Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Vision and Image Understanding 99 (2005) 151–174
Computer Vision and Image Understanding
www.elsevier.com/locate/cviu

Least-squares 3D reconstruction from one or more views and geometric clues ☆

Etienne Grossmann [a,*], José Santos-Victor [b,*]

[a] *UK Center for Visualization and Virtual Environments, 1 Quality St, Lexington, KY 40507, USA*
[b] *Instituto Superior Técnico, Instituto de Sistemas e Robótica, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal*

Received 24 January 2004; accepted 12 January 2005
Available online 19 February 2005

Abstract

We present a method to reconstruct from one or more images a scene that is rich in planes, alignments, symmetries, orthogonalities, and other forms of geometrical regularity. Given image points of interest and some geometric information, the method recovers least-squares estimates of the 3D points, camera position(s), orientation(s), and eventually calibration(s). Our contributions lie (i) in a novel way of exploiting some types of symmetry and of geometric regularity, (ii) in treating indifferently one or more images, (iii) in a geometric test that indicates whether the input data uniquely defines a reconstruction, and (iv) a parameterization method for collections of 3D points subject to geometric constraints. Moreover, the reconstruction algorithm lends itself to sensitivity analysis. The method is benchmarked on synthetic data and its effectiveness is shown on real-world data.

Keywords: Single-view 3D reconstruction; Unicity of reconstruction; Least-squares estimation; Performance analysis

☆ This work has been funded by FCT Programa Operacional Sociedade de Informação (POSI), within QCA III, by FCT PRAXIS XXI Grant BD/19594/99, by project CAVIAR: IST-2001-37540 and by the Center for Visualization and Virtual Environments.
* Corresponding authors.
E-mail addresses: etienne@cs.uky.edu (E. Grossmann), jasv@isr.ist.utl.pt (J. Santos-Victor).

1077-3142/$ - see front matter
doi:10.1016/j.cviu.2005.01.002

1. Introduction

This article addresses the problem of 3D reconstruction of scenes that display properties of planarity, orthogonality, parallelism, and symmetry, such as the man-made scenes in Fig. 1. This problem is of interest for archeology, urbanism, architecture, virtual reality, reverse engineering, etc.

Knowing geometric properties of the scene allows to perform 3D reconstruction in situations that general methods [31] cannot treat. For example, a reconstruction may be obtained from a single image [10,28,38,43] or from multiple views with too few point correspondence.

1.1. Current approaches

Methods for 3D reconstruction from image points and geometric clues can be roughly classified as "model-based" and "constraint-based." The former reconstruct assemblages of primitive shapes, such as parallelepipeds, prisms, cylinders, while the latter exploit geometric properties of the scene-planarity, orthogonality, etc.

In *model-based methods* [11,26,29], the scene is defined as in CAD [45] systems. For example, the buildings in Fig. 1 can be decomposed in parallelepipeds, prisms, and truncated pyramids. By fitting the model to image data, its dimensions, position, and orientation are determined. The fact that the scene must be decomposable in primitive shapes is the main limitation of model-based methods.

In *constraint-based methods*, this limitation does not exist, as any shape can be reconstructed as long as there are enough geometric properties to define a unique reconstruction. Geometric properties are either detected automatically [5,12,13,50] or, as in the present work, given by the user [1,2,10,38,43]. Some forms of symmetry have been exploited [24,37,49] and, in theory, general polynomial constraints on the

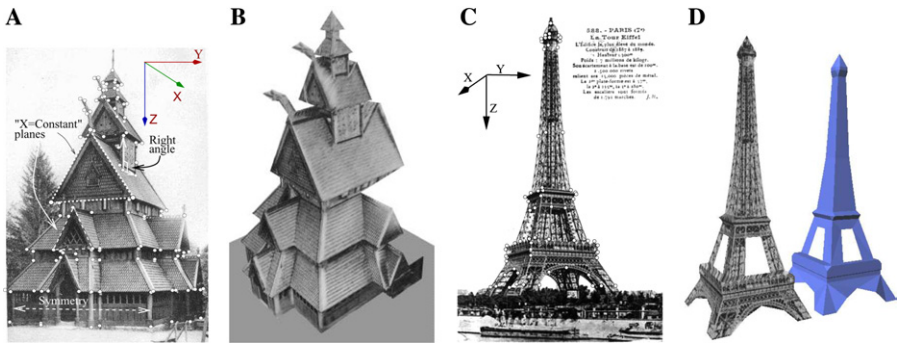


Fig. 1. (A) A man-made scenes, rich in planarities, parallelisms, orthogonalities, and symmetry: the outlined triangles lie on vertical planes, there are many horizontal planes and there is an approximate symmetry with respect to a vertical "*X*–*Z*" plane. (B) This reconstruction is obtained by using the 2D points (white dots) in the left image at left and some geometric information known à-priori. The 3D points (white dots at left) are estimated and visualized with textured planar surfaces. (C) A man-made object, rich in symmetry. (D) Reconstruction, with and without texture.

3D points could be used [2,3]. Most often, however, only planarity, alignment, known angles, and parallelism are used [2,4,9,28,35,38,40,43].

Amongst constraint-based methods, computation differs between single-view and multi-view methods. The latter [2,4,40] being usually modifications of classical 3D reconstruction methods [31].

Single-view constraint-based methods [9,35,38,43] rely on the possibility of expressing geometric properties as linear constraints on the estimated quantities. For this reason, the 3D directions orthogonal or parallel to planes and edges of interest should be estimated before the reconstruction itself. These 3D directions will be called "dominant directions" in the sequel.

Knowing dominant directions beforehand allows to express planarity by a linear constraint. Fig. 2, left, shows that, given a dominant direction $\mathbf{v} \in \mathbb{R}^3$, two 3D points $\mathbf{X}_1, \mathbf{X}_2 \in \mathbb{R}^3$ belong to a plane normal to $\mathbf{v}$ if and only if

$$\mathbf{v}^\top \mathbf{X}_1 = \mathbf{v}^\top \mathbf{X}_2. \tag{1}$$

In single-view constraint-based methods, some dominant directions are estimated before the reconstruction [9,35,38,43]. This is done by estimating vanishing points of dominant directions and then calibrating the camera(s), e.g., with the technique of Caprile and Torre [6]. Once dominant directions are known, the geometric information is expressed by linear equalities, as in Eq. (1).

More linear constraints, provided by the images points are added. The resulting system of linear equations is solved to obtain the reconstruction.

From this general idea, many variations exist. Sturm and Maybank [42,43] use linear equations in which the height (i.e., "intercept") of planes appears; their iterative method has the advantage of not requiring all dominant directions to be known beforehand. Methods from Oxford [9,10,33] make measurements in a plane and along a third direction, then obtain the reconstruction by assembling the measured elements. Wang et al. [47] further extend this approach based on measurements taken on planes and along lines. Kushal et al. [27] allow spheres, frustums, and solids of revolution in the reconstruction. Shum et al. [38,39] select some geometric

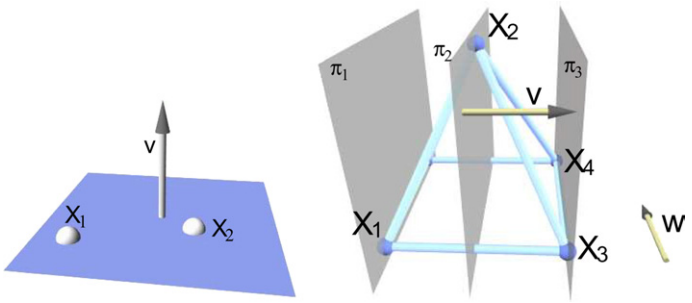


Fig. 2. (Left) The equation $\mathbf{v}^\top \mathbf{X}_1 = \mathbf{v}^\top \mathbf{X}_2$ expresses the fact that $\mathbf{X}_1$ and $\mathbf{X}_2$ belong to a plane with normal $\mathbf{v}$. (Right) The symmetry of the pyramid is expressed by the equality of distances between pairs of parallel planes. For example, the signed distances between the pairs of planes (π_1, π_2) and (π_2, π_3) are opposite.

constraints to be verified exactly by the reconstruction, the remainder being verified approximately; many images may be used by inserting a previous reconstruction in a new view. The multi-view method of Cipolla et al. [7] estimate camera calibrations and orientations from vanishing points like single-view methods. Finally, omnidirectional sensors have been used [16,38,39,42] as well as pinhole cameras.

These methods share common limitations.

- The collection of geometric properties that are exploited—usually planarity and orthogonality—is restricted.
- Methods are designed for either one or for many images, but not for both situations—except in [38].
- The precision of the obtained reconstruction is unknown, except in [10,33].

1.2. Contributions

The method presented here addresses these shortcomings by exploiting a new type of geometric constraint, by working with one or more images alike and by computing the reconstruction together with an estimate of its precision. Moreover, we also provide means to determine whether the user-provided data defines a unique 3D reconstruction.

