875	19.05	0.0	515.971	11.889	3.267
515.875	28.575	0.0	515.591	20.120	-3.854
563.5	19.05	0.0	559.544	11.644	-3.282
563.5	-31.75	-31.75	560.507	-33.378	-28.832
507.938	-12.7	31.75	508.842	-17.897	34.472

Table 10: Structure estimation from known ego-motions (Section 5.1): Some sample results for Landmark II (the model house). We compare reconstruction results using inner camera invariants (*'ICI Results'*) with calibration data. All coordinates (X, Y and Z) are in mm.

denominator are of comparable order and neither is too small.

A example calibration data set has parameters: R_x = -0.375^o, R_y = -4.795^o, R_z = 0.649^o, t_x = -694.802 mm, t_y = -24.189 mm, and t_z = 1041.215 mm. We use typical triplets of 3-D world points such that their 2-D image coordinates fulfil the above criterion: the images of the second and third points are not too close to the image of the first. Let us consider cross-sections of the signed difference between the left- and right-hand sides of Equation 17, while varying the external parameters. We refer to this signed difference as E_J and E_K for inner camera invariants J_{ijk} and K_{ijk}, respectively (Equation 38). We show the zero line, the intersection of which with the curve(s) would indicate the position of the minimum of the corresponding component of the optimization function (${E_J}^2$ and ${E_K}^2$). In these examples, we consider E_J alone; the treatment of E_K follows on exactly the same lines. For all these graphs, we plot the varying quantity (R_y, t_x or t_z) on the horizontal axis, and E_J on the vertical axis. Figures 7(a), (b) and (c) show the variation in the cross-section of E_J with the variation in one external parameter (R_y, t_x and t_z, respectively). In each of these cases, the slope of the E_J cross-section with respect to the variation in the external parameter, is such that we can easily locate its zero-crossing using a non-linear optimization scheme (Levenberg-Marquardt, for example). Figures 7(a) and (b) consider the above calibration set. Figure 7(a) shows the variation of the E_J plot with R_y varying between -10^o and 10^o. In Figure 7(b) considers t_x varying between -800 mm and -600 mm. The corresponding set for Figure 7(c) has the following parameters: R_x = 0.003^o, R_y = -28.727^o, R_z = 0.316^o, t_x = -434.520 mm, t_y = 18.000 mm, and t_z = 822.988 mm. (Let us refer to the above two calibration sets as

Reference Camera Station: Calibration Results: R_x = 1.080, R_y = 0.837, R_z = -0.317		
Camera Station I: Calibration Results: R_x = 0.831, R_y = 0.672, R_z = 4.701		
Results with Inner Camera Invariants: Observation Set A: Points 9, 17, 19, 22		
Parameter	*Calib.*	*ICI*
R_z	4.701	5.005
X_{12} / Y_{14}	1.0	0.956
Y_{12} / Y_{14}	0.3	0.302
X_{13} / Y_{14}	-1.0	-0.942
Y_{13} / Y_{14}	1.0	0.998
X_{14} / Y_{14}	0.7	0.683
Camera Station II: Calibration Data: R_x = 1.146, R_y = 0.624, R_z = 21.009		
Results with Inner Camera Invariants: Observation Set B: Points 11, 14, 18, 20		
Parameter	*Calib.*	*ICI*
R_z	21.009	22.191
X_{12} / Y_{14}	-1.3	-1.239
Y_{12} / Y_{14}	0.3	0.309
X_{13} / Y_{14}	0.7	0.679
Y_{13} / Y_{14}	0.3	0.304
X_{14} / Y_{14}	-1.3	-1.234

Table 11: Simultaneous computation of pose and structure from planar motion, and constrained rotation about the **Z**-axis (Section 6.1): We have compared results using inner camera invariants (*'ICI'*) with calibration results (*'Calib.'*). All angles are in degrees.

Calibration Set I and Calibraiton Set II, respectively.) Here, we vary t_y from 800 mm to 900 mm.

Let us now consider the effect of pixel noise on computations involving inner camera invariants. We illustrate the fact that such pixel errors do not result in unbounded errors in the constraint equations for the optimization process (used in pose computation or structure estimation, for example). We have successively experimented with injecting an increasing amount of zero mean Gaussian noise in the pixel measurements (u_i, u_j, u_k etc.). For the zero-noise case, the zero-crossing of the E_J surface cross-section indicates the solution, the calibration value of the external parameter. For the cases of noise, we plot histograms of the distribution of zero-crossings of the E_J surface cross-section. For each of these cases, we have taken the number of histogram bins as 10. We have experimented with pixel noise of the order of $\sigma = 1$ and 2.5 pixels.

