3D Modelling Using Geometric Constraints: A Parallelepiped Based Approach

Marta Wilczkowiak, Edmond Boyer, and Peter Sturm

MOVI–GRAVIR–INRIA Rhône-Alpes, 38330 Montbonnot, France,
Firstname.Surname@inrialpes.fr,
http://www.inrialpes.fr/movi/people/Surname

Abstract. In this paper, efficient and generic tools for calibration and 3D reconstruction are presented. These tools exploit geometric constraints frequently present in man-made environments and allow camera calibration as well as scene structure to be estimated with a small amount of user interactions and little *a priori* knowledge. The proposed approach is based on primitives that naturally characterize rigidity constraints: parallelepipeds. It has been shown previously that the intrinsic metric characteristics of a parallelepiped are dual to the intrinsic characteristics of a perspective camera. Here, we generalize this idea by taking into account additional redundancies between multiple images of multiple parallelepipeds. We propose a method for the estimation of camera and scene parameters that bears strong similarities with some self-calibration approaches. Taking into account prior knowledge on scene primitives or cameras, leads to simpler equations than for standard self-calibration, and is expected to improve results, as well as to allow structure and motion recovery in situations that are otherwise under-constrained. These principles are illustrated by experimental calibration results and several reconstructions from uncalibrated images.

1 Introduction

This paper is about using partial information on camera parameters and scene structure, to simplify and enhance structure from motion and (self-) calibration. We are especially interested in reconstructing man-made environments for which constraints on the scene structure are usually easy to provide. Constraints such as parallelism, orthogonality or length ratios, are often available, and we show that they are especially powerful if they can be encapsulated in higher-level primitives. Concretely, man-made environments are rich in elements that may be defined as parallelepipeds or parallelograms. Encoding constraints using these, reinforces them, simplifies the user interactions necessary to provide them, and allows an easier analysis of issues such as the existence of solutions.

In the following, we briefly review some existing works that use constraints on scene structure. Geometric information about the scene can be used in many different ways. In a seminal work, Caprile and Torre [2] used cuboids, i.e. parallelepipeds with right angles, to estimate some camera parameters. Their approach is based on vanishing points defined by the cuboid's projected edges. Such

A. Heyden et al. (Eds.): ECCV 2002, LNCS 2353, pp. 221–236, 2002.

vanishing points correspond to perpendicular directions in space and therefore impose constraints on the transformation between 3D space and the image plane. Following this idea, several approaches making use of vanishing points and lines, have been proposed to either partially calibrate cameras or reconstruct scenes [4,14,5,12]. However, dealing with individual vanishing points does not allow to fully exploit the redundancy contained in the input, i.e. that all the vanishing points stem from the projection of a single parallelepiped. In contrast to the above mentioned approaches, we do not compute vanishing points or lines explicitly, but projection matrices such that a parallelepiped's projection fits the concerned image points. In [7], different kinds of primitives, including cubes, are used for 3D reconstruction. However, this approach, cast as a bundle adjustment, has only a very simple initialization step. In [3], parallelepipeds are used for calibration in AR applications. The proposed approach has a limited application domain since the camera must satisfy a strong constraint – unit aspect ratio – and only partial knowledge on the parallelepiped – angles – is used.

In addition to using scene constraints, our approach also takes into account any constraint on intrinsic parameters of any camera involved. There is a perfect duality between the intrinsic parameters of a perspective camera and those of a parallelepiped [20]. We formalize this in a generic framework, in which cameras and parallelepipeds are treated in a dual manner. One of the main issues of this paper is the use of multi-view and multi-parallelepiped constraints, as opposed to using knowledge on single images or parallelepipeds only. Multi-view or multi-parallelepiped constraints arise when it is known that several entities share properties (e.g. two views with identical aspect ratio, or two parallelepipeds with identical shape). These constraints are incorporated in our dual framework. For the various types of constraints, it is shown which types of equations can be obtained. In many practical circumstances, (self-) calibration of intrinsic cameras and parallelepiped parameters can be done by solving linear equation systems.

Our approach has some similarities with methods based on planar patterns and homographies [18,21]. While more flexible than standard calibration techniques, homography based approaches require either Euclidean information or, for self-calibration, many images in general position [19]. The approach presented in this paper works for a small number of images (a single image for example) and can make use of any metric information on calibration primitives *independently*, for example one angle between directions, or one length ratio, give additional constraints. In this sense, our approach is a generalization of plane-based methods with metric information to three-dimensional parallelepipedic patterns.

While the main contributions of the paper concern the estimation of intrinsic camera and parallelepiped parameters, we also propose methods for subsequent pose estimation, as well as ways of enhancing reconstructions with primitives other than parallelepipeds. The complete system allows calibration and 3D model acquisition from a small number of arbitrary images, taken for instance from the Internet, with a reasonable amount of user interaction.

