

# Using Geometric Constraints through Parallelepipeds for Calibration and 3D Modeling

Marta Wilczkowiak, Peter Sturm, and Edmond Boyer

**Abstract**—This paper concerns the incorporation of geometric information in camera calibration and 3D modeling. Using geometric constraints enables more stable results and allows us to perform tasks with fewer images. Our approach is motivated and developed within a framework of semi-automatic 3D modeling, where the user defines geometric primitives and constraints between them. It is based on the observation that constraints, such as coplanarity, parallelism, or orthogonality, are often embedded intuitively in parallelepipeds. Moreover, parallelepipeds are easy to delineate by a user and are well adapted to model the main structure of, e.g., architectural scenes. In this paper, first a duality that exists between the shape parameters of a parallelepiped and the intrinsic parameters of a camera is described. Then, a factorization-based algorithm exploiting this relation is developed. Using images of parallelepipeds, it allows us to simultaneously calibrate cameras, recover shapes of parallelepipeds, and estimate the relative pose of all entities. Besides geometric constraints expressed via parallelepipeds, our approach simultaneously takes into account the usual self-calibration constraints on cameras. The proposed algorithm is completed by a study of the singular cases of the calibration method. A complete method for the reconstruction of scene primitives that are not modeled by parallelepipeds is also briefly described. The proposed methods are validated by various experiments with real and simulated data, for single-view as well as multiview cases.

**Index Terms**—3D modeling, calibration, geometric constraints.

## 1 INTRODUCTION

EFFICIENT 3D modeling from images is one of the most challenging issues in computer vision. The tremendous research effort made to develop feasible methods has proven that recovering 3D structures from 2D images is a difficult and often underconstrained problem. Several reasons account for that, including the fundamental fact that, without any prior information on cameras or on the scene to recover, a Euclidean reconstruction is not possible at all [1]. This is why knowledge of the acquisition process or of the scene is required. A number of approaches have been proposed to exploit prior information, both on camera and scene parameters. Such prior information not only solves the projective ambiguity in the reconstruction but also usually stabilizes the sensitive reconstruction process. Furthermore, it often leads to simple and direct solutions for the estimation of both camera and scene parameters, which may eventually be adjusted nonlinearly for higher accuracy. The method proposed in this paper is based on the observation that constraints such as coplanarity, parallelism, or orthogonality are often embedded intuitively in parallelepipeds. Moreover, parallelepipeds are easy to delineate by a user and are well adapted to model the main structure of, e.g., architectural scenes. Using parallelepipeds to constrain the calibration and reconstruction process enables modeling from small sets of images, in particular from single images, thus making possible reconstructions

from images not originally taken for that purpose, such as archival images or images from the Internet.

An exhaustive review of the literature on using prior information for self-calibration and Euclidean reconstruction is beyond the scope of this paper. We will concentrate on works which have somehow inspired the method we propose, especially direct approaches giving a good first estimate of camera and scene parameters. There is a large variety of information which can be incorporated into a 3D modeling process. This can be simple knowledge of camera intrinsic parameters or pose (stationarity, pure translation, etc.) or of global 3D scene structure (calibration patterns); it can also be information on scene elements such as points, lines, and planes, as well as on high-level primitives like cubes, prisms, cylinders, etc. Nonetheless, whatever the information is, it can be used at any stage of the 3D modeling process, including the initial calibration, pose estimation, model reconstruction, or an additional nonlinear adjustment of the initial estimate at each step.

**Approaches based on calibration patterns.** Classical calibration approaches are based on known positions of points in 3D space or known calibration patterns [2]. Unfortunately, such information relies on specific acquisition systems and is thus seldom available in general situations. The use of prior knowledge on some intrinsic parameters, i.e., self-calibration, offers the opportunity to build more flexible systems.

**Self-calibration.** In standard self-calibration algorithms [3], [4], [5], [6], 3D reconstruction is done in three steps, recovering, in order, the projective, affine, and Euclidean strata, the projective-affine step being considered as the most nonlinear and, thus, the most difficult step. One of the main problems is critical motion sequences, for which self-calibration does not have a unique solution [7]. This problem has been dealt with by restraining the camera motions [8], [9], [10] or by incorporating prior knowledge on the camera [11]

• M. Wilczkowiak is with the Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, UK. E-mail: mw373@cam.ac.uk.

• P. Sturm and E. Boyer are with INRIA Rhône-Alpes, 655 Avenue de l'Europe, 38330 Montbonnot, France.  
E-mail: {Peter.Sturm, Edmond.Boyer}@inrialpes.fr.

Manuscript received 5 Jan. 2004; revised 10 June 2004; accepted 25 June 2004; published online 13 Dec. 2004.

Recommended for acceptance by C. Taylor.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0006-0104.

or on the scene. But, to get stable results for self-calibration, a large number of images is usually necessary.

**Structure and motion.** The basic constraint is that back-projection lines (planes) associated with corresponding image points (lines) intersect in a single space point (line). This observation allows us to formulate the matching tensors, which compactly describe two, three, and four view geometry. When more views are accessible, it is necessary to combine results computed from small subsets of images, which decreases the accuracy of results. An overview of tensor-based structure&motion methods can be found in [12].

Another category of approaches allows the simultaneous recovery of cameras and 3D models via the factorization of a measurement matrix of image points [13], [14], lines [15], [16], or similar methods using planes in the scene [17], [18]. Factorization methods suffer from missing data, i.e., when a primitive is not seen in all images, although some ways of dealing with this problem have been proposed [13], [16]. Using only the above backprojection constraints, it is only possible to recover the scene up to a projective or affine transformation.

**Incorporating Euclidean scene constraints.** A large variety of geometric constraints can disambiguate the projective reconstruction to a Euclidean one and allow us to decrease the number of images required to obtain a satisfying reconstruction. Many of them can be easily incorporated into a self-calibration framework. A common constraint is given by vanishing points of mutually orthogonal directions, as defined by known cubical structures [19], [20], [21] or by dominating scene directions [22]. Also, knowing the Euclidean structure of scene planes is useful in this context, through rectified planes [23], maps [24], or known plane-to-image homographies [25], [26]. It is also possible to use multiple images of unknown planes, but more images in general position are needed here [27], [28].

When cameras are calibrated, it is relatively easy to reconstruct a 3D structure. However, and as mentioned previously, using geometric constraints may dramatically improve the reconstruction quality, especially when a single or only a few images are considered [29]. Even simple constraints can be very efficient, e.g., in [30], [31], vanishing lines of planes and coplanarity constraints are used for single image reconstruction. However, in general, dealing with different types of scene objects and constraints is a complicated problem. Some authors prefer to model the scene by simple primitives like points, lines, and planes and constraints between them, such as incidence, parallelism, orthogonality, etc. Some direct approaches using the bilinear character of many useful constraints were proposed in [32], [33], [34]. The results can be improved using nonlinear methods applying penalty terms corresponding to the constraints [35], constrained optimization techniques [36], [37], [38], or a minimal scene parameterization [39], [40]. Yet a different approach consists of high-level scene descriptions using complex primitives like cubes, prisms, cylinders, etc. [41], [42]. Recently, some effort has been devoted to the automatic detection of such primitives [43]. All these methods ensure, by the strong inherent geometric constraints, that the final models are visually correct.

**The proposed approach.** In this paper, we address the first part of the 3D modeling process—intrinsic and extrinsic calibration (pose/motion estimation). In particular, we study the use of a specific calibration primitive: the parallelepipeds.

Parallelepipeds are frequently present in man-made environments and they naturally encode the scene’s affine structure. Any information about their Euclidean structure (angles or ratios of edge lengths), possibly combined with information about camera parameters, may allow us to recover the entire scene’s Euclidean structure. We propose an elegant formalism to incorporate such information, in which camera parameters are dual to parallelepiped parameters, i.e., any knowledge about one entity provides constraints on the parameters of the others. Hence, the image of a known parallelepiped defines the camera parameters and, reciprocally, a calibrated image of a parallelepiped defines its Euclidean shape (up to size). In this paper, we synthesize previous work on parallelepipeds [44], [45] and propose more elegant and efficient approaches.

Camera and parallelepiped parameters are recovered in two steps. First, a factorization-based approach is used to compute their intrinsic and orientation (rotation) parameters. The usual problems of factorization methods—missing data and unknown scale factors—are dealt with rather easily. Then, position and size parameters are recovered simultaneously using linear least squares. The use of well-constrained calibration primitives allows us to obtain good calibration results even from as little as one image. However, depending on the available constraints, singularities might occur. These are described in a detailed catalogue.

Our calibration approach is conceptually close to self-calibration, especially to methods that upgrade an affine structure to Euclidean [5], [6] or methods considering special camera motions [8], [9], [10]. The way Euclidean information on a parallelepiped is used is also similar to vanishing point-based methods [19], [20], [21], [22]. Some properties of our algorithm are also common with plane-based approaches [25], [26], [27], [28], [17], [18]. While more flexible than standard calibration techniques, plane-based approaches still require either Euclidean information or, for self-calibration, many images in general position [27], or at least one plane visible in all images [17]. In this sense, our approach is a generalization of plane-based methods with Euclidean information to three-dimensional parallelepipedic patterns. Finally, our approach can be compared to methods using complex primitives for scene representation. However, unlike most such methods, we use the parallelepiped parameters directly to solve the calibration problem, without requiring nonlinear optimization.

While the main contributions of the paper concern the estimation of camera and parallelepiped parameters, we show that the proposed method can be easily combined with an approach for enhancing reconstructions with primitives other than parallelepipeds [34]. The complete system allows for both calibration and 3D model acquisition from a small number of images with a reasonable amount of user interaction.

The paper is organized as follows: Section 2 gives definitions and some background. Section 3 introduces the concept of camera-parallelepiped duality. Calibration using parallelepipeds and a study on the singular configurations are described in Sections 4 and 5. Sections 6 and 7 describe our approaches for pose estimation and 3D reconstruction. Experimental results are presented in Section 8.

## 2 PRELIMINARIES

### 2.1 Camera Parameterization

We represent cameras using the pinhole model. The projection from a 3D point  $\mathbf{P}$  to a 2D image point  $\mathbf{p}$  is expressed by:  $\mathbf{p} \sim \mathbf{M}\mathbf{P}$ , where  $\mathbf{M}$  is a  $3 \times 4$  matrix, which can be decomposed as:

$$\mathbf{M} = \mathbf{K}(\mathbf{R} \quad \mathbf{t}).$$

The  $3 \times 4$  matrix  $(\mathbf{R} \quad \mathbf{t})$  encapsulates the camera's pose in the world coordinate system or its extrinsic parameters: The rotation matrix  $\mathbf{R}$  represents its orientation and the vector  $-\mathbf{R}^\top \mathbf{t}$  its position. The  $3 \times 3$  calibration matrix  $\mathbf{K}$  or, equivalently,  $\omega \sim \mathbf{K}^{-\top} \mathbf{K}^{-1}$  represents the camera's intrinsic parameters:

$$\mathbf{K} = \begin{pmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\omega \sim \mathbf{K}^{-\top} \mathbf{K}^{-1} \sim \begin{pmatrix} 1 & 0 & -u_0 \\ 0 & \tau^2 & -\tau^2 v_0 \\ -u_0 & -\tau^2 v_0 & \tau^2 \alpha_v^2 + u_0^2 + \tau^2 v_0^2 \end{pmatrix}, \quad (1)$$

where  $\alpha_u$  and  $\alpha_v$  stand for the focal length, expressed in horizontal and vertical pixel dimensions,  $s$  is a skew parameter considered as equal to zero in the following,  $(u_0, v_0)$  are the pixel coordinates of the principal point, and  $\tau = \frac{\alpha_u}{\alpha_v}$  is the camera's aspect ratio.  $\omega$  represents the IAC (image of the absolute conic) and is commonly used to express constraints on the intrinsic parameters. In the following, the term *camera axes* will be used for the axes of the camera coordinate system, i.e., the coordinate system attached to the camera's optical center, two of them being parallel to pixel edges and the third one being orthogonal to the image plane (the optical axis).

