Multi-view constraints between collineations: application to self-calibration from unknown planar structures

Ezio Malis and Roberto Cipolla

Engineering Department, Cambridge University
Trumpington Street, CB2 1PZ Cambridge
{em240,cipolla}@eng.cam.ac.uk
http://www-svr.eng.cam.ac.uk

Abstract. In this paper we describe an efficient method to impose the constraints existing between the collineations which can be computed from a sequence of views of a planar structure. These constraints are usually not taken into account by multi-view techniques in order not to increase the computational complexity of the algorithms. However, imposing the constraints is very useful since it allows a reduction in the geometric errors in the reprojected features and provides a consistent set of collineations which can be used for several applications such as mosaicing, reconstruction and self-calibration. In order to show the validity of our approach, this paper focus on self-calibration from unknown planar structures proposing a new method exploiting the consistent set of collineations. Our method can deal with an arbitrary number of views and an arbitrary number of planes and varying camera internal parameters. However, for simplicity this paper will only discuss the case with constant camera internal parameters. The results obtained with synthetic and real data are very accurate and stable even when using only a few images.

Keywords: Self-calibration, Homography.

1 Introduction

The particular geometry of features lying on planes is often the reason for the inaccuracy of many computer vision applications (structure from motion, self-calibration) if it is not taken explicitly into account in the algorithms. Introducing some knowledge about the coplanarity of the features and about their structure (metric or topological) can improve the quality of the estimates [12]. However, the only prior geometric knowledge on the features that will be used here is their coplanarity. Two views of a plane are related by a collineation. Using multiple views of a plane we obtain a set of collineations which are not independent. If there are multiple planes in the scene there will be a set of collineations for each plane and again some constraints between the different

sets. In order to avoid solving non-linear optimisation problems, the constraints existing within a set of collineation and between sets have often been neglected. However, these multi-view constraints can be used to improve the estimation of the collineations matrices as in [15], where multiple planes (≥ 2) are supposed to be viewed in the images. In this paper we analyse the constraints existing between a set of collineations induced by a simple plane in the image but it is very easy to extend our analysis to the case of multiple planes. Imposing the constraint is useful since it allows the reduction of the geometric error in the reprojected features and provides a consistent set of collineations which can be used for several applications as mosaicing, reconstruction and self-calibration.

In this paper we will focus on camera self-calibration. Camera self-calibration from views of a generic scene has been widely investigated and the two main approaches are based on the properties absolute conics [13] [10] or on some algebraic error [7] [4]. Depending on the a priori information provided the self-calibration algorithms can be classified as follows. Algorithms that use some knowledge of the observed scene: identifiable targets of known shape [8], metric structure of planes [11]. Algorithms that exploit particular camera motions: translating camera or rotating camera [5]. Algorithms that suppose known some of the camera parameters: some fixed camera parameters (i.e. skew zero, unit ratio ...), varying camera parameters [10] [9]. Camera self-calibration from planar scenes with known metric structure has been investigated in several papers. However, it is interesting to develop flexible techniques which do not need any a priori knowledge about the camera motion as in [5] or metric knowledge of the planar scene. A method for self-calibrating a camera from views of planar scenes without knowing their metric structure was proposed in [14]. Triggs developed a self-calibration technique based on some constraints involving the absolute quadric and the scene-plane to image-plane collineations. However, in practice it is not possible to estimate these collineations without knowing the metric structure of the plane. Only the collineations with respect to a reference view (a key image) can be used to self-calibrate a camera with constant internal parameters. As noticed by Triggs, inaccurate measurements or poor conditioning in the key image contribute to all the collineations reducing the numerical accuracy or the stability of the method. The aim of this paper is to investigate how to improve the self-calibration from planar scenes with unknown metric structure. We will not use any key image but all the images of the sequence are treated equally averaging the uncertainty over all of them.

This paper is organised as follows. In Section 2 we review the relationship existing between two views of coplanar features and some properties of the collineation matrices. In Section 3 we generalise the two-view geometry to multiple views introducing the super-collineation matrix to describe a set of collineations. Then, we describe a simple algorithm to impose the constraints existing between the collineation of the set. Finally, we describe some constraints on the camera internal parameters which can be used for self-calibration. In Section 4 we give the results obtained with both synthetic and real data.

2 Two-view geometry of a plane

In this section we describe the relationship between two views of a planar structure. Each camera performs a perspective projection of a point $\mathbf{x} \in \mathbb{P}^3$ (with homogeneous coordinates $\mathbf{x} = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T$) to an image point $\mathbf{p} \in \mathbb{P}^2$ (with homogeneous coordinates $\mathbf{p} = \begin{bmatrix} u & v & 1 \end{bmatrix}^T$) measured in pixels: $\mathbf{p} \propto \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \mathbf{x}$, where $\mathbf{R}$ and $\mathbf{t}$ represent the displacement between the frame $\mathcal{F}$ attached to the camera and an absolute coordinate frame $\mathcal{F}_0$, and $\mathbf{K}$ is a non-singular (3×3) matrix containing the intrinsic parameters of the camera:

$$\mathbf{K} = \begin{bmatrix} fk_u & -fk_u \cot(\theta) & u_0 \\ 0 & fk_v / \sin(\theta) & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

where u_0 and v_0 are the coordinates of principal point (in pixels), f is the focal length (in metres), k_u and k_v are the magnifications respectively in the $\vec{u}$ and $\vec{v}$ direction (in pixels/metres) and θ is the angle between these axes.