1.2.1. A new way of exploiting some types of regularity and symmetry

Beyond planarity, linear equations can express some forms of symmetry. Fig. 2B shows the pyramid at the top of Fig. 1A. If $\mathbf{X}_1$, $\mathbf{X}_2$, and $\mathbf{X}_3$ are the corners of the front face and $\mathbf{v}$ is parallel to $\mathbf{X}_3 - \mathbf{X}_1$, then the equation:

$$\mathbf{v}^\top(\mathbf{X}_1 - \mathbf{X}_2) = \mathbf{v}^\top(\mathbf{X}_2 - \mathbf{X}_3), \tag{2}$$

expresses the symmetry of the pyramid with respect to a plane with normal $\mathbf{v}$. Likewise, the squareness of the base of the pyramid is expressed by:

$$\mathbf{v}^\top(\mathbf{X}_1 - \mathbf{X}_3) = \mathbf{w}^\top(\mathbf{X}_4 - \mathbf{X}_3), \tag{3}$$

where $\mathbf{w}$ is a vector parallel to the "X" axis.

In geometric terms, these equations express equalities of *signed distances between pairs of planes*. This notion is distinct from what is usually called "symmetry" or "repetition," but we will nevertheless use these terms. The interested reader may compare our approach to that of Rothwell et al. [37], Zabrodsky and Weishal [49] or Hong et al. [24]. Symmetries like that expressed in Eqs. (2) and (3) are common in man-made scenes; using them is important because it allows to compute reconstructions that cannot be obtained using planarity and orthogonality properties alone, as illustrated in Figs. 1B and D.

1.2.2. A unified framework for one or more images

As noted above, methods either use a single view or many views, but most methods usually cannot handle both cases. One way of extending a single-view method is

to treat multiple views sequentially as in [38], inserting information from previous views at each reconstruction stage.

However, it is easy to adapt a single-view method to use a set of sparse observations from multiple images, so that iterative procedures can be avoided. We show, in Section 3, how a single system of equations can represent all the visual and geometric information given by a sparse multi-view dataset.

1.2.3. Testing the unicity of the reconstruction

Datasets involving many views and many types of geometric constraints do not necessarily define a unique 3D object. Various parts of the scene may not be rigidly connected, so that there exist a continuum of shapes that verify the geometric constraints and project to identical 2D image points. It is [38,43] important to test whether the reconstruction is unique and we propose a method for doing so.

Various approaches are possible. Those based on a threshold on the conditioning of the data [38] raise the problem of choosing the threshold and always incur a non-zero probability of error. Automatic theorem proving [3,30] could be a valid alternative, but it also requires thresholding and would thus be sensible to noise. This is especially frustrating because the property of unicity of a reconstruction should depend only on the geometric information, and not on the noise in the observed image points.

The method presented here is insensitive to noise and uses only common linear algebra tools. It consists in building a synthetic reconstruction problem with the same properties as the original problem. We show that this noiseless reconstruction problem allows to check whether the original problem has a unique solution.

This test for unicity is algebraic and does not indicate whether the setup is near a critical position, this issue being addressed below.

1.2.4. Parameterization of constrained 3D points

There exist many reasons to compute the least-square estimate. For example, if the errors in the observed image points are i.i.d. Gaussian random variable, then the least squares is also the maximum-likelihood estimate. The weaker assumption of independence of errors allows to estimate the covariance of the least-squares estimator [22]. In that case, one may estimate the error committed in real-world reconstructions and detect critical positions in which the estimation problem is ill posed.

The least-squares reconstruction defined by:

$$(\mathbf{X}, R, \mathbf{T}, K) = \arg\min_{\mathbf{X},R,\mathbf{T},K} \|\mathbf{x} - \mathscr{X}(\mathbf{X}, R, \mathbf{T}, K)\|_2^2, \tag{4}$$

where $\mathbf{x} \in \mathbb{R}^{2N}$ is the vector of input image points, $\mathbf{X} \in \mathbb{R}^{3N}$, R, $\mathbf{T}$, and K are the estimated 3D points, camera orientations, translations, and calibrations, respectively; $\mathscr{X}(\,)$ is the perspective projection. In this problem, the 3D points are subject to geometric constraints. To solve the problem with commonplace unconstrained optimization tools, a smooth parameterization of the feasible set is needed. It is thus necessary to parameterize smoothly a collection of 3D points subject to geometric constraints.

Few such parameterization schemes have been proposed. In Bondyfalat et al. [3], some 3D points coordinates are the parameters from which remaining coordinates

are computed. This approach requires polynomial manipulations and may lead to a high condition number. Cornou et al. [8] apply constraints sequentially to 3D points and the constraints should thus be ordered in one way or another. In the present work, the dominant directions need to be ordered, but no particular order is given to the geometric constraints or the points. Since the presented parameterization it is based on the singular value decomposition and other common linear algebra operations, its implementation is not overly complicated. As in [8], this parameterization is easy to plug in the optimization scheme that yields the final least-squares reconstruction.

Having stated our contributions, the article is organized as follows: Section 2, defines the notation and the computation of vanishing points, dominant directions, camera calibration(s), and orientation(s), which is similar to that in Refs. [10,19,38,43]. Section 3 shows how the input data are encoded in a system of linear equations. Section 4 presents the test for unicity and the computation of the reconstruction. The parameterization of 3D points and least-squares reconstruction are in Section 5. Benchmarking and real-world results are shown in Section 6. Finally, Section 7 presents some conclusions and possible extensions.

2. Notations and definitions

This section defines the observation model, input data, and preliminary computations [6,19] and discusses the issue of calibration.

2.1. Observation model

Image points (2D observations) $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathbb{R}^2$ are obtained by perspective projection [23] of 3D points $\mathbf{X}_1, \ldots, \mathbf{X}_N$. Coordinates of 3D points and dominant directions are taken in a basis formed by three linearly independent dominant directions $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$. With this convention, the perspective projection is written [23]:

$$\begin{bmatrix} \mathbf{x}_m \\ 1 \end{bmatrix} = \lambda_m^{-1} V_f \left(\mathbf{X}_m - \mathbf{T}_f \right) + \begin{bmatrix} \varepsilon_m \\ 0 \end{bmatrix}, \tag{5}$$

where $f \in \{1, \ldots, F\}$ is the index of the image in which $\mathbf{x}_m$ has been observed, λ_m is the "depth," V_f is a 3×3 whose columns are the vanishing points of the three dominant directions in which coordinates are taken, $\mathbf{T}_f$ and $\mathbf{X}_m$ are the coordinates of the fth optical center and of the mth point, respectively. Finally, ϵ_m is an additive error term in the observations.

2.2. Input data

We now define the input data provided by the user, which consists in.

- The coordinates $\mathbf{x}_1, \ldots, \mathbf{x}_N$ of some image points, as defined in Eq. (5). The index of the image in which $\mathbf{x}_m$ is observed is written φ_m.

- Planarities: each plane is defined by the indices $\{m,n,\ldots\} \subset \{1,\ldots,N\}$ of the points it contains and by its orientation, given either by two indices i, j of distinct dominant directions parallel to the plane, or by the index of i its normal. The parallelism of planes is expressed by the equality of their orientation.
- Alignments: a known alignment of points is expressed by constraining them to lay in two distinct planes parallel to the alignment.
- Symmetry and other forms of regularity: known ratios of signed distances between pairs of parallel planes are specified by giving the ratio $\alpha \in \mathbb{R}$, the (indices of) directions i, j along which the distances are measured, and two pairs (m,n), (p,q) of (indices of) points contained in each of the involved planes.
- It is assumed that the dominant directions are two-by-two distinct and that three of them either form an orthonormal basis or at least are linearly independent. The dominant directions may be subject to other constraints.

Before reaching the core of this article, some preliminary computations are presented that only use previously published methods.

2.3. Preliminary computations

This section shows how to estimate the dominant directions $\mathbf{v}_i$ from vanishing points which are computed using [19]. Also, the issue of camera calibration is addressed.

2.3.1. Dominant directions

We assume the vanishing points have been obtained by one of the numerous available methods for this purpose. Since the 3D coordinates will be taken in the basis formed by the first three dominant directions, one has $\mathbf{v}_1 = [100]^\top$, $\mathbf{v}_2 = [010]^\top$, and $\mathbf{v}_3 = [001]^\top$.

Other dominant directions are estimated from vanishing points, using the relations

$$\mathbf{g}_i^f \sim V_f \mathbf{v}_i \iff \mathbf{v}_i \sim V_f^{-1} \mathbf{g}_i^f,$$

where $\sim$ indicates equality up to a scale factor and $V_f = [\mathbf{g}_1^f\, \mathbf{g}_2^f\, \mathbf{g}_3^f]$ is the projection matrix of image f (Eq. (5)). Known angles and planarities may further constrain the dominant directions [21].

The minimum requirement for the presented method is that the vanishing points $\mathbf{g}_1^f\, \mathbf{g}_2^f\, \mathbf{g}_3^f$ and all the dominant directions $\mathbf{v}_i$, be computable. In the remainder of this article, it is assumed that this requirement is met.

2.3.2. Calibration: Euclidean vs. affine reconstruction

Camera calibration calibration is important because it allows to obtain a Euclidean, rather than affine, reconstruction [14].