The paper is organized as follows. Section 2 gives definitions and some background. Calibration using parallelepipeds is studied in section 3. Sections 4 and 5 describe our approaches for pose estimation and 3D reconstruction.

2 Preliminaries

2.1 Camera Parameterization

We assume that all cameras can be described by the pinhole model. The projection from a 3D point P to the 2D image point p is expressed by: $p \sim M \cdot P$, where M is a 3x4 matrix, which can be decomposed as: $M = K \cdot [R\ t]$. The 3×4 matrix $[R\ t]$ encapsulates the relative orientation R and translation t between the camera and the world coordinate system. The matrix K is the 3×3 calibration matrix containing the camera's intrinsic parameters:

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

where α_u and α_v stand for the focal length, expressed in horizontal and vertical pixel dimensions, s is a skew parameter considered here as equal to zero, and u_0, v_0 are the pixel coordinates of the principal point. In the following, we will also use the IAC (image of the absolute conic) representation of the intrinsic parameters, namely the matrix $\omega \sim K^{-T} \cdot K^{-1}$.

2.2 Parallelepiped Parameterization

A parallelepiped is defined by twelve 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 the three angles between parallelepiped edges ($\theta_{12}, \theta_{23}, \theta_{13}$). The internal characteristics of a parallelepiped can be expressed by a matrix $\tilde{\Lambda}$ which represents the transformation between a canonic cube and the shape of the concerned parallelepiped (see [20] for details):

$$\tilde{\Lambda} = \begin{pmatrix} l_1 & l_2 c_{12} & l_3 c_{13} & 0 \\ 0 & l_2 s_{12} & l_3 \frac{c_{23} - c_{13} c_{12}}{s_{12}} & 0 \\ 0 & 0 & l_3 \sqrt{\frac{s_{12}^2 - c_{13}^2 s_{12}^2 - (c_{23} - c_{13} c_{12})^2}{s_{12}^2}} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

where $s_{ij} = \sin\theta_{ij}$ and $c_{ij} = \cos\theta_{ij}$. This matrix encodes all Euclidean characteristics of the parallelepiped. The analogous entity to a camera's IAC ω, is the matrix μ, defined by:

$$\mu \sim \Lambda^T \cdot \Lambda \sim \begin{pmatrix} l_1^2 & l_1 l_2 \cos\theta_{12} & l_1 l_3 \cos\theta_{13} \\ l_1 l_2 \cos\theta_{12} & l_2^2 & l_2 l_3 \cos\theta_{23} \\ l_1 l_3 \cos\theta_{13} & l_2 l_3 \cos\theta_{23} & l_3^2 \end{pmatrix},$$

where Λ is the upper left 3×3 matrix of $\tilde{\Lambda}$.

2.3 One Parallelepiped in a Single View

In this section, we briefly recall the duality between the intrinsic characteristics of a camera and those of a parallelepiped, as established in [20]. Without loss of generality, suppose that the scene coordinate system is attached to a parallelepiped. The projection of the parallelepiped is then defined by the relative pose $[R\ t]$ between the parallelepiped and the camera, the camera calibration matrix K and the parallelepiped's intrinsic parameters, given by $\tilde{\Lambda}$. In the following, we denote by C the matrix whose columns $C_{i,i\in[1..8]}$ are the homogeneous coordinates of a canonic cube's vertices. Thus, image projections $p_{i\in[1..8]} = [u_i\ v_i\ 1]$ of the parallelepiped's vertices satisfy:

$$\begin{pmatrix} \alpha_1 u_1 & \dots & \alpha_8 u_8 \\ \alpha_1 v_1 & \dots & \alpha_8 v_8 \\ \alpha_1 & \dots & \alpha_8 \end{pmatrix} = \tilde{X} \cdot C = \tilde{X} \cdot \begin{pmatrix} 1 & \dots & -1 \\ 1 & \dots & -1 \\ 1 & \dots & -1 \\ 1 & \dots & 1 \end{pmatrix}, \tag{1}$$

where the 3×4 matrix $\tilde{X}$, is defined up to a scalar factor by:

$$\tilde{X} \sim M \cdot \tilde{\Lambda} \sim K \cdot [R\ t] \cdot \tilde{\Lambda}. \tag{2}$$

We call this matrix the *canonic projection matrix*. Five image points and one image direction are, in general, sufficient to compute its eleven independent entries. Additional points make the computation more stable.