2.1 The collineation matrix in projective space

Let $\mathcal{F}_i$ and $\mathcal{F}_j$ be two frames attached respectively to the image $\mathcal{I}_i$ and $\mathcal{I}_j$. The two views of a planar object are related by a collineation matrix in projective space. Indeed, the image coordinates $\mathbf{p}_{ik}$ of the point $\mathcal{P}_k$ in the image $\mathcal{I}_i$ can be obtained from the image coordinates $\mathbf{p}_{jk}$ of the point $\mathcal{P}_k$ in the image $\mathcal{I}_j$:

$$\mathbf{p}_{ik} \propto \mathbf{G}_{ij} \mathbf{p}_{jk} \tag{2}$$

where the collineation matrix $\mathbf{G}_{ij}$ is a (3×3) matrix defined up to scalar factor which can be written as:

$$\mathbf{G}_{ij} \propto \mathbf{K}_i \mathbf{H}_{ij} \mathbf{K}_j^{-1} \tag{3}$$

where $\mathbf{H}_{ij}$ is the corresponding homography matrix in the Euclidean space. Homography and collineation are generally used to indicate the same projective transformation from $\mathbb{P}^n$ to $\mathbb{P}^n$ (in our case $n = 2$). In this paper we will use the term "homography" to indicate a collineation expressed in Euclidean space.

A relationship similar to equation (2) exists between the projections $\mathbf{l}_{ik}$ and $\mathbf{l}_{jk}$ in the two images $\mathcal{I}_i$ and $\mathcal{I}_j$ of a 3D line $\mathcal{L}_k$:

$$\mathbf{l}_{ik} \propto \mathbf{G}_{ij}^{-\top} \mathbf{l}_{jk} \tag{4}$$

The estimation of the collineation matrix is possible both from equation (2) and/or equation (4). However, for simplicity we will analyse only the case of points since the same results can be applied for lines.

2.2 The homography matrix in Euclidean space

The homography matrix can be written as a function of the camera displacement and the normal to the plane [2]:

$$\mathbf{H}_{ij} = \mathbf{R}_{ij} + \frac{\mathbf{t}_{ij}\,\mathbf{n}_j^\top}{d_j} \tag{5}$$

where $\mathbf{R}_{ij}$ and $\mathbf{t}_{ij}$ are respectively the rotation and the translation between the frames $\mathcal{F}_i$ and $\mathcal{F}_j$, $\mathbf{n}_j$ is the normal to the plane π expressed in the frame $\mathcal{F}_j$ and d_j is the distance of the plane π from the origin of the frame $\mathcal{F}_j$. From (3), $\mathbf{H}_{ij}$ can be estimated from $\mathbf{G}_{ij}$ if we know the camera internal parameters of the two cameras:

$$\mathbf{H}_{ij} \propto \mathbf{K}_i^{-1}\mathbf{G}_{ij}\mathbf{K}_j \tag{6}$$

Three important properties of the homography matrix will be extended to the multi-view geometry in the next section:

1. the Euclidean homography matrix is *not* defined up to a scale factor. If the homography is multiplied by a scalar γ ($\mathbf{H}' = \gamma\mathbf{H}$), this scalar can be easily recovered. If $\text{svd}(\mathbf{H}') = \begin{bmatrix}\sigma_1 & \sigma_2 & \sigma_3\end{bmatrix}$ are the singular values of $\mathbf{H}'$ in decreasing order, $\sigma_1 \geq \sigma_2 \geq \sigma_3 > 0$, then γ is the median singular value of $\mathbf{H}'$: $\gamma = \text{median}(\text{svd}(\mathbf{H}')) = \sigma_2$. Indeed, the matrix $\mathbf{H}$ has a unit singular value [16] and this property can be used to normalise the homography matrix.
2. from equation (5) it is easy to show that the homography matrix satisfies the following equation $\forall k > 0$ (where $[\mathbf{n}_i]_\times$ and $[\mathbf{n}_j]_\times$ are the skew symmetric matrices associated with vectors $\mathbf{n}_i$ and $\mathbf{n}_j$ which represent the normal to the plane expressed respectively in the image frame $\mathcal{F}_i$ and $\mathcal{F}_j$):

   $$[\mathbf{n}_i]_\times^k\,\mathbf{H}_{ji}^T = \mathbf{H}_{ij}\,[\mathbf{n}_j]_\times^k \tag{7}$$

   This equation provides useful constraints. If $k = 1$, the matrix $[\mathbf{n}_i]_\times\,\mathbf{H}_{ji}^T = [\mathbf{n}_i]_\times\,\mathbf{R}_{ij}$ has similar properties to the essential matrix (i.e. $\mathbf{E} = [\mathbf{t}]_\times\mathbf{R}$). Indeed, this matrix has two equal singular values and one equal to zero. This means two constraints each homography on the camera internal parameters [6] which can be used for the self-calibration as in [9]. If $k = 2$, knowing that $[\mathbf{n}]^2 = \mathbf{n}\mathbf{n}^T - \mathbf{I}$, equation (7) can be written:

   $$\mathbf{n}_i\mathbf{n}_i^T\mathbf{H}_{ji}^T - \mathbf{H}_{ij}\mathbf{n}_j\mathbf{n}_j^T = \mathbf{H}_{ji}^T - \mathbf{H}_{ij} \tag{8}$$

   and provides equations that will be used to compute $\mathbf{n}_i$ and $\mathbf{n}_j$.
3. a very important relation can be obtained from equation (7) (with $k = 1$) and will be used to compute $\mathbf{n}_i$ and $\mathbf{n}_j$:

   $$[\mathbf{n}_i]_\times = \mathbf{H}_{ij}\,[\mathbf{n}_j]_\times\,\mathbf{H}_{ij}^T \tag{9}$$

   Indeed, since $\det(\mathbf{M})\mathbf{M}\,[\mathbf{v}]_\times\,\mathbf{M}^T = \left[\mathbf{M}^{-\top}\mathbf{v}\right]_\times$ then:

   $$\mathbf{n}_i = \mathbf{Q}_{ij}\mathbf{n}_j \tag{10}$$

   where:

   $$\mathbf{Q}_{ij} = \det(\mathbf{H}_{ij})\,\mathbf{H}_{ij}^{-\top} \tag{11}$$

3 Multi-view geometry of a plane

In this section we describe the relationships between several views of a planar structure. We will point out that a super matrix of 2D collineations among m views has rank 3 and we will show how to enforce the rank property in an iterative procedure. The properties of the corresponding super matrix of 2D homographies provide the necessary constraint for the self-calibration of the camera internal parameters. In what follows we will describe the case when only one planar structure is used but the extension to more than one plane is straightforward.