In the present work, calibration is estimated using [6], if three dominant directions are mutually orthogonal and if the camera-scene distance is not too big relative to the size of the scene. Otherwise, an affine reconstruction is obtained. Note that other

Table 1
Summary of notations

$\mathbf{x}_m \in \mathbb{R}^2$	Pixel coordinates of 2D observations
$\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N) \in \mathbb{R}^{2N}$	Pixel coordinates of all 2D observations
$\mathbf{X}_m \in \mathbb{R}^3$	Coordinates in the basis $(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$ of 3D points
$\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_N) \in \mathbb{R}^{3N}$	Coordinates of all 3D points
$\mathbf{T}_f \in \mathbb{R}^3$	Camera translation(s), in the basis ($\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$)
$\mathbf{T} = (\mathbf{T}_1, \ldots, \mathbf{T}_F) \in \mathbb{R}^{3F}$	All camera translations
$\varphi_m \in \{1, \ldots, F\}$	Index of image in which $\mathbf{x}_m$ is observed
$\mathbf{v}_i \in \mathbb{R}^3, \|\mathbf{v}_i\| = 1$	Dominant directions
$\mathbf{v} = (\mathbf{v}_1, \ldots, \mathbf{v}_D) \in \mathbb{R}^{3D}$	All dominant directions
$\mathbf{g}_i^f \in \mathbb{R}^3, \|\mathbf{g}_i^f\| = 1$	Vanishing points of dominant directions
$V_f = [\mathbf{g}_1^f \mathbf{g}_2^f \mathbf{g}_3^f]$	Projection matrix of image number f

calibration methods could be used, e.g., from known angles [23], from parallelepipeds [48] or from tracked 3D points [15].

When calibration is available, the observation model is written:

$$\begin{bmatrix} \mathbf{x}_m \\ 1 \end{bmatrix} = \lambda_m^{-1} K_f R_f (\mathbf{X}_m - \mathbf{T}_f) + \begin{bmatrix} \varepsilon_m \\ 0 \end{bmatrix}, \tag{6}$$

where R_f is the orientation of the camera, represented by an orthogonal matrix, and K_f is the matrix of intrinsic parameters [6]:

$$K_f = \begin{bmatrix} \rho_f & 0 & u_{0f} \\ 0 & \rho_f & v_{0f} \\ 0 & 0 & 1 \end{bmatrix}, \tag{7}$$

where ρ_f is the focal length and $[u_{0f}, v_{0f}]$ is the principal point. Table 1 summarizes the notation used throughout the paper.

3. Representing visual and geometric information

We now reach the core of this paper. This section shows how the input data, the newly computed dominant directions $\mathbf{v}_i (i \in \{1 \ldots D\})$ and vanishing points $\mathbf{g}_i^f (i \in \{1 \ldots 3\},\ f \in \{1 \ldots F\})$ define a system of linear equations on the unknown $\mathbf{X}_m$ and $\mathbf{T}_f$.

3.1. Using geometric information

The two types of geometric constraints are.

(i) Planarity: saying that points $\mathbf{X}_m$ and $\mathbf{X}_n$ belong to a plane with normal $\mathbf{v}_i$, is equivalently expressed by the equation [10,38]

$$\mathbf{v}_i^\top (\mathbf{X}_m - \mathbf{X}_n) = 0. \tag{8}$$

(ii) Ratio of distances between pairs of parallel planes: Eqs. (2) and (3) are generalized to relate distances that are not necessarily equal:

$$\mathbf{v}_i^\top(\mathbf{X}_m - \mathbf{X}_n) + \alpha\mathbf{v}_j^\top(\mathbf{X}_p - \mathbf{X}_q) = 0, \tag{9}$$

where α is the known ratio of signed distances. In geometric terms, this equation fixes the ratio of signed distances between two pairs of parallel planes. One distance is taken along $\mathbf{v}_i$, the other along $\mathbf{v}_j$. One should only take $\mathbf{v}_i \neq \mathbf{v}_j$ when performing Euclidean reconstruction, because affine transformations in general do not leave Eq. (9) invariant unless $\mathbf{v}_i$ and $\mathbf{v}_j$ are collinear. This type of constraint can express symmetry (see Fig. 2, right), squareness, repetition of patterns, etc.

In addition to the constraints above, it is convenient to add three linear constraints to fix the origin of the coordinate system to the center of gravity of the reconstruction. Joining these new constraint to all the geometric information, expressed as Eqs. (8) and (9), yields a single system

$$B(\mathbf{v})\mathbf{X} = \mathbb{O}_{M\times 1}, \tag{10}$$

where $\mathbf{v} = [\mathbf{v}_1^\top, \ldots, \mathbf{v}_N^\top]^\top$ is a $3D \times 1$ vector holding all the dominant directions and $B(\mathbf{v})$ is a $M \times 3N$ matrix holding the coefficients of the equations, which depends only on the geometric constraints and on the dominant directions.

The most important property of the matrix-valued function $B(\mathbf{v})$ is that its rank is maximal for almost all values of $\mathbf{v}$. This implies that the nature—i.e., the degrees of freedom—of the set of points that verify Eq. (10) does not change when the noiseless dominant directions are substituted by their noisy estimates.

The notion of degrees of freedom of the shape defined by the geometric information is conveyed by the dimension of the subspace of vectors $\mathbf{X}$ that verify Eq. (10). This dimension is the *corank* of B [41].

To illustrate this notion, consider Figs. 3A and B, in which two points, $\mathbf{X}_1$ and $\mathbf{X}_2$ are constrained to lie on two planes with normals $\mathbf{v}_i$ and $\mathbf{v}_j$, respectively. If these vectors are not collinear, then the points are constrained to lie on a line. Now, if they are collinear, then the points are only constrained to lie on a plane and the corank of B is increased by one.

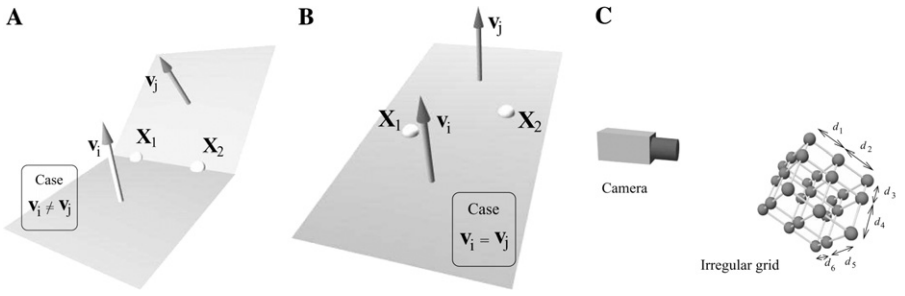


Fig. 3. (A) If the dominant directions $\mathbf{v}_i$ and $\mathbf{v}_j$ differ, then points $\mathbf{X}_1$ and $\mathbf{X}_2$ that belong simultaneously to a plane with normal $\mathbf{v}_i$ and a plane with normal $\mathbf{v}_j$ are constrained to a line. (B) If the dominant directions $\mathbf{v}_i$ and $\mathbf{v}_j$ are equal, $\mathbf{X}_1$ and $\mathbf{X}_2$ are only constrained to a plane, and there is one extra degree of freedom. (C) Points on an irregular grid and camera position used for benchmarking the reconstruction method.

The property of maximal rank of $B(\mathbf{v})$ is stated as:

Lemma 1. *The rank of $B(\mathbf{v})$ is maximal for all values of $\mathbf{v} = (\mathbf{v}_1, \ldots, \mathbf{v}_D)$ except on a subset of empty interior. The proof is in Appendix A.*

3.2. Visual information

Going back to 3D reconstruction, we now use the image projections $\mathbf{x}_1, \ldots, \mathbf{x}_N$ to add linear constraints on $\mathbf{X}$ and on the $\mathbf{T}_f$.

Recalling that the collinearity of two 3D vectors is expressed by the nullity of their cross product, one has

$$\underbrace{\begin{bmatrix}\mathbf{x}_m \\ 1\end{bmatrix} \times V_f}_{S_m}(\mathbf{X}_m - \mathbf{T}_f) = -\underbrace{\begin{bmatrix}\mathbf{x}_m \\ 1\end{bmatrix} \times \begin{bmatrix}\varepsilon_m \\ 0\end{bmatrix}}_{\text{Small error term}} \simeq \mathbb{O}_{3\times 1}. \tag{11}$$

The right-hand term is due to noise and is ignored for now. The left-hand term is a linear function *of rank two* in the unknown 3D point and camera translations, all other quantities being known. Grouping all observations, one gets

$$\underbrace{\begin{bmatrix} S_1 & & \\ & \ddots & \\ & & S_N \end{bmatrix}}_{A} \mathbf{X} - \underbrace{\begin{bmatrix} \mathbf{e}_{\varphi_1}^\top \otimes S_1 \\ \vdots \\ \mathbf{e}_{\varphi_N}^\top \otimes S_N \end{bmatrix}}_{L} \mathbf{T} = A\mathbf{X} + L\mathbf{T} = \mathbb{O}_{3N\times 1} \tag{12}$$

where φ_m is the index of the image in which $\mathbf{x}_m$ is observed, $\mathbf{T} = [\mathbf{T}_1^\top, \ldots, \mathbf{T}_F^\top]^\top \in \mathbb{R}^{3F}$ is the vector of camera translations, $\mathbf{e}_f$ is the $F \times 1$ vector with all zero elements, except for element f, which is one and $\otimes$ is the Kronecker product.

Joining together the geometric information and 2D observations, one obtains

$$\underbrace{\begin{bmatrix} B & \mathbb{O} \\ A & L \end{bmatrix}}_{\mathscr{A}(\mathbf{v},\mathbf{x})} \begin{bmatrix}\mathbf{X} \\ \mathbf{T}\end{bmatrix} = \mathbb{O}, \tag{13}$$

which summarizes all the information provided by the user. The matrix on the left will be called $\mathscr{A}(\mathbf{v}, \mathbf{x})$, in order to emphasize that it is computed from the dominant directions $\mathbf{v}$, and from the 2D image points $\mathbf{x}$. The interested reader may wish to compare Eq. (13) with the equation used by Rother [36, Eq. 13] to represent information.

