Linear Pose Estimate from Corresponding Conics

I. Frosio[1], A. Alzati[2], M. Bertolini[2], C. Turrini[2] and N. A. Borghese[1]

[1] *Computer Science Dept., University of Milan, Via Comelico 29/41 – 20135 Milan (Italy)*

[2] *Mathematic Dept., University of Milan, Via Saldini, 50 - 20133 Milan (Italy)*

Abstract - We propose here a new method to recover the orientation and position of a plane by matching at least three projections of a conic lying on the plane itself. The procedure is based on rearranging the conic projection equations such that the non linear terms are eliminated. It works with any kind of conic and does not require that the shape of the conic is known a-priori. The method was extensively tested using ellipses, but it can also be used for hyperbolas and parabolas. It was further applied to pairs of lines, that can be viewed as a degenerate case of hyperbola, without requiring the correspondence problem to be solved first. Critical configurations and numerical stability have been analyzed through simulations. The accuracy of the proposed algorithm was compared to that of traditional algorithms and of a trinocular vision system using a set of landmarks.

Index Terms - Conics, multi-view geometry, pose estimate, coplanar pairs of lines, critical configurations.

1 INTRODUCTION

Conics are considered, together with points and lines, one of the fundamental features in computer vision [1, 2]. They are often found on real objects and frequently adopted as reliable markers for calibration and for tracking objects in 3D space. Although points and lines lead to simpler mathematical formulation, conics contain more geometrical information: for instance, given two or more views, the correspondence between the images of the same conic is unique under some weak constraints [1]. Moreover, as the projection of a conic is again a conic, the mathematics is somewhat simplified and elegant at the same time. Lastly, since a conic is

represented by a symmetric matrix, its manipulation is generally easy [2]. Because of these

reasons, a wide range of applications based on conics has been described in literature, including

camera calibration [3], object reconstruction [4] and tracking of surgical instruments [5], robotic

arms [6] as well as human hands wearing gloves inside an immersive virtual environment [2].

Most papers on conics focus on the problem of estimating the pose of a conic in 3D Euclidean

space using a single camera [5-7]. In this case, some additional information about the curve, like

its shape or its size, is needed. For instance [7] describes a method for tracking a circle in 3D

space from a single projection, supposing that the radius of the circle is known; focus of the

paper is in the development of a reliable algorithm for tracking the elliptical projection of the

circle. In [6], a real-time algorithm for pose estimate of complex objects from a single camera is

presented; it is based on tracking simple features, including the conic projections of circles and

spheres and it uses a model of the object, based on kinematic chains, to recover the 3D pose of

the entire object.

When two or more cameras are used, a priori knowledge about the conic is not necessary to

track it [8-10, 20-21]. In [8, 11, 12] conics as well as higher order algebraic curves are considered,

but the system of equations associated to the problem includes many non linear terms: multiple

solutions are possible and a time consuming, iterative optimization algorithm is necessary.

In [9], the problem of recovering the pose of quadratic-curves (including conics) from two

projections is analyzed. The solution is defined as the intersection between the two cones

associated to the projections of the curves onto two cameras. This leads to a non linear system

of nine equations with nine unknowns. A similar approach is described in [10].

A simpler solution is proposed by Quan in [1], where the pencil of quadrics including the two

cones is first determined. The algorithm searches the unique degenerate quadric of the pencil,

that is represented by a 4x4 matrix of rank two. This is composed of two planes, one of which contains the unkown conic. Although the resulting equations are non-linear, a closed-solution is derived under reasonable hypotheses. However, this can be numerically unstable as recognized by the authors who stated that they "can never have a perfect rank two" [1].

In this paper, we propose a different approach to determine the pose of a conic observed by at least three cameras. Rearranging the projection equations we derive linear relationships and obtain a robust and simple solution. From the intersection of the estimated plane and the projection cones of the conics images, we also determine the position of the conic in 3D space. No a priori knowledge is required on the shape or the dimension of the conic. The method has been extensively applied to both ellipses and pairs of intersecting lines, that constitutes a degenerated case of hyperbola. The experimental results on simulated and real data demonstrate the accuracy and reliability achieved with the proposed method that does not suffers from some instabilities typical of [1]. Morover, it can be also used with pairs of intersecting lines without solving the line correspondence problem first.

The paper is organized as follows. Section 2 provides geometrical background; the method is described in Section 3. Section 4 gives details about the fitting algorithms, whereas Section 5 and 6 report the experimental results on real and simulated data; these are discussed in Section 7. A conclusion is finally drawn in Section 8.

2 NOTATION AND BACKGROUND MATERIAL

Let E^3 denote the 3-dimensional Euclidean space where a point is represented in homogeneous coordinates as $\mathbf{x}^T = (x_1, x_2, x_3, x_4)$ and the corresponding affine coordinates are $(x_1/x_4, x_2/x_4, x_3/x_4)$ for $x_4 \neq 0$. Each camera realizes a perspective projection, described by a 3x4

matrix of rank three, defined for the i-th camera as [14]:

$$\mathbf{P}_i = \mathbf{K}_i \left[\mathbf{R}_i^T \mid -\mathbf{R}_i^T \mathbf{t}_i \right] = \left[\mathbf{\Pi}_i \mid \mathbf{v}_i \right] \tag{1},$$

where the matrix $\mathbf{K}_i$ contains the camera internal parameters, $\mathbf{R}_i$ is the 3x3 orientation matrix of

the camera and $\mathbf{t}_i$ the position of its projection center in E^3. These are supposed to be known. By

by $\Lambda_{a,b,c} \subset E^3$ we will denote a plane, not passing through the origin, described by:

$$ax_1 + bx_2 + cx_3 = x_4 \tag{2}.$$

A conic γ on $\Lambda_{a,b,c}$ can be obtained as the intersection of a quadric surface with $\Lambda_{a,b,c}$. In

particular, γ is given here by the intersection of $\Lambda_{a,b,c}$ with the cone Γ_i projecting γ from a point

$\mathbf{t}_i \notin \Lambda_{a,b,c}$. A cone is in fact a degenerated quadric surface, represented by a 4x4 matrix of rank 3.

Let us define an arbitrary reference frame Ψ in $\Lambda_{a,b,c}$. In this frame, γ is represented in

homogenous coordinates by the equation:

$$\mathbf{y}^T \mathbf{L}_\gamma \mathbf{y} = 0 \tag{3}$$

for a suitable 3x3 symmetric matrix, $\mathbf{L}_\gamma$. Analogously the cone Γ_i in E^3 is represented by the

equation:

$$\mathbf{x}^T \mathbf{M}_{\Gamma i} \mathbf{x} = 0 \tag{4}$$

for a suitable 4x4 symmetric matrix $\mathbf{M}_{\Gamma i}$.