### 2.2 Parallelepiped Parameterization

A parallelepiped is defined by 12 parameters: six extrinsic parameters describing its orientation and position and six intrinsic parameters describing its Euclidean shape: three dimension parameters (edge lengths  $l_1, l_2$ , and  $l_3$ ) and three angles between edges ( $\theta_{12}, \theta_{23}, \theta_{13}$ ). These intrinsic parameters are illustrated in Fig. 1. The parallelepiped may be represented compactly by a  $4 \times 4$  matrix  $\mathbf{N}$ :

$$\mathbf{N} = \begin{pmatrix} \mathbf{S} & \mathbf{v} \\ \mathbf{0}^\top & 1 \end{pmatrix} \underbrace{\begin{pmatrix} l_1 & l_2 c_{12} & l_3 c_{13} & 0 \\ 0 & l_2 s_{12} & l_3 \frac{c_{23} - c_{13} c_{12}}{s_{12}} & 0 \\ 0 & 0 & l_3 \sqrt{\frac{s_{12}^2 - c_{13}^2 s_{12}^2 - (c_{23} - c_{13} c_{12})^2}{s_{12}^2}} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\tilde{\mathbf{L}}},$$

where  $\mathbf{S}$  is a rotation matrix and  $\mathbf{v}$  a vector, representing the parallelepiped's pose (extrinsic parameters). The  $4 \times 4$  matrix  $\tilde{\mathbf{L}}$  represents the parallelepiped's shape (intrinsic parameters) with:  $c_{ij} = \cos \theta_{ij}$ ,  $s_{ij} = \sin \theta_{ij}$ ,  $\theta_{ij} \in ]0, \pi[$ ,  $l_i > 0$ .

The matrix  $\tilde{\mathbf{L}}$  represents the affine transformation between a canonic cube and a parallelepiped with the given shape. Concretely, a vertex  $(\pm 1, \pm 1, \pm 1)^\top$  of the canonic cube is mapped, by  $\tilde{\mathbf{L}}$ , to a vertex of our parallelepiped's intrinsic

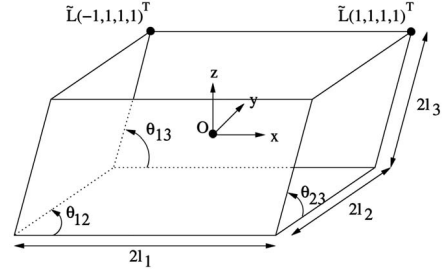


Fig. 1. Parameterization of a parallelepiped:  $2l_i$  are the edge lengths,  $\theta_{ij}$  are the angles between nonparallel edges.

shape. Then, the pose part of  $\mathbf{N}$  maps the vertices into the world coordinate system.

Other parameterizations for  $\tilde{\mathbf{L}}$  may be chosen, but the above one is attractive due to its upper triangular form. This underlines the fact that  $\tilde{\mathbf{L}}$  plays the same role for the parallelepiped as the calibration matrix  $\mathbf{K}$  for a camera.

Analogous to a camera's IAC  $\omega$  is the matrix  $\mu$ , defined by:

$$\mu \sim \mathbf{L}^\top \mathbf{L} \sim \begin{pmatrix} l_1^2 & l_1 l_2 \cos \theta_{12} & l_1 l_3 \cos \theta_{13} \\ l_1 l_2 \cos \theta_{12} & l_2^2 & l_2 l_3 \cos \theta_{23} \\ l_1 l_3 \cos \theta_{13} & l_2 l_3 \cos \theta_{23} & l_3^2 \end{pmatrix}, \quad (2)$$

where  $\mathbf{L}$  is the upper left  $3 \times 3$  matrix of  $\tilde{\mathbf{L}}$ .

Hence, there is a symmetry between the intrinsic parameters of cameras and parallelepipeds ((1) and (2)). The only difference is that, in some cases, the *size* of a parallelepiped matters, as will be explained in the following. As for cameras, the fact that  $K_{33} = 1$  allows us to fix the scale factor in the relation  $\omega \sim \mathbf{K}^{-\top} \mathbf{K}^{-1}$  and, thus, to extract  $\mathbf{K}$  uniquely from the IAC  $\omega$ , e.g., using Cholesky decomposition. As for parallelepipeds, however, we have no such constraint on its "calibration matrix"  $\mathbf{L}$ , so the relation  $\mu \sim \mathbf{L}^\top \mathbf{L}$  gives us a parallelepiped's Euclidean shape, but not its (absolute) size. This does not matter in general since we are usually only interested in reconstructing a scene up to some scale. However, when reconstructing several parallelepipeds, one needs to recover at least their *relative* sizes.

There are many possibilities to define the size of a parallelepiped. We choose the following definition, motivated by the equations underlying our calibration and reconstruction algorithms below: The *size* of a parallelepiped is defined as  $s = (\det \mathbf{L})^{1/3}$ . This definition is actually directly linked to the parallelepiped's volume:  $s^3 = \det \mathbf{L} = \text{Vol}/8$  (the factor 8 arises since our canonic cube has an edge length of 2).

## 3 PROJECTIONS OF PARALLELEPIPEDS

### 3.1 One Parallelepiped in a Single View

In this section, we introduce the concept of duality between the intrinsic parameters of cameras and parallelepipeds. Consider the projection of a parallelepiped's vertices into a camera. Let  $\mathbf{C}_{i, i \in [1, 8]}$  be the homogeneous coordinates of the canonic cube's vertices. Using results from Section 2.2, the projection of the corresponding vertex in the image is:

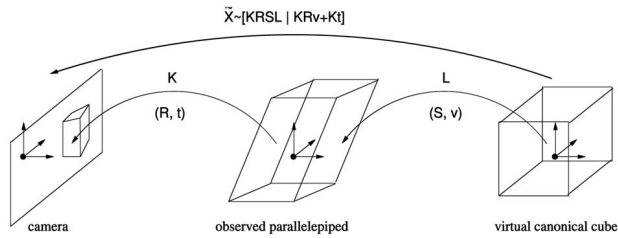


Fig. 2. The projection of the canonic parallelepiped (cube) into the image. Matrices  $K$ ,  $L$  correspond to intrinsic parameters of camera and parallelepiped and  $(R, \mathbf{t})$ ,  $(S, \mathbf{v})$  correspond to extrinsic parameters of camera and parallelepiped, respectively.

$$\mathbf{p}_i \sim \mathbf{MP}_i = \underbrace{K(R \quad \mathbf{t}) \begin{pmatrix} S & \mathbf{v} \\ \mathbf{0}^\top & 1 \end{pmatrix} \tilde{L}}_{\tilde{X}} \mathbf{C}_i. \quad (3)$$

The matrix  $\tilde{X}$  will be called the *canonic projection matrix*. It represents a perspective projection that maps the vertices of the canonic cube onto the image points of the parallelepiped's vertices. This is illustrated in Fig. 2. Given image points for at least six vertices,<sup>1</sup> the canonic projection matrix can be computed [2], even without prior knowledge on intrinsic or extrinsic parameters. Our calibration and pose algorithms are based on the link between the canonic projection matrix  $\tilde{X}$  (which we suppose given from now on) and the camera's and parallelepiped's intrinsic and extrinsic parameters.

Let us consider this in more detail. First, we may identify the *relative pose* between camera and parallelepiped in (3), represented by the following  $3 \times 4$  matrix:

$$(R \quad \mathbf{t}) \begin{pmatrix} S & \mathbf{v} \\ \mathbf{0}^\top & 1 \end{pmatrix} = (RS \quad R\mathbf{v} + \mathbf{t}).$$

Second, let us consider the leading  $3 \times 3$  submatrix  $X$  of the canonic projection matrix  $\tilde{X}$ , which is given by:  $X \sim K(RS)L$ .

Due to the orthogonality of the rotation matrices  $R$  and  $S$ , it is simple to derive the following relation between the camera's IAC  $\omega$  and the corresponding entity  $\mu$  of the parallelepiped:

$$X^\top \omega X \sim \mu. \quad (4)$$

This equation establishes an interesting duality between the intrinsic parameters of a camera and those of a parallelepiped. It shows (unsurprisingly) that knowing the parallelepiped's shape  $\mu$  allows us to calibrate the camera. Conversely, knowing the camera's intrinsic parameters allows us to compute the parallelepiped's Euclidean shape, also from a single image. Moreover, even partial information about one set of intrinsic parameters allows us to form equations on the other set [44].

In the next sections, we generalize the use of this duality for calibration and pose estimation to the case of multiple parallelepipeds seen in multiple cameras and to the use of *partial* knowledge about the camera's or parallelepiped's intrinsic parameters. Before doing so, let us describe a few

1. In theory, five image points and one image direction are sufficient to determine the 11 parameters of a projection matrix. Additional points make the computation more stable.

interesting links between our and other (self-) calibration scenarios.

Classical self-calibration usually proceeds in two main steps: First, a projective reconstruction of the scene is obtained from image correspondences. Then, this is upgraded to a Euclidean reconstruction using the available prior knowledge on intrinsic parameters. Sometimes, an intermediate upgrade to an affine reconstruction is performed.

In our scenario, we have a 3D reconstruction of the scene already from a *single* rather than multiple images, which is, furthermore, of *affine* rather than projective nature: We know that the observed parallelepiped's shape is that of a cube, up to some affine transformation. Analogously, our canonic projection matrix is equal to the true one up to an affine transformation. Hence, self-calibration in our scenario does not need to recover the plane at infinity, which is known to be the hardest part of self-calibration. Indeed, our calibration method is somewhat similar to the affine-to-Euclidean upgrade of stratified self-calibration approaches, e.g., [5], [6].

Similarities also exist with (self-) calibration approaches based on special camera motions: Calibrating a rotating camera [8], [9] is more or less equivalent to self-calibrating a camera in general motion once the affine structure is known. Other approaches recover the affine structure by first performing pure translations and then general motions [10], [46].

Our approach is similar to all these. In the following sections, we show how it allows us to efficiently combine the usual self-calibration constraints with constraints on scene structure. This enables us to perform calibration (and 3D reconstruction) from very few images; one image may actually be sufficient.