For the graphs of Figure 8, we consider the variation of R_y from -10^o to 10^o (the calibration value is $R_y =$

	3-D point	X/Z	Y/Z	Z/Z
Calib. Values	1	-0.1222	0.0850	1.0000
	2	-0.1676	-0.0147	1.0000
	3	0.0816	-0.0269	1.0000
	4	0.0730	0.1745	1.0000
	5	0.1918	0.0402	1.0000
	6	-0.0324	-0.0213	1.0000
ICI Results	1	-0.1222	0.0850	1.0000
	2	-0.1676	-0.0147	1.0000
	3	0.0816	-0.0269	1.0000
	4	0.0730	0.1745	1.0000
	5	0.1918	0.0402	1.0000
	6	-0.0324	-0.0213	1.0000

Table 12: Simultaneous pose and structure from the image invariants: Section 6 (We consider $N = 8$ and $n = 6$ for this example). Results for the 3-D points in the synthetic image sequence. We show a comparison of values obtained using inner camera invariants (*'ICI Results'*) with calibration values (*'Calib. Values'*)

	Rotation	a	b	c
Calib. Results	R_2	-0.0198	0.4184	0.0052
	R_3	0.6336	0.4153	0.1418
	R_4	-0.6775	0.4214	-0.1323
	R_5	-0.6266	-0.5241	0.2451
	R_6	0.6846	-0.5724	-0.1136
	R_7	0.6757	-0.4010	-0.0693
	R_8	0.6588	-0.0729	0.0156
ICI Results	R_2	-0.0198	0.4184	0.0052
	R_3	0.6336	0.4153	0.1418
	R_4	-0.6775	0.4214	-0.1323
	R_5	-0.6266	-0.5241	0.2451
	R_6	0.6846	-0.5724	-0.1136
	R_7	0.6757	-0.4010	-0.0693
	R_8	0.6588	-0.0729	0.0156

Table 13: Simultaneous pose and structure from the inner camera invariants (Section 6): Results for the synthetic sequence. (We have considered $N = 8$ and $n = 6$ for our experiments). We show results for the rotations R_i, $i = 2, ..., 8$ relative to the first camera frame. The rotation matrices are expressed in the from of the *Rodriques matrix* [24, 4]. a, b and c are 3 independent parameters of the Rodriques matrix. We compare values obtained using inner camera invariants (*'ICI Results'*) with calibration results (*'Calib. Results'*).

	Trans.	t_x/t_z	t_y/t_z	t_z/t_z
Calib. Results	$\mathbf{t_2}$	-1.9292	-0.0592	1.0000
	$\mathbf{t_3}$	-0.7082	0.9681	1.0000
	$\mathbf{t_4}$	-0.5570	-0.6902	1.0000
	$\mathbf{t_5}$	0.9171	-0.5141	1.0000
	$\mathbf{t_6}$	0.8079	0.7223	1.0000
	$\mathbf{t_7}$	0.7202	1.0682	1.0000
	$\mathbf{t_8}$	0.0614	1.5600	1.0000
ICI Results	$\mathbf{t_2}$	-1.9292	-0.0592	1.0000
	$\mathbf{t_3}$	-0.7082	0.9681	1.0000
	$\mathbf{t_4}$	-0.5570	-0.6902	1.0000
	$\mathbf{t_5}$	0.9171	-0.5141	1.0000
	$\mathbf{t_6}$	0.8079	0.7223	1.0000
	$\mathbf{t_7}$	0.7202	1.0682	1.0000
	$\mathbf{t_8}$	0.0614	1.5600	1.0000

Table 14: Simultaneous pose and structure from the inner camera invariants: Section 6. (We consider the case of $N = 8$ and $n = 6$). Results for the translational vectors $\mathbf{t_i}$, $i = 2, ..., 8$, with respect to the first camera frame, for the synthetic image sequence. We compare results using inner camera invariants (*'ICI Results'*) with calibration results (*'Calib. Results'*).