3 METHOD

3.1 Reconstruction of a plane from three or more views of a conic

Here we show how to compute the parameters of $\Lambda_{a,b,c}$ from the projections of a conic

$\gamma \in \Lambda_{a,b,c}$ on the image plane of three cameras. Let us indicate with γ_i the projection onto the i-th

camera; this is also a conic [14]. $\mathbf{M}_{\Gamma 1}, \mathbf{M}_{\Gamma 2}, \mathbf{M}_{\Gamma 3}$ indicate the matrices of the cones Γ_1, Γ_2 and Γ_3

that project γ onto γ_1, γ_2, γ_3 through the projection centers of the three cameras, $\mathbf{t_1}$, $\mathbf{t_2}$, $\mathbf{t_3}$. Given the matrix $\mathbf{M_{\gamma i}}$, associated to the conic γ_i measured on the i-th camera in the camera reference frame, and the camera projection matrix $\mathbf{P_i}$, the matrix $\mathbf{M_{\Gamma i}}$ is given by [14]:

$$\mathbf{M_{\Gamma i}} = \mathbf{P_i^T M_{\gamma i} P_i} \tag{5}.$$

Let us indicate with $\mathbf{L_i}$ the matrix of the conic generated by the intersection of Γ_i with $\Lambda_{a,b,c}$. Since the intersection of the three cones Γ_1, Γ_2 and Γ_3 with $\Lambda_{a,b,c}$ produces the same conic γ, the coefficients of the matrices $\mathbf{L_1}$, $\mathbf{L_2}$ and $\mathbf{L_3}$ should be equal up to a multiplicative factor. In the following, we will show how to use this observation to compute a, b and c in (2).

We first show how to get an easy-to-manipulate representation of γ in Ψ. Let us consider three points of $\Lambda_{a,b,c}$: $[1, 0, 0, a]^T$, $[0, 1, 0, b]^T$, $[0, 0, 1, c]^T$ and suppose that in Ψ they have homogenous coordinates $[1, 0, 0]^T$, $[0, 1, 0]^T$ and $[0, 0, 1]^T$. With this choice of Ψ, we can define a 4x3 matrix, $\mathbf{N}$, that expresses the relationship between a point in E^3 and a point in Ψ:

$$\mathbf{x} = \left[\begin{array}{c} \mathbf{I_3} \\ \hline a \quad b \quad c \end{array} \right] \mathbf{y} = \mathbf{Ny} \tag{6}.$$

where $\mathbf{I_3}$ is the 3x3 identity matrix. Substituting (6) into (4) we obtain:

$$\mathbf{y^T N^T M_{\Gamma i} Ny} = 0 \tag{7}.$$

Comparing (7) and (3), we get:

$$\mathbf{L_i} = \mathbf{N^T M_{\Gamma i} N} \tag{8},$$

where the $\mathbf{L_i}$ matrix contains the coefficients of γ, in the reference frame Ψ on $\Lambda_{a,b,c}$, computed starting from its projection onto the i-th camera, γ_i. This relationship is defined up to a non zero scalar value and establishes a relationship between the cones $\{\Gamma_i\}_{i=1..3}$ and the conic γ on $\Lambda_{a,b,c}$.

From (5) and (8), the matrix associated to the conic obtained cutting the cone Γ_i with $\Lambda_{a,b,c}$ can

also be written as:

$$\mathbf{L_i} = \mathbf{N}^T \mathbf{P_i^T} \mathbf{M}_{\gamma i} \mathbf{P_i} \mathbf{N} \tag{9};$$

as $\mathbf{L_1}$, $\mathbf{L_2}$ and $\mathbf{L_3}$ should be equal up to a multiplicative factor, we can write:

$$\begin{aligned}
\mathbf{L_1} &= \rho_1 \mathbf{L_2} \Rightarrow \mathbf{N^T P_1^T M}_{\gamma 1} \mathbf{P_1 N} = \rho_1 \mathbf{N^T P_2^T M}_{\gamma 2} \mathbf{P_2 N} \\
\mathbf{L_3} &= \rho_3 \mathbf{L_2} \Rightarrow \mathbf{N^T P_3^T M}_{\gamma 3} \mathbf{P_3 N} = \rho_3 \mathbf{N^T P_2^T M}_{\gamma 2} \mathbf{P_2 N}
\end{aligned} \tag{10},$$

where ρ_1 and ρ_3 are unknown multiplicative factors. Each line of (10) represents a set of nine

non-linear equations. However, since each $\mathbf{L_i}$ is symmetric, we have a total of twelve non-linear

equations with five unknowns: a, b, c, ρ_1, and ρ_3.

To sort out the non-linearity, we first rewrite (9) as:

$$\mathbf{L_i} = \mathbf{A_i} + \begin{bmatrix} a & b & c \end{bmatrix}^T \mathbf{b_i^T} + \mathbf{b_i} \begin{bmatrix} a & b & c \end{bmatrix} + \begin{bmatrix} a & b & c \end{bmatrix}^T d_i \begin{bmatrix} a & b & c \end{bmatrix} \tag{11},$$

where $\mathbf{A_i}$ is a suitable 3x3 symmetric matrix, $\mathbf{b_i}$ is a suitable 3x1 vector and:

$$d_i = \mathbf{v_i^T} \mathbf{M}_{\gamma i} \mathbf{v_i} \tag{12},$$

with $\mathbf{v_i} = -\mathbf{K_i} \cdot \mathbf{R_i}^T \cdot \mathbf{t_i}$ (cf. (1)). Eq. (11) is therefore the sum of one constant, one linear term in [a, b, c]

and a quadratic one.