### 3.2 $n$ Parallelepipeds in $m$ Views

Let us now consider the general case where  $n$  parallelepipeds are seen by  $m$  cameras. Let  $\tilde{X}_{ik}$  be the canonic projection matrix associated with the projection of the  $k$ th parallelepiped in the  $i$ th camera and  $\lambda_{ik}$  a scale factor such that (3) can be written as a component-wise equality:

$$\lambda_{ik} \tilde{X}_{ik} = K_i(R_i \quad \mathbf{t}_i) \begin{pmatrix} S_k & \mathbf{v}_k \\ \mathbf{0}^\top & 1 \end{pmatrix} \tilde{L}_k. \quad (5)$$

We may gather these equations for all  $m$  cameras and  $n$  parallelepipeds into the following single matrix equation:

$$\underbrace{\begin{bmatrix} \lambda_{11} \tilde{X}_{11} & \cdots & \lambda_{1n} \tilde{X}_{1n} \\ \vdots & \ddots & \vdots \\ \lambda_{m1} \tilde{X}_{m1} & \cdots & \lambda_{mn} \tilde{X}_{mn} \end{bmatrix}}_{\mathcal{X}_{3m \times 4n}} = \underbrace{\begin{bmatrix} K_1(R_1 \quad \mathbf{t}_1) \\ \vdots \\ K_m(R_m \quad \mathbf{t}_m) \end{bmatrix}}_{\mathcal{M}_{3m \times 4}} \underbrace{\left[ \begin{pmatrix} S_1 & \mathbf{v}_1 \\ \mathbf{0}^\top & 1 \end{pmatrix} \tilde{L}_1 \quad \cdots \quad \begin{pmatrix} S_n & \mathbf{v}_n \\ \mathbf{0}^\top & 1 \end{pmatrix} \tilde{L}_n \right]}_{\mathcal{S}_{4 \times 4n}}. \quad (6)$$

This equation naturally leads to the idea of a factorization-based calibration algorithm, which will be developed in Section 4. It is based on the following observation: The matrix  $\mathcal{X}$  contains all information that can be recovered from the parallelepipeds' image points alone (below, we

discuss the issue of computing the scale factors  $\lambda_{ik}$ . In analogy with [13], we call it the *measurement matrix*. Since the measurement matrix is the product of a “motion matrix”  $M$  of four columns, with a “shape matrix”  $S$  of four rows, its rank can be four at most (in the absence of noise).

We might aim at extracting intrinsic and extrinsic parameters directly from a rank-4-factorization of  $\mathcal{X}$ . One step of factorization-based methods for structure and motion recovery is to disambiguate the factorization’s result: In general, for a rank- $r$ -factorization, motion and shape are recovered up to a transformation represented by an  $r \times r$  matrix (here, this would be a 3D projective transformation). The ambiguity can be reduced using, e.g., constraints on intrinsic camera parameters (see more details in Section 4). In our case, we observe that the  $4 \times 4$  subblocks of the shape matrix  $S$  are affine transformations. We would have to include this constraint into the disambiguation, but, nevertheless, the result would not, in general, exactly satisfy the affine form of these subblocks. We thus cut the problem into two steps, which allows us to easily guarantee that the subblocks of the shape matrix will be affine transformations. In the first step (Section 4), we consider a “reduced measurement matrix” consisting of the leading  $3 \times 3$  submatrices of the  $\tilde{X}_{ik}$ . We extract *intrinsic* and *orientation* parameters of our cameras and parallelepipeds based on a rank-3-factorization and a disambiguation stage using calibration and scene constraints. In the second step (Section 6), we then estimate the *position* of cameras and parallelepipeds, as well as the parallelepipeds’ *size*.

Just as a sidenote, we observe that, for two views  $i$  and  $j$  and a parallelepiped  $k$ , the infinite homography between the two views is given by the product  $X_{ik}X_{jk}^{-1}$ .

#### 4 ESTIMATING INTRINSIC AND ORIENTATION PARAMETERS BY FACTORIZATION

In this section, we concentrate on the computation of the cameras’ and parallelepipeds’ intrinsic parameters and orientation (rotation), based on (6) and the observations concerning it, cf. the previous section. As mentioned, we first restrict our attention to the leading  $3 \times 3$  submatrices of the  $\tilde{X}_{ik}$ , as in Section 3.1 for the establishment of the duality between intrinsic parameters of cameras and parallelepipeds. We thus deal with the following subpart of (6):

$$\underbrace{\begin{bmatrix} \lambda_{11}X_{11} & \cdots & \lambda_{1n}X_{1n} \\ \vdots & \ddots & \vdots \\ \lambda_{m1}X_{m1} & \cdots & \lambda_{mn}X_{mn} \end{bmatrix}}_{\mathcal{X}'_{3m \times 3n}} = \underbrace{\begin{bmatrix} K_1R_1 \\ \vdots \\ K_mR_m \end{bmatrix}}_{\mathcal{M}_{3m \times 3}} \underbrace{[S_1L_1 \cdots S_nL_n]}_{S'_{3 \times 3n}}. \quad (7)$$

In the following, we describe the different steps of our factorization-based method. We first deal with the problem of missing data. Then, we describe how to compute the scale factors  $\lambda_{ik}$ , needed to construct the measurement matrix  $\mathcal{X}'$ . The factorization itself is described in Section 4.3, followed by the most important aspect: how to disambiguate the factorization’s result in order to extract intrinsic and orientation parameters. A summary of these steps and a discussion of minimal cases and singularities is provided at the end of this section and in Section 5. The subsequent

computation of position parameters and parallelepiped size is dealt with in Section 6.

##### 4.1 Missing Data

As is usual with factorization approaches, our method might suffer from the problem of missing data, i.e., missing  $X_{ik}$ . Indeed, in practice, the condition that all parallelepipeds are seen in all views is usually not satisfied. However, each missing matrix  $X_{ik}$  can be deduced from others if there is one camera  $j$  and one parallelepiped  $l$  such that  $X_{jl}$ ,  $X_{jk}$ , and  $X_{il}$  are known. The missing matrix can be computed using:

$$X_{ik} \sim X_{il} (X_{jl})^{-1} X_{jk}. \quad (8)$$

Several equations of this type can be used simultaneously to increase the accuracy. Care has to be taken since (8) is defined up to scale only. This problem can be circumvented very simply though by normalizing all  $X_{ik}$  to unit determinant.

These observations motivate a simple recursive method<sup>2</sup> to compute missing matrices  $X_{ik}$ : At each iteration, we compute the one for which most equations of type (8) are available. Previously computed matrices  $X_{ik}$  can be involved at every successive iteration of this procedure.

##### 4.2 Recovery of Scale Factors

The reduced measurement matrix  $\mathcal{X}'$  in (7) is, in the absence of noise, of rank 3, being the product of a matrix of three columns and a matrix of three rows. This, however, only holds if a correct set of scale factors  $\lambda_{ik}$  is used. For other problems, these are often nontrivial to compute, see, e.g., [28], [14]. In our case, however, this turns out to be rather simple.

Let us first write  $A_i = K_iR_i$  and  $B_k = S_kL_k$ . What we know is that (in the absence of noise) there exist matrices  $A_i$ ,  $i = 1..m$  and  $B_k$ ,  $k = 1..n$  such that:  $\forall i, k : X_{ik} \sim A_iB_k$ . Since this equation is valid up to scale only, we also have:  $\forall i, k : X_{ik} \sim (a_iA_i)(b_kB_k)$  for any nonzero scale factors  $a_i$ ,  $i = 1..m$  and  $b_k$ ,  $k = 1..n$ . Consequently, this is also true for the scale factors  $a_i$  and  $b_k$  that satisfy:

$$\det(a_iA_i) = \det(b_kB_k) = 1.$$

Note that we do not need to know these scale factors; it is sufficient to know they exist!

Hence, there exist scale factors  $a_i$  and  $b_k$  with:

$$\forall i, k : X_{ik} \sim a_i b_k A_i B_k, \quad (9)$$

$$\forall i, k : \det(a_i b_k A_i B_k) = \det(a_i A_i) \det(b_k B_k) = 1. \quad (10)$$

As for the sought for scale factors  $\lambda_{ik}$ , we use those that give  $\det(\lambda_{ik}X_{ik}) = 1$ . They are computed as:

$$\lambda_{ik} = (\det X_{ik})^{-1/3}.$$

Due to (9), we have  $\lambda_{ik}X_{ik} \sim a_i b_k A_i B_k$  and, since the determinants of both sides of this equation are equal (they are both equal to 1, cf. the definition of  $\lambda_{ik}$  and (10)), the equation not only holds up to scale, but component-wise:<sup>3</sup>

$$\forall i, k : \lambda_{ik}X_{ik} = (a_iA_i)(b_kB_k).$$

2. Compare with the analogous method in [18].

3. Two nonsingular  $3 \times 3$  matrices that are equal up to scale and whose determinants are equal are also equal component-wise.

This means that the measurement matrix in (7), with the scale factors  $\lambda_{ik}$  as described here, is of rank 3: It is the product of one matrix of three columns (the  $a_i A_i$  stacked on top of each other) and one of three rows (the  $b_k B_k$  side-by-side).

In the following, we assume that the  $X_{ik}$  are already scaled to unit determinant, i.e., that  $\lambda_{ik} = 1$ . Equation (7) becomes:

$$\underbrace{\begin{bmatrix} X_{11} & \cdots & X_{1n} \\ \vdots & \ddots & \vdots \\ X_{m1} & \cdots & X_{mn} \end{bmatrix}}_{\mathcal{X}'_{3m \times 3n}} = \underbrace{\begin{bmatrix} a_1 K_1 R_1 \\ \vdots \\ a_m K_m R_m \end{bmatrix}}_{\mathcal{M}'_{3m \times 3}} \underbrace{\begin{bmatrix} b_1 S_1 L_1 & \cdots & b_n S_n L_n \end{bmatrix}}_{S'_{3 \times 3n}}. \quad (11)$$

The scale factors  $a_i$  and  $b_k$  do not matter for now; all that counts is that they exist and that the measurement matrix  $\mathcal{X}'$  containing the normalized  $X_{ik}$  is of rank 3 at most and can thus be factorized as shown below.

### 4.3 Factorization

As usual, we use the SVD (Singular Value Decomposition) to obtain the low-rank factorization of the measurement matrix. Let the SVD of  $\mathcal{X}'$  be given as:

$$\mathcal{X}'_{3m \times 3n} = U_{3m \times 3n} \Sigma_{3n \times 3n} V_{3n \times 3n}^\top.$$

The diagonal matrix  $\Sigma$  contains the singular values of  $\mathcal{X}'$ :  $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{3n}$ . In the absence of noise,  $\mathcal{X}'$  is of rank 3 at most and  $\sigma_4 = \cdots = \sigma_{3n} = 0$ . If noise is present,  $\mathcal{X}'$  is of full rank in general. Setting all singular values to zero, besides the three largest ones, leads to the best rank-3 approximation of  $\mathcal{X}'$  (in the sense of the Frobenius norm).