	3-D point	X/Z	Y/Z	Z/Z
Calib. Values	1	-0.1614	-0.1342	1.0000
	2	-0.0941	-0.1648	1.0000
	3	-0.0316	-0.1932	1.0000
	4	0.0264	-0.2196	1.0000
	5	0.0806	-0.2443	1.0000
	6	-0.1362	-0.0597	1.0000
ICI Results	1	-0.1593	-0.1327	1.0000
	2	-0.0938	-0.1647	1.0000
	3	-0.0315	-0.1926	1.0000
	4	0.0271	-0.2255	1.0000
	5	0.0795	-0.2350	1.0000
	6	-0.1354	-0.0591	1.0000

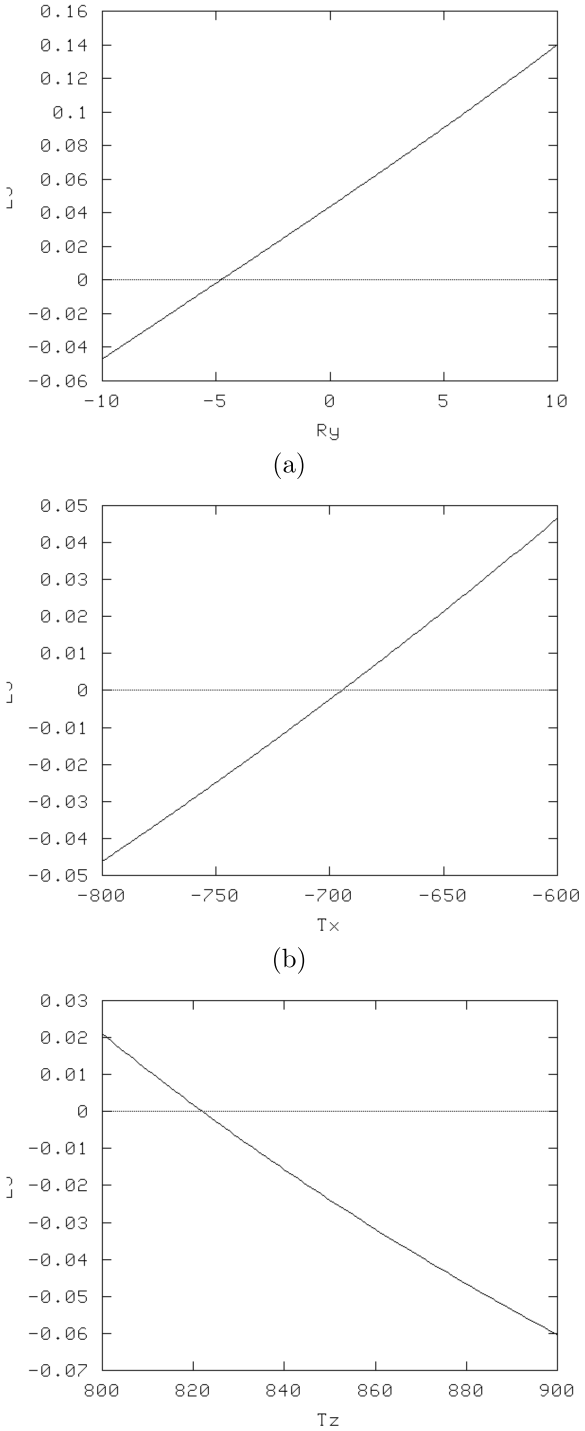
Table 15: Simultaneous pose and structure from the inner camera invariants (Section 6). This example considers $N = 8$ and $n = 6$) on the real image sequence: results for the 3-D points. We compare results using inner camera invariants (*'ICI Results'*) with calibration values (*'Calib. Values'*).

	Rotation	a	b	c
Calib. Results	R_2	-0.0107	-0.0018	0.0017
	R_3	-0.0186	-0.0220	0.0008
	R_4	0.0171	-0.0260	0.0038
	R_5	-0.0110	-0.0667	0.0067
	R_6	-0.0138	-0.0747	0.0050
	R_7	0.0216	-0.1278	0.0065
	R_8	0.0200	-0.1379	0.0155
ICI Results	R_2	-0.0107	-0.0018	0.0017
	R_3	-0.0184	-0.0222	0.0008
	R_4	0.0174	-0.0260	0.0038
	R_5	-0.0110	-0.0677	0.0065
	R_6	-0.0137	-0.0749	0.0050
	R_7	0.0216	-0.1284	0.0065
	R_8	0.0201	-0.1370	0.0159