We choose now the reference frame in E^3 such that the projection center of the second camera

lies in the origin. This implies that $\mathbf{t_2} = [0\ 0\ 0]^T$ and, therefore, $d_2 = 0$ and $\mathbf{P_2} = [\mathbf{\Pi_2} | \mathbf{0}]$. With this

choice, the product of matrix $\mathbf{P_2 N}$ in (9) gives:

$$\mathbf{P_2 N} = \begin{bmatrix} \mathbf{\Pi_2} | \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T \end{bmatrix} \begin{bmatrix} \mathbf{I_3} \\ \hline a & b & c \end{bmatrix} = \mathbf{\Pi_2} \tag{13},$$

which does not depend on a, b, c. Therefore, $\mathbf{L_2}$ can be re-written as:

$$\mathbf{L_2} = \mathbf{\Pi_2^T} \mathbf{M}_{\gamma_2} \mathbf{\Pi_2} \tag{14}$$

and it can be computed from the coefficients of the conic observed by the second camera and from the projection matrix of the same camera. With this choice, the right members of (10) do not contain anymore the unknowns a, b, c. Combining equations (10) we can now eliminate the non-linear terms appearing on the left side. In particular, we consider the following combination of (10):

$$d_3\mathbf{L_1} - d_1\mathbf{L_3} = (d_3\rho_1 - d_1\rho_3)\mathbf{L_2} \tag{15}.$$

Substituting (11) into (15), we get:

$$d_3\left\{\mathbf{A_1} + \begin{bmatrix} a \\ b \\ c \end{bmatrix}\mathbf{b_1^T} + \mathbf{b_1}\begin{bmatrix} a \\ b \\ c \end{bmatrix}^T + \begin{bmatrix} a \\ b \\ c \end{bmatrix}d_1\begin{bmatrix} a \\ b \\ c \end{bmatrix}^T\right\} +$$

$$-d_1\left\{\mathbf{A_3} + \begin{bmatrix} a \\ b \\ c \end{bmatrix}\mathbf{b_3^T} + \mathbf{b_3}\begin{bmatrix} a \\ b \\ c \end{bmatrix}^T + \begin{bmatrix} a \\ b \\ c \end{bmatrix}d_3\begin{bmatrix} a \\ b \\ c \end{bmatrix}^T\right\} = (d_3\rho_1 - d_1\rho_3)\mathbf{L_2} \Rightarrow \tag{16}.$$

$$(d_3\mathbf{b_1} - d_1\mathbf{b_3})\begin{bmatrix} a \\ b \\ c \end{bmatrix}^T + \begin{bmatrix} a \\ b \\ c \end{bmatrix}(d_3 b_1^T - d_1 b_{31}^T) + (d_3\mathbf{A_1} - d_1\mathbf{A_3}) = (d_3\rho_1 - d_1\rho_3)\mathbf{L_2}$$

Eq. (16) is constituted of nine linear equations, in the five unknowns a, b, c, ρ_1 and ρ_3. However, as both the right and left sides of (16) are constituted by a 3x3 symmetric matrix, there are only six independent equations.

Notice that the 6x5 matrix $\mathbf{E}$ associated to the linear system (16) is not of maximal rank, as the columns of ρ_1 and ρ_3 are linearly dependent. However, for a general choice of the positions of the cameras and of the conic, one has $rank(\mathbf{E}) = 4$. To show this it is enough to prove that $rank(\mathbf{E}) = 4$ for a particular choice of the set up, and this indeed occurs in all the examples we have considered; in fact, for a general principle in algebraic geometry, either $rank(\mathbf{E})<4$ occurs for every choice, or only for some particular configurations. In other words, in the space Σ

parametrizing all configurations of cameras and conics, those which correspond to $rank(\mathbf{E})<4$ are a subvariety whose dimension is less than that of Σ.
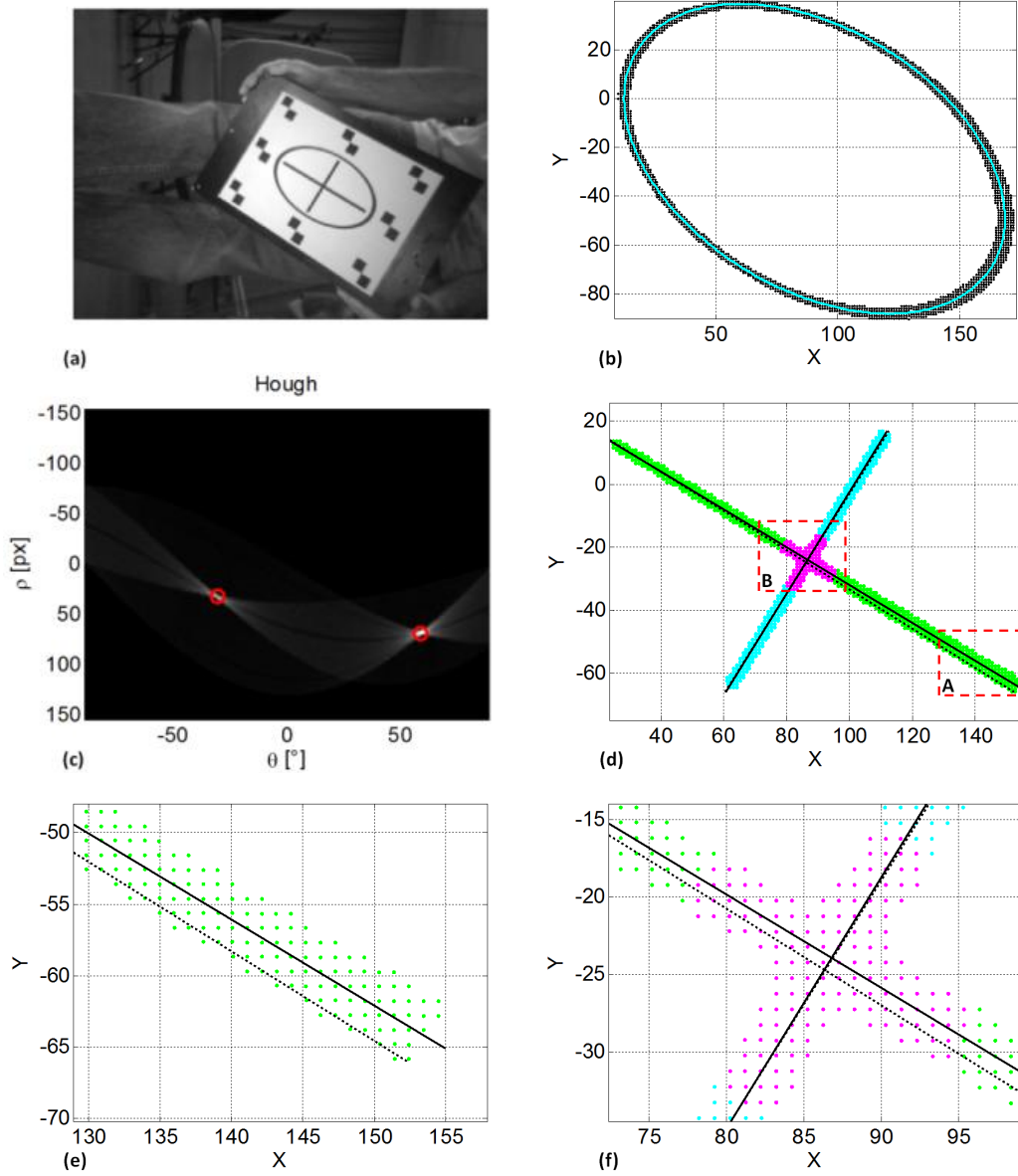

Fig. 1. Panel (a) shows one of the images used for validation. Panel (b) shows the points sampled over the ellipse in panel (a) and the fitted ellipse for the same image. Panel (c) represents the HT of the points on the intersecting segments for the same image; two local maxima are highlighted by circles. In panel (d), the points associated to the first line are depicted in green, those to the second one in cyan, whereas magenta points are not classified to any of the two; the lines identified by HT are represented as black, dashed lines $[(\rho_1, \theta_1) = (26\ pixel, 58°),\ (\rho_2, \theta_2) = (86\ pixel, -32°)]$, whereas the continuous lines are those identified by the proposed algorithm $[(\rho_1, \theta_1) = (25.48\ pixel, 58.99°),\ (\rho_2, \theta_2) = (86.73pixel, -31.62°)]$. Panels (e) and (f) are zooms of the rectangles A and B in panel (d); they highlight the difference between the lines estimated with the HT and the proposed method.