In the following, we consider the rank-3 approximation of  $\mathcal{X}'$  (for ease of notation, we denote this also as  $\mathcal{X}'$ ):

$$\mathcal{X}' = U_{3m \times 3n} \text{diag}(\sigma_1, \sigma_2, \sigma_3, 0, \dots, 0) V_{3n \times 3n}^\top.$$

In the matrix product on the right, only columns of  $U$  and rows of  $V^\top$  corresponding to nonzero  $\sigma_j$  contribute. Hence:

$$\mathcal{X}' = U'_{3m \times 3} \text{diag}(\sigma_1, \sigma_2, \sigma_3) \left( V'^\top \right)_{3 \times 3n},$$

where  $U'$  (resp.  $V'$ ) consists of the first three columns of  $U$  (resp.  $V$ ). Let us define  $U'' = U' \text{diag}(\sqrt{\sigma_1}, \sqrt{\sigma_2}, \sqrt{\sigma_3})$  and  $V'' = V' \text{diag}(\sqrt{\sigma_1}, \sqrt{\sigma_2}, \sqrt{\sigma_3})$ . Thus, we have:  $\mathcal{X}' = U'' V''^\top$ . This represents a decomposition of the measurement matrix  $\mathcal{X}'$  into a product of a matrix of three columns ( $U''$ ) with a matrix of three rows ( $V''^\top$ ). Note, however, that this decomposition is not unique. For any nonsingular  $3 \times 3$  matrix  $T$ , the following is also a valid decomposition:

$$\mathcal{X}' = (U'' T^{-1}) (T V''^\top).$$

Making the link with (11), we obtain:

$$\begin{bmatrix} a_1 K_1 R_1 \\ \vdots \\ a_m K_m R_m \end{bmatrix} \begin{bmatrix} b_1 S_1 L_1 & \cdots & b_n S_n L_n \end{bmatrix} = (U'' T^{-1}) (T V''^\top). \quad (12)$$

Let us decompose matrices  $U''$  and  $V''$  in  $3 \times 3$  submatrices:  $U''^\top = [U_1^\top \cdots U_m^\top]$  and  $V''^\top = [V_1^\top \cdots V_n^\top]$ . Equation (12) thus becomes:

$$\begin{bmatrix} a_1 K_1 R_1 \\ \vdots \\ a_m K_m R_m \end{bmatrix} \begin{bmatrix} b_1 S_1 L_1 & \cdots & b_n S_n L_n \end{bmatrix} = \begin{bmatrix} U_1 T^{-1} \\ \vdots \\ U_m T^{-1} \end{bmatrix} \begin{bmatrix} T V_1^\top & \cdots & T V_n^\top \end{bmatrix}. \quad (13)$$

How to estimate  $T$  is explained in Section 4.4. Once a correct estimate is given, we can directly extract the matrices  $A_i = a_i K_i R_i$  and  $B_k = b_k S_k L_k$  from which, in turn, the individual rotation and calibration matrices can be recovered by Cholesky or QR-decompositions. The Cholesky decomposition of  $A_i A_i^\top$ , e.g., results in an upper triangular matrix  $M_i = a_i K_i$ . Based on the requirement  $K_{i,33} = 1$ , we can compute the unknown scale factor  $a_i$  as  $a_i = M_{i,33}$ . The calibration matrix is finally obtained as  ${}^4 K_i = \frac{1}{a_i} M_i$ .

As for the parallelepipeds, there is no constraint similar to  $K_{i,33} = 1$  on the entries of their calibration matrices  $L_k$ . Hence, we can compute them only up to the unknown scale factors  $b_k$ . This means that we can compute the *shape* of each parallelepiped, but not (yet) their *size* (or volume). In Section 6, we explain how to compute their (relative) size.

We now briefly discuss the structure and geometric signification of matrix  $T$ . Note that  $T$  actually represents the nontranslational part of a 3D affine transformation (its upper left  $3 \times 3$  submatrix). This is just another expression of the previously mentioned fact that, due to the observation of parallelepipeds, we directly have an affine reconstruction (of scene and cameras).

The matrix  $T$  can only be computed up to an arbitrary rotation and scale: For any rotation matrix  $R$  and scale factor  $s$ ,  $T' = sRT$  cannot be distinguished from  $T$  in the factorization since  $T'^{-1} T' \sim T^{-1} T$ . This ambiguity is natural and expresses the fact that the global Euclidean reference frame for the reconstruction of parallelepipeds and cameras can be chosen arbitrarily. Without loss of generality, we may thus assume that  $T$  is upper triangular. This highlights the fact that our estimation problem has only five degrees of freedom (six parameters for an upper triangular  $3 \times 3$  matrix minus one for the free scale), which can also be explained in more geometric terms: As explained previously, our problem is somewhat equivalent to self-calibration with known affine structure. The five degrees of the problem can thus be interpreted as the coefficients of the absolute conic on the plane at infinity.

### 4.4 Disambiguating the Factorization

We now deal with the estimation of the unknown transformation  $T$  appearing in (13). As will be seen below and as is often the case in self-calibration problems, it is simpler to not directly estimate  $T$ , but the symmetric and

4. In overconstrained situations, the computed calibration matrices will not, in general, exactly satisfy the constraints used for their computation. The best way of dealing with this would be a constrained nonlinear optimization.

positive definite  $3 \times 3$  matrix  $Z$  defined as:  $Z = T^T T$ . (We may observe that  $Z$  represents the absolute conic on the plane at infinity.) Once  $Z$  is estimated,  $T$  may be extracted from it using Cholesky decomposition. As described above,  $T$  is defined up to a rotation and scale, so the upper triangular Cholesky factor of  $Z$  can directly be used as the estimate for  $T$ .

The matrix  $Z$  (and, thus,  $T$ ) can be estimated in various ways, using any information about the cameras or the parallelepipeds, e.g., prior knowledge of the relative positioning of some entities. Here, we concentrate on exploiting prior information of the intrinsic parameters of cameras and parallelepipeds. In the following, we consider two types of information, first for cameras and then for parallelepipeds:

- knowledge of the actual value of some intrinsic parameter for some camera or parallelepiped,
- knowledge that two or more cameras (or parallelepipeds) have the same value for some intrinsic parameter. We also sometimes speak of “constant” intrinsic parameters.

#### 4.4.1 Using Information on Camera Intrinsic

From (13), we have:  $a_i K_i R_i = U_i T^{-1}$ . Due to the orthogonality of  $R_i$ , we get:

$$a_i^2 \underbrace{K_i K_i^T}_{\omega_i^{-1}} = U_i \underbrace{T^{-1} T^{-T}}_{Z^{-1}} U_i^T.$$

Neglecting the unknown scale factor  $a_i$  and taking the inverse of both sides of the equation, we obtain (note that the  $U_i$  are not orthogonal in general):

$$\omega_i \sim U_i^{-T} Z U_i^{-1}. \quad (14)$$

We are now ready to formulate constraints on  $Z$  based on information on the cameras' intrinsics.

**Known Values of Camera Intrinsic.** Knowing the aspect ratio and principal point coordinates of a camera  $i$  and substituting  $\omega_i$  according to (14) and (1), the following linear constraints on  $Z$  can be written:

$$\begin{aligned} \tau_i^2 (U_i^{-T} Z U_i^{-1})_{11} - (U_i^{-T} Z U_i^{-1})_{22} &= 0 \\ u_{i,0} (U_i^{-T} Z U_i^{-1})_{11} + (U_i^{-T} Z U_i^{-1})_{13} &= 0 \\ v_{i,0} (U_i^{-T} Z U_i^{-1})_{22} + (U_i^{-T} Z U_i^{-1})_{23} &= 0. \end{aligned}$$

A known value of the focal length  $\alpha_v$  can only be used to formulate linear equations if all the other intrinsics are also known. In such a fully calibrated case, other algorithms [25] might be better suited, so we neglect that case in the following.

**Constant Camera Intrinsic.** In the case when two cameras  $i$  and  $j$  are known to have the same, yet unknown value for one intrinsic parameter, we in general obtain *quadratic* equations on  $Z$ . For example, the assumption of equal aspect ratios leads to the quadratic equation:

$$(U_i^{-T} Z U_i^{-1})_{11} (U_j^{-T} Z U_j^{-1})_{22} = (U_j^{-T} Z U_j^{-1})_{11} (U_i^{-T} Z U_i^{-1})_{22}.$$

The situation is different if *all* intrinsic parameters of two (or more) views are known to be identical. In that case, we can obtain linear equations instead of quadratic ones, as shown in

[8]: The matrices  $U^i$  are first scaled such as to have unit determinant. Then, we can write the following component-wise matrix equality between any pair  $(i, j)$  of views:

$$U_i^{-T} Z U_i^{-1} - U_j^{-T} Z U_j^{-1} = 0_{3 \times 3}.$$

This represents six *linear* equations on  $Z$  for each pair of views, among which four are independent.

#### 4.4.2 Information on Parallelepipeds

From (13), we have:  $b_k S_k L_k = T V_k^T$ . Due to the orthogonality of  $S_k$ , we get:

$$b_k^2 \underbrace{L_k^T L_k}_{\mu_k} = V_k \underbrace{T^T T}_Z V_k^T.$$

Neglecting the unknown factor  $b_k$ :

$$\mu_k \sim V_k Z V_k^T.$$

Knowledge of parallelepiped intrinsic can be used in analogous ways as for camera parameters. For example, suppose we know the length ratio of two parallelepiped edges  $r_{uv} = \frac{l_u}{l_v}$ . Referring to (2), we get the following *linear* equation on  $Z$ :

$$r_{k,uv}^2 \mu_{k,vv} - \mu_{k,uu} = r_{k,uv}^2 (V_k Z V_k^T)_{vv} - (V_k Z V_k^T)_{uu} = 0.$$

Similarly, the assumption that  $\theta_{uv}$  is a right angle, i.e.,  $\cos \theta_{uv} = 0$ , also gives a *linear equation*:

$$\mu_{k,uv} = (V_k Z V_k^T)_{uv} = 0.$$

A known angle  $\theta_{uv}$  that is not a right angle does not lead to a linear, but a *bilinear* equation [44].

Like for cameras, *quadratic* equations may be derived from assumptions about two or more parallelepipeds having the same, yet unknown value for some intrinsic parameter. Also, two parallelepipeds having the same shape give a set of *linear* equations on  $Z$ , even if the parallelepipeds are of different size. Equal size of parallelepipeds gives an additional *linear* equation, but which constrains relative pose rather than intrinsic parameters.

Currently, we only exploit constraints on individual parallelepipeds (right angles and length ratios) since they are easier to provide for the user.

#### 4.5 Complete Algorithm

1. Estimate the canonical projection matrices  $\tilde{X}_{ik}$ .
2. Compute missing  $X_{ik}$ .
3. Normalize the  $X_{ik}$  to unit determinant.
4. Construct the measurement matrix and compute its SVD.
5. From the SVD, extract the matrices  $U_i$  and  $V_k$ .
6. Establish a linear equation system on  $Z$  based on prior knowledge of intrinsic parameters of cameras and parallelepipeds and solve it to least squares.
7. If  $Z$  is positive, definite extract  $T$  from  $Z$  using Cholesky decomposition.
8. Extract the  $K_i, R_i, L_k, S_k$  from the  $U_i T^{-1}$  and the  $T V_k^T$  using, e.g., QR-decomposition. Note that, at this stage, the  $L_k$  can only be recovered up to

TABLE 1  
Structures of  $\omega$  Depending on Prior Knowledge

Known camera parameters			
(A) Zero skew	(B) $\tau$	(C) $u_0, v_0$	(D) $\tau, u_0, v_0$
$\begin{pmatrix} a & 0 & d \\ 0 & b & e \\ d & e & c \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & d \\ 0 & 1 & e \\ d & e & c \end{pmatrix}$	$\begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & c \end{pmatrix}$

scale, i.e., the parallelepipeds' (relative) sizes remain undetermined.