Table 16: Simultaneous pose and structure from the inner camera invariants (Section 6). We consider $N = 8$ and $n = 6$ here, for the real image sequence. Results for the rotations R_i, $i = 2, ..., 8$ are relative to the first camera frame. We compare our results with inner camera invariants (*'ICI Results'*) with calibration results (*'Calib. Results'*). The rotation matrices are expressed in the from of the *Rodriques matrix* [24, 4]. a, b and c are 3 independent parameters of the Rodriques matrix

	Trans.	t_x/t_z	t_y/t_z	t_z/t_z
Calib. Results	$\mathbf{t_2}$	0.7895	-0.0642	1.0000
	$\mathbf{t_3}$	0.8717	0.0854	1.0000
	$\mathbf{t_4}$	0.8944	0.0410	1.0000
	$\mathbf{t_5}$	0.6587	0.0195	1.0000
	$\mathbf{t_6}$	0.6430	0.0395	1.0000
	$\mathbf{t_7}$	0.5757	-0.0076	1.0000
	$\mathbf{t_8}$	0.6144	0.0251	1.0000
	$\mathbf{t_2}$	0.7879	-0.0642	1.0000
ICI Results	$\mathbf{t_3}$	0.8853	0.0851	1.0000
	$\mathbf{t_4}$	0.8806	0.0403	1.0000
	$\mathbf{t_5}$	0.6693	0.0193	1.0000
	$\mathbf{t_6}$	0.6410	0.0392	1.0000
	$\mathbf{t_7}$	0.5658	-0.0077	1.0000
	$\mathbf{t_8}$	0.6089	0.0244	1.0000

Table 17: Simultaneous pose and structure from the inner camera invariants (Section 6) using $N = 8$ and $n = 6$, for the real image sequence: results for the translational vectors $\mathbf{t_i}$, $i = 2, ..., 8$ with respect to the first camera frame. We compare results using inner camera invariants (*'ICI Results'*) with calibration results (*'Calib. Results'*).

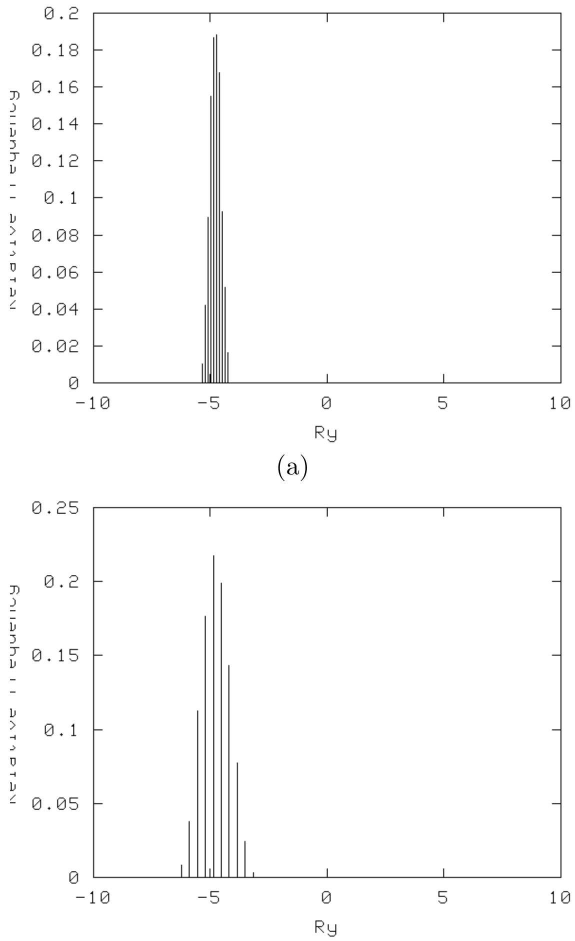


(a)

(b)

(c)

Figure 7: A plot of a cross-section of E_J for varying values of external parameters (a) R_y, (b) t_x, and (c) t_z: the zero-noise case.