3.1 The super-collineation matrix

If m images of an unknown planar structure are available, it is possible to compute $m(m-1)$ collineations (m collineations are always equal to the identity matrix). Let us define the super-collineation matrix as follows:

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_{11} & \cdots & \mathbf{G}_{1m} \\ \vdots & \ddots & \vdots \\ \mathbf{G}_{m1} & \cdots & \mathbf{G}_{mm} \end{bmatrix} \tag{12}$$

with $\dim(\mathbf{G}) = (3m, 3m)$ and $\text{rank}(\mathbf{G}) = 3$. The rank of $\mathbf{G}$ can not be less than three since $\mathbf{G}_{ii} = \mathbf{I}_3 \; i \in \{1, 2, 3, ..., m\}$, and cannot be more than three since each row of the matrix can be obtained from a linear combination of three others rows:

$$\mathbf{G}_{ij} = \mathbf{G}_{ik}\mathbf{G}_{kj} \qquad \forall i, j, k \in \{1, 2, 3, ..., m\} \tag{13}$$

This is a very strong constraint which is generally never imposed. Indeed, it would require a complex nonlinear minimisation algorithm over all the images. The constraints (13) can be summarised by the following equation:

$$\mathbf{G}^2 = m\,\mathbf{G} \tag{14}$$

Then, matrix $\mathbf{G}$ has 3 nonzero equal eigenvalues $\lambda_1 = \lambda_2 = \lambda_3 = m$ and $3(m-1)$ null eigenvalues $\lambda_4 = \lambda_5 = ... = \lambda_{3m} = 0$. If we can impose the constraint $\mathbf{G}^2 = m\;\mathbf{G}$ (with $\mathbf{G}_{ii} = \mathbf{I}_3 \; i = 1, 2, 3, ..., m$) then this is in fact equivalent to imposing the constraints $\mathbf{G}_{ij} = \mathbf{G}_{ik}\mathbf{G}_{kj}$.

Imposing the constraints In order to impose the constraint, we exploit the properties of the super-collineation matrix. Let $\mathbf{p}_{ij}$ be the j-th point ($j = \{1, 2, 3, ..., n\}$) of the i-th image ($i = \{1, 2, 3, ..., m\}$). The j-th point in all the images can be represented by the vector of dimension $(3m, 1)$ (which we will call a super-point): $\mathbf{p}_j^T = \begin{bmatrix} \mathbf{p}_{1j}^T & \mathbf{p}_{2j}^T & \cdots & \mathbf{p}_{mj}^T \end{bmatrix}$. Generalising equation (2) we obtain:

$$\boldsymbol{\Gamma}_j \mathbf{p}_j = \mathbf{G}\mathbf{p}_j \tag{15}$$

where $\boldsymbol{\Gamma}_j = \text{diag}(\gamma_{1j}\mathbf{I}_3, \gamma_{2j}\mathbf{I}_3, ..., \gamma_{mj}\mathbf{I}_3)$ is a diagonal matrix relative to the set of points j. Then, multiplying both sides of equation (15) by $\mathbf{G}$ we have:

$$\mathbf{G}\boldsymbol{\Gamma}_j \mathbf{p}_j = \mathbf{G}^2\mathbf{p}_j = m\mathbf{G}\mathbf{p}_j = m\boldsymbol{\Gamma}_j\mathbf{p}_j \tag{16}$$

The vector $\tilde{\mathbf{p}}_j = \boldsymbol{\Gamma}_j \mathbf{p}_j$ (representing the homogeneous coordinates of the point j in all the images) is an eigenvector of $\mathbf{G}$ corresponding to the eigenvalue m:

$$\mathbf{G}\tilde{\mathbf{p}}_j = m\tilde{\mathbf{p}}_j \tag{17}$$

As a consequence any super-point can be obtained as a linear combination of the eigenvectors of $\mathbf{G}$ corresponding to the eigenvalue $\lambda = m$:

$$\tilde{\mathbf{p}}_j = \alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2 + \alpha_3\mathbf{x}_3 \tag{18}$$

The matrix $\mathbf{G}$ can always be diagonalised and thus three linearly independent eigenvectors always exist, i.e., $\exists \mathbf{X} : \mathbf{X}^{-1}\mathbf{G}\mathbf{X} = diag(\lambda_1, \lambda_2, ..., \lambda_{3m})$. The columns of the matrix $\mathbf{X}$ are in fact eigenvectors of $\mathbf{G}$. Since $\mathbf{X}$ is nonsingular, the eigenvectors of $\mathbf{G}$ are linearly independent and span the space $\mathbb{R}^{3m}$. That means that an initial estimation $\widehat{\mathbf{p}} \in \mathbb{R}^{3m}$ of the super-point $\tilde{\mathbf{p}}$ can be written as $\widehat{\mathbf{p}} = \alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2 + \alpha_3\mathbf{x}_3 + ... + \alpha_{3m}\mathbf{x}_{3m}$. The real super-point $\tilde{\mathbf{p}}$ is an eigenvector of $\mathbf{G}$ corresponding to the largest eigenvalue $\lambda = m$. We can thus use a well-known algorithm to find an eigenvector of $\mathbf{G}$ starting from $\widehat{\mathbf{p}}$. Lets multiply our vector by $\frac{1}{m}\mathbf{G}$:

$$\widehat{\mathbf{p}}^1 = \frac{1}{m}\mathbf{G}\widehat{\mathbf{p}} = \frac{1}{m}(\alpha_1\mathbf{G}\mathbf{x}_1 + \alpha_2\mathbf{G}\mathbf{x}_2 + \alpha_3\mathbf{G}\mathbf{x}_3 + ... + \alpha_{3m}\mathbf{G}\mathbf{x}_{3m}) \tag{19}$$

and then replace each $\mathbf{G}\mathbf{x}_k$ with its corresponding $\lambda_k\mathbf{x}_k$. Factoring out λ_1 we have:

$$\widehat{\mathbf{p}}^1 = \frac{\lambda_1}{m}\left(\alpha_1\mathbf{x}_1 + \frac{\lambda_2}{\lambda_1}\alpha_2\mathbf{x}_2 + \frac{\lambda_3}{\lambda_1}\alpha_3\mathbf{x}_3 + ... + \frac{\lambda_{3m}}{\lambda_1}\alpha_{3m}\mathbf{x}_{3m}\right) \tag{20}$$