Although Eq. (13) brings us very close to the reconstruction, we first discuss the unicity of the solution.

4. Test of unicity and algebraic reconstruction

Having transformed the input data into a system of linear equations in the unknown $\mathbf{X}_m$ and $\mathbf{T}_f$, we show how to determine whether the reconstruction is uniquely

defined and, in the affirmative, how to compute it. We begin by precisely stating what is meant by "uniquely defined."

4.1. Unicity

The unicity of reconstruction is a property of the geometric information one has about a set of 3D points. This property is true when, given perspective projections of the 3D points of interest, it is possible to determine their position and that of the cameras uniquely, up to a Euclidean or affine transformation. of the matrix-valued function $B(\mathbf{v})$.

Definition 2. The geometric information encoded in a matrix-valued function $B(\mathbf{v})$ defines a unique reconstruction iff a collection of 2D observations $\mathbf{x} \in \mathbb{R}^{2N}$, determines uniquely the original 3D points $\mathbf{X}$, camera translations $\mathbf{T}$ and projection matrices $V_1, \ldots, V_F$, up to a Euclidean (or affine) transformation.

4.1.1. Noiseless case

Given noiseless observations $\mathbf{x}_m$, a unique reconstruction is defined if and only if the nullspace of the matrix $\mathscr{A}(\mathbf{v}, \mathbf{x})$ of Eq. (13) has dimension one, i.e., if its corank is one. The rank of the matrix $\mathscr{A}(\mathbf{v}, \mathbf{x})$ obtained from noiseless observations $\mathbf{x}_m$ thus determines the unicity of reconstruction.

4.1.2. Real-world case

The above method only works with noiseless 2D data. In real-world situations, errors in the observations $\mathbf{x}_m$ turn up in the matrices $B(\mathbf{v})$, A and L, and $\mathscr{A}(\mathbf{v}, \mathbf{x})$ usually has full rank.

For this reason, we propose a test that is absolutely insensitive to noise. It is based on the following lemma, which is similar to Lemma 1, but relates to the rank of $\mathscr{A}(\mathbf{v}, \mathbf{x})$.

Lemma 3. *The rank of the matrix $\mathscr{A}(\mathbf{v}, \mathbf{x})$ is maximal for all possible dominant directions $\mathbf{v}$ and noiseless observations $\mathbf{x}$, obtained by the perspective projection of 3D points that verify $B(\mathbf{v})\mathbf{X} = \mathbb{O}$, except for a subset with empty interior. The proof is given in Appendix A.*

This lemma implies that, given any collection of noiseless 2D points $\mathbf{x}'_m$ which are the projections of a collection of 3D points $\mathbf{X}'_m$ that verify the geometric properties given by the user, it is possible to test whether the original—noisy—setup is reconstructible.

The test consists in building a vector $\mathbf{x}'$ of noiseless observations and testing whether $\mathscr{A}(\mathbf{v}, \mathbf{x}')$, defined as in Eq. (13) has a nullspace of dimension one.

It is easy to produce $\mathbf{x}'$: dominant directions $\mathbf{v}_i$ and projection matrices V_f are available, all that is missing is an appropriate collection of 3D points $\mathbf{X}'_m$ and camera translations $\mathbf{T}'_f$. The former is obtained by building an orthogonal matrix U whose columns form a basis of the nullspace of $B(\mathbf{v})$ [17] and taking $\mathbf{X}' = U\mathbf{V}'$, where $\mathbf{V}'$ can be any vector, e.g., given by a pseudo-random number generator [32]. Camera translations $\mathbf{T}'$ are also randomly generated and noiseless observations $\mathbf{x}'$ are computed using Eq. (5).

According to Lemma 3, if the reconstruction is uniquely defined, then there is *zero probability* that the rank of $\mathscr{A}(\mathbf{v},\mathbf{x}')$ is not maximal. The rank of $\mathscr{A}(\mathbf{v},\mathbf{x}')$ thus indicates, *with probability one*, the unicity of the reconstruction.

One could argue that, due to numerical approximations in the representation of real values in a computer, the probability of test yielding an erroneous result is in practice non-zero. This argument is not valid in practice, because the computation of the rank is a stable operation [17]. We are not aware of any failure despite extensive usage of this criterion. Altogether, the procedure for determining whether the input is sufficient is:

Criterion 4. Necessary and sufficient condition for the visual and geometric information to define a unique reconstruction

(1) Generate a random vector $\mathbf{V}$ conformable with U, with components, e.g., in $[-1, 1]$. Define $\mathbf{X}' = (\mathbf{X}'_1, \ldots, \mathbf{X}'_N) = U\mathbf{V}$.
(2) Generate random camera positions $\mathbf{T}' = (\mathbf{T}'_1, \ldots, \mathbf{T}'_F)$. Use these camera positions to project the points $\mathbf{X}'_m$ according to Eq. (5), without the noise term. Join these noiseless projections in a vector $\mathbf{x}' = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$.
(3) Build the matrix $\mathscr{A}(\mathbf{v},\mathbf{x}')$ as in Eq. (13).
   (a) If its nullspace has dimension 1, then the visual and geometric information define a unique reconstruction.
   (b) If its nullspace has dimension greater than 1, then information is lacking and the user should provide more observations or more geometric constraints.
   (c) If $\mathscr{A}(\mathbf{v},\mathbf{x}')$ has full rank, then the geometric information is contradictory and the user should remove some of the constrains.

4.2. Reconstruction

This section shows how to obtain a reconstruction that verifies exactly all the geometric properties, assuming the geometric information defines a unique reconstruction.

First, the linear system

$$AU\mathbf{V} - L\mathbf{T} = [AU|L]\begin{bmatrix}\mathbf{V}\\ \mathbf{T}\end{bmatrix} = \mathbb{O}_{3N\times 1}. \tag{14}$$

is built, where U is a matrix whose columns form an orthonormal basis of the nullspace of $B(\mathbf{v})$, e.g., that used in the previous section.

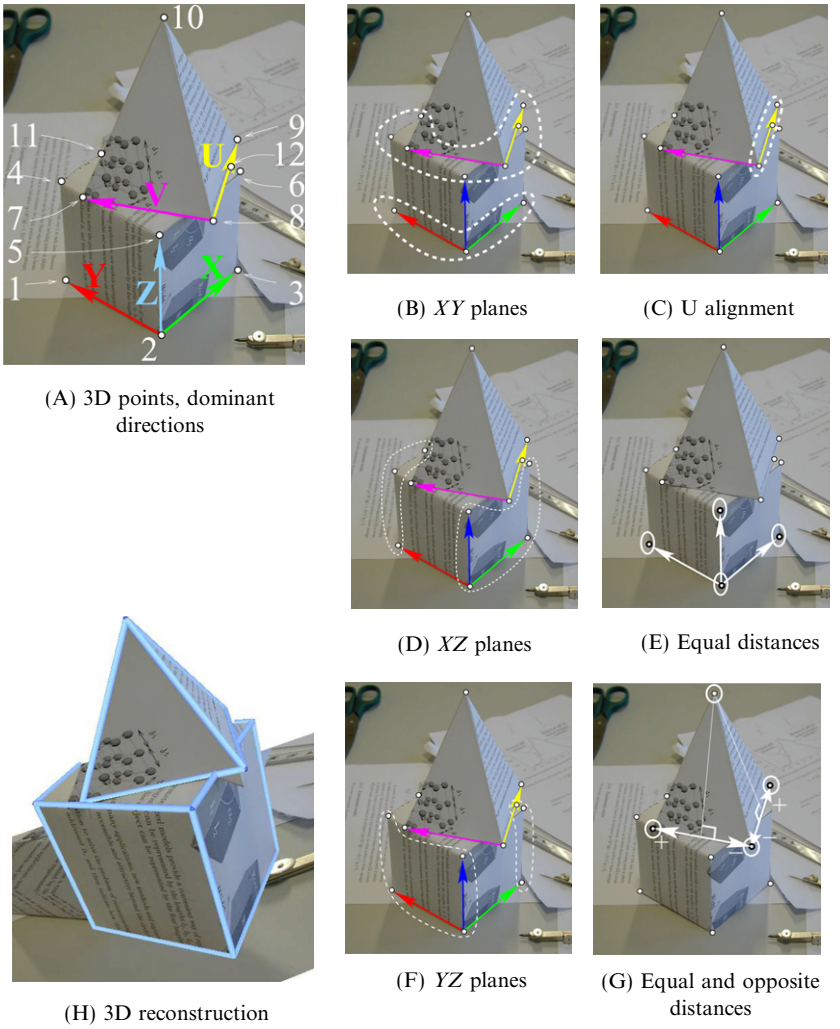
This system is solved in the total least-squares sense [17,25], by assigning to $(\mathbf{V},\mathbf{T})$ the singular vector of $[AU|L]$ corresponding to the smallest singular value. The 3D points given by $\mathbf{X} = U\mathbf{V}$ verify all the geometric constraints given by the user and approximately (exactly in the absence of noise) verify the observations in Eq. (12).

4.3. Illustrative example

Before showing how least-squares estimation is performed, we illustrate how the principles described in this section are put in practice.