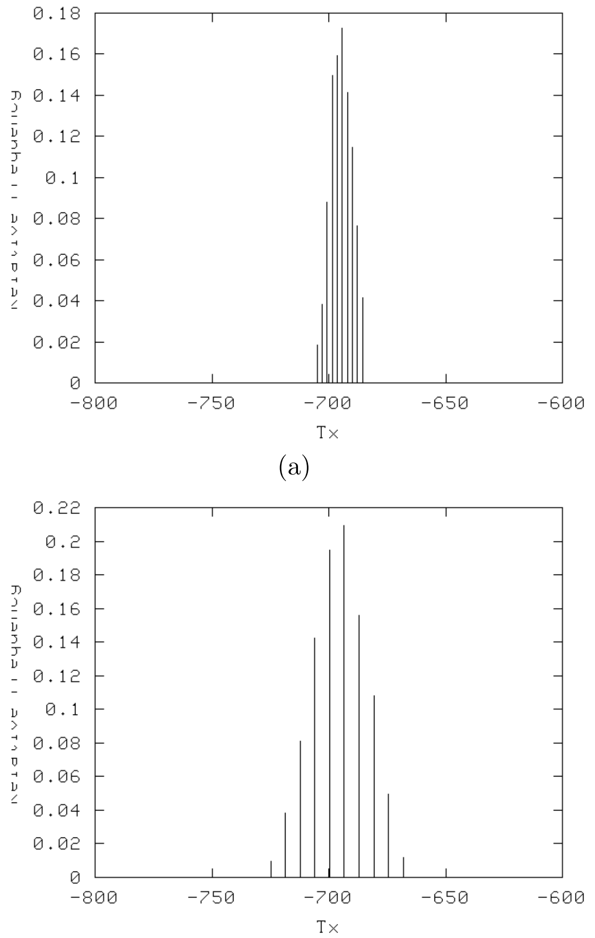
(a)

(b)

Figure 8: A plot of a 10-bin histogram of the distribution of the zero-crossings of the corresponding cross-section of the E_J surface. The varying external parameter is R_y. (a) and (b) consider the noise to be of the order of σ pixels, $\sigma = 1$ and 2.5, respectively.

-4.795^o) holding the other external parameters at their calibration values (Calibration Set I). We have experimented with zero mean Gaussian noise added to the image points with the standard deviation $\sigma = 1$ and 2.5 pixels. In Figures 8(a) and (b), we plot a distribution of the relative frequencies of occurrence of the zero-crossings of the E_J surface cross section for the case of noise: $\sigma = 1$ and 2.5, respectively. It is to be noted that due to the addition of noise, the value of R_y which satisfies the constraint equation does not shift appreciably. The range of these values varies from -5.425^o to -4.178^o for the $\sigma = 1$ case, to -6.404^o to -2.999^o for $\sigma = 2.5$. In both cases, the distribution peaks around the actual calibration value.

We now show results with varying the translation



(a)

(b)

Figure 9: A 10-bin histogram of the distribution of the zero-crossings of the E_J surface cross-section, corresponding to possible solutions, for varying values of external parameter t_x. $\sigma = 1$ and 2.5 in plots (a) and (b), respectively.