In a similar way, iterating the procedure k times we obtain:

$$\widehat{\mathbf{p}}^k = \frac{\lambda_1^k}{m^k}\left(\alpha_1\mathbf{x}_1 + \left(\frac{\lambda_2}{\lambda_1}\right)^k\alpha_2\mathbf{x}_2 + \left(\frac{\lambda_3}{\lambda_1}\right)^k\alpha_3\mathbf{x}_3 + ... + \left(\frac{\lambda_{3m}}{\lambda_1}\right)^k\alpha_{3m}\mathbf{x}_{3m}\right) \tag{21}$$

This algorithm will converge to the eigenvector of $\mathbf{G}$ corresponding to the highest eigenvalue since all the fractions λ_k/λ_1 that are less than unit in magnitude become smaller as we raise to higher powers. In our case, if we knew exactly the super-collineation matrix, the algorithm would converge after only one iteration since we have $\lambda_1 = \lambda_2 = \lambda_3 = m$ and $\lambda_k = 0 \ \forall \ 3 < k \leq 3m$ and the new estimated super-point will satisfy the constraint of being an eigenvector of $\mathbf{G}$ which means that the noise has been reduced.

In practice, the real super-collineation matrix $\mathbf{G}$ is unknown and we must use an approximation $\widehat{\mathbf{G}}$ estimated from the noisy points in the images. The algorithm used is the following. We start with a set of n points $\widehat{\mathbf{p}}_j$ $(j = 1, 2, 3, ..., n)$ and compute the super-collineation matrix $\widehat{\mathbf{G}}$ solving independently the linear problem of estimating each block $\widehat{\mathbf{G}}_{ij}$ from equation (2). It is not necessary that all the points are visible in all the images. Then, we compute a new set of super-points trying to impose the constraint. The better the estimate of $\mathbf{G}$ we obtain the faster the algorithm will converge and the more accurate will be the results. At iteration k the algorithm is:

(i) estimate the super-collineation matrix

$$\widehat{\mathbf{p}}_j(k) \;\Rightarrow\; \widehat{\mathbf{G}}(k) \tag{22}$$

(ii) compute the new super-point

$$\widehat{\mathbf{p}}_j(k+1) = \frac{1}{m}\widehat{\mathbf{G}}(k)\widehat{\mathbf{p}}_j(k) \tag{23}$$

This algorithm treats all the images with the same priority without using any key image and forces the rank 3 constraint on $\mathbf{G}$. We now show some simulation results which demonstrate the validity of our approach (we will describe in the next section 3.2 how to use the consistent set of collineation matrix in order to perform the self-calibration of the camera).

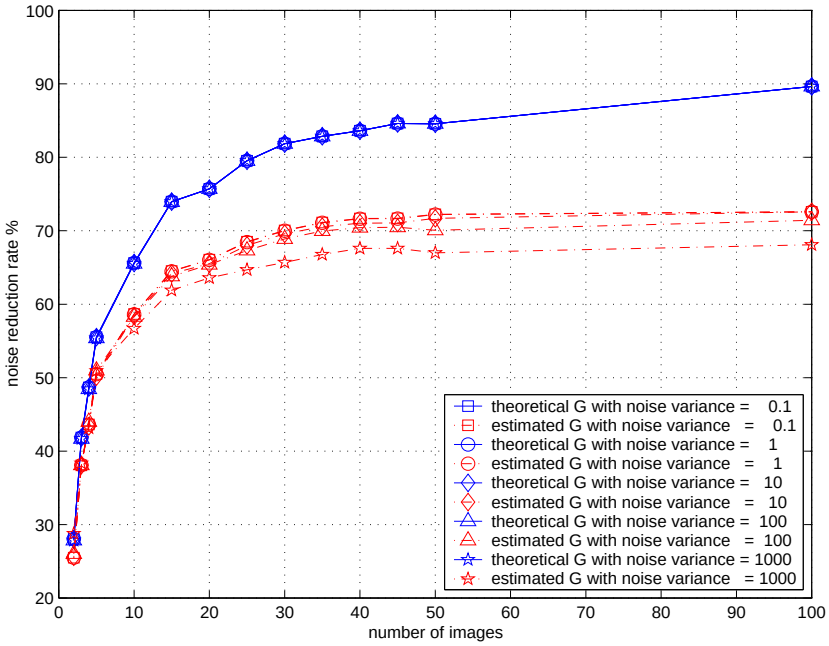


Fig. 1. Geometric noise reduction obtained imposing the rank 3 constraint on the super-collineation matrix.

Simulation results The algorithm was tested on a simulated planar grid of 64 points. It converges after 1 or 2 iterations imposing the constraint on the rank of the matrix $\mathbf{G}$. The error between the theoretical position of the points (which is never used in the algorithm but only for its evaluation) and the transformed coordinates is greatly reduced. Figure 1 shows the results obtained from random views with a random axis of rotation and a 30 degrees angle of rotation with respect to a fixed position. On the horizontal axis are given the number of

images used and on the vertical axis the corresponding rate of noise reduction. The continuous line gives us an upper bound of the reduction rate that could be possible if the super-collineation was known exactly. This reduction is practically independent of the level of noise and all continuous lines are superposed in Figure 1. The dashed dotted lines represent the results obtained estimating the super-collineation matrix with differents level of noise ($0.1 \leq \sigma^2 \leq 1000$). The rate of noise reduction does not vary significantly with the amplitude of the noise. However, when the level of noise increases the reduction rate decreases since the estimation of the super-collineation matrix becomes less accurate. Finally, we obtain only a small improvement when increasing the number of the images from 50 to 100. In the simulation results described in section 4.1 we obtain very similar results varying the camera internal parameters and the angle of rotation between the images.