For further derivations, let us also introduce the leading 3×3 sub-matrices X and Λ of $\tilde{X}$, $\tilde{\Lambda}$ respectively, such that:

$$X \sim K \cdot R \cdot \Lambda. \tag{3}$$

The matrix X captures all geometric information given by the projection of a parallelepiped. From equation (3), it is simple to derive the following relation (using $\omega = K^{-T} \cdot K^{-1}$ and $\mu = \Lambda^T \cdot \Lambda$):

$$X^T \cdot \omega \cdot X \sim \mu. \tag{4}$$

This equation establishes the duality between the intrinsic characteristics of a camera and those of a parallelepiped. The matrix X, which can be determined from image information only, transforms one set of intrinsics into the other. Thus, ω and μ can be seen as different representations of the IAC.

2.4 n Parallelepipeds in m Views

We now consider the more general case where n parallelepipeds are seen by m cameras. In the scene coordinate frame (chosen arbitrarily), the coordinates P_{kl} of the lth vertex ($l \in [1..8]$) of the kth parallelepiped are:

$$P_{kl} = \begin{pmatrix} S_k & v_k \\ 0^T & 1 \end{pmatrix} \cdot \tilde{\Lambda}_k \cdot C_l, \tag{5}$$

where S_k and v_k stand for the rotation and translation of the kth parallelepiped. In a way similar to the single view case (cf. equation (1)), the projections $p_{ikl,l\in[1..8]}$ of these vertices in the i^{th} view are defined by:

$$p_{ikl} \sim \; K_i \cdot [R_i \; t_i] \cdot P_{kl} \sim K_i \cdot [R_i \cdot S_k \quad R_i \cdot v_k + t_i] \cdot \tilde{\Lambda}_k \cdot C_l. \tag{6}$$

Denoting $\tilde{X}_{ik}$ the canonic projection matrix of the kth parallelepiped into the ith view, we have: $X_{ik} \sim K_i \cdot R_i \cdot S_k \cdot \Lambda_k$. Thus, using: $Y_{ik} = X_{ik}^{-1}$, we obtain the two following forms of the duality equation (4):

$$X_{ik}^T \cdot \omega_i \cdot X_{ik} \sim \mu_k \quad (a) \qquad \Leftrightarrow \qquad \omega_i \sim Y_{ik}^T \cdot \mu_k \cdot Y_{ik} \quad (b). \tag{7}$$

Note that we can derive similar expressions for the two-dimensional equivalents of the parallelepipeds, parallelograms. The dimension reduction in that case does not allow for a full duality between parallelograms and cameras, but parallelograms are still useful for calibration (due to lack of space, we will not describe this in this paper, although we use it in practice).

3 Calibration of Intrinsic Parameters

In the previous section, the duality between the intrinsic parameters of a parallelepiped and that of a camera was introduced. Here, we consider a generic situation where n parallelepipeds are viewed by m cameras, and we study how to exploit the duality relations for the calibration task. Interestingly, using parallelepipeds as natural calibration objects offers several advantages over standard self-calibration approaches [15]:

- Fewer correspondences are needed; five and a half points extracted *per image* are sufficient, and even fewer inter-image correspondences are needed. For instance, the calibration of two cameras that view a parallelepiped from opposite viewpoints, is possible.
- The approach is based on Euclidean information about parallelepipeds or cameras which are easy to obtain (skew parameter equal to zero, right angles, *etc.*). Using such information ensures stability and robustness by limiting the number of singularities.
- Projections of parallelepipeds naturally enclose affine information, thus reduce the algebraic complexity when solving the calibration problem. Indeed, our calibration method is somewhat similar to self-calibration approaches that consider special camera motions [11,6,1] or to approaches that first recover the affine structure, e.g. [9,16].

In the following, we first study how to parameterize the calibration problem in a consistent way when n parallelepipeds are viewed by m cameras. We then explicit the constraints that can be derived from prior knowledge.

3.1 Parameterization of the Calibration Problem

When considering n parallelepipeds and m cameras, the question that naturally arises is how many independent unknowns are needed to represent all unknown intrinsic parameters? In general, each camera i and each parallelepiped k has five independent unknowns (the elements of ω_i and μ_k, minus the scale factors). Thus, n parallelepipeds and m cameras lead to $5n + 5m$ unknowns. However, each set of unknowns ω_i (resp. μ_k) is related to every set of unknowns of μ_k (resp. ω_i) via the duality equations (7). The duality between two sets of intrinsic parameters leads to 5 independent bilinear equations obtained by rewriting equations (7) as:

$$\mu_k \wedge \left(X_{ik}^T \cdot \omega_i \cdot X_{ik}\right) = 0 \quad \text{or} \quad \omega_i \wedge \left(Y_{ik}^T \cdot \mu_k \cdot Y_{ik}\right) = 0,$$

where the "cross product" operations are carried out between all possible pairs of corresponding matrix elements. Hence, one may obtain up to $5mn$ different bilinear equations in the $5n+5m$ unknowns, which are however not independent: all $5n + 5m$ unknowns can actually be parameterized using only 5 of them, corresponding e.g. to the intrinsic parameters of a single camera or parallelepiped (the parameters of the other entities can be computed by successive applications of the appropriate duality equations). Thus, the calibration problem for m cameras seeing n parallelepipeds, can be reduced to the estimation of a single set of intrinsic parameters, belonging to one camera or parallelepiped.