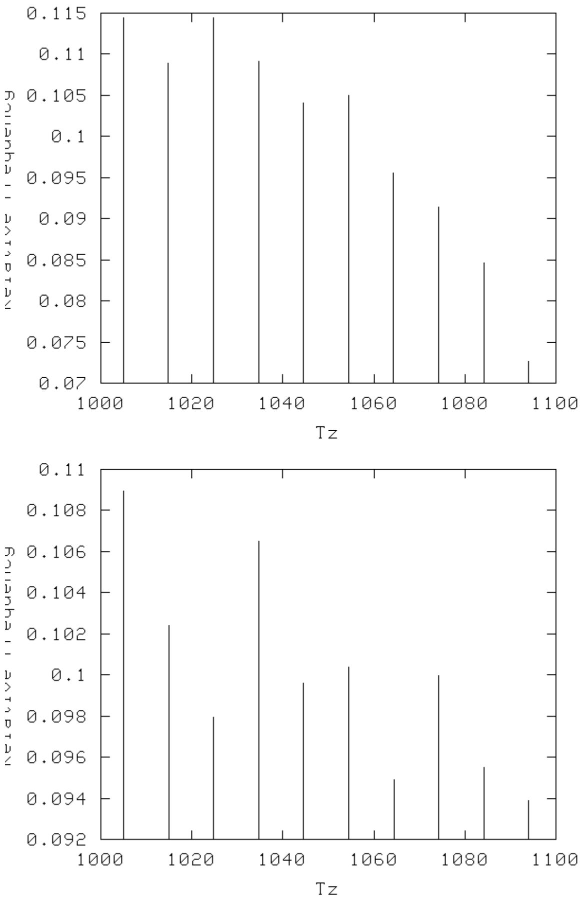


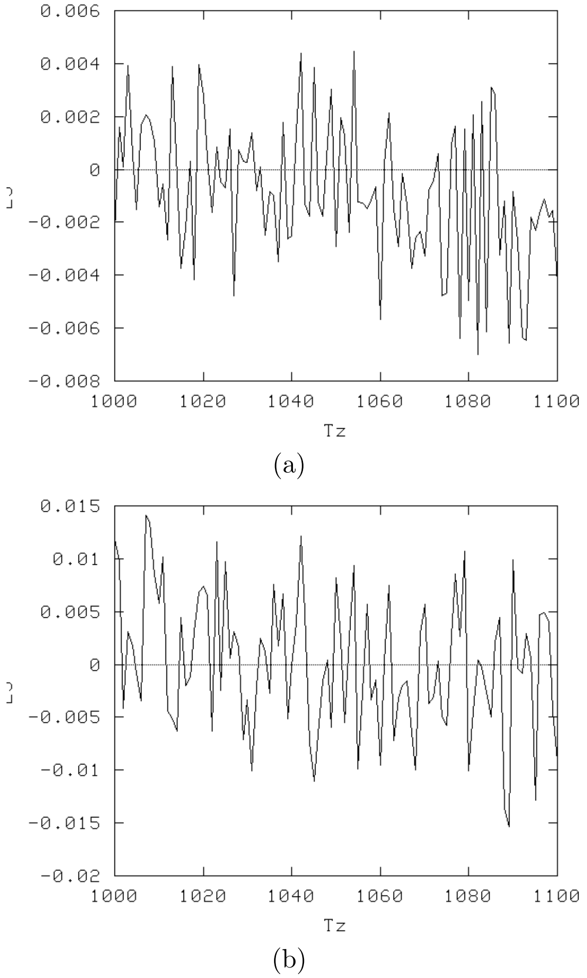
Figure 10: A plot of the distribution of the zero-crossings of the E_J surface cross-section, for noise of the order of $\sigma = 1$ and 2.5, respectively. The varying external parameter is t_z, and the parameters correspond to Calibration Set I. These plots illustrate the fact that for variation in t_z, good results with inner camera invariants are not guaranteed (details in text).

parameters. We vary t_x between -800 mm and -600 mm, and t_z between 1000 mm and 1100 mm, respectively (the calibration values are -694.802 mm and 1041.215 mm, respectively: Calibration Set I). Figures 9(a) and (b) show a distribution of the zero-crossings of the E_J surface cross-section for the case of noise: $\sigma = 1$ and 2.5, respectively. We see similar noise resilience as in the case of R_y.

We now show an interesting case when the use of inner camera invariants may fail. This is for a variation in t_z for world points. The above calibration set (Calibration Set I) presents such an example. For the plots in Figure 10, we have again considered a 10-bin histogram of the distribution of the E_J zero-crossings, for varying values of t_z. The two plots correspond to noise of the order of 1 and 2.5 pixels, respectively. For these plots, we observe that we do not have a well defined solution of the constraint equation.

In Figure 11, we show the cross-section of the E_J surface, corresponding to the distribution plots in Figure 10. Note that for a small translation of 3-D points along the view direction the corresponding 2-D image points may not change appreciably. As a consequence the variation of t_z does not change the value of J_{ijk} and E_J significantly and the slope of the constraint equation in this case is one order of magnitude less than the other cases (R_y and t_x). Hence the effect of pixel noise is more pronounced in this case.