To estimate a, b, c, ρ_1 and ρ_3 the system (16) is solved through the More-Penrose pseudoinverse. Resorting to the pseudoinverse is actually necessary, as the matrix of the linear system (16) has rank four. The null space of the linear system is composed by a vector with null components associated to the unknowns a, b and c, and non null components for ρ_1 and ρ_3.

==We remark that the parametric representation of $\Lambda_{a,b,c}$ given in (2) does not allow representing the planes through the origin. Therefore, all the 3D planes that pass through the center of the second camera cannot be determined (cf. Fig. 7). Notice, however, that this issue is easil solved moving the origin of the reference system to a different camera.==

3.2 Application of the proposed method to pairs of intersecting lines

The method described in 3.1 can be applied to any kind of conic γ. In particular, no assumption has been made on the rank of the matrix $\mathbf{L_\gamma}$ representing γ in $\Lambda_{a,b,c}$. The approach can therefore be applied also to pairs of lines that can be interpreted as a degenerate form of hyperbola [14]. Representing each line in homogenous coordinates and in polar form, the equation of a degenerate hyperbola is:

$$[y_1 c(\vartheta_1) + y_2 s(\vartheta_1) - \rho_1 y_3][y_1 c(\vartheta_2) + y_2 s(\vartheta_2) - \rho_2 y_3] = 0 \qquad (17),$$

where c(.) and s(.) indicate respectively cos(.) and sin(.), the parameters $(\rho_i, \theta_i)_{i=1..2}$ represent a line passing at a distance ρ_i from the origin at an angle $\pi/2 - \theta_i$ with respect to the x axis. This equation is rearranged as:

$$\mathbf{y^T L_\gamma y} = \mathbf{y^T} \begin{bmatrix} 2c(\theta_1)c(\theta_2) & [c(\theta_1)s(\theta_2) + s(\theta_1)c(\theta_2)] & -[\rho_1 c(\theta_2) + c(\theta_1)\rho_2] \\ [c(\theta_1)s(\theta_2) + s(\theta_1)c(\theta_2)] & 2s(\theta_1)s(\theta_2) & -[\rho_1 s(\theta_2) + s(\theta_1)\rho_2] \\ -[\rho_1 c(\theta_2) + c(\theta_1)\rho_2] & -[\rho_1 s(\theta_2) + s(\theta_1)\rho_2] & 2\rho_1\rho_2 \end{bmatrix} \mathbf{y} = 0 \qquad (18),$$

which is a quadratic form associated to a symmetric matrix with null determinant and rank two. Therefore it represents a degenerate conic, and more specifically, as the discriminant is

positive, a degenerate hyperbola (cf. (3)).

3.3 Traditional linear reconstruction of a plane from pairs of correspondent lines

A more traditional approach, that will be considered in the following for comparison with the proposed method, is described here. Once the correspondence between lines is solved, estimating the pose of $\Lambda_{a,b,c}$ becomes a linear problem [14]. In fact, for any unknown line, l_k, in 3D space, its projection l_{ki} and l_{kj} on the i-th and j-th cameras, generates two planes π_{ki} and π_{kj}. The intersection of π_{ki} and π_{kj} provides the line l_k in 3D space. The orientation of the plane containing the conic can be obtained as the cross product of the director cosines of the pair of intersecting 3D lines considered. This procedure is repeated for each pair of cameras and the final plane orientation is computed as the average orientation. The plane position is finally computed imposing that the plane passes through the 3D point given by the triangulation of the intersection point of the pair of lines on each camera.

4 CONICS FITTING

In this paragraph, we describe first how to fit an ellipse to a set of N points, $\{\mathbf{p}_j = (x_j, y_j)\}_{j=1..N}$, measured on the image plane of a camera, but the same method could be used without modifications to fit hyperbolas or parabolas. Notice, however, that these conics are less frequently encountered in a real scenario, and therefore they have not been considered here. Although the methodology is well known [15, 16, 19], the procedure is reported here for sake of completeness. We then describe the method that we have developed for robust fitting of intersecting lines. In both cases, we assume that points are measured on a calibrated camera and that lens distortion has been adequately corrected by a proper distortion model, whose parameters have been estimated during the calibration phase.

4.1 Ellipse fitting

We indicate with the vector $\mathbf{e}=[e_1, e_2, \ldots, e_6]^T$ the ellipse of equation

$e_1 \cdot x^2 + e_2 \cdot x \cdot y + e_3 \cdot y^2 + e_4 \cdot x + e_5 \cdot y + e_6 = 0$. Once $\mathbf{e}$ has been fixed, the algebraic error of the measured j-

th point $\mathbf{p}_j$ is given by $e_1 \cdot x_j^2 + e_2 \cdot x_j \cdot y_j + e_3 \cdot y_j^2 + e_4 \cdot x_j + e_5 \cdot y_j + e_6$. Considering the entire set of points, the

total algebraic error can be written as:

$$E = \mathbf{e}^T \mathbf{A}^T \mathbf{A} \mathbf{e} \qquad (19),$$

where $\mathbf{A}$ is a Nx6 matrix, whose j-th row is $[x_j^2, x_j \cdot y_j, y_j^2, x_j, y_j, 1]$. Minimizing E with respect to $\mathbf{e}$

allows estimating the ellipse coefficients. Nevertheless, to avoid the trivial solution $\mathbf{e} = \mathbf{0}$, the

parameter vector $\mathbf{e}$ has to be constrained; accordingly to the LIN algorithm [15], we use the

constraint $||\mathbf{e}||^2 = 1$. The resulting constrained minimization problem is formulated using the

the Lagrange multipliers which leads to the minimization of the following Lagrangian function:

$$L = \mathbf{e}^T \mathbf{A}^T \mathbf{A} \mathbf{e} + \lambda \left(\mathbf{e}^T \mathbf{e} - 1 \right) \qquad (20)$$

instead of (19). Setting the derivatives of L with respect to $\mathbf{e}$ and λ equal to zero, we get:

$$\begin{cases} \partial L / \partial \mathbf{e} = 2\mathbf{A}^T \mathbf{A} \mathbf{e} + 2\lambda \mathbf{e} = 0 \\ \partial L / \partial \lambda = \mathbf{e}^T \mathbf{e} - 1 = 0 \end{cases} \qquad (21),$$