Fig. 4A shows the original image with 12 identified points. Five dominant directions X, Y, Z, U, and V, written $\mathbf{v}_1, \ldots, \mathbf{v}_5$ are used: $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$ are fixed to [1, 0, 0], [0, 1, 0], and [0, 0, 1], respectively. Moreover $\mathbf{v}_4$ is constrained to be orthogonal to $\mathbf{v}_3$ and $\mathbf{v}_5$ is defined as $\mathbf{v}_3 \times \mathbf{v}_4$.

Figs. 4B, D, and F show the points that are grouped in XY, XZ, and YZ planes, respectively. The geometric information given by the lower plane in Fig. 4B, which holds three points, translates into the following two rows of B:



(A) 3D points, dominant directions

(B) XY planes

(C) U alignment

(D) XZ planes

(E) Equal distances

(F) YZ planes

(G) Equal and opposite distances

(H) 3D reconstruction

Fig. 4. A simple example: Image points and dominant directions (A), illustrated geometric information (B–G), and reconstruction (H).

$$\begin{bmatrix} \mathbf{v}_3^\top & -\mathbf{v}_3^\top & \mathbb{0} & \mathbb{0} & \cdots & \mathbb{0} \\ \mathbf{v}_3^\top & \mathbb{0} & \mathbf{v}_3^\top & \mathbb{0} & \cdots & \mathbb{0} \end{bmatrix}.$$

The upper plane in that figure, which holds 7 points would likewise contribute 6 rows to B, and the planes in Figs. 4D and F will contribute more rows in the same manner.

Fig. 4C highlights the alignment of points 8, 9, and 12. This geometric information is expressed in B by the rows

$$\begin{bmatrix} \overbrace{\mathbb{0} \quad \cdots \quad \mathbb{0}}^{\times 7} & \mathbf{v}_5^\top & -\mathbf{v}_5^\top & \mathbb{0} & \mathbb{0} & \mathbb{0} \\ \mathbb{0} \quad \cdots \quad \mathbb{0} & \mathbf{v}_5^\top & \mathbb{0} & \mathbb{0} & \mathbb{0} & -\mathbf{v}_5^\top \\ \mathbb{0} \quad \cdots \quad \mathbb{0} & \mathbf{v}_3^\top & -\mathbf{v}_3^\top & \mathbb{0} & \mathbb{0} & \mathbb{0} \\ \mathbb{0} \quad \cdots \quad \mathbb{0} & \mathbf{v}_3^\top & \mathbb{0} & \mathbb{0} & \mathbb{0} & -\mathbf{v}_3^\top \end{bmatrix}.$$

Note that the last two rows are not needed, since they express that points 8, 9, and 12 belong to a XY plane, a fact which has been expressed earlier. However, these rows may be kept with no adverse effect.

Since the lower object is approximately cubic, the signed distances in Fig. 4E are equal, which is expressed by adding to B the rows

$$\begin{bmatrix} \mathbf{v}_2^\top & (\mathbf{v}_1 - \mathbf{v}_2)^\top & -\mathbf{v}_1^\top & \mathbb{0} & \mathbb{0} & \mathbb{0} & \cdots & \mathbb{0} \\ \mathbb{0} & (\mathbf{v}_1 - \mathbf{v}_3)^\top & -\mathbf{v}_1^\top & \mathbb{0} & \mathbf{v}_3^\top & \mathbb{0} & \cdots & \mathbb{0} \end{bmatrix}.$$

The two planar symmetries of the pyramid (Fig. 4F) are expressed by

$$\begin{bmatrix} \mathbb{0} & \cdots & \mathbb{0} & \overbrace{\mathbf{v}_5^\top}^{7} & \mathbf{v}_5^\top & \mathbb{0} & \overbrace{-2\mathbf{v}_5^\top}^{10} & \mathbb{0} & \mathbb{0} \\ \mathbb{0} & \cdots & \mathbb{0} & \mathbb{0} & \mathbf{v}_4^\top & \mathbf{v}_4^\top & -2\mathbf{v}_4^\top & \mathbb{0} & \mathbb{0} \end{bmatrix},$$

and the fact that its base is square is expressed, e.g., by

$$[\mathbb{0} \quad \cdots \quad \mathbb{0} \quad \overbrace{\mathbf{v}_5^\top}^{7} \quad (\mathbf{v}_4 - \mathbf{v}_5)^\top \quad -\mathbf{v}_4^\top \quad \mathbb{0} \quad \mathbb{0} \quad \mathbb{0}].$$

Note that, if none of the last three constraints were used, then the reconstruction would not be well defined, as point 10 would be "dangling" and matrix $\mathscr{A}(\mathbf{v}, \mathbf{x}')$ of Criterion 4 would have corank two. Any one one of these constraints is sufficient for the reconstruction to be well defined, but unless all three constraints are used, the reconstruction would lack some of the properties of the pyramid.

Needless to say, B also has three rows

$$\begin{bmatrix} \mathbf{v}_1^\top & \cdots & \mathbf{v}_1^\top \\ \mathbf{v}_2^\top & \cdots & \mathbf{v}_2^\top \\ \mathbf{v}_3^\top & \cdots & \mathbf{v}_3^\top \end{bmatrix}$$

that ensure that the center of mass of the reconstruction coincides with the origin of coordinates.

Altogether, the matrix $\mathscr{A}(\mathbf{v}, \mathbf{x}')$ of Criterion 4 has corank one, which indicates that the geometric information defines a unique reconstruction. As said above, slightly less information would have been sufficient to uniquely define the reconstruction, but it would then lack some properties.

5. Least-squares estimation

This section shows how to compute the 3D reconstruction that verifies all the given geometric constraints and minimizes the squared error in the image plane. Because the method is iterative, it is initialized with the estimate computed in the previous section.

As stated in Eq. (4), the least-squares reconstruction is defined by

$$(\mathbf{X}, R, \mathbf{T}, K) = \arg\min_{\mathbf{X},R,\mathbf{T},K} \|\mathbf{x} - \mathscr{X}(\mathbf{X}, R, \mathbf{T}, K)\|_2^2,$$

where $\mathscr{X}(\,)$ is the perspective projection defined in Eq. (5) and the 3D points $\mathbf{X}$ are subject to the geometric constraints provided by the user. In order to use, e.g., the efficient and well-known Levenberg–Marquardt algorithm it is necessary to parameterize the feasible set of the optimization problem. The main difficulty lies in the parameterization of the constrained 3D [3,8,11] points, which is the main topic of this section.

5.1. Parameterization of the estimated quantities

A differentiable parameterization of the collection 3D points is obtained by parameterizing the dominant directions and then the space of collections of 3D points that verify all the geometric properties. The mapping, with parameters $\theta_1, \ldots, \theta_D$ for the directions, and $\mathbf{V} \in \mathbb{R}^M$ for the 3D points, takes the form:

$$\begin{aligned} &\mathbf{v}_i = \mathbf{v}_i(\theta_i) \quad \text{for all } i \in \{1, \ldots, D\} \\ &\mathbf{X} = U(\mathbf{v}_1, ..., \mathbf{v}_D)\mathbf{V}. \end{aligned} \tag{15}$$

where the $3N \times M$ matrix $U(\mathbf{v}_1, \ldots, \mathbf{v}_D)$ forms an orthonormal basis of the nullspace of the matrix $B(\mathbf{v}_1, \ldots, \mathbf{v}_D)$ of geometric constraints. The differentiable function $U(\mathbf{v}_1, \ldots, \mathbf{v}_D)$ is defined in Section 5.1.2 and the $\mathbf{v}_i$ are defined below.

5.1.1. Parameterization of the dominant directions

We assume that the $\mathbf{v}_i$ are ordered so that they can be computed sequentially, starting from $\mathbf{v}_1$. Each $\mathbf{v}_i$ is defined by one of the following rules:

Constraint type	Parameters
Known direction: $\mathbf{v}_i =$ Known, e.g., $\mathbf{v}_1 = [1, 0, 0]^\top$, so that no parameter θ_i is needed.	0
Arbitrary direction: $\mathbf{v}_i = \theta_i/\|\theta_i\|$ for some $\theta_i \in \mathbb{R}^3 \setminus \{[0,0,0]^\top\}$.	3
Fixed angle: $\mathbf{v}_i$ is constrained to verify $\mathbf{v}_i^\top \mathbf{v}_j = \cos(\alpha)$, for some previously computed $\mathbf{v}_j$ (i.e., $j < i$) and some fixed α. $\mathbf{v}_i$ is computed by projecting orthogonally onto the 3D circle $\{\mathbf{v} \mid \mathbf{v}^\top \mathbf{v}_j = \cos(\alpha), \|\mathbf{v}\| = 1\}$ a vector $\theta_i \in \mathbb{R}^3$.	3
Coplanarity: $\mathbf{v}_i$ is coplanar with two direction $\mathbf{v}_j$ and $\mathbf{v}_k$, for some $j < i$, $k < i$. This is implemented in the same way as a fixed angle, by taking $\mathbf{v}_i^\top (\mathbf{v}_j \times \mathbf{v}_k) = \cos(\pi/2)$.	3
Cross-product: $\mathbf{v}_i$ is constrained to be orthogonal to two directions, $\mathbf{v}_j$ and $\mathbf{v}_k$, and is defined, e.g., by $\mathbf{v}_i = \mathbf{v}_j \times \mathbf{v}_k/\|\mathbf{v}_j \times \mathbf{v}_k\|$, so that no parameter θ_i is needed.	0

Although this parameterisation is less general than arbitrary angle and planarity constraints between directions, it has the advantage of always being feasible (one cannot give incompatible constraints) and of defining explicitly a differentiable parameterisation.