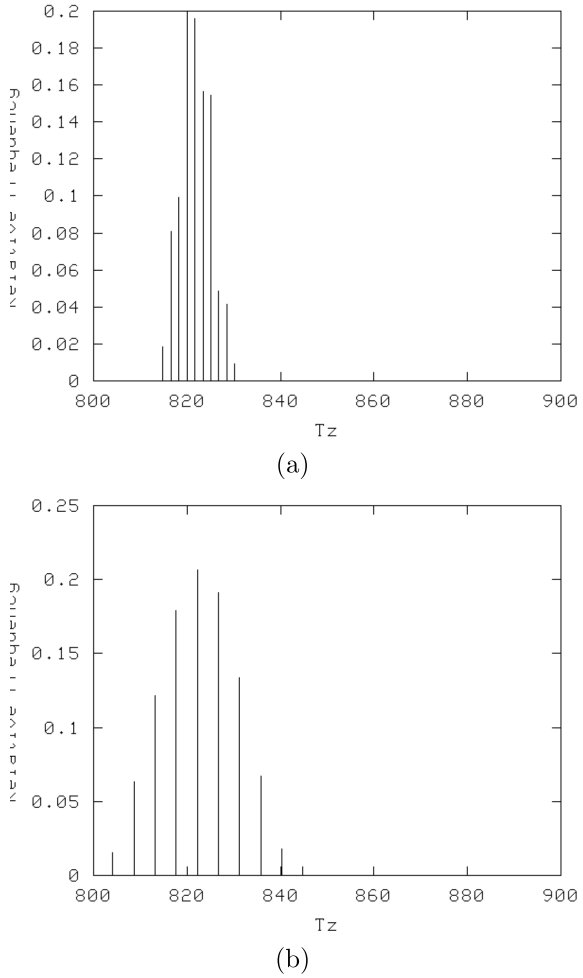
We show results for a set of calibration data where we get good results for t_z (Calibration Set II). Fig-



(a)

(b)

Figure 11: A plot of a cross-section of the E_J surface, for varying values of t_z: the noise is of the order of σ pixels, $\sigma = 1$ and 2.5 in plots (a) and (b), respectively. These correspond to the distribution plots in Figure 10.



(a)

(b)

Figure 12: A plot of a 10-bin histogram of the distribution of the zero-crossings of the corresponding cross-section of the E_J surface, for varying external parameter t_z (Calibration Set II). $\sigma = 1$ and 2.5 in plots (a) and (b), respectively.

ure 12 shows the distributions of the E_J zero-crossings corresponding to the two cases of noise, respectively. In general, compuation of t_z with inner camera invariants may be problematic for situations with a small field of view.

In this section, we have examined the stability of inner camera invariants. First, we have shown above that the variation in the E_J values due to pixel noise does not cause the solution of the constraint equation for E_J to vary appreciably. We consider noise of the order of $\sigma = 1$ and 2.5 pixels, respectively. These graphs help us to find out the experimental upper bound on how far away from the calibration values, an optimization routine can converge, in the presence of pixel noise.

9 Conclusion

The constraints that we proposed in this paper are independent of the internal parameters of the camera. Changes of the internals of the camera does not affect the results of localization, motion or structure. The proposed approaches based on these constraints are totally different from the previous methods and give promising results. We show experimental results of the robustness and stability of inner camera invariants. We believe that these invariants are suitable for a wide range of computer vision applications.

References

[1] M. Armstrong, A. Zisserman, and R. Hartley. Self-Calibration from Image Triplets. In *Proc. European Conf. on Computer Vision*, pages 3–16, 1996.

[2] P. Beardsley and A. Zisserman. Affine Calibration of Mobile Vehicles. In *Proc. Europe-China Workshop on Geometrical Modeling and Invariants for Computer Vision*, pages 214–221, 1995.

[3] P.A. Beardsley, A. Zisserman, and D.W. Murray. Sequential Updating of Projective and Affine Structure from Motion. *Intl. J. Computer Vision*, 23(3):235–260, 1997.

[4] W. Faig. Calibration of Close Range Photogrammetry Systems: Mathematical Formulation. *Photogrammetric Engineering and Remote Sensing*, 41(12):1479–1486, 1975.

[5] O. Faugeras. Stratification of 3-D Vision: Projective, Affine, and Metric Representations. *J. of Opt. Soc. of Am. Series A*, 12(3):465–484, March 1995.

[6] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint.* The MIT Press, 1996.

[7] O.D. Faugeras. What can be seen in Three Dimensions with an Uncalibrated Stereo Rig? In *Proc. European Conf. on Computer Vision*, pages 563–578, 1992.

[8] R. M. Haralick, C. N. Lee, K. Ottenberg, and M. Nölle. Analysis and Solutions of the Three Points Perspective Pose Estimation Problem. In *Proc. Computer Vision and Pattern Recognition*, pages 592–598, 1991.