3.2 The super-homography matrix

Let us define the super-homography matrix in the Euclidean space as:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{11} & \cdots & \mathbf{H}_{1m} \\ \vdots & \ddots & \vdots \\ \mathbf{H}_{m1} & \cdots & \mathbf{H}_{mm} \end{bmatrix} \tag{24}$$

with $\dim(\mathbf{H}) = (3m, 3m)$ and $\text{rank}(\mathbf{H}) = 3$. The super-homography matrix can be obtained from the super-collineation matrix and the camera parameters:

$$\mathbf{H} = \mathbf{K}^{-1}\mathbf{G}\mathbf{K} \tag{25}$$

where ($\dim(\mathbf{K}) = (3m, 3m)$ and $\text{rank}(\mathbf{K}) = 3m$):

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{K}_n \end{bmatrix} \tag{26}$$

is the matrix containing the internal parameters of all the cameras. It should be noticed that if the constraint $\mathbf{G}^2 = m\mathbf{G}$ was imposed, then the constraint $\mathbf{H}^2 = m\mathbf{H}$ is automatically imposed which means that the following constraints are satisfied:

$$\mathbf{H}_{ij} = \mathbf{H}_{ik}\mathbf{H}_{kj} \tag{27}$$

Unlike the super-collineation matrix, the super-homography matrix is *not* defined up to a diagonal similarity. Indeed, if σ_{ij} denotes the median singular value of the matrix $\mathbf{H}_{ij}$ we can build the following matrix which contain all the coefficients of normalisation: $\mathbf{D} = \text{diag}(\sigma_{11}\mathbf{I}_3, \sigma_{12}\mathbf{I}_3, ..., \sigma_{1m}\mathbf{I}_3)$. The super homography matrix is thus normalised as follows:

$$\mathbf{H} = \mathbf{D}\tilde{\mathbf{H}}\mathbf{D}^{-1} \tag{28}$$

From this equation we can easily see that the constraint $\mathbf{H}^2 = m\mathbf{H}$ holds. In the presence of noise, normalising $\mathbf{H}$ with equation (28) will conserve the rank constraint of the matrix since it is a similarity transformation.

3.3 Super-homography decomposition

After normalisation, the homography matrix can be decomposed as:

$$\mathbf{H} = \mathbf{R} + \mathbf{T}\mathbf{N}^T \tag{29}$$

where:

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_{11} & \cdots & \mathbf{R}_{1m} \\ \vdots & \ddots & \vdots \\ \mathbf{R}_{m1} & \cdots & \mathbf{R}_{mm} \end{bmatrix}, \mathbf{T} = \begin{bmatrix} \frac{\mathbf{t}_{11}}{d_1} & \cdots & \frac{\mathbf{t}_{1m}}{d_m} \\ \vdots & \ddots & \vdots \\ \frac{\mathbf{t}_{m1}}{d_1} & \cdots & \frac{\mathbf{t}_{mm}}{d_m} \end{bmatrix}, \mathbf{N} = \begin{bmatrix} \mathbf{n}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{n}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{n}_m \end{bmatrix} \tag{30}$$

with $\dim(\mathbf{R}) = (3m, 3m)$, $\text{rank}(\mathbf{R}) = 3$, $\dim(\mathbf{T}) = (3m, m)$ and $\dim(\mathbf{N}) = (3m, m)$. Matrix $\mathbf{R}$ is a symmetric matrix, $\mathbf{R} = \mathbf{R}^T$ and $\mathbf{R}^2 = m\mathbf{R}$. As a consequence not only are the three largest eigenvalues $\lambda_1 = \lambda_2 = \lambda_3 = m$ but also the three larg est singular values are $\sigma_1 = \sigma_2 = \sigma_3 = m$. In [2] and [16] are presented two different methods for decomposing the homography matrix, computed from two views of a planar structure, following equation (5). In general, there are two possible solutions but the ambiguity can be resolved by adding more images. Here we present a method to decompose any set of homography matrices. Equation (10) can be generalised as follow:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{11} & \cdots & \mathbf{Q}_{1m} \\ \vdots & \ddots & \vdots \\ \mathbf{Q}_{m1} & \cdots & \mathbf{Q}_{mm} \end{bmatrix} = \mathbf{W}\mathbf{H}^T\mathbf{W}^{-1} \tag{31}$$

where $\mathbf{W} = \text{diag}(1, \det(\mathbf{H}_{21}), ..., \det(\mathbf{H}_{m1}))$ and $\dim(\mathbf{Q}) = (3m, 3m)$ and $\text{rank}(\mathbf{Q}) = 3$. Matrix $\mathbf{Q}$ has similar properties to the matrix $\mathbf{H}$, for example, it has an eigenvalue $\lambda = m$ of multiplicity three. The vector $\mathbf{n}$ is an eigenvector of $\mathbf{Q}$ corresponding to the eigenvalue $\lambda = m$:

$$\mathbf{Q}\mathbf{n} = m\mathbf{n} \tag{32}$$

where $\mathbf{n} = [\mathbf{n}_1^T \mathbf{n}_2^T ... \mathbf{n}_m^T]^T$. The vector can be written as a linear combination of the eigenvectors $\mathbf{n} = x\ \mathbf{v}_1 + y\ \mathbf{v}_2 + z\ \mathbf{v}_3 = \mathbf{V}\mathbf{x}$, where $\mathbf{x} = \begin{bmatrix} x\ y\ z \end{bmatrix}^T$ is a vector containing three unknowns and $\mathbf{V} = \begin{bmatrix} \mathbf{v}_1\ \mathbf{v}_2\ \mathbf{v}_3 \end{bmatrix}$ is a known matrix. Imposing the constraint $\|\mathbf{n}_k\| = 1$ and the constraints given by equation (8) we obtain:

$$\mathbf{V}_i\mathbf{x}\mathbf{x}^T\mathbf{V}_i^T\mathbf{H}_{ji}^T - \mathbf{H}_{ij}\mathbf{V}_j\mathbf{x}\mathbf{x}^T\mathbf{V}_j^T = \mathbf{H}_{ji}^T - \mathbf{H}_{ij} \tag{33}$$

from which is possible to compute the unknown matrix $\mathbf{x}\mathbf{x}^T$ and then, by singular values decomposition, the original unknown which is $\mathbf{x}$. Once find $\mathbf{x}$, the normals to the plane are extracted from $\mathbf{H}$ and knowing that $\mathbf{R}\mathbf{N} = \mathbf{N}\mathbf{O}_m$ we find:

$$\mathbf{T} = \mathbf{H}\mathbf{N} - \mathbf{0}_{3m}\mathbf{N} \tag{34}$$

$$\mathbf{R} = \mathbf{H}(\mathbf{N}\mathbf{N}^T - \mathbf{I}_{3m}) + \mathbf{N}^T\mathbf{0}_{3m}\mathbf{N} \tag{35}$$

3.4 Camera self-calibration

The super-homography can of course be used in many applications. In this section we use the properties of the set of homography matrices to self-calibrate the cameras. It should be noticed that we avoid the use of a bundle adjustment technique to impose the rank 3 constraint on the super-homography (as explained in section 3) and thus we considerably simplify the algorithm. In this case, the only unknowns are the camera internal parameters. Each independent homography will provide us two constraints on the parameters according to equation (7). Indeed, if σ_{ij}^I and σ_{ij}^{II} are the two non-zero singular values of $[\mathbf{n}_i]_\times \mathbf{H}_{ji}^T$ our self-calibration method is based on the minimisation of the following cost function [4][9]:

$$\mathcal{C} = \sum_{i=1}^{m} \sum_{j=1}^{m} \frac{\sigma_{ij}^I - \sigma_{ij}^{II}}{\sigma_{ij}^I} \tag{36}$$

A minimum of 3 independent homography matrices (4 images) is sufficient to recover the focal length and the principal point supposing $r = 1$ and $\theta = \pi/2$ and a minimum of 4 independent homography matrices (5 images) is sufficient to recover all the parameters.

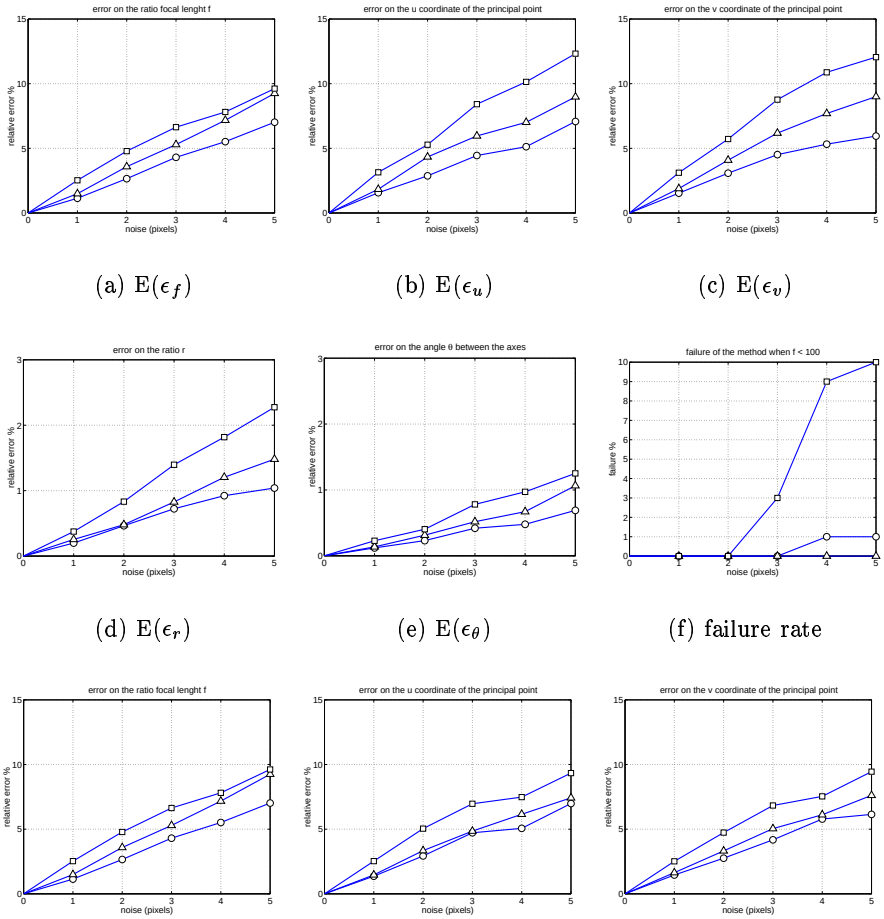
4 Experiments

The self-calibration algorithm has been tested on synthetic and real images. The results obtained with a calibration grid were compared with the standard Faugeras-Toscani method [3]. Our self-calibration algorithm is the following:

1. Match corresponding points in m images of a planar structure;
2. Compute the super-collineation imposing the rank 3 constraint using the algorithm described in Section 3.1;
3. Using an initial guess of the camera parameters compute the normalised super-homography matrix as described in Section 3.2;
4. Decompose the super-homography matrix and find the normal to the plane as described in Section 3.3;
5. Compute a new set of camera parameters which minimise the cost function given in Section 3.4 and go to step 3.