Let us choose a parallelepiped to parameterize the calibration problem. We denote μ_0 the corresponding matrix of intrinsic parameters. Solving for μ_0 requires therefore at least five constraints coming from prior knowledge on any of the $m + n$ entities involved (parallelepipeds and cameras). The following section explains how to do this.

3.2 Using Prior Knowledge

As explained before, we assume that n parallelepipeds are seen by m cameras. The geometric information that can be computed from the projection of a parallelepiped k in image i is enclosed in the canonic projection matrix $X_{ik} = Y_{ik}^{-1}$. From such matrices and prior information, we can derive constraints on the calibration by using the duality equations. In this section, we will consider prior knowledge on the elements of ω_i or μ_k, and how they constrain the elements of μ_0. Such knowledge comes from information on the intrinsic parameters of camera i or parallelepiped k. It should be noticed here that the relations between intrinsic parameters and matrix elements are not always linear (see for example the matrix μ given in section 2.2). In particular, a known angle of a parallelepiped gives in general a quadratic constraint on several elements of the associated matrix μ. However, most other types of prior information likely to be used in practice, lead to linear constraints.

The duality equations between μ_0 and image i, respectively parallelepiped k, are:

$$\begin{cases} \omega_i \sim Y_{i0}^T \cdot \mu_0 \cdot Y_{i0}, \\ \mu_k \sim X_{ik}^T \cdot Y_{i0}^T \cdot \mu_0 \cdot Y_{i0} \cdot X_{ik} \quad i \in [0..m-1]. \end{cases} \tag{8}$$

which may be rewritten in several different forms, by nested applications of the duality equations, e.g.:

$$\begin{cases} \omega_i \sim Y_{il}^T \cdot X_{jl}^T \cdot Y_{j0}^T \cdot \mu_0 \cdot Y_{j0} \cdot X_{jl} \cdot Y_{il}, \\ \mu_k \sim X_{jk}^T \cdot Y_{jl}^T \cdot X_{0l}^T \cdot Y_{00}^T \cdot \mu_0 \cdot Y_{00} \cdot X_{0l} \cdot Y_{jl} \cdot X_{jk}. \end{cases} j \in [0..m-1],\ l \in [0..n-1] \tag{9}$$

Such forms of the duality equations do in principle not provide additional independent equations on the elements of μ_0. However, they are useful for example in the case where the parallelepiped associated to μ_0 is occluded in view i (e.g. due to occlusion), thus X_{i0} is not available. Using extensions of the duality equations such as (9), knowledge on camera i can still be used to estimate μ_0.

We therefore derive the statements given in the following two paragraphs. When we speak of independent equations, we mean that they are independent in general, i.e. non-singular, situations.

Known elements of ω_i or μ_k.

1. Knowing that an element of ω_i is equal to zero (e.g. a null skew parameter) gives 1 linear equation on the elements of μ_0 (e.g. via (8)).
2. Knowing that an element of $\mu_{k\neq 0}$ is equal to zero (e.g. a right angle in the parallelepiped k) also leads to 1 linear constraint on the elements of μ_0. If $k = 0$, then the number of unknowns is reduced by 1.
3. Knowing a single element of ω_i or μ_k simply cancels the scale ambiguity for the corresponding entity.
4. Knowing p elements of ω_i leads to $(p-1)$ independent linear constraints on the elements of μ_0. Indeed, these p known elements form $(p-1)$ independent pairs. The ratio of each such pair of elements gives a linear constraint on the elements of μ_0 (via the appropriate duality relation).
5. In a similar way, knowing p elements of $\mu_{k\neq 0}$ leads to $(p-1)$ independent linear constraints on the elements of μ_0. If $k = 0$, then the number of unknowns is reduced to $(6-p)$.

In [20], we describe in detail which types of prior knowledge on intrinsic parameters may be used in the above ways. As mentioned before, any of the above constraints can be enforced using one or several redundant equations of the types (8) and (9) for example. Note that due to the above facts, an acquisition system with five different cameras viewing an arbitrary parallelepiped can be fully calibrated under the assumption of the skew parameters being zero. Equivalently, a system with one camera viewing five parallelepipeds with one right angle, or two parallelepipeds with three right angles (cuboids) can be fully calibrated.

Constant elements of ω_i or μ_k.