5.1.2. Parameterization of the constrained points

We now define a differentiable function $U(\mathbf{v}_1,..,\mathbf{v}_D)$ to complete the mapping in Eq. (15). The main issue is the differentiability, since it is well known that, although it is possible to compute a matrix $U(\mathbf{v}_1, .. , \mathbf{v}_D)$ that verifies

$$B(\mathbf{v}_1, .., \mathbf{v}_D)U(\mathbf{v}_1, .., \mathbf{v}_D) = \mathbb{O}_{3N\times M} \quad \text{and } U(\mathbf{v}_1, .., \mathbf{v}_D)^\top U(\mathbf{v}_1, .., \mathbf{v}_D) = I_M, \tag{16}$$

these relations do not define a unique—let alone continuous or differentiable—function $U(\mathbf{v}_1, .. , \mathbf{v}_D)$.

However, under some smoothness conditions verified by $B(\mathbf{v}_1, .. , \mathbf{v}_D)$, it is possible to define locally a function verifying Eq. (16). This is done by fixing arbitrarily its value U_0 at an arbitrary point $(\mathbf{v}_{01}, \ldots , \mathbf{v}_{0D})$ and defining:

$$U(\mathbf{v}_1, .., \mathbf{v}_D) = \arg\min_U \left\{ \|U - U_0\|_F \mid U \text{ verifies Eq. (16)} \right\}. \tag{17}$$

Proof of the differentiability of this function is a generalization of the single-variable case [34,44] and can be done with the implicit function theorem [18]. In practice, $U(\mathbf{v}_1, .. , \mathbf{v}_D)$ is computed by

$$U(\mathbf{v}_1, .., \mathbf{v}_D) = U_1 \mathscr{U} \mathscr{V}^\top, \tag{18}$$

where U_1 is a unitary matrix whose columns form a basis of the nullspace of $B(\mathbf{v}_1, .. , \mathbf{v}_D)$, and $\mathscr{U}$, $\mathscr{V}$ are given by the SVD decomposition of $U_1^\top U_0$: $U_1^\top U_0 \overset{\text{SVD}}{=} \mathscr{U} D \mathscr{V}^\top$. Note that the value of Eq. (18) is independent of the choice of U_1, which can be given by any nullspace or SVD computing algorithm.

The derivative of $U(\mathbf{v}_1,..,\mathbf{v}_D)$ with respect to the ith component of the vector $(\mathbf{v}_1,\ldots,\mathbf{v}_D)$ at $(\mathbf{v}_1^0,\ldots,\mathbf{v}_D^0)$ is given by:

$$U_i' = -B^+ B_i' U. \tag{19}$$

where B_i' is the derivative of B with respect to the ith component and B^+ is the pseudo-inverse. A more general formula exists [18] for the derivatives at other points than $(\mathbf{v}_1^0,\ldots,\mathbf{v}_D^0)$, but it is not needed here.

The other parameters—camera translation, orientation, and calibration are commonly used in optimisation problems and their parameterization does not need to be detailed here. We have thus defined a differentiable parameterization of all the involved quantities.

5.2. Parameter estimation and assessment of accuracy

This parameterization can now be plugged in any least-squares optimization (''bundle adjustment'') method to obtain the least-squares estimates of camera parameters and constrained 3D points.

We used for this purpose the Levenberg–Marquardt algorithm [32], which yields a local minimum of the sum-of-squared-error function starting from the reconstruction given in Section 4.2. The only caveat worth mentioning is that, since the parameterization of $U(\mathbf{v}_1,..,\mathbf{v}_D)$ is local, the value of U_0 is updated at each outer loop of optimization. With this implementation detail, the information in this section should allow the interested reader to reproduce our results.

Finally, since the reconstruction is obtained by minimizing a differentiable function it is straightforward to apply the method of [22] to estimate the covariance of the estimated parameters.

5.3. Summary

We may now summarize the steps that lead from a dataset to a least-squares reconstruction.

(1) Preliminary computations: Estimate the vanishing points, projection matrices, and dominant directions $\mathbf{v}_i$. If possible, compute the calibration so that the $\mathbf{v}_i$ are expressed in a Euclidean coordinate system.
(2) Build the matrix $B(\mathbf{v})$ in Eq. (10) and a matrix U_0 whose columns form an othonormal basis of the nullspace of $B(\mathbf{v})$.
(3) Check the unicity of the reconstruction.
   (a) Generate random vectors $\mathbf{V}'$ and $\mathbf{T}'$, define $\mathbf{X}' = U_0\mathbf{V}'$.
   (b) Compute the perspective projections $\mathbf{x}'_m$ from $\mathbf{X}'$, $\mathbf{T}'$, and the projection matrices $V_1,\ldots,V_F$. Build the matrices A' and L' as in Eq. (12).
   (c) Check that the data defines a unique reconstruction, i.e., that the corank of

$$\mathscr{A}(\mathbf{v},\mathbf{x}') = \begin{bmatrix} B & \mathbb{O} \\ A' & L' \end{bmatrix}$$

   is one. If this is not the case, warn the user and stop.

(4) Reconstruction: compute $(\mathbf{V},\mathbf{T})$, the least singular vector of $[AU_0|L]$ and set $\mathbf{X} = U_0\mathbf{V}$.
(5) Least-squares: find iteratively the $\mathbf{X}$, $\mathbf{T}$, R, and K (in the calibrated case), using the parameterization described in Section 5.1 in conjunction with Levenberg–Marquardt.

6. Experimental results

Having detailed the simple dataset in Fig. 4, we proceed with the performance evaluation of the proposed reconstruction method and will then show results obtained in datasets of significant size.

6.1. Benchmarking

First, the effect of errors in the observations is gauged by benchmarking the method on synthetic data.

Benchmarking is done using the setup shown in Fig. 3C, consisting of 27 points on a $3 \times 3 \times 3$ irregular grid. The lengths of the edges of the grid are uniform random variables in [0.5, 1]. The grid is translated and scaled so that its center of mass be $[0\ 0\ 0]^\top$ and the mean norm of its vertices be 1. The world origin has coordinates $\mathbf{T} = [T_1 T_2 5]^\top$, where the T_i are $N(0, 0.5)$ random variables. The camera is randomly rotated by 30° to 60° around an axis in the "X–Y"' plane. The observations are given by Eq. (5), where the natural logarithm of the focal length is a $N(0, 0.1)$ random variable.

The noise terms, given by a Gaussian pseudo-random number generator, are i.i.d. terms with amplitude varying from 0.0 (no noise) to 2.24% (33 dB) of the amplitude of the observations. For comparison, the noise usually ranges from 0.3 (50 dB) to 1.0% (40 dB) in real-world data.

For each noise level, both estimators are run on 50 different datasets and the error is measured. Fig. 5 shows the curves of the error in the 3D points (left), camera orientation (middle) and translation (right). The error in 3D points and camera

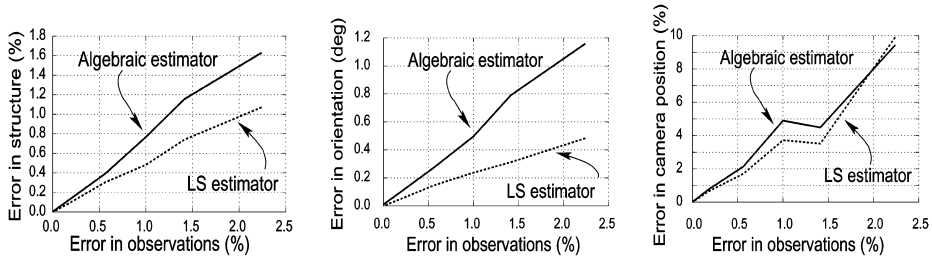


Fig. 5. Error in the reconstruction as a function of the error in the observations: error in the estimated 3D points (left), the orientation of the cameras (middle, measured in degrees), and the translation of the cameras (right).

translations is measured in percents of the amplitude of these quantities, so that these results are invariant to the size of the setup. The error in camera orientation is measured by the mean angle, in degrees, between the columns of the true and the estimated orientation matrices, R_f, in Eq. (5).

Fig. 5 shows that the proposed method behaves well and that it does not break down in very high noise levels. In real-world cases, the algebraic and least-squares reconstructions are indistinguishable by eye, so that the former may be used in situations where precision is less important or computation resource is scarce.

Although, we do not compare our method to others, it is generally admitted that least-squares reconstruction is more accurate than "linear" methods, and that multi-view reconstruction is more accurate when geometric constraints are used. This last fact is, e.g., confirmed in [2] for polynomial constraints, in [1] for planar constraints and in [20] for planar and angular constraints.

6.2. Real-world reconstructions

We now present results obtained from real-world datasets, with the aim of further illustrating the capabilities of the presented method.

6.2.1. Tour Eiffel

Figs. 1C and D illustrates how symmetry allows to uniquely define a reconstruction. Horizontal planes are easily identified, but very few vertical planes, on the first floor only, can be used, so that previous [10,38,43] constraint-based methods cannot treat this dataset. However, symmetries with respect to vertical planes allow to define a unique reconstruction.