4.1 Simulations of a planar grid

The planar grid used for the simulations in section 3 was used to test the self-calibration algorithm. The experimental setup is as close as possible to the one proposed by Triggs [14]. The cameras roughly fixate a point on the plane from randomly generated orientations varying $\pm 30^\circ$ in each of the three axes. The nominal camera calibration is $f = 1000$, $r = 1$, $\theta = 90^\circ$, $u = 250$ and $v = 250$. The plane contains 64 points projected into a 500×500 image. The camera calibration varies randomly about the nominal values of $\sigma_f = \pm 30\%$, $\sigma_r = \pm 10\%$, $\sigma_\theta = 0.5^\circ$ and $\sigma_u = \sigma_v = \pm 75$ pixels (σ_f and σ_r are standard deviations of log-normal distributions while σ_θ, σ_u and σ_v of normal ones).



(a) E(ϵ_f) (b) E(ϵ_u) (c) E(ϵ_v)

(d) E(ϵ_r) (e) E(ϵ_θ) (f) failure rate

(g) E(ϵ_f) r=1, θ=90° (h) E(ϵ_u) r=1, θ=90° (i) E(ϵ_v) r=1, θ=90°

Fig. 2. Simulations results of camera self-calibration using a planar grid. The graphs (a), (b), (c), (d) and (e) show the mean of the errors on the camera internal parameters (respectively f, θ, r, u, v) obtained with the self-calibration algorithm. Graph (f) shows the failure rate of the algorithm. Finally, the graphs (g), (h) and (i) show the mean of the errors on the camera internal parameters (respectively f, u, v) obtained supposing $\theta = 90°$ and $r = 1$.

In Figure 2 we give the results obtained using 6 (lines marked with a square), 8 (lines marked with a triangle) and 10 (lines marked with a circle) images and supposing all the camera parameters unknowns. The figure represents the mean error computed on 100 trials with different parameters and different camera positions for each level of noise (the standard deviation of the Gaussian noise added to the coordinates of the points is increased from 0 to 5). The error on the principal point is given as a percentage of the focal length. The errors on

the camera parameters increase with the noise and decrease with the number of images used for self calibration. Using six images the results are still satisfactory even if the failure rate of the method increases rapidly (a failure of the method occurs when the obtained focal length is less than 100 pixels). This can be explained since it is known that with few images there is the risk of degenerate configurations [11]. The results obtained compare very favourably to the results obtained by Triggs in [14], especially considering the failure rate of the method. For example, using 10 images we obtain a mean error more than 50 % smaller than the error obtained by Triggs. Finally, in Figure 2(g), (h) and (i) we give the results of our method when r and θ are fixed to their nominal values. The error on the focal length is practically the same. On the other hand, the error on the location of the principal point is reduced since the non-linear search is now done in a three-dimensional space reducing the risk of local minima.

4.2 Self-calibration from images of a grid and comparison with the standard Faugeras-Toscani calibration method

A sequence (26 images of dimension (640×480)) of a calibration grid was taken using a Fuji MX700 camera with a 7mm lens. Figure 3 shows three images of the sequence. The corners of the black squares are used to compute the super-collineation matrix in order to self-calibrate the camera with our method.

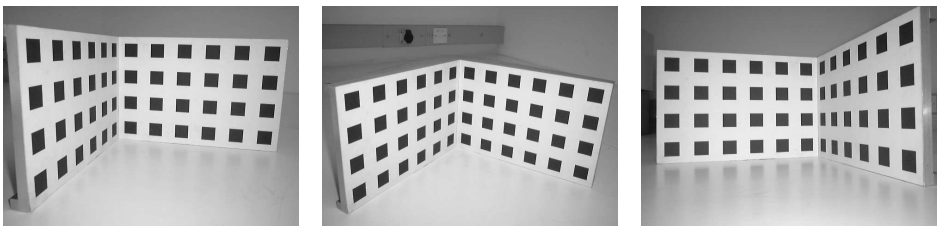


Fig. 3. Three images of the sequence taken with a digital camera. The calibration grid allows the "ground truth" to compare our method with the standard Faugeras-Toscani method. The main advantage of using our method is that we don't need any knowledge of the 3D structure of the grid to calibrate the camera. On the other hand, at least five images are needed.

Table 1 gives the results for the following experiment:

- non-planar calibration: the mean and the standard deviation on 26 images of the grid calibrated with the standard Faugeras-Toscani method initialised with the DLT linear method [1]. In this case we use both planes to calibrate the camera;

- planar self-calibration: the mean and the standard deviation on 50 tests using m images (m = 6,8,10) randomly chosen between the 26 images of the grid. The same tests are repeated using the right plane alone, the left plane alone and then again with r and θ fixed to nominal values.

The results are very good and agree with the simulations. They are more accurate than the results obtained by Triggs with a sequence of images of a calibration grid. In our case, the angle of rotation between the images of the sequence can be greater than 60° which has in general the effect to improve the results. However, this is not always true since the planes can be very close to the optical center of the camera (see in Figure 3) and in this case the estimation of the collineations is not accurate. The calibration obtained using the right plane is again very similar to the calibration obtained using the left plane. As we expected the accuracy decreases as we decrease the number of images but the worst result (obtained using only 6 images of the grid) is only an error of 2% on the focal length.

calibration method	f	r	θ	u	v
DLT linear	685 ± 3	1.0005 ± 0.0033	90.00 ± 0.14	322 ± 5	229 ± 4
Faugeras-Toscani	685 ± 3	1.0003 ± 0.0022	90.00 ± 0.16	322 ± 5	229 ± 4
right plane (10 im)	680 ± 8	0.9976 ± 0.0088	89.23 ± 0.59	318 ± 8	230 ± 8
left plane (10 im)	680 ± 6	0.9943 ± 0.0058	89.89 ± 0.30	320 ± 7	232 ± 4
right plane (8 im)	681 ± 12	0.9950 ± 0.0105	89.21 ± 0.80	315 ± 11	232 ± 9
left plane (8 im)	678 ± 12	0.9969 ± 0.0075	90.06 ± 0.24	327 ± 14	233 ± 3
right plane (6 im)	686 ± 13	0.9891 ± 0.0126	89.63 ± 0.69	312 ± 18	231 ± 11
left plane (6 im)	685 ± 10	0.9886 ± 0.0147	89.80 ± 0.59	339 ± 18	232 ± 7
Faugeras-Toscani	685 ± 3	1 ± 0	90 ± 0	322 ± 6	229 ± 4
right plane (10 im)	679 ± 6	1 ± 0	90 ± 0	318 ± 5	224 ± 8
left plane (10 im)	675 ± 6	1 ± 0	90 ± 0	325 ± 4	232 ± 4
right plane (8 im)	687 ± 6	1 ± 0	90 ± 0	323 ± 3	231 ± 6
left plane (8 im)	676 ± 4	1 ± 0	90 ± 0	343 ± 27	231 ± 12
right plane (6 im)	676 ± 8	1 ± 0	90 ± 0	314 ± 9	227 ± 5
left plane (6 im)	677 ± 18	1 ± 0	90 ± 0	327 ± 34	230 ± 31