This algorithm allows us to calibrate a set of cameras using very little prior knowledge. Indeed, as mentioned in this section, all constraints provided by knowledge of cameras and parallelepipeds can be expressed in terms of the five independent parameters of the matrix  $Z$ . Thus, to calibrate the whole system, it is in general sufficient to know values of a total of only five intrinsic parameters of cameras or parallelepipeds. That is why, in practice, we only use the associated linear equations. In most cases, they are sufficient to find a unique solution. In some minimal cases, when the available linear constraints are insufficient, quadratic equations might be used to find a unique solution or a finite set of solutions.

## 5 SINGULARITIES

Many calibration or self-calibration algorithms are subject to more or less severe singularities, i.e., there exist situations where the algorithm is bound to fail. Furthermore, even in situations that are not exactly singular, but close to a singularity, the results usually become very unstable. In this section, we examine the singularities for the linear calibration algorithm described in Section 4.5. We separately study the singularities for a parallelepiped being seen by one and multiple cameras.

### 5.1 One Parallelepiped in a Single View

We have studied all possible combinations of a priori knowledge, on both camera and parallelepiped intrinsic parameters leading to linear equations (see Sections 4.1.1 and 4.4.2). We first formulate the meaning of a singularity in terms of the ingredients of the calibration algorithm. The existence of a singularity in our case means that (4) has more than one solution for  $\omega$  and  $\mu$  conforming to all available a priori information, i.e., that there is at least one solution different from the true one. It is easy to show that the existence of a singularity does not depend on the relative *position* of the camera and the parallelepiped, only on the relative *orientation* and the a priori knowledge on camera and parallelepiped intrinsic parameters. Proofs for the following results are given in [47].

Table 1 explains the four considered cases of different prior knowledge on camera intrinsics. Table 2 shows all singularities for nearly all combinations of known camera and parallelepiped parameters. Singularities are explained in geometric terms by describing the relative orientation of the parallelepiped with respect to the camera. In the following paragraphs, we give a few comments on different cases of prior knowledge on the parallelepiped. Several singular situations that might occur in practice are illustrated in Fig. 3.

TABLE 2  
Singular Relative Orientation (One Parallelepiped in One View) for Various Combinations of Prior Knowledge

Case	Conditions for singularity
A-3-1	$\mathbf{u}$ is orthogonal to the $\mathbf{x}$ or $\mathbf{y}$ camera axis
B-3-1	$\mathbf{u}$ is parallel to the optical axis $\mathbf{z}$
C-3-1	$\mathbf{u}$ is parallel to any of the three camera axes
D-3-1	$\mathbf{u}$ is parallel to the optical axis
A-3-0	always (3 constraints for 4 camera intrinsics)
B-3-0	any parallelepiped edge is parallel to the image plane
C-3-0	any parallelepiped edge is parallel to any camera axis
D-3-0	any parallelepiped edge is parallel to the optical axis
A-2-2	no intuitive geometrical interpretation
B-2-2	$(\mathbf{u} \parallel \Pi)$ and $(\mathbf{w} \parallel \text{optical axis or } \Pi)$
C-2-2	$(\mathbf{u} \parallel \mathbf{x}$ or $\mathbf{y}$ axis and $\angle(\mathbf{w}, \Pi) = 45^\circ$ ) or $(\mathbf{u} \parallel \mathbf{z}$ and $\mathbf{w} \parallel \Pi$ and $\angle(\mathbf{w}, \mathbf{x}) = \angle(\mathbf{w}, \mathbf{y}) = 45^\circ$ ) never!
D-2-2	never!
A-2-1	always (three constraints for four camera intrinsics)
B-2-1	$\mathbf{u} \parallel \Pi$
C-2-1	$(\mathbf{u} \parallel \text{to either camera axis})$ or $(\mathbf{u}$ and $\mathbf{w} \parallel \Pi)$ or $(\mathbf{u}$ and $\mathbf{w} \perp \text{to } \mathbf{x}$ or $\mathbf{y}$ camera axis)
D-2-1	$\mathbf{u}$ and $\mathbf{w}$ are parallel to the image plane
A-2-0	always (two constraints for four camera intrinsics)
B-2-0	always (two constraints for three camera intrinsics)
C-2-0	$\mathbf{u}$ orthogonal to the $\mathbf{x}$ or $\mathbf{y}$ camera axis or $\parallel \Pi$
D-2-0	$\mathbf{u}$ parallel to image plane or $\parallel$ optical axis

Cases are denoted X-Y-Z, where  $X \in \{A, B, C, D\}$  refers to Table 1 and Y(Z) is the number of known right angles (length ratios).  $\Pi$  means the image plane.

1. *Three right angles, two length ratios (cases \*-3-2 in Table 2).* In this case, the Euclidean structure of the parallelepiped is completely given (up to scale) and it can be used as a classical calibration object. There are singularities proper to the use of a parallelepiped, but, of course, the generic singularities described in [48] apply here, too.
2. *Three right angles, one length ratio (cases \*-3-1).* In Table 2,  $\mathbf{u}$  represents the direction of the parallelepiped's edge which is not "involved" in the known length ratio.
3. *Two right angles (cases \*-2-\*).* Here, the parallelepiped can be visualized as built around two rectangles sharing an edge  $\mathbf{u}$ . In Table 2,  $\mathbf{w}$  is one of the edges not parallel to  $\mathbf{u}$ .

### 5.2 One Parallelepiped in Multiple Views

Two observations are useful to characterize the singularities in the case when one parallelepiped is seen in multiple images:



Fig. 3. Examples of singular and nonsingular configurations for calibration based on a parallelepiped with three right angles (the house's main body). Left: The parallelepiped's vertical edge is parallel to the camera's  $\mathbf{y}$ -axis; this configuration is singular if the camera aspect ratio and principal point are not given (cf. cases B-3-0 and C-3-0 in Table 2). Middle: The parallelepiped's vertical edge is parallel to the image plane; this configuration is singular if the camera's principal point is not given (case B-3-0 in Table 2). Right: A rotation of the camera as shown here removes these singularities.



- The way the canonic parallelepiped projection matrix  $\tilde{X}$  is computed implies that there is only an affine ambiguity in the calibration process. Thus, the singularities of calibration of images viewing one parallelepiped are equivalent to singularities of generic self-calibration when the plane at infinity is known.
- It is natural to suppose that the prior knowledge of the intrinsic parameters is the same for all cameras used, thus all matrices  $\omega_i$  are assumed here to belong to the same among the groups defined in Table 1.

These observations make it possible to adapt the studies on critical motions for self-calibration. In particular, we use the results presented in [49] for scenarios with a known plane at infinity. Depending on the type of knowledge about the camera (cf. Table 1), the following rotations are singular for calibration:

- Always critical with fewer than five cameras. Critical motions for five or more cameras are hard to describe.
- Critical if the cameras' optical axes point in at most two different directions.
- Critical if one axis of each camera is pointing in some common direction.
- Critical if the optical axes of all cameras point in the same direction.

Results for the cases A, B, D were given in [49] and the proof of case C is given in [47].

## 6 ESTIMATING POSITION AND SIZE

In this section, we propose an algorithm for the estimation of the (relative) positions of the cameras and parallelepipeds, as well as the (relative) sizes of the parallelepipeds. After Section 4, this concludes our complete method for intrinsic and extrinsic calibration.

Consider (5):

$$\lambda_{ik}\tilde{X}_{ik} = K_i(R_i \quad \mathbf{t}_i) \begin{pmatrix} S_k & \mathbf{v}_k \\ \mathbf{0}^\top & 1 \end{pmatrix} \tilde{L}_k.$$

The leading  $3 \times 3$  subpart of the two sides of the equation were used in Section 4 to compute the intrinsic camera parameters  $K_i$  and the rotation matrices  $R_i$  and  $S_k$ . The parallelepipeds' intrinsic parameters  $L_k$  were computed up to scale only, i.e., up to the "size" of the parallelepipeds.

Let us consider this in detail. In the following, we suppose that the matrices  $\tilde{X}_{ik}$  are already scaled such that the submatrices  $X_{ik}$  have unit determinant, as in Section 4, i.e.,  $\lambda_{ik} = 1$ . Let  $\bar{K}_i$  and  $\bar{L}_k$  be the calibration matrices scaled to unit determinant. We know all matrices in the following equation:  $X_{ik} = \bar{K}_i R_i S_k \bar{L}_k$ .

What we don't know is the size  $s_k$  of the parallelepipeds. Let us observe the following:

$$\tilde{L}_k = \begin{pmatrix} s_k \bar{L}_k & \mathbf{0} \\ \mathbf{0}^\top & 1 \end{pmatrix} \sim \begin{pmatrix} \bar{L}_k & \mathbf{0} \\ \mathbf{0}^\top & 1/s_k \end{pmatrix}.$$

We may now rewrite (5):

$$\tilde{X}_{ik} = \bar{K}_i(R_i \quad \mathbf{t}_i) \begin{pmatrix} S_k & \mathbf{v}_k \\ \mathbf{0}^\top & 1 \end{pmatrix} \begin{pmatrix} \bar{L}_k & \mathbf{0} \\ \mathbf{0}^\top & 1/s_k \end{pmatrix}.$$

Let  $\mathbf{x}_{ik}$  be the fourth column of  $\tilde{X}_{ik}$ . We have the following equation:

$$\mathbf{x}_{ik} = \bar{K}_i(R_i \quad \mathbf{t}_i) \begin{pmatrix} S_k & \mathbf{v}_k \\ \mathbf{0}^\top & 1 \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ 1/s_k \end{pmatrix} = \frac{1}{s_k} \bar{K}_i(R_i \mathbf{v}_k + \mathbf{t}_i).$$

From this, we get an equation that is linear in all unknowns ( $s_k$ ,  $\mathbf{t}_i$ , and  $\mathbf{v}_k$ ):

$$s_k \mathbf{x}_{ik} - \bar{K}_i R_i \mathbf{v}_k - \bar{K}_i \mathbf{t}_i = \mathbf{0}.$$

The unknowns can be computed using linear least squares: minimizing the sum of the squared  $L_2$  norms of the vectors on the left hand side of the above equation, over all camera-parallelepiped pairs. The estimates for the  $s_k$ ,  $\mathbf{t}_i$ , and  $\mathbf{v}_k$  are, of course, defined up to a single global scale. Note that, at this stage, missing data are not an issue any more, contrary to the computations in Section 4.