[9] R. Hartley. Ambiguous Configurations for 3-View Projective Reconstruction. In *Proc. European Conf. on Computer Vision*, 2000.

[10] R. I. Hartley. Self-Calibration from Multiple Views with a Rotating Camera. In *Proc. European Conf. on Computer Vision*, 1994.

[11] A. Heyden and K. Åström. Flexible calibration: Minimal cases for auto-calibration. In *ICCV99*, pages 350–355, 1999.

[12] A. Heyden and K. Åström. Euclidean Reconstruction from Image Sequences with Varying and Unknown Focal Length and Principal Point. In *Proc. Computer Vision and Pattern Recognition*, 1997.

[13] R. Horaud and G. Csurka. Self-Calibration and Euclidean Reconstruction Using Motion of a Stereo Rig. In *Proc. Intl. Conf. Computer Vision*, pages 96–104, 1998.

[14] F. Kahl and B. Triggs. Critical Motions in Euclidean Structure from Motion. In *Proc. Computer Vision and Pattern Recognition*, pages II:366–372, 1999.

[15] S. Maybank and O. Faugeras. A Theory of Self-Calibration of a Moving Camera. *Intl. J. Computer Vision*, 8(2):123–151, 1992.

[16] S.J. Maybank and A. Shashua. Ambiguity in Reconstruction from Images of Six Points. In *Proc. Intl. Conf. Computer Vision*, pages 703–708, 1998.

[17] T. Moons, L. Van Gool, M. Proesmans, and E. Pauwels. Affine Reconstruction from Perspective Image Pairs With A Relative Object-Camera Translation in Between. *IEEE Trans. on Pattern Anal. and Machine Intell.*, 18(1):77–83, January 1996.

[18] M. Pollefeys and L. Van Gool. A Stratified Approach to Self-Calibration. In *Proc. Computer Vision and Pattern Recognition*, 1997.

[19] M. Pollefeys, L. Van Gool, and A. Oosterlinck. The Modulus Constraint: A New Constraint for Self-Calibration. In *Proc. Intl. Conf. Pattern Recognition*, pages 349–353, 1996.

[20] M. Pollefeys, L. Van Gool, and M. Proesmans. Euclidean 3-D Reconstruction from Image Sequences with Variable Focal Lengths. In *Proc. European Conf. on Computer Vision*, pages 31–42, 1996.

[21] M. Pollefeys, R. Koch, and L. Van Gool. Self-Calibration and Metric Reconstruction in Spite of Varying and Unknown Internal camera parameters. In *Proc. Intl. Conf. Computer Vision*, pages 90–95, 1998.

[22] W. H. Press, S. A. Teukolsky, W. T. Vettering, and B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1993.

[23] L. Quan and Z. Lan. Linear $N \geq 4$-Point Pose Determination. In *Proc. Intl. Conf. Computer Vision*, pages 778–783, 1998.

[24] C. C. Slama(Ed.). *Manual of Photogrammetry.* American Society of Photogrammetry and Remote Sensing, Falls Church, Virginia, USA, 1980.

[25] P. Sturm. Critical Motion Sequences for Monocular Self-Calibration and Uncalibrated Euclidean Reconstruction. In *Proc. Computer Vision and Pattern Recognition*, pages 1100–1105, 1997.

[26] B. Triggs. Autocalibration and the Absolute Quadric. In *Proc. Computer Vision and Pattern Recognition*, 1997.

[27] Roger Y. Tsai. A Versatile Camera Calibration Technique for High-Accuracy 3-D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses. *IEEE Journal of Robotics and Automation*, RA-3(4):323–344, August 1987.

[28] M. Werman, S. Banerjee, S. Dutta Roy, and M. Qiu. Robot Localization using Uncalibrated Camera Invariants. In *Proc. Computer Vision and Pattern Recognition*, pages II:353–359, 1999.

[29] C.S. Wiles and M. Brady. Ground Plane Motion Camera Models. In *Proc. European Conf. on Computer Vision*, pages II:238–247, 1996.

[30] C. Zeller and O. Faugeras. Camera Self-Calibration from Video Sequences: the Kruppa Equations Revisited. Technical report, Research Report No. 2793, INRIA, 1996.

[31] A. Zisserman, P. Beardsley, and I. Reid. Metric Calibration of a Stereo Rig. In *Proc. IEEE Workshop on Representation of Visual Scenes*, 1995.