Eq. (21) highlights that a solution $\mathbf{e}$ is an eigenvector of the matrix $\mathbf{A}^T \mathbf{A}$. More specifically, six

eigenvectors of $\mathbf{A}^T \mathbf{A}$ can be computed. Since we do not constrain the conic to any specific type,

each eigenvector can represent an ellipse, a hyperbola or a parabola, depending on the value

assumed by $\Delta = e_2^2 - 4e_1e_2$. The eigenvector representing the best fitting ellipse is that associated to

the smallest absolute eigenvalue, among all the solutions characterized by $\Delta = e_2^2 - 4e_1e_3 < 0$.

To obtain a numerically more stable result, as suggested in [16], the coordinates of the ellipse

points are normalized before computing $\mathbf{e}$; the points are shifted such that their barycenter is

coincident with the origin of the reference system and a scale factor is applied such that the mean distance of the points from the origin is equal to $\sqrt{2}$. Ellipse fitting is performed on these normalized data and the true coefficients are obtained applying the inverse scale factor and shift. From the vector **e**, the matrix of the ellipse is then obtained as:

$$\mathbf{L} = \begin{bmatrix} e_1 & e_2/2 & e_4/2 \\ e_2/2 & e_3 & e_5/2 \\ e_2/2 & e_5/2 & e_6 \end{bmatrix} \tag{22}.$$

A typical example of fitted ellipse is shown in Fig. 1.

4.2 Fitting pairs of straight lines

A traditional approach for identifying set of lines in an image is based on the Hough transform (HT) [17]. This is a 2D discretized accumulation space, where the i-th cell with coordinates (θ_i, ρ_i) represents the line $x \cdot c(\theta_i) + y \cdot s(\theta_i) = \rho_i$. The HT is computed from the set of points, extracted from an image, that belong to any of the lines. The value contained in each cell is initialized to zero and for each extracted point, all the cells representing a line passing through that point are increased by one. After all the points have been considered, the local maxima of the HT indicate the lines in the image (Fig. 1c). HT usually produces a reliable estimate of the lines whose parameters are determined with an accuracy which depends on the cells size. When these are large, like in Fig. 1c, local maxima of the HT can be easily identified; however the parameters accuracy is not sufficient for high accuracy measurements. If the cell size decreases, the HT tends to become sparse and noisy and the identification of the local maxima problematic. Moreover, also the computational cost increases. For this reason, HT is used here only to compute approximately the position of the pairs of intersecting lines.

On the other hand, a highly accurate estimate of the polar coordinates (θ, ρ) of a line through

a set of points can be obtained through *total least squares* (also known as *orthogonal* or *Deming*

regression). However, this approach computes with high accuracy the parameters of a single line

but it cannot estimate the parameters of two lines together. ==To overcome this problem, we have==

==adopted== the following three steps procedure. First, we use the HT to identify the two lines with

an angular and spatial resolution of 1° and 1 pixel respectively. Then, we assign each point

$\{\mathbf{p}_j\}_{j=1..N}$ either to the first line or to the second one or to none of them, depending on the

distance of $\mathbf{p}_i$ from the two lines deteremined through the HT. Critical pixels close (less than ten

pixels in the present case) to the intersection of the two lines are not assigned to any of them

(Fig. 1d). Notice that the classification induced by this method did always provide a correct

classification of the data during the validation of the method. In the third phase, we accurately

estimate the parameters of each line considering only the subset of points assigned to it through

total least squares. Lastly the matrix of the degenerate conic is derived as described in 3.2. A

typical result is shown in Fig. 1d-f. In Appendix A, we show how the *total least squares* estimate

of the line parameters (θ, ρ) can be computed in closed form through algebraic manipulation.

5. RESULTS ON REAL DATA

We have extensively tested the method with a trinocular system composed of three Basler

A601f B&W cameras, 656x491 pixels @8bpp, equipped with 8mm lens, externally triggered. A

ring illuminator of red leds was mounted around the lens of each camera to increase the overall

contrast making image binarization more robust. The three cameras looked approximately at

the point [0mm 0mm 700mm] and were disposed as shown in Fig. 2a. Each camera was

calibrated through the Zhang method implemented in Matlab [18]. To this aim, 33 images of a

7x5 chessboard of 140mm x 200mm, with a random pose in 3D space, were acquired by the

three cameras. The standard deviation of the projection error never exceeded 0.06 pixels for

each camera, with a maximum error of 0.25 pixels. The registration error was of the same order
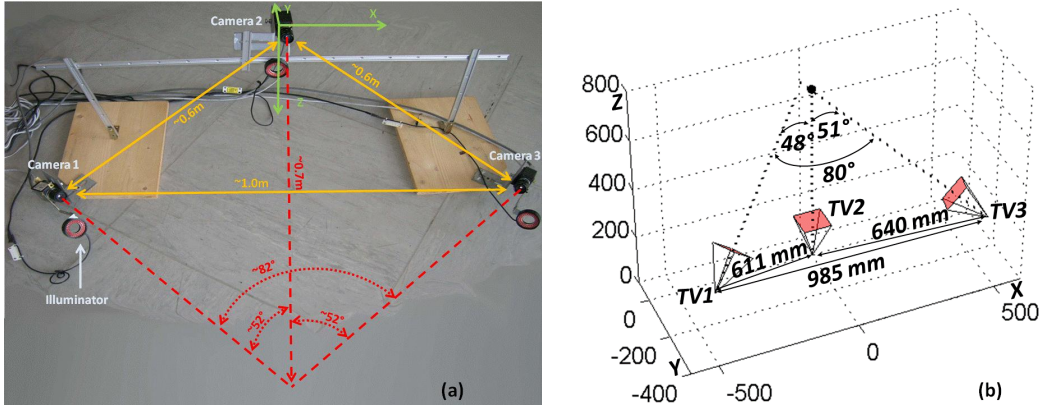
of magnitude.

Fig. 2. In panel (a), the acquisition system used for validation is shown. Panel (b) shows the set-up used in simulation. The cameras projection center, in mm, is respectively [-474, -310, 230], [0, 0, 0] and [511 -310 230]. The cameras look respectively at the points [-50, 0, 700], [0, 0, 700] and [-50, 0, 750] and have a normalized focal length of 800 pixels.