## 7 3D RECONSTRUCTION

The presented calibration approach is well adapted to interactive 3D reconstruction from a few images. It has a major advantage over other methods: simplicity. Indeed, only a small amount of user interaction is needed for both calibration and reconstruction: A few points must be picked in the image to define the primitives' image positions. It thus seems to be an efficient and intuitive way to build models from images of any type, in particular from images taken from the Internet for which no information about the camera is known. For the reconstruction of points not belonging to the parallelepiped, we use an iterative approach described in [34]. This approach is actually independent from the calibration method, although it uses the same input in the first step. It allows us to propagate the information on points, lines, and planes defining the model. All the accessible information is processed simultaneously. Interestingly, it allows 3D models to be computed from nonoverlapping photographs (see, e.g., Fig. 10).

The following section illustrates this approach with results obtained by solving linear systems only. Note that, in order to refine the results, nonlinear optimization, taking into account prior information, might be applied.

## 8 EXPERIMENTAL RESULTS

### 8.1 Synthetic Data

The main goal of our experiments with synthetic data is to study the performance of the calibration algorithm in the proximity of singular configurations. In this paper, we only report on experiments on the minimal case: One parallelepiped seen in one camera. Additional experiments, evaluating calibration results in the proximity of singular positions with respect to different numbers of parallelepipeds, as well as different types of prior information, are described in [47].

Tests were performed with synthetic  $600 \times 400$  images, taken by a camera with the following intrinsic parameters:

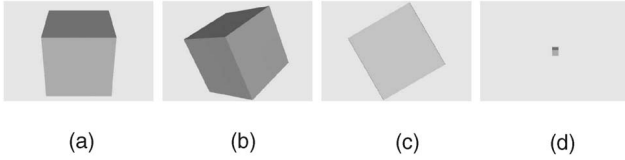


Fig. 4. Parallelepiped orientations in the experiment. (a) Initial orientation ( $x$  axes of parallelepiped and camera are parallel), (b) intermediate orientation, (c) final orientation (the  $y$  and  $z$  axes of the parallelepiped are parallel to the image plane), (d) minimal parallelepiped size.

$(\alpha_u, \alpha_v, s, u_0, v_0) = (1, 000, 1, 000, 0, 300, 200)$ . Parallelepiped parameters were varying over the different tests. The most important parameter of the experiments is the relative orientation parallelepiped-camera. For a given orientation, six parallelepiped vertices were projected into the images and random Gaussian noise was added to image points (for the presented results, noise was of standard deviation 1 pixel). For a given setting (relative orientation, standard deviation of noise, etc.), 100 such data sets were created randomly and used as input for calibration. Calibration was considered to have failed if any of the estimated matrices  $\omega$  or  $\mu$  were not positive definite (in that case,  $K$  or  $L$  cannot be retrieved). We give results by indicating the number of failures, as well as median values for estimated parameters (computed using valid calibration results only). Prior information used was: The parallelepiped has only right angles and known camera parameters were  $(s, \tau) = (0, 1)$  (i.e., case B-3-0 in Section 5). This is one of the minimal cases for the calibration.

Tests were performed for different orientations of the parallelepiped, as illustrated in Fig. 4. Orientation varies continuously from that shown in Fig. 4a ( $x$  axes of parallelepiped and camera are parallel) to that of Fig. 4c (the  $y$  and  $z$  axes of the parallelepiped are parallel to the image plane). The continuous rotation between the two positions is parameterized by an angle ranging from  $0^\circ$  (Fig. 4a) to  $90^\circ$  (Fig. 4c), see the horizontal axis of the graphs in Fig. 5. According to Section 5, both extremal orientations are singular. We also varied the size of the parallelepiped: Maximal and minimal sizes are shown in Fig. 4a and Fig. 4d, respectively.

The results of the calibration method described here are compared to calibration based on vanishing points [19]; vanishing points were determined using the method [50]. In the case tested here, both methods use the same constraints (three right angles); the difference is that [19] uses individually estimated vanishing points, whereas our method, via the estimation of the canonical projection matrix, accounts for the fact that the three vanishing points of a parallelepiped are not fully independent.

Fig. 5 shows the number of successful calibrations, the median of estimated values for  $\alpha_v$ , and errors on the estimated pose (median of errors on the angle of the estimated rotational part of pose). Results are shown with respect to the orientation of the parallelepiped (horizontal axis of graphs, see above) and its size (vertical axis, unit is coverage of image by parallelepiped, from 5 percent to 50 percent).

Results are shown for both the parallelepiped-based approach (first column) and the vanishing point approach

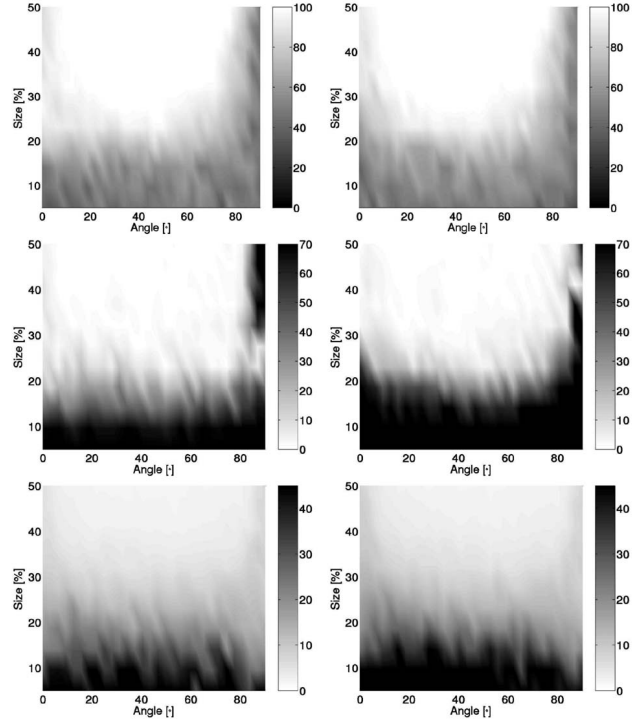


Fig. 5. Calibration results as a function of the size and relative camera-parallelepiped rotation angle. The two columns correspond to parallelepiped-based and vanishing point-based calibration, respectively. From top to bottom, the graphs show: 1) the number of successful calibrations, 2) median values of relative errors on the estimation of  $\alpha_v$ , 3) median rotation errors.

(second column). The effect of singular cases is clearly visible in the upper two rows: Calibration often fails for orientations within  $10^\circ$  of the singular ones. However, for the intermediate range of orientations, the relative error of calibration is smaller than 5 percent for both methods, when the parallelepiped covers more than 20 percent of the image. Results in case of successful calibration are slightly better for the parallelepiped method.

## 8.2 Results on Real Scenes

We present 3D reconstruction results of our methods for indoor and outdoor scenes. These examples correspond to situations where automatic methods are bound to fail: Small sets of images are used and occlusions are frequent. Each reconstruction was performed in two steps: First, one or more parallelepipeds were used to calibrate the intrinsic and extrinsic camera parameters; second, scene points and geometric constraints were used for the reconstruction (cf. Section 7). Results from single as well as multiple images are shown.

### 8.2.1 Kitchen Scene

Fig. 6a shows the original image used for the 3D reconstruction. The image was taken with a small focal length, leading, therefore, to a slight optical distortion which was not corrected here. Calibration was based on the cupboard in the central part of the image. Note that the camera was almost frontoparallel with respect to the cupboard and, thus, close to a singular situation. Prior information used for calibration was: right parallelepiped angles, zero camera skew, and

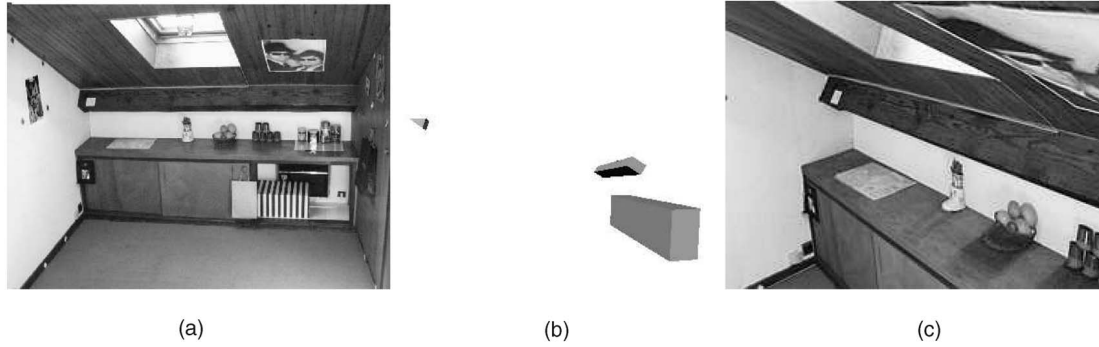


Fig. 6. Kitchen scene: (a) The original image, (b) the modeled parallelepipeds and camera pose, (c) textured model seen from a different viewpoint.

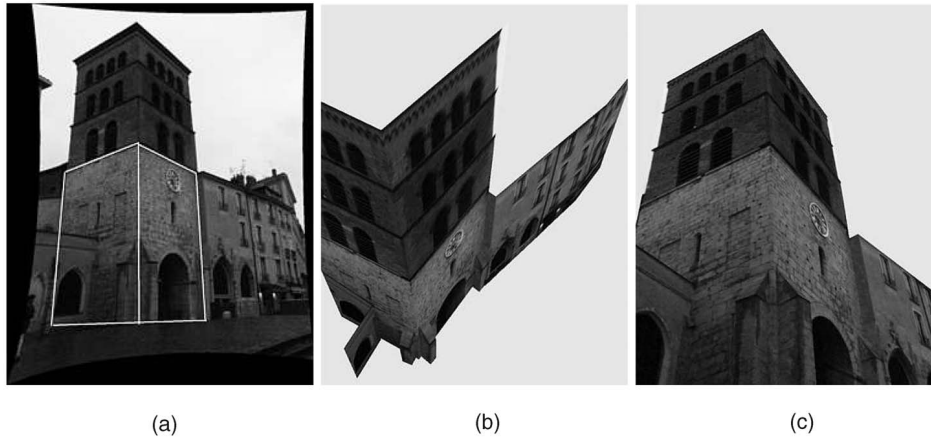


Fig. 7. Notre Dame square scene: (a) The original image, (b) and (c) screen-shots of the model obtained using the image on the left only.

principal point in the image center. The full model was reconstructed using 29 points, constrained by two parallelepipeds (the cupboard and the wooden belt), three parallelograms, and six collinearity and coplanarity constraints. Fig. 6b shows the reconstructed parallelepipeds as well as the camera pose. Due to the fact that a single image was used, only the shape of the parallelepipeds can be reconstructed at first, but not their relative size. This was then done using additional constraints such as coplanarity, which were also used to reconstruct the positions of additional points. Fig. 6c shows the complete textured model rendered from a new viewpoint.

### 8.2.2 Notre Dame Square Scene

In this section, we present reconstructions of the Notre Dame Square in Grenoble. Here, radial image distortion was corrected offline.

**Reconstruction from one image.** The image and the calibration parallelepiped are shown in Fig. 7a. Prior information used for calibration as: right parallelepiped angles, zero camera skew, and principal point in the image center. The final model is composed of 42 points, three parallelepipeds, four parallelograms, and four lines and planes. Rendered views of the model are shown in Fig. 7.