Symmetry is enforced using three corner points on each horizontal level and the top-most point. Eqs. (2) and (3) guaranty the symmetry of the figure. Altogether, there are $N = 77$ points, 56 planes and 45 known length ratios. The object has $Q = 49$ degrees of freedom, i.e., U has 49 columns. The error (measured as in the previous section) in the 3D points and camera orientation are approximately 0.52% and 0.21°.

6.2.2. Saint-Michel

The example in Fig. 6 shows that single-view reconstruction needs not be limited to block-like or highly regular objects. Horizontal coplanarity relations hold for

Fig. 6. (Left) Indoor scene with 114 identified points. (Middle, right) Reconstruction obtained from these points and 39 known planes, shown with and without texture.

points on the floor, for points on the wall that can be connected along a horizontal row of bricks and for some points on the window. Vertical planes are identified on the walls and arcades. Altogether, $N = 114$ points, 39 planes and no length ratios are used. The scene has $Q = 173$ degrees of freedom. In this model, the principal point of the camera was estimated. The error in the 3D points and camera orientation are approximately 5.5% and 0.49°.

6.2.3. Folkemuseum

In this model, shown in Figs. 1A and B, $N = 122$ points were identified and there are 75 planes and 26 known length ratios. The model has $Q = 124$ degrees of freedom. The error in the 3D points and camera orientation are approximately 1.53% and 0.31°.

6.2.4. Multiple-view indoors

Fig. 7 shows two indoor images with very little overlap, taken from nearly the same point, in almost perpendicular directions. No intrinsic parameter can reliably be estimated because two vanishing points are almost at infinity. This is thus an affine reconstruction. There are $N = 61$ points, 35 planes and one known ratio of lengths: the distance from the point marked "A" in the first image to that marked "A-prime" in the second image is equal to that from point "B" (first image) to point "B-prime" second image. There is a unique reconstruction despite no 3D point is visible in both images. The error in the 3D points is approximately 3.01%.

6.2.5. Multiple-view outdoors—"Conciergerie"

Fig. 8 (top) shows two outdoors image with some overlap. Seventy-two points (24 in the first image, 48 in the second) and 21 planes are identified; two known length ratios are given, to express that the spikes on the walls–represented by the small inclined planes near the top of the reconstruction—stick out by the same amount on the left and front wall; without this information, the reconstruction would not be uniquely defined. The error in the 3D points and camera orientation are approximately 2.05% and 0.26°.

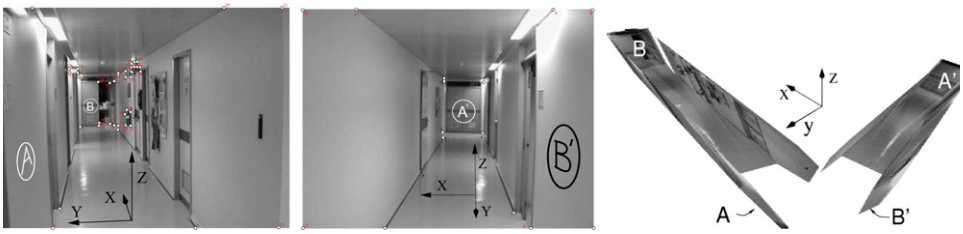


Fig. 7. (Left, middle) Two views with no overlapping points or lines. However, because some planes—the floor, ceiling and a wall—are visible in both images and because the length of the hall is equal on both sides, it is possible to obtain a reconstruction (Right).

Fig. 8. (Left, middle) Two outdoor images. (Right) Untextured reconstruction.

7. Conclusions

We have presented an algebraic method for the of 3D reconstruction of geometrically constrained scenes. Its good behavior is shown on synthetic and real-world data. This method improves over the current state-of-the art by exploiting a wider range of geometric constraints and by handling multi-view datasets naturally. A parameterization of 3D points subject to geometric constraints was defined that allows to obtain the least-squares reconstruction and to estimate its precision. Moreover, a geometric test that depends only on the geometric information determines whether the input data defines a unique reconstruction.

The proposed method is applicable in a wide range of situations and can still be extended in many ways. For example, the camera positions and motion could be constrained linearly just like the 3D points, with very little change in the proposed framework. Furthermore, like other methods that require the preliminary computation of dominant directions, it could benefit from different calibration techniques in the single- [23,48,46] or multi-view [15] cases.

Finally, it should be noted that two original mathematical techniques were introduced in this article. First, in testing the unicity, we use the fact that the outcome of the test does not depend on the particular setup, but only on the geometric information, to devise a test that is insensitive to noise. Second, we use a differentiable representation of the nullspace of a matrix-valued function to parameterize a collection of constrained 3D points.

Appendix A. Demonstrations

We show in this section some important properties of the systems of linear constraints used in this article.

First we show that the rank of $B(\mathbf{v})$ is maximal for almost all possible values of the dominant directions $\mathbf{v}_1, \ldots, \mathbf{v}_D$. This statement is imprecise because it omits that the $\mathbf{v}_i$ are not arbitrary: some are fixed (e.g., $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$), all have unit norm, and other constraints may be imposed [21].

These constraints are expressed by a polynomial equation $C(\mathbf{v}) = 0$, where $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_D)$ and $C(\mathbf{v})$ is a polynomial. This polynomial is a sum of non-nega-

tive terms that are zero iff the constraint is met. For example the condition $\mathbf{v}_i = \mathbb{O}_{3\times 1}$ yields the term $\|\mathbf{v}_1 - [1 \quad 0 \quad 0]^\top\|^2$; $\|\mathbf{v}_i\| = 1$ yields: $(\|\mathbf{v}_i\|^2 - 1)^2$; known angles: $(\mathbf{v}_i^\top \mathbf{v}_j - \cos\beta)^2$(for some known β). The dominant directions are thus restricted to belong to the set $\mathscr{V} = \{\mathbf{v} \in \mathbb{R}^{3D} \mid C(\mathbf{v}) = 0\}$.

Note that $C(\mathbf{v})$ is needed only for the purpose of this demonstration and not for actually testing the unicity of the reconstruction.

Lemma 1. *The rank of $B(\mathbf{v})$ is maximal for all values of $\mathbf{v} = (\mathbf{v}_1, \ldots, \mathbf{v}_D) \in \mathscr{V}$ except for a subset of empty interior in the topology of $\mathscr{V}$ induced by that of $\mathbb{R}^{3D}$.*

This lemma holds because for any integer r, the sum of the squared minors of $B(\mathbf{v})$ of size r or more is either identically zero on $\mathscr{V}$ or zero on a subset of $\mathscr{V}$ that has zero interior for the topology of $\mathscr{V}$ induced by the usual topology of $\mathbb{R}^3$. This is shown using simple analytical arguments: we call $M_r(\mathbf{v})$ the sum of squared determinants of all minors of size r or greater of the matrix $B(\mathbf{v})$. Thus, the rank of $B(\mathbf{v})$ is r iff $M_r(\mathbf{v}) = 0$ and $M_{r-1}(\mathbf{v}) \neq 0$. Define $\mathscr{M}_r = \{\mathbf{v} \in \mathbb{R}^{3D} \mid M_r(\mathbf{v}) = 0\}$; $\mathscr{M}_r \cap \mathscr{V}$ is a closed set of $\mathbb{R}^{3D}$ and is thus a closed set of (the topological subspace) $\mathscr{V}$. If $\mathscr{M}_r$ contains a non-empty open set of $\mathscr{V}$, then $M_r(\mathbf{v})$ and $C(\mathbf{v})$ coincide on that open set and thus coincide on the whole set $\mathscr{V}$. In that case, $M_r(\mathbf{v})$ is identically zero on $\mathscr{V}$. Thus, for all r, $M_r(\mathbf{v})$ is either identically zero on $\mathscr{V}$ or $M_r(\mathbf{v})$ is zero at most on a closed set of empty interior. That is, all minors of B of size r or greater have zero determinant either for all values in $\mathscr{V}$ or at most on a set of empty interior.

The following lemma states a similar property for the matrix $\mathscr{A}$, when the observations are noiseless. While $B(\mathbf{v})$ is a function of the dominant directions alone, $\mathscr{A}(\mathbf{v}, \mathbf{x})$ also involves a vector of noiseless observations $\mathbf{x}$.

Lemma 3. *The rank of the matrix $\mathscr{A}(\mathbf{v}, \mathbf{x})$ is maximal for all possible dominant directions $\mathbf{v}$ and noiseless observations $\mathbf{x}$ obtained by the perspective projection of 3D points that verify $B(\mathbf{v})\mathbf{X} = \mathbb{O}$, except for a subset with empty interior.*

This proof is similar to the previous : we define the set $\mathscr{V} \subset \mathbb{R}^{3D+2N}$ of pairs $(\mathbf{v}, \mathbf{x})$ such that $C(\mathbf{v}) = 0$ and $\mathbf{x}$ is the perspective projection of a collection of 3D points $\mathbf{X}$ that verifies $B(\mathbf{v})\mathbf{X} = \mathbb{O}$. We call $M_r(\mathbf{v}, \mathbf{x})$ the sum of the squared determinants of all minors of size r or greater of the matrix $\mathscr{A}(\mathbf{v}, \mathbf{x})$. The lemma follows from the fact that $M_r(\mathbf{v}, \mathbf{x})$ is either identically zero for all $(\mathbf{v}, \mathbf{x}) \in \mathscr{V}$ or is zero at most on a subset of $\mathscr{V}$ of zero interior for the topology induced on $\mathscr{V}$ by that of $\mathbb{R}^{3D+2N}$.