1. Knowing that an element of ω_i (resp. μ_k) is constant for $i = [0..m-1]$ (resp. $k = [0..n-1]$) does not give any information.
2. Knowing that two elements of ω_i (resp. μ_k) are constant for $i = [0..m-1]$ (resp. $k = [0..n-1]$) gives $(m-1)$ (resp. $n-1$) independent quadratic constraints on the elements of μ_0. They can be obtained by writing the two element ratios using the bilinear equations (8) with any independent pair of images (resp. any independent pair of parallelepipeds).
3. Knowing that *all* elements of ω_i (resp. μ_k) are constant for $i = [0..m-1]$ (resp. $k = [0..n-1]$) gives $4(m-1)$ (resp. $4(n-1)$) linear independent equations on the elements of μ_0. Indeed, as shown by Hartley [11], we can turn the quadratic equations into linear ones. If all X^{ik} and their inverses Y^{ik} are scaled such as to have the same determinant, then we may write down the following matrix equation between any pair (i, j) of views, that holds *exactly*, i.e. not only up to scale, as usual:

$$Y_{i0}^T \cdot \mu_0 \cdot Y_{i0} = Y_{j0}^T \cdot \mu_0 \cdot Y_{j0},$$

and respectively between any pair (k, l) of parallelepipeds:

$$X_{ik}^T \cdot Y_{i0}^T \cdot \mu_0 \cdot Y_{i0} \cdot X_{ik} = X_{il}^T \cdot Y_{i0}^T \cdot \mu_0 \cdot Y_{i0} \cdot X_{il}, \quad i \in [0..m-1].$$

This leads to 5 linear equations for each pair of views (resp. parallelepipeds) among which 4 are independent.

Consequently, an acquisition system where a camera views two identical parallelepipeds or where two identical cameras view one parallelepiped can be fully calibrated if the cameras' skew parameters are equal to zero. Note also the special case where the camera parameter u_0 (or equivalently v_0) is constant. This leads to quadratic equations since this parameter can be written as the ratio of two elements of ω, corresponding to case 2 above. Again, each of the above constraints can be enforced using several redundant equations taken from different versions of the duality equations.

3.3 Calibration Algorithm

Our approach consists of two stages. First, all available linear equations are used to determine μ_0 (the system is solved using SVD). If there is a unique solution, then we are done (from μ_0, all the camera and parallelepiped intrinsics can be computed using the X_{ik}). If however, the linear system is under-constrained, then the quadratic equations arising from constant but unknown intrinsics can be used to reduce the ambiguity in the solution. The decision if the system is under-constrained, may be taken on the basis of a singular value analysis. This also gives the degree of the ambiguity (dimension of the solution space). In practice, this is usually two or lower. Hence, two quadratic equations are in general sufficient to obtain a finite number of solutions (if more than the

minimum number of equations are available, then a best initial solution might be found using a RANSAC-type approach [8]). Once the matrices ω_i and μ_k are estimated, the matrices K_i and Λ_i can be computed via Cholesky decomposition.

Note that singular cases may exist, where calibration can not be achieved, Singular cases for one parallelepiped seen in one image, are described in [20].

3.4 Comparison with Vanishing Point Based Approaches

A popular way to calibrate from known parallel lines is to use vanishing points of perpendicular directions. Indeed, the perpendicularity condition gives a constraint on the intrinsic camera parameters. However, computing vanishing point positions in the image appears to be a process sensitive to noise. Our approach avoids such difficulties by computing projection matrices and hiding therefore vanishing point computation in a well defined estimation problem[1]. In the following, we give some numerical evidence suggesting this principle.

We have applied our approach on simulated images of a cube, taken for different orientations, and compared the calibration results with a non-linear approach where vanishing points were computed using the MLE estimator described in [14]. A 2 pixels Gaussian noise was added to the vertices' projections, prior to running the algorithms. Figure 1 shows median values for 500 tests of the relative error on the estimated focal length for both methods. In figure 1–(a) six vertices were used and thus one vanishing point's position was non-linearly optimized in the second approach. In figure 1–(b) seven vertices were used and all vanishing point positions were optimized. The graphs show that our *linear* method obtains similar results to the *non-linear* approach in non-singular situations, and better ones close to a singular situation (90° rotation of the cube).