To validate the method an ad hoc, high precision mechanical system could be used to

position the plane with sufficiently high accuracy and obtain the ground truth poses. However,

this solution would be too costly and we have resorted to an indirect estimate as the best fitting

plane of a redundant set of fiducial markers coplanar with the plane acquired with the same

trinocular system.

A set of patterns was printed on an A4 sheet, stitched to a thick wooden plane (Fig. 1a). These

were: an ellipse with axis lengths of 178mm and 109mm and line thickness of 5mm; two

orthogonal segments of length 159mm and 97mm, thickness of 5mm; eight pairs of squares

(fiducial markers) of 17mm x 17mm, placed on the perimeter of a rectangle of 243mm x 167mm.

The ellipse center, the intersection of the two segments and the barycenter of the fiducial

markers were coincident.

We moved randomly the wooden plane inside the working volume, acquiring triplets of images at 1Hz. After discarding all the frames in which the patterns were not completely visible to all cameras, a total of 140 triplets of images were obtained for the analysis. The Harris corner detector was used to identify the accurate 2D position of the fiducial markers and their 3D position was computed through standard triangulation [14]. For each pose, the ground truth plane was computed from the Principal Component Analysis (PCA) of the 3D position of the eight fiducial markers. The average distance of the fiducial markers from the estimated plane, measured over the 140 frames, was 0.06mm with a maximum error of 0.32mm.

To determine the plane orientations with the proposed method, we first identified the area delimited by the fiducial markers and binarized it through the threshold set by the Otsu procedure. The pixels of the resulting largest connected component belong to the ellipse, whereas those of the second largest connected component to the pairs of intersecting segments. The mean number of pixels of the ellipse was 1935 ± 339, whereas that of the two segments was 1064 ± 185. Fig. 1 shows a typical example of the identified sets of points. A conic was fitted to these sets of points according to the procedure described in Section 4 and the plane containing the conic determined with the procedure described in Section 3.

The accuracy of the proposed algorithm was compared with the non-linear method of Quan described in [1], which allows estimating the pose of a plane when two cameras observe the same conic in 3D space. Quan's method was applied here to each of the three pairs of cameras that were available in our trinocular system. The pose estimate was averaged over all the pairs to compare the results with those provided by the method described here. We remark that Quan's method could not be applied to pairs of lines, since in this case, all the quadrics of the

Quan's pencil are singular, hence the determinants of the associated matrices are identically zero, and a 0/0 division is obtained. When pairs of lines were considered, results for the proposed method have been compared with the procedure described in Section 3.3, that requires the correspondence problem to be solved in advance.

For each method, the error on plane orientation was computed as the angle between the normal to the ground truth plane and that to the estimated plane. The 3D position error was measured as the distance between the barycenter of the eight fiducial markers and the estimated center of the ellipse or with the intersection point of the pair of segments.

The measured errors for all the methods considered here are reported numerically in table I and graphically in Figure 3. As it can be appreciated, when the proposed algorithm is adopted, the error with the intersecting lines is generally larger than that obtained using ellipses. Moreover, the error is highly variable across trials.

When Quan's method is applied to one pair of cameras, its accuracy is lower than our method, for both plane orientation and position (not reported here). When three cameras are used, on the other hand, Quan's method achieves a slightly higher accuracy with respect to the proposed method, although not significant from a statistical point of view. Similar results are obtained for intersection lines, where the linear method in Section 3.3 is applied.

In both cases, however, critical configurations produced very high error in few specific frames: for instance at frame #93 an orientation error of 111° was obtained with the method in Section 3.4, whereas 5% of the frames had an orientation error larger than 4° with the proposed method. As a result, error distribution is asymmetric and it contains outliers. For this reason, median and Inter Quartile Range (IQR) values have been reported in Table I.

TABLE I. MEDIAN AND IQR OF THE ORIENTATION AND POSITION ERRORS MEASURED OVER 140 FRAMES.

| | ERROR: PROPOSED METHOD | | | | ERROR: OTHER METHODS | | | |
| | ELLIPSE | | PAIR OF LINES | | QUAN(ELLIPSE) | | LINEAR (PAIR OF LINES) | |
	ANGLE [°]	DIST [MM]	ANGLE [°]	DIST [MM]	ANGLE [°]	DIST [MM]	ANGLE [°]	DIST [MM]
MEDIAN	0.06	0.01	0.36	-0.08	0.04	-0.01	0.17	-0.06
IQR	0.04	0.14	0.59	0.20	0.08	0.04	0.06	0.19
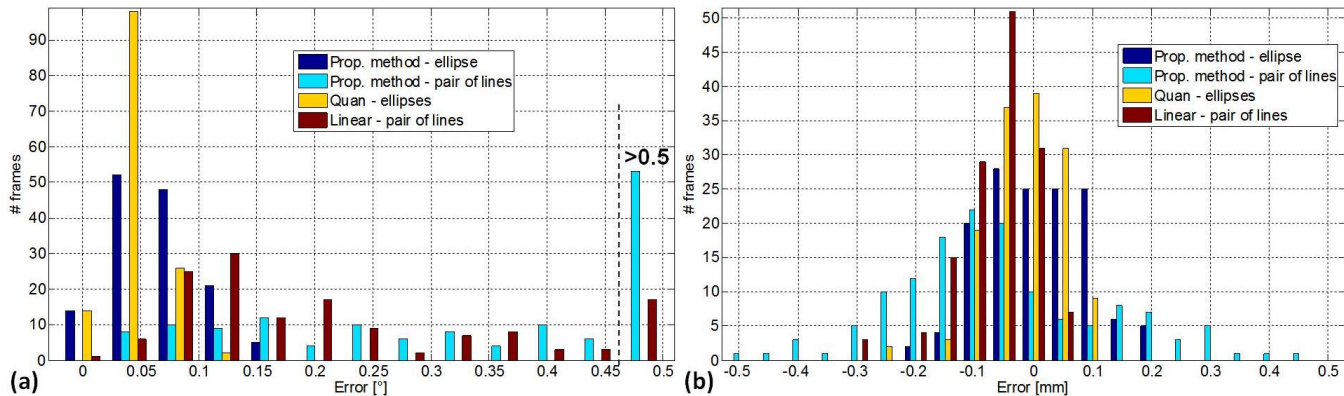
Fig 3. In panel (a), the histogram of the orientation error is plotted for the proposed method applied to ellipses and pairs of lines, for the Quan's method [1] and for linear reconstruction of 3D lines, with bins of 0.04°; panel (b) shows the histogram of the error on plane 3D position, with bins of 0.05mm.