In full rigor, these lemmas should be adapted to floating-point computation by providing bounds (certainly very low) on the probability that they fail.

References

[1] A. Bartoli, P. Sturm, Constrained structure and motion from multiple uncalibrated views of a piecewise planar scene, Int. J. Comput. Vis. 52 (1) (2003) 45–64.

[2] D. Bondyfalat, S. Bougnoux, Imposing euclidean constraint during self-calibration processes, in: 3D Structure from Multiple Images of Large-scale Environments, ECCV Workshop, Freiburg, Germany, 1998, pp. 224–235.

[3] D. Bondyfalat, B. Mourrain, T. Papadopoulo, An application of automatic theorem proving in computer vision, in: Automated Deduction in Geometry, 1998, pp. 207–231.
[4] B. Boufama, R. Mohr, F. Veillon, Euclidean constraints for uncalibrated reconstruction, in: Intl. Conf. on Computer Vision, Berlin, Germany, 1993, pp. 466–470.
[5] H. Cantzler, R. Fisher, M. Devy, Improving architectural 3D reconstruction by plane and edge constraining, in: British Machine Vision Conf., 2002, pp. 43–52.
[6] B. Caprile, V. Torre, Using vanishing points for camera calibration, Intl. J. Comput. Vis. 4 (1990) 127–140.
[7] R. Cipolla, E. Boyer, 3D model acquisition from uncalibrated images, in: IAPR Workshop on Machine Vision Applications, 1998, pp. 559–568.
[8] S. Cornou, M. Dhome, P. Sayd, Architectural reconstruction with multiple views and geometric constraints, in: British Machine Vision Conf., 2003.
[9] A. Criminisi, Accurate visual metrology from single and multiple uncalibrated images, PhD thesis, University of Oxford, 1999.
[10] A. Criminisi, I. Reid, A. Zisserman, Single view metrology, in: Intl. Conf. Computer Vision, 1999, Corfu, Greece, pp. 434–441.
[11] P. Debevec, C. Taylor, J. Malik, Modeling and Rendering Architecture from Photographs: A hybrid geometry- and image-based approach, Technical Report CSD/96-893, University of California at Berkeley, 1996.
[12] A. Dick, P. Torr, R. Cipolla, Automatic 3D modelling of architecture, in: British Machine Vision Conf., 2000, pp. 372–381.
[13] A. Dick, P. Torr, S. Ruffle, R. Cipolla, Combining single-view reconstruction and multiview stereo for architectural scenes, in: Intl. Conf. on Computer Vision 2001, 2001, pp. 268–274.
[14] O. Faugeras, Stratification of three-dimensional vision: Projective, affine and metric representation, J. Opt. Soc. Am. A 12 (7) (1995) 465–484.
[15] O. Faugeras, Q. Luong, S. Maybank, Camera self calibration: Theory and experiments, in: Eur. Conf. on Computer Vision, Santa Margherita, Italy, 1992.
[16] J. Gaspar, E. Grossmann, J. Santos-Victor, Interactive reconstruction from an omnidirectional image, In Intl. Symposium on Intelligent Robotic Systems, 2001.
[17] G.H. Golub, C.F.V. Loan, Matrix Computations. Johns Hopkins Studies in the Mathematical Sciences, third ed., The Johns Hopkins University Press, Baltimore, MD, USA, 1996.
[18] E. Grossmann, Maximum Likelihood 3D Reconstruction From One or More Uncalibrated Views Under Geometric Constraints, PhD thesis, IST, 2002.
[19] E. Grossmann, D. Ortin, J. Santos-Victor, Single and multi-view reconstruction of structured scenes, in: Asian Conf. on Computer Vision, 2002, pp. 228–234.
[20] E. Grossmann, J. Santos-Victor, "Dual representations" for vision-based 3D reconstruction, in: British Machine Vision Conference, 2000, British Machine Vision Association, Bristol, pp. 516–526.
[21] E. Grossmann, J. Santos-Victor, Maximum likelihood 3D reconstruction from one or more images under geometric constraints, in: British Machine Vision Conf., 2002, pp. 343–352.
[22] R. Haralick, Propagating covariance in computer vision, in: Workshop on Performance Characteristics of Vision Algorithms, Cambridge, 1996, pp. 1–12.
[23] R. Hartley, A. Zissermann, Multiple View Geometry in Computer Vision, Cambridge University Press, 2000.
[24] W. Hong, A.Y. Yang, K. Huang, Y. Ma, On symmetry and multiple view geometry: Structure, pose and calibration from a single image, Intl. J. Comput. Vis. 60 (2004) 241–265.
[25] S.V. Huffel, J. Vandewalle, The Total Least-squares Problem: Computational Aspects and Analysis, SIAM, 1991.
[26] D. Jelinek, C.J. Taylor, Reconstruction of linearly parameterized models from single images with a camera of unknown focal length, IEEE Trans. Pattern Recognit. Mach. Intell. 23 (7) (2001) 767–774.
[27] A.M. Kushal, G. Chanda, K. Shrivastava, M. Gupta, S. Sanyal, T.S. Ram, P. Kalra, S. Banerjee, Multilevel modelling and rendering of architectural scenes, in: Eurographics, 2003.
[28] D. Liebowitz, A. Criminisi, A. Zisserman, Creating architectural models from images, in: EuroGraphics, 1999, pp. 39–50.

[29] D.G. Lowe, Fitting parameterized three-dimensional models to images, IEEE Trans. Pattern Recognit. Mach. Intell. 14 (5) (1991) 441–450.
[30] S. Petitjean, Algebraic geometry and computer vision: Polynomial systems, real and complex roots, J. Math. Imag. Vis. 10 (1999) 191–220.
[31] M. Pollefeys, R. Koch, L.V. Gool, Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters, in: Proc. 6th Intl. Conf. on Computer Vision, 1998, pp. 90–95.
[32] W.H. Press, S.A. Teutolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes, the Art of Scientific Computing, second ed., Cambridge University Press, NY, 1992.
[33] I. Reid, A. Zisserman, Goal-directed video metrology, in: Eur. Conf. on Computer Vision, 1996, pp. 647–658.
[34] W. Rheinboldt, On the computation of multidimensional solution manifolds of parametrized equations, Numerische Math. 53 (1988) 165–181.
[35] D. Robertson, R. Cipolla, An interactive system for constraint-based modelling, in: British Machine Vision Conf., 2000, pp. 536–545.
[36] C. Rother, Linear multi-view reconstruction of points, lines, planes and cameras, using a reference plane, in: Intl. Conf. on Computer Vision, 2003, pp. 1210–1217.
[37] C. Rothwell, D. Forsyth, A. Zisserman, J. Mundy, Extracting Projective Information from Single Views of 3D Point Sets, Technical Report 1973/93, Oxford University Department of Engineering Science, 1993.
[38] H. Shum, M. Han, R. Szeliski, Interactive construction of 3D models from panoramic mosaics, in: IEEE Conf. on Computer Vision and Pattern Recognition, 1998, pp. 427–433.
[39] H. Shum, R. Szeliski, S. Baker, M. Han, P. Anandan, Interactive 3D modeling from multiple images using scene regularities, in: 3D Structure from Multiple Images of Large-scale Environments, ECCV Workshop, Freiburg, Germany, 1998, pp. 236–252.
[40] G. Sparr, Euclidean and affine structure/motion for uncalibrated cameras from affine shape and subsidiary information, in: 3D Structure from Multiple Images of Large-scale Environments, ECCV Workshop, Freiburg, Germany, 1998, pp. 187–207.
[41] G.W. Stewart, Introduction to Matrix Computations, Academic press, 1973.
[42] P. Sturm, A method for 3D reconstruction of piecewise planar objects from single panoramic images, in: OMNIVIS—CVPR Workshop on Omnidirectional Vision, Hilton Head Island, 2000, pp. 119–126.
[43] P.F. Sturm, S.J. Maybank, A method for interactive 3D reconstruction of piecewise planar objects from single views, in: British Machine Vision Conf., 1999, pp. 265–274.
[44] P.K. und, V. Mehrmann, Smooth factorizations of matrix valued functions and their derivatives, Numerische Mathe. 60 (1991) 115–132.
[45] F. van den Heuvel, Trends in cad-based photogrammetric measurement, Intl. Archiv. Photogrammetry Remote Sens. 33 (5/2) (2000) 852–863.
[46] G. Wang, H.-T. Tsui, Z. Hu, F. Wu, Camera calibration and 3D reconstruction from a single view based on scene constraints, Image Vis. Comput., 2003 (submitted).
[47] G. Wang, Z. Hu, F. Wu, H.-T. Tsui, Single view metrology from scene constraints. Image Vis. Comput., 2003 (submitted).
[48] M. Wilczkowiak, E. Boyer, P. Sturm, Camera calibration and 3D reconstruction from single images using parallelepipeds, in: Intl. Conf. on Computer Vision, 2001, pp. 142–148.
[49] H. Zabrodsky, D. Weinshall, Using bilateral symmetry to improve 3D reconstruction from image sequences, Comput. Vis. Image Understand. 67 (1) (1997) 48–57.
[50] M. Zucchelli, J. Santos-Victor, H. Christensen, Multiple plane segmentation using optical flow, in: British Machine Vision Conf., 2002, pp. 313–322.