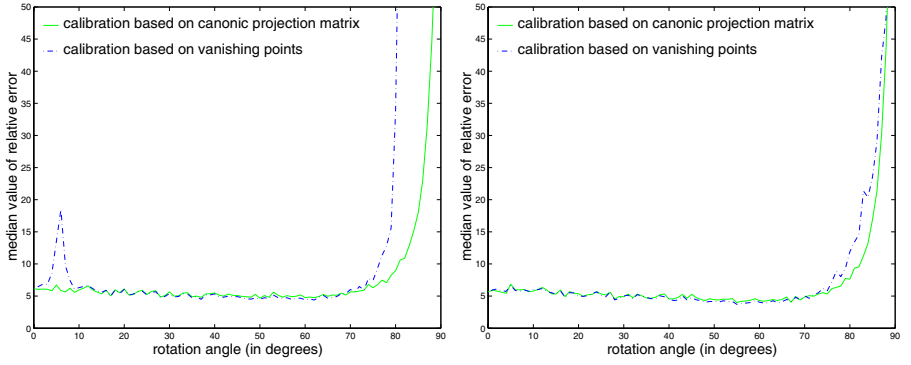

Fig. 1. Relative error on the estimated parameter α_v as a function of the cube rotation angle: (a) estimation using 6 cube vertices; (b) estimation using 7 cube vertices.

[1] The vanishing points of a parallelepiped's directions correspond to the columns of matrix X.

4 Pose Estimation

The step following the intrinsic calibration in the reconstruction process is pose estimation. The different cameras must indeed be localized with respect to a common coordinate frame in order to achieve consistent reconstructions. To this purpose, we extend the plane-based method presented in [17].

4.1 The Scale Ambiguity

If the intrinsic parameters of the cameras ($K_{i,i\in[0..m-1]}$) and the parallelepipeds ($\Lambda_{k,k\in[0..n-1]}$) are known, then from every matrix X_{ik} (see Section 2.4), we can compute the matrix $\tilde{X}'_{ik}$, which represents relative pose:

$$\tilde{X}'_{ik} \sim K_i^{-1} \cdot \tilde{X}_{ik} \cdot \tilde{\Lambda}_k^{-1} \sim [R_i S_k \quad R_i v_k + t_i]. \tag{10}$$

The left 3×3 submatrix of $\tilde{X}'_{ik}$ will be denoted X'_{ik}. Note that Λ_k and $\tilde{X}_{ik}$ are only defined up to scale. For matrices $\tilde{X}_{ik}$ used in the position recovery step, this scale factor has to be computed. It can be fixed for one parallelepiped and computed for others, for which *a priori* information about the relative scale between them and the reference one is available[2]. Afterwards, the matrices Λ_k and $\tilde{X}_{ik}$ are scaled such that the X'_{ik} have unit determinant. Then, X'_{ik} represents the rotation between the i^{th} camera and the k^{th} parallelepiped.

4.2 Rotational Part of the Pose

The matrix $X'_{ik} = R_i \cdot S_k$ represents the relative rotation between the i^{th} camera and the k^{th} parallelepiped. In practice, X'_{ik} will not be a perfect rotation matrix, but this can easily be corrected using SVD [13].

Let us first consider the case where all parallelepipeds are seen in all views. Then, all matrices X'_{ik} can be grouped and written as:

$$\underbrace{\begin{pmatrix} X'_{0,0} & X'_{0,1} & \cdots & X'_{0,n-1} \\ X'_{1,0} & X'_{1,1} & \cdots & X'_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ X'_{m-1,0} & X'_{m-1,1} & \cdots & X'_{m-1,n-1} \end{pmatrix}}_{X'} = \underbrace{\begin{pmatrix} R_0 \\ R_1 \\ \vdots \\ R_{m-1} \end{pmatrix}}_{R} \underbrace{\begin{pmatrix} S_0 & S_1 & \cdots & S_{n-1} \end{pmatrix}}_{S} \tag{11}$$

The matrices R_i and S_k can be extracted by factorizing X', due to the fact that its rank is 3. The factorization leads to solutions defined up to a global rotation. One might thus attach the reference frame to any camera or parallelepiped.

Missing data. In practice, the condition that all parallelepipeds are seen in all views can not always be satisfied and thus, some data might be missing in

[2] Note that the scale factor could also be recovered for each parallelepiped seen by at least two cameras, after estimation of the cameras' relative positions.

the matrix X'. However, each missing relative orientation X'_{ik} between camera i and parallelepiped k can be deduced from others if there is one camera i' and one parallelepiped k' such that the rotations $X'_{i'k'}$, $X'_{i'k}$ and $X'_{ik'}$ are known. The missing matrix can then be computed using:

$$X'_{ik} = X'_{ik'} \ (X'_{i'k'})^T X'_{i'k}. \tag{12}$$

Several equations of this type may be used simultaneously to increase accuracy. Also, knowing that two parallelepipeds k and k' have a common orientation can be imposed by fixing $S_{k'} = S_k$, and eliminating the k^{th} column from matrix X'. This can be useful when dealing with missing data.

4.3 Translational Part of the Pose

We assume that the scale ambiguity mentioned in section 4.1 is solved. Denoting w_{ik} the 4^{th} column of matrix X'_{ik}, and $v'_k = R_i v_k$, then camera positions can be determined by minimizing the sum of terms $\parallel w_{ik} - R_i v_k - t_i \parallel$ over all available image-parallelepiped pairs. This can be done using linear least squares [17].