6. RESULTS ON SIMULATED DATA

To deeper investigate and validate the results obtained on real data, and to identify the critical configurations for the proposed, Quan's and the linear reconstruction methods, we have resorted to accurately controlled simulations. We have first determined the uncertainty on the conics parameters induced by fitting and then investigated the effect of several set-up parameters on the estimate accuracy of the plane, taking into account such uncertainty.

6.1 Error in conics fitting

We considered a set of 420 ellipses, whose coefficients are equal to those computed in the validation described in Section 5 over the three cameras and points along each ellipse were

sampled with a constant step to obtain a number of points approximately equal to the real-case.

A random quantity with uniform distribution in the range of -2.5 to 2.5 pixels was then added

to each coordinate of these sampled points, to get an elliptical line with thickness of about 5

pixels, like in the real scenario (see Fig. 1b for a visual comparison). Lastly, a zero mean

Gaussian noise with standard deviation of 0.06 pixels was added to each point, to simulate the

residual calibration and measurement errors.

The fitting accuracy was quantified by the distance between the center of the true and the

fitted ellipses, by the difference of their axes length, and by the angle between the axes of the

true and of the fitted ellipse. These data are reported in Table II. The same procedure was

adopted for pairs of lines; in this case, we measured the distance between the intersection of the

true and fitted lines and the difference of orientation for each line. These quantities are also

reported in Table II.

TABLE II. MEAN ABSOLUTE VALUE (MAE) AND STANDARD DEVIATION OF THE NUMBER OF POINTS, ERROR ON THE CENTER POSITION, AXES LENGTH AND ORIENTATION FOR A TYPICAL SET OF 420 SIMULATED ELLIPSES AND PAIRS OF LINES USED IN SECTION 5.

		ELLIPSES		PAIRS OF LINES	
		MAE	STD	MAE	STD
# POINTS		1935	339	1064	185
ERRORS	CENTER [PIXELS]	0.066	0.039	0.022	0.025
	ORIENTATION [°]	0.07	0.06	0.04	0.05
	AXES LENGTH [PIXELS]	0.082	0.062	---	---

6.2 Critical geometrical configurations for the proposed algorithm: set-up

The set-up in Fig. 2b, similar to that used in Section 5, was used for evaluating through

simulations the critical aspects of the proposed algorithm. The center of the working volume

was assumed to be $\mathbf{P_0}$ = [0mm 0mm 700mm]^T but the cameras did not look exactly at this point

to avoid symmetries that may hide numerical issues associated to critical configurations.

In each simulation, the conic γ was positioned on a plane in 3D space and analytically

projected through (9) onto the image plane of the three cameras, to get γ_1, γ_2 and γ_3. Noise was

added to their parameters, coherently with the fitting errors reported in Table II. The conics

γ_1, γ_2 and γ_3 were finally used to estimate $\Lambda_{a,b,c}$ through the method described in Section 3, and

the resulting pose errors was then computed.

In each simulation, the ellipse was initially centered in $\mathbf{P_0}$ and it lied on a plane parallel to XY;

its axes were set equal to 109mm and 178mm respectively. In the following, we will refer to this

reference ellipse as $\gamma^e{}_0$. Similarly we define $\gamma^l{}_0$, the pair of intersecting lines intersecting in $\mathbf{P_0}$ at

an angle of 90 degrees and lying on a plane parallel to XY. In the different simulations, $\gamma^e{}_0$ and

$\gamma^l{}_0$ will be differently modified to investigate several geometrical aspects.

In the first simulation, we modify the axes ratio of $\gamma^e{}_0$ to assess its impact on the accuracy of

the estimated parameters. In particular, we consider the following axes ratio: (1; 0.9; 0.75 and

0.5) maintaining the same area of 109 x 178 π mm^2. For each axes ratio, 50,000 different conics, γ,

were generated rotating $\gamma^e{}_0$ around the Z axis by a random angle (uniform distribution between

0° and 360°). Each conic γ was then translated by a random quantity (standard deviation of

25mm) in X and Y directions and further rotated around two axes parallel to X and Y through

the conic center. These rotations were distributed as a zero mean Gaussian with standard

deviation of 10°. With the same sequence of geometrical transformations we have investigated

the effect of the angle between the two intersecting lines on the plane estimate. Angles of 10°,

20°, 40°, 60° and 90° (the latter is the one used in Section 5) were considered.

In the second simulation, we have investigated how the closeness of $\Lambda_{a,b,c}$ to the projection

center of one of the cameras influences the accuracy. To this aim, we have rotated γ^e_0 and γ^l_0 around an axis parallel to the Y through $\mathbf{P_0}$ by an angle, α, ranging from -100° to 100° with steps of 1°. For $\alpha = \pm 90°$ the plane was vertical and passed through the projection center of the second camera; for $\alpha \approx -45°$ and $\alpha \approx 43°$ it passed respectively through the center of the first and third camera. For each α, 1,000 different conics γ were generated from the rotated γ^e_0 and γ^l_0, further rotating it by a random angle (uniformly distributed from 0° to 360°) around an axis normal to $\Lambda_{a,b,c}$ through $\mathbf{P_0}$. The conic was then translated inside the $\Lambda_{a,b,c}$ plane by a random vector (standard deviation of 25mm in each direction) and finally projected on the cameras image plane.

In the third simulation, we investigated the situation in which $\Lambda_{a,b,c}$ passes close to the projection center of two cameras. To this aim, we have rotated γ^e_0 and γ^l_0 around an axis through $\mathbf{P_0}$ and parallel to the X axis by an angle, β, from -61° to -51° with steps of 0.25°; for $\beta \approx -56.5°$ the plane passed simultaneously through the center of the first and third camera.

The fourth simulation was aimed to highlight a particular critical configuration associated to pairs of intersecting lines that explains the large orientation error beyond the 95th percentile (Table I). In this simulation, γ^e_0 and γ^l_0 are rotated around $\mathbf{P_0}$ by a small angle ζ, such that $\Lambda_{a,b,c}$ is not anymore parallel to the XY plane; we have somehow arbitrary chosen $\Lambda_{a,b,c}$ with normal [0.099 0.099 0.990]. We then further rotate the conics around $\mathbf{P_0}$ by an angle δ, ranging from 0° to 180° with steps of 1°. For each value of δ, 1,000 samples of γ were generated translating the conics along a random vector inside the $\Lambda_{a,b,c}$ plane (standard deviation of 25mm in each direction).