**Reconstruction from multiple images.** The sequence used for the reconstruction is composed of 15 images whose sizes vary from  $768 \times 1,024$  to  $960 \times 1,280$  pixels. Calibration was based on three parallelepipeds (shown in Fig. 8a, Fig. 8b, Fig. 8c, Fig. 8d). Prior information used was: right angles for parallelepipeds 1 and 2, zero camera skew, unit aspect ratios

and centered principal points for all images. Parallelepiped 3 is relatively small in those images where both parallelepipeds 2 and 3 appear. Consequently, the estimation of its vertices is unstable and, thus, information about its intrinsic parameters was not used for calibration. Calibration was performed in two steps. First, the proposed linear factorization approach was applied. Second, the parameters of cameras and parallelepipeds obtained from the previous step were nonlinearly optimized by minimizing the reprojection error of vertices in a bundle adjustment.

Then, scene elements were added and reconstructed so that the final model is composed of 194 points, 19 planes, and 25 lines. The mean reprojection error over all the model points was about 8 pixels (only linear methods were used for reconstruction). As expected, the largest errors occurred in images calibrated using parallelepiped 3.

For comparison, an unconstrained bundle adjustment, using the Levenberg-Marquardt optimization method, was performed over all the model points and the camera focal lengths. This reduced the reprojection error to 2 pixels. It did not, however, reduce the small artifacts occurring in the final model.

The calibration primitives and cameras reconstructed using the factorization method, the parallelepiped-based nonlinear optimization, and the point-based nonlinear optimization are shown, respectively, in Fig. 8e, Fig. 8f, Fig. 8g. Rendered views of the model reconstructed using the parallelepiped-based calibration are shown in Fig. 8h and Fig. 8j.

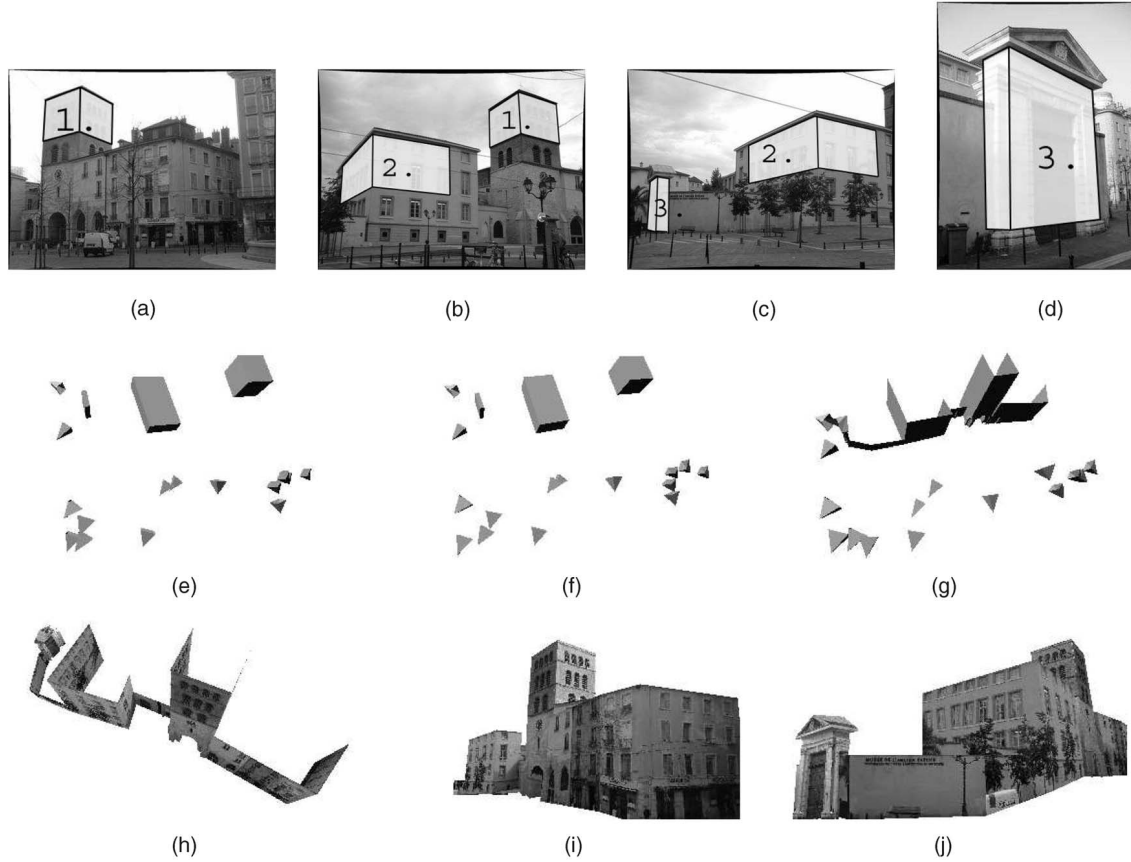


Fig. 8. Notre Dame Square scene: (a)-(d) Four images from the 15 used for the reconstruction. Parallelepipeds used for the reconstruction are marked in white. (e) Cameras and parallelepipeds as estimated by the proposed linear factorization method. (f) Camera and parallelepiped parameters after nonlinear optimization. (g) Cameras and 194 model points optimized by an unconstrained nonlinear method. (h)-(j) Synthetic viewpoints of the textured model.

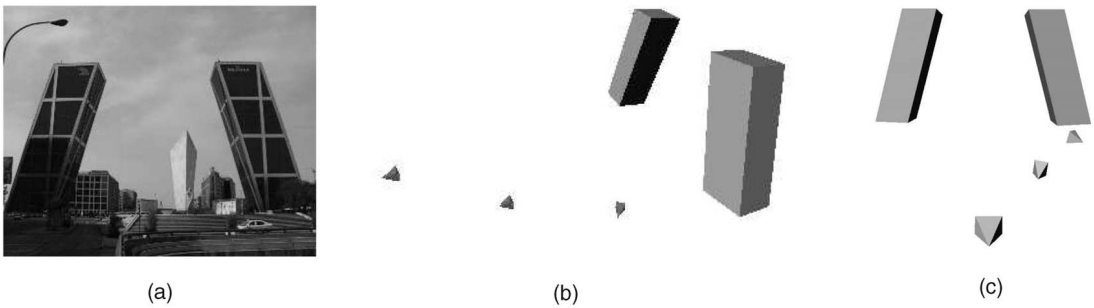


Fig. 9. Kio towers in Madrid: (a) The original image; (b) and (c) reconstructed model and camera poses.

8.2.3 Kio Towers

Reconstruction was based on three images and two calibration primitives. One of the images used for the reconstruction is shown in Fig. 9a. Information used for calibration was: two right angles in each tower, zero camera skew, unit aspect ratio, and centered principal point. The reconstructed cameras and primitives are shown in Fig. 9.

8.2.4 Opposite Viewpoints Scene

Fig. 10 shows the reconstruction of a modern building from two images taken from completely opposite viewpoints. The parallelepiped used for calibration and the estimated cameras' positions are shown in the two original images (Fig. 10a and Fig. 10b). In the first image, intersections of

lines were computed to obtain the six points required to define a parallelepiped (see Fig. 10a). The parallelepiped and the cameras reconstructed by the factorization algorithm are shown in Fig. 10c. New viewpoints of the whole model, composed of 32 points, 13 parallelograms, and six planes, are shown in Fig. 10d, Fig. 10e, and Fig. 10f.

9 CONCLUSION

We have presented an approach for calibration, pose estimation, and 3D model acquisition from several uncalibrated images based on user-provided geometric constraints on the scene. Useful constraints, such as parallelism, coplanarity, and right angles, can often be

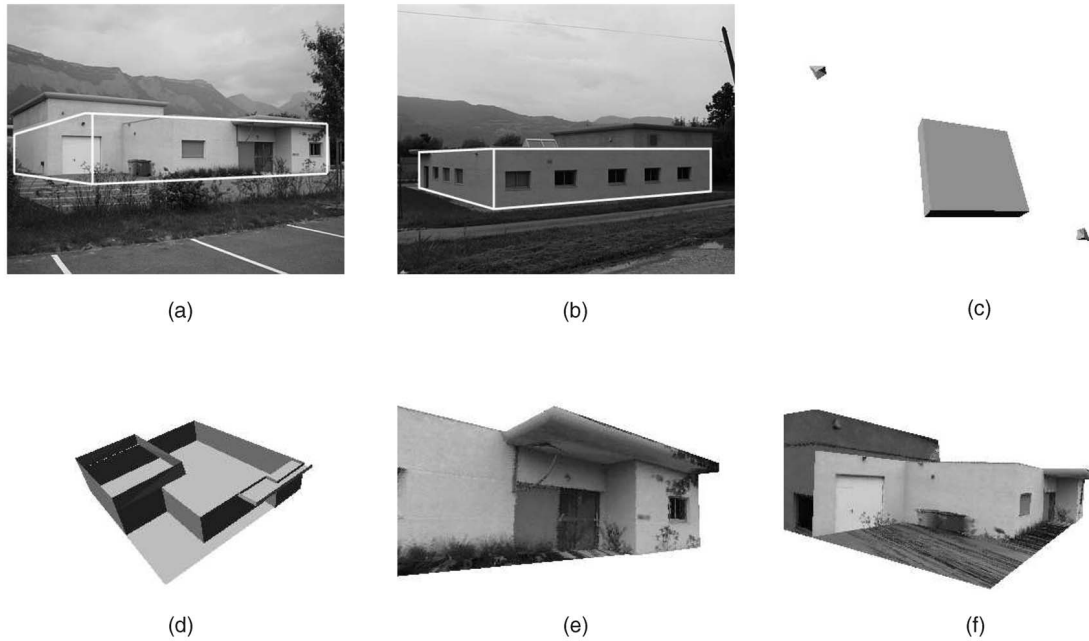


Fig. 10. Opposite viewpoints scene: (a), (b) The original images used for the reconstruction. (c) The reconstruction scenario with the computed model and the cameras' positions. (d), (e), (f) New viewpoints of the model.

nicely modeled via parallelepipeds. Especially, this allows us to couple together constraints between several neighboring scene primitives (points, lines, planes), which potentially brings about higher stability than only using constraints between pairs of primitives. The projections of parallelepipeds already encode the affine structure of the scene. Metric information (length ratios and angles) is then combined with prior information on camera parameters in a self-calibration type approach, performing complete calibration and pose estimation. This is formulated in a factorization framework. The usual problems of missing data and unknown scale factors are dealt with relatively easily and a satisfying solution can already be obtained with a small number of images and correspondences (starting from four correspondences per image pair or six per image and parallelepiped).

A detailed study on singular cases of this approach is also presented: Singularities are derived theoretically and the impact on the method's performance due to the proximity to singular configurations is shown by simulated experiments. Experiments with real images show that our calibration approach gives excellent initial results for general 3D model reconstruction methods.

We believe that an approach such as the one presented here is a useful tool for easily calibrating cameras using images of unknown though constrained scenes. Also, it allows us to efficiently obtain models of the global structure of scenes (including camera pose), which are good starting points for more automatic reconstruction methods.