5 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. In this section, we briefly present a linear algorithm that we have implemented to construct realistic 3D models. It can use linear constraints from single or multiple images.

5.1 Reconstruction Algorithm

The algorithm is composed of three main successive steps:

- Reconstruction of parallelepipeds used for calibration: their intrinsics are recovered in the calibration phase and position/orientation by pose estimation. The 3D positions of their vertices can then be computed using (6).
- Reconstruction of points visible in several images (given correspondences): using any method described for example in [10].
- Reconstruction of points visible in one image only: by solving systems of linear constraints like coplanarity, collinearity, or the fact that points belong to a parallelogram, such constraints being defined with respect to points already reconstructed [20].

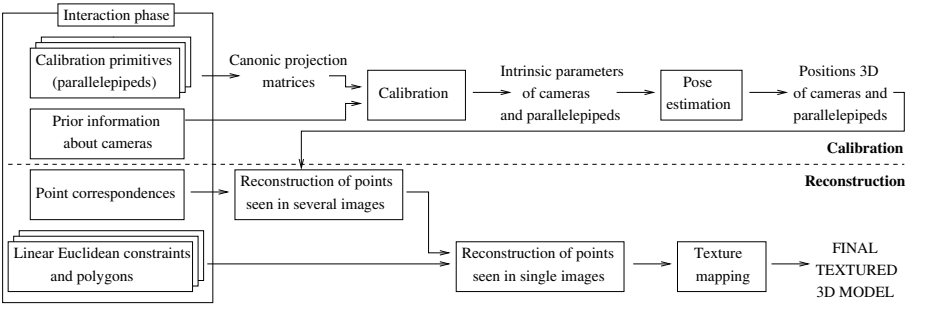

Fig. 2. The calibration and reconstruction algorithms.

This approach is actually independent from the calibration method, although it uses the same input in the first step. Interestingly, it allows 3D models to be computed from non-overlapping photographs (see e.g. Fig. 5). The global scheme of our system is shown in Figure 2.

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

5.2 Experimental Results

Known scene. In order to estimate the accuracy of the calibration and reconstruction methods, a scene containing several boxes with known dimensions (see Fig. 3) was used. Three images were taken with an off-the-shelf digital camera, and with varying zoom. Calibration and reconstruction of the scene were achieved using different calibration scenarios. Table 1 gives the relative errors on length ratios l_x/l_z, l_y/l_z of the reconstructed primitives 0, 1, 2 (see left part of Fig.3 for notations), as well as the relative errors on the estimated aspect ratio. We assume here that the principal point is the image center and that skew parameters are equal to zero. The following scenarios were tested:

1. Cameras are independently calibrated: for every image $i \in [0..2]$, we parameterize the problem in the unknown elements of ω_i (i.e. α_{ui}, α_{vi}) and we use the information about right angles of parallelepipeds seen in that view.
2. Parallelepipeds are independently calibrated: for every parallelepiped $k \in [0..2]$, we parameterize the problem in the unknown terms of μ_k (i.e. l_{xk}/l_{zk}, l_{yk}/l_{zk}) and we use the information that the skew parameter is equal to zero and that the principal point is the image center.
3. We parameterize the problem in terms of μ_0, and use the known principal points, but no prior knowledge on the parallelepipeds.
4. Same as scenario 3, but also using known values of the camera aspect ratios.

These results confirm the intuition that, on average, errors do not depend on the entity chosen for parameterization. However, using a fundamental entity with respect to which all the constraints are expressed, results in a more uniform

Fig. 3. Images and parallelepipeds used for calibration.

distribution of the errors, and naturally, adding known information on the aspect ratio, reduces the errors.

To visually evaluate the quality of the reconstruction, we show screenshots of the reconstructed scene. Figure 4–(a) shows the orientation of boxes, which was recovered without any assumption about their relative rotation. To solve the scale ambiguity between boxes 0 and 2, we used only a coplanarity constraint between one of the points of box 2 and the base-plane of box 0. Figure 4–(b) shows the roof of the house, which was reconstructed as an arbitrary parallelepiped having several vertices in common with the cubic base of the house. These figures show that the reconstructed scene is qualitatively correct.

(a) (b)

Fig. 4. Reconstructed boxes.

Table 1. Average relative errors on the estimated camera aspect ratios and the estimated length ratios of the parallelepipeds.