Table 1. Results using digital images of the grid (statistics on 50 tests)

After the camera has been calibrated, the 3D reconstruction of the planes was realized. If $\mathbf{n}_{rj}$ and $\mathbf{n}_{lj}$ are respectively the normal to the right and left plane in the frame attached to the image $\mathcal{I}_j$, the angle between them is:

$$\phi_j = \cos^{-1}(\mathbf{n}_{rj}^T \mathbf{n}_{lj})$$

This angle should be the same for all the images $j = 1, 2, 3, ..., m$. In order to verify the quality of the reconstruction results we can compute the mean and the standard deviation σ over all the images. For example, the results obtained with a sequence of $m = 10$ images were:

$$\nu = \frac{1}{m}\sum_{j=1}^{m}\phi_j = 89.84, \qquad \sigma = \frac{1}{m}\sqrt{\sum_{j=1}^{m}(\phi_j - \nu)^2} = 0.21$$

4.3 Self-calibration from images of a facade

In this experiment, in order to test our algorithm in very extreme conditions, only four images of a facade (see Figure 4) were taken with the same digital camera. With such images the localisation of the corners was not accurate and with only four images we can only calibrate the focal length and the principal point (thus we fixed $r = 1$ and $\theta = 90°$). The results obtained using our self-calibration with 56 points (the corners of the windows on the facade) are $f = 678$ (1 % of the mean focal length obtained with the Faugeras-Toscani method and the calibration grid), $u = 355$ and $v = 216$. The results are very good even using images of a roughly planar structure.

Fig. 4. Four images of a facade. The corners of the windows (marked with a white cross) belong roughly to a plane. They are used to compute the super-collineation matrix from which it is possible to self-calibrate the camera.

5 Conclusion

In this paper we presented an efficient technique to impose the constraints existing within a set of collineation matrices computed from multiple views of a planar structure. The obtained set of collineations can be used for several applications such mosaicing, reconstruction and self-calibration from planes. In this paper we focused on self-calibration proposing a new method which does not need any a priori knowledge of the metric structure of the plane. The method

was tested both with synthetic and real images and the obtained results are very good. However, the method could be improved by imposing further constraints in order to obtain not only a consistent set of collineations matrices but also a consistent set homography matrices. The method could also be improved using a probabilistic model for the noise.

Acknowledgements

This work was supported by an EC (ESPRIT) grant no. LTR26247 (VIGOR). We would like to thank Paulo Mendonca and Kenneth Wong for their kind help.

References

1. Y. I. Abdel-Aziz and H. M. Karara. Direct linear transformation into object space coordinates in close-range photogrametry. In *Proc. Symp. Close-Range Photogrametry*, pages 1–18, University of Illinois, Urbana, 1971.
2. O. Faugeras and F. Lustman. Motion and structure from motion in a piecewise planar environment. *International Journal of Pattern Recognition and Artificial Intelligence*, 2(3):485–508, 1988.
3. O.D. Faugeras and G. Toscani. The calibration problem for stereo. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 15–20, June 1986.
4. R. Hartley. Estimation of relative camera positions for uncalibrated cameras. In G. Sandini, editor, *Proc. European Conf. on Computer Vision*, volume 588 of *Lecture Notes in Computer Science*, pages 579–587. Springer-Verlag, May 1992.
5. R. Hartley. Self-calibration from multiple views with a rotating camera. In *Proc. European Conf. on Computer Vision*, pages 471–478, May 1994.
6. R. Hartley. Minimising algebraic error in geometric estimation problem. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 469–476, 1998.
7. S. Maybank and O. Faugeras. A theory of self-calibration of a moving camera. *International Journal of Computer Vision*, 8(2):123–151, 1992.
8. J. Mendelsohn and K. Daniilidis. Constrained self-calibration. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 581–587, 1999.
9. P.R.S. Mendonca and R. Cipolla. A simple techinique for self-calibration. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 500–505, 1999.
10. M. Pollefeys, R. Koch, and L. VanGool. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. *International Journal of Computer Vision*, 32(1):7–25, August 1999.
11. P.F. Sturm and S.J. Maybank. On plane-based camera calibration: A general algorithm, singularities, applications. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 432–437, 1999.
12. R. Szeliski and P. Torr. Geometrically constrained structure from motion: Points on planes. In *European Workshop on 3D Structure from Multiple Images of Large-Scale Environments (SMILE)*, pages 171–186, Freiburg, Germany, June 1998.
13. B. Triggs. Autocalibration and the absolute quadric. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 609–614, 1997.
14. B. Triggs. Autocalibration from planar scenes. In *Proc. European Conf. on Computer Vision*, pages 89–105, 1998.
15. L. Zelnik-Manor and M. Irani. Multi-view subspace constraints on homographies. In *Proc. IEEE Int. Conf. on Computer Vision*, volume 1, pp. 710–715, Sept. 1999.
16. Z. Zhang and A. R. Hanson. Scaled euclidean 3D reconstruction based on externally uncalibrated cameras. In *IEEE Symposium on Computer Vision*, pp 37–42, 1995.