## ACKNOWLEDGMENTS

This work was partially supported by the European project VISIRE (IST-1999-10756).

## REFERENCES

- [1] O. Faugeras, "What Can Be Seen in Three Dimensions with an Uncalibrated Stereo Rig?" *Proc. European Conf. Computer Vision*, pp. 563-578, 1992.
- [2] R. Tsai, "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 364-374, 1986.
- [3] S. Maybank and O. Faugeras, "A Theory of Self Calibration of a Moving Camera," *Int'l J. Computer Vision*, vol. 8, no. 2, pp. 123-151, 1992.
- [4] B. Triggs, "Autocalibration and the Absolute Quadric," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 609-614, 1997.
- [5] R. Hartley, "Euclidean Reconstruction from Uncalibrated Views," *Proc. DARPA-ESPRIT Workshop Applications of Invariants in Computer Vision*, pp. 187-202, 1993.
- [6] M. Pollefeys and L. Van Gool, "A Stratified Approach to Metric Self-Calibration," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 407-412, 1997.
- [7] P. Sturm, "Critical Motion Sequences for Monocular Self-Calibration and Uncalibrated Euclidean Reconstruction," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 1100-1105, 1997.
- [8] R.I. Hartley, "Self-Calibration of Stationary Cameras," *Int'l J. Computer Vision*, vol. 22, no. 1, pp. 5-23, 1997.
- [9] L. de Agapito, R. Hartley, and E. Hayman, "Linear Self-Calibration of a Rotating and Zooming Camera," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 15-21, 1999.
- [10] M. Armstrong, A. Zisserman, and P. Beardsley, "Euclidean Structure from Uncalibrated Images," *Proc. British Machine Vision Conf.*, vol. 2, pp. 509-518, 1994.
- [11] A. Zisserman, D. Liebowitz, and M. Armstrong, "Resolving Ambiguities in Auto-Calibration," *Philosophical Trans. Royal Soc. London, Series A*, vol. 356, no. 1740, pp. 1193-1211, 1998.
- [12] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge Univ. Press, 2000.
- [13] C. Tomasi and T. Kanade, "Shape and Motion from Image Streams under Orthography: A Factorization Method," *Int'l J. Computer Vision*, vol. 9, no. 2, pp. 137-154, 1992.
- [14] P. Sturm and B. Triggs, "A Factorization Based Algorithm for Multi-Image Projective Structure and Motion," *Proc. European Conf. Computer Vision*, pp. 709-720, 1996.
- [15] B. Triggs, "Factorization Methods for Projective Structure and Motion," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 845-851, 1996.

- [16] D. Martinec and T. Pajdla, "Structure from Many Perspective Images with Occlusions," *Proc. European Conf. Computer Vision*, pp. 355-369, 2002.
- [17] C. Rother, S. Carlsson, and D. Tell, "Projective Factorization of Planes and Cameras in Multiple Views," *Proc. Int'l Conf. Pattern Recognition*, pp. 737-740, 2002.
- [18] P. Sturm, "Algorithms for Plane-Based Pose Estimation," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 1010-1017, 2000.
- [19] B. Caprile and V. Torre, "Using Vanishing Points for Camera Calibration," *Int'l J. Computer Vision*, vol. 4, pp. 127-140, 1990.
- [20] R. Cipolla and E. Boyer, "3D Model Acquisition from Uncalibrated Images," *Proc. IAPR Workshop Computer Vision*, pp. 559-568, 1998.
- [21] C. Chen, C. Yu, and Y. Hung, "New Calibration-Free Approach for Augmented Reality Based on Parameterized Cuboid Structure," *Proc. Int'l Conf. Computer Vision*, pp. 30-37, 1999.
- [22] J. Kosecka and W. Zhang, "Video Compass," *Proc. European Conf. Computer Vision*, pp. 476-491, 2002.
- [23] D. Liebowitz and A. Zisserman, "Combining Scene and Auto-Calibration Constraints," *Proc. Int'l Conf. Computer Vision*, pp. 293-300, 1999.
- [24] D. Bondyfalat, T. Papadopoulou, and B. Mourrain, "Using Scene Constraints during the Calibration Procedure," *Proc. Int'l Conf. Computer Vision*, pp. 124-130, 2001.
- [25] P. Sturm and S. Maybank, "On Plane-Based Camera Calibration: A General Algorithm, Singularities, Applications," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 432-437, 1999.
- [26] Z. Zhang, "Flexible Camera Calibration by Viewing a Plane from Unknown Orientations," *Proc. Int'l Conf. Computer Vision*, pp. 666-673, 1999.
- [27] B. Triggs, "Autocalibration from Planar Scenes," *Proc. European Conf. Computer Vision*, pp. 89-105, 1998.
- [28] E. Malis and R. Cipolla, "Camera Self-Calibration from Unknown Planar Structures Enforcing the Multi-View Constraints between Collineations," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, Sept. 2002.
- [29] B. Boufama, R. Mohr, and F. Veillon, "Euclidean Constraints for Uncalibrated Reconstruction," *Proc. Int'l Conf. Computer Vision*, pp. 466-470, 1993.
- [30] A. Criminisi, I.D. Reid, and A. Zisserman, "Single View Metrology," *Int'l J. Computer Vision*, vol. 40, no. 2, pp. 123-148, 2000.
- [31] P. Sturm and S. Maybank, "A Method for Interactive 3D Reconstruction of Piecewise Planar Objects from Single Images," *Proc. British Machine Vision Conf.*, pp. 265-274, 1999.
- [32] H.-Y. Shum, M. Han, and R. Szeliski, "Interactive Construction of 3D Models from Panoramic Mosaics," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 427-433, 1998.
- [33] E. Grossmann and J.S. Victor, "Single and Multi-View Reconstruction of Structured Scenes," *Proc. Asian Conf. Computer Vision*, pp. 93-104, 2002.
- [34] M. Wilczkowiak, P. Sturm, and E. Boyer, "The Analysis of Ambiguous Solutions in Linear Systems and Its Application to Computer Vision," *Proc. British Machine Vision Conf.*, pp. 53-62, 2003.
- [35] C. McGlone, "Bundle Adjustment with Object Space Geometric Constraints for Site Modeling," *Proc. SPIE Conf. Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision II*, pp. 25-36, 1995.
- [36] R. Szeliski and P. Torr, "Geometrically Constrained Structure from Motion: Points on Planes," *Proc. SMILE Workshop*, pp. 171-186, 1998.
- [37] P. McLauchlan, X. Shen, A. Manassis, P. Palmer, and A. Hilton, "Surface-Based Structure-from-Motion Using Feature Groupings," *Proc. Asian Conf. Computer Vision*, pp. 699-705, 2000.
- [38] E. Grossmann and J. Santos-Victor, "Dual Representations for Vision-Based 3D Reconstruction," *Proc. British Machine Vision Conf.*, pp. 516-526, 2000.
- [39] D. Bondyfalat and S. Bougnoux, "Imposing Euclidean Constraints during Self-Calibration Processes," *Proc. SMILE Workshop*, pp. 224-235, 1998.
- [40] M. Wilczkowiak, G. Trombettoni, C. Jermann, P. Sturm, and E. Boyer, "Scene Modeling Based on Constraint System Decomposition Techniques," *Proc. Int'l Conf. Computer Vision*, pp. 1004-1010, 2003.
- [41] P. Debevec, C. Taylor, and J. Malik, "Modeling and Rendering Architecture from Photographs: A Hybrid Geometry and Image-Based Approach," *Proc. SIGGRAPH*, pp. 11-20, 1996.
- [42] D. Jelinek and C. Taylor, "Reconstruction of Linearly Parameterized Models from Single Images with a Camera of Unknown Focal Length," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 7, pp. 767-774, July 2000.
- [43] A. Dick, P. Torr, S. Ruffle, and R. Cipolla, "Combining Single View Recognition and Multiple View Stereo for Architectural Scenes," *Proc. Int'l Conf. Computer Vision*, pp. 268-274, 2001.
- [44] M. Wilczkowiak, E. Boyer, and P. Sturm, "Camera Calibration and 3D Reconstruction from Single Images Using Parallelepipeds," *Proc. Int'l Conf. Computer Vision*, pp. 142-148, 2001.
- [45] M. Wilczkowiak, E. Boyer, and P. Sturm, "3D Modelling Using Geometric Constraints: A Parallelepiped Based Approach," *Proc. European Conf. Computer Vision*, pp. 221-236, 2002.
- [46] M. Pollefeys, L. Van Gool, and M. Proesmans, "Euclidean 3D Reconstruction from Image Sequences with Variable Focal Lengths," *Proc. European Conf. Computer Vision*, pp. 31-42, 1996.
- [47] M. Wilczkowiak, P. Sturm, and E. Boyer, "Using Geometric Constraints through Parallelepipeds for Calibration and 3D Modelling," INRIA, Rapport de Recherche 5055, 2003.
- [48] T. Buchanan, "The Twisted Cubic and Camera Calibration," *Computer Vision, Graphics, and Image Processing*, vol. 42, no. 1, pp. 130-132, 1988.
- [49] F. Kahl, B. Triggs, and K. Åström, "Critical Motions for Auto-Calibration when Some Intrinsic Parameters Can Vary," *J. Math. Imaging and Vision*, vol. 13, no. 2, pp. 131-146, 2000.
- [50] D. Liebowitz and A. Zisserman, "Metric Rectification for Perspective Images of Planes," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 482-488, 1998.



**Marta Wilczkowiak** received the PhD degree from INPG (National Polytechnical Institute of Grenoble, France) in 2004 after receiving the MSc degree in engineering from Warsaw University of Technology in 2000 (first class grade). She is currently a research associate at the University of Cambridge, United Kingdom, in the Computer Vision and Robotics Group. Her research interests are in image-based 3D modeling, including camera (self-)calibration, 3D reconstruction, and, more recently, texture description for region filling and classification.



**Peter Sturm** received the PhD degree from INPG (National Polytechnical Institute of Grenoble, France) in 1997, after receiving the MSc degrees from INPG and the Technical University of Karlsruhe, both in 1994. After a two-year postdoctoral at Reading University, he joined INRIA as a senior researcher in 1999. He has acted as a program committee member for ICCV, CVPR, ECCV, ICIP, ICPR, ACCV, and several other conferences and (co)authored more than 70 scientific publications. He received the SPECIF award for 1998 for his PhD thesis (given to one French PhD thesis in computer science per year). His current main research topics are in computer vision, specifically, related to camera (self-)calibration, 3D reconstruction, and motion estimation, both for traditional perspective cameras and omnidirectional sensors.



**Edmond Boyer** received the MS degree from the Ecole Nationale Supérieure de l'Electronique et de ses Applications, France, in 1993 and the PhD degree in computer science from the Institut National Polytechnique de Lorraine, France, in 1996. He has been with the Institut National de la Recherche en Informatique et Automatique (INRIA) since 1993. He is currently an assistant professor in computer science at the University Joseph Fourier, Grenoble, France, and a researcher at INRIA Rhône-Alpes, Grenoble. His research interests include 3D modeling, silhouettes, motion capture, motion recognition, and multiple camera environments.