Scenario	av. rel. err. τ[%]			av. rel. err. length ratio [%]		
	K_1	K_2	K_3	l_{x0}/l_{z0} l_{y0}/l_{z0}	l_{x1}/l_{z1} l_{y1}/l_{z1}	l_{x2}/l_{z2} l_{y2}/l_{z2}
1	3.41	4.13	0.10	3.21 5.14	5.43 14.47	5.41 2.08
2	3.57	3.15	9.55	0.45 6.97	0.74 5.13	13.20 3.32
3	5.23	6.12	6.24	2.67 7.82	3.91 0.49	7.57 0.31
4	3.76	4.60	4.57	2.35 6.01	3.24 0.97	5.77 0.64

Outdoor scene. Figure 5 shows the reconstruction of an entire building from just two images taken from opposite viewpoints. The parallelepiped used for calibration and the estimated camera positions are shown in the two original images 5–(a),(b). In the first image, intersections of lines were computed to obtain the six points required to define a parallelepiped (see Fig. 5–(a)). The reconstruction was then done according to the two following steps. First, vertices of the reference parallelepiped were reconstructed during the calibration step. Second, the rest of the scene was modeled using primitives depicted by the user.

Fig. 5. Building reconstruction: (a) Original photos used for the reconstruction; (b) the reconstruction scenario with the computed model and camera positions; circles correspond to parallelepiped vertices, crosses and rectangles to points reconstructed from the first and second image respectively; (c),(d) Details of the model.

6 Conclusion

We have presented a framework for calibration and 3D model acquisition from several arbitrary images based on projections of parallelepipeds. Our system uses geometric constraints, provided interactively by the user, for camera calibration, pose estimation, and modeling of 3D scene structure. It combines available information on the scene structure (angles and length ratios in parallelepipeds) and on the intrinsic camera parameters with multi-view constraints.

In practice, our method leads to calibration and reconstruction which can be obtained by solving only linear equations. Future work on this topic concerns: the automatization of the process and thus the reduction of interaction needed to define primitives in several images; refinement of an approximate model; automatic surface generation from reconstructed points.

References

1. Armstrong, M., Zisserman, A., Beardsley, P.: Euclidean Structure from Uncalibrated Images. BMVC (1994) 509–518
2. Caprile, B., Torre, V.: Using Vanishing Points for Camera Calibration. IJCV **4** (1990) 127–140
3. Chen, C.-S., Yu, C.-K., Hung, Y.-P.: New Calibration-free Approach for Augmented Reality Based on Parametrized Cuboid Structure. ICCV (1999) 30–37
4. Cipolla, R., Boyer, E.: 3D model acquisition from uncalibrated images. IAPR Workshop on Machine Vision Applications, Chiba, Japan (1998) 559–568
5. Criminisi, A., Reid, I., Zisserman, A.: Single View Metrology. ICCV (1999) 434–442
6. de Agapito, L., Hartley, R., Hayman, E.: Linear selfcalibration of a rotating and zooming camera. CVPR (1999) 15–21
7. Debevec, P.E., Taylor, C.J., Malik, J.: Modeling and Rendering Architecture from Photographs: a Hybrid Geometry-and Image-Based Approach. SIGGRAPH (1996) 11–20
8. Fischler, M.A., Bolles, R.C.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Graphics and Image Processing **24(6)** (1981) 381–395
9. Hartley, R.I.: Euclidean Reconstruction from Uncalibrated Views. DARPA–ESPRIT Workshop on Applications of Invariants in Computer Vision, Azores, Portugal (1993) 187–202
10. Hartley, R.I., Sturm, P.: Triangulation. CVIU **68(2)** (1997) 146–157
11. Hartley, R.I.: Self-calibration of Stationary Cameras. IJCV **22(1)** (1997) 5–23
12. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2000)
13. Kanatani, K.: Statistical Optimization for Geometric Computation: Theory and Practice. Elsevier Science (1996)
14. Liebowitz, D., Zisserman, A.: Metric Rectification for Perspective Images of Planes. CVPR (1998) 482–488
15. Maybank, S.J., Faugeras, O.D.: A Theory of Self Calibration of a Moving Camera. IJCV **8(2)** (1992) 123–151
16. Pollefeys, M., Van Gool, L.: A Stratified Approach to Metric Self-Calibration. CVPR (1998) 407–412

17. Sturm, P.: Algorithms for Plane-Based Pose Estimation. CVPR (2000) 1010–1017
18. Sturm, P., Maybank, S.: On Plane-Based Camera Calibration: A General Algorithm, Singularities, Applications. CVPR (1999) 432–437
19. Triggs, B.: Autocalibration and the Absolute Quadric. CVPR (1997) 609–614
20. Wilczkowiak, M., Boyer, E., Sturm, P.: Camera Calibration and 3D Reconstruction from Single Images Using Parallelepipeds. ICCV (2001) 142–148
21. Zhang, Z.: Flexible Camera Calibration By Viewing a Plane From Unknown Orientations. ICCV (1999) 666–673