6.3 Critical geometrical configurations for the proposed algorithm: results

The median and the 95th percentile of the orientation and position errors of the first simulation are summarized in Figs. 4. For the ellipses, the highest accuracy was obtained for an axes ratio equal to 1 (circles) and it decreases with the decrease of the axes ratio. For pairs of lines, the highest accuracy was achieved when the two lines were orthogonal and decreases with the decrease of their angle. This is further investigated in Figs. 5 and 6, where the estimated normal to the plane is projected onto the true plane. When circles are used the error assumes an isotropic distribution with small amplitude; with the decrease of the axes ratio the error tends to be elongated in the direction of the minor axis and increases in amplitude. A similar result is obtained for pair of lines and it is represented by the green points in Figs. 6a-e; in this case, the error increases when the intersection angle decreases from 90° to 10°. Fig. 6 also highlights that, in a significant number of cases (black points in the figure), the error is orthogonal to one of the two lines, thus meaning that the estimated plane is rotated around one of the two lines with respect to the true one. Such behavior is associated to a geometrically critical configuration which will be analyzed in the fourth simulation.

The median orientation and absolute position error of the second simulation are shown in Fig. 7 as a function of the plane rotation, α, around an axis parallel to the Y axis. Notice that the error is not influenced by the closeness of the plane to the center of the first or third camera as no peak is present for $\alpha \approx$ -45°, 43°. On the other hand, when the plane is close to the center of the second camera, the position error increases for both ellipses and pairs of lines (see the peaks close to ±90°). Moreover, the error increases with α for the ellipses as these tend to be elongated in the view of the second camera.

Fig. 8 refers to the third simulation: it shows the plane estimation errors for different

rotations, β, around an axis parallel to the X axis. A peak in the error is evident when the plane

is close to the center of camera 1 and camera 3: in this situation the orientation error decreases

while the position error increases. Position error is expressed here through the Inter-Quartile

Range (IQR) because its median is close to zero for any value of β. IQR is typically used for

reliable dispersion estimate for non Gaussian distributions and / or in presence of outliers as in

this case (cf. Fig. 4).
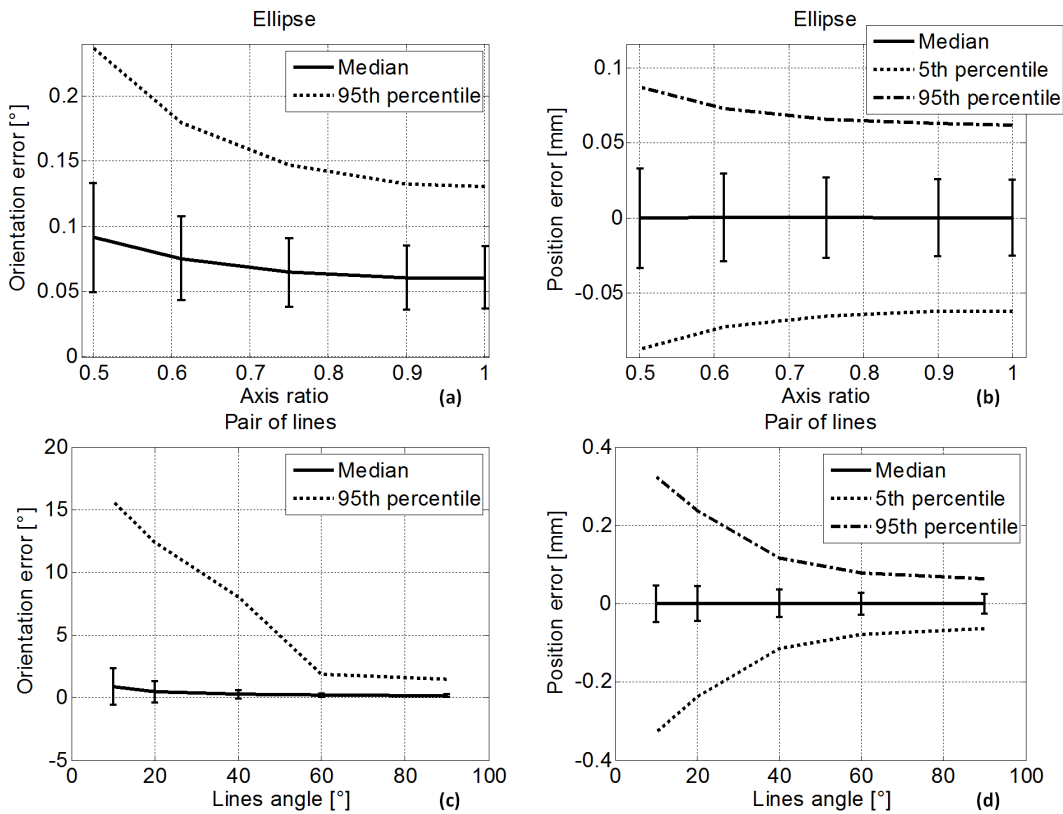
Fig. 4. In panel (a), the median and the 95th percentile of the orientation error are reported as a function of the ellipse axes

ratio; in panel (b), the median position error together with the 5th and 95th percentile are shown. In panel (c), the median

orientation error and its 95th percentile are reported as a function of the angle between the two lines of the degenerate

hyperbola; in panel (c), the median position error together with the 5th and 95th percentile are shown. The vertical bars

indicate the inter quartile range.

Fig. 9 refers to the fourth simulation. It shows the plane estimation error for different rotations, δ, of the conic around $\mathbf{P_0}$ and highlights some criticalities in the set-up. When the ellipse is used, both position and orientation errors exhibit a one period oscillation for δ increasing from 0° to 180°. This is due to the interaction between the inclination of the plane $\Lambda_{a,b,c}$ with respect to the plane XY, (angle ζ), and the rotation angle δ, that change the apparent eccentricity of the ellipse viewed by the cameras. A similar trend is shown by the position error when pairs of intersecting lines are used, while the orientation error has, in this case, a particular behavior with six clear peaks (Fig. 9a).

The peaks are associated to the situation in which one of the two intersecting lines gets close in 3D space to one of the lines that join the center of two cameras. This situation is illustrated in Fig. 10, where one of the two intersecting lines, K_2, and the line joining camera 1 and 3, K_{13}, belongs to the same plane Π that contains the point of intersection of K_2 and K_{13}, $\mathbf{Q}$. In this configuration, γ_1 and γ_3 would allow estimating the plane Π but not K_2 and therefore the conic itself. In fact, any other line in Π generates the same projections onto these two cameras. By a geometrical point of view, adding a third camera out of Π should resolve the degeneracy, but it does not as shown by experimental results. This is due to a degeneracy induced by algebraic manipulation as explained in details in the next section. Notice that this critical configuration occurs also when the intersection point $\mathbf{Q}$ lies at infinity, that is when the line K_2 is parallel to K_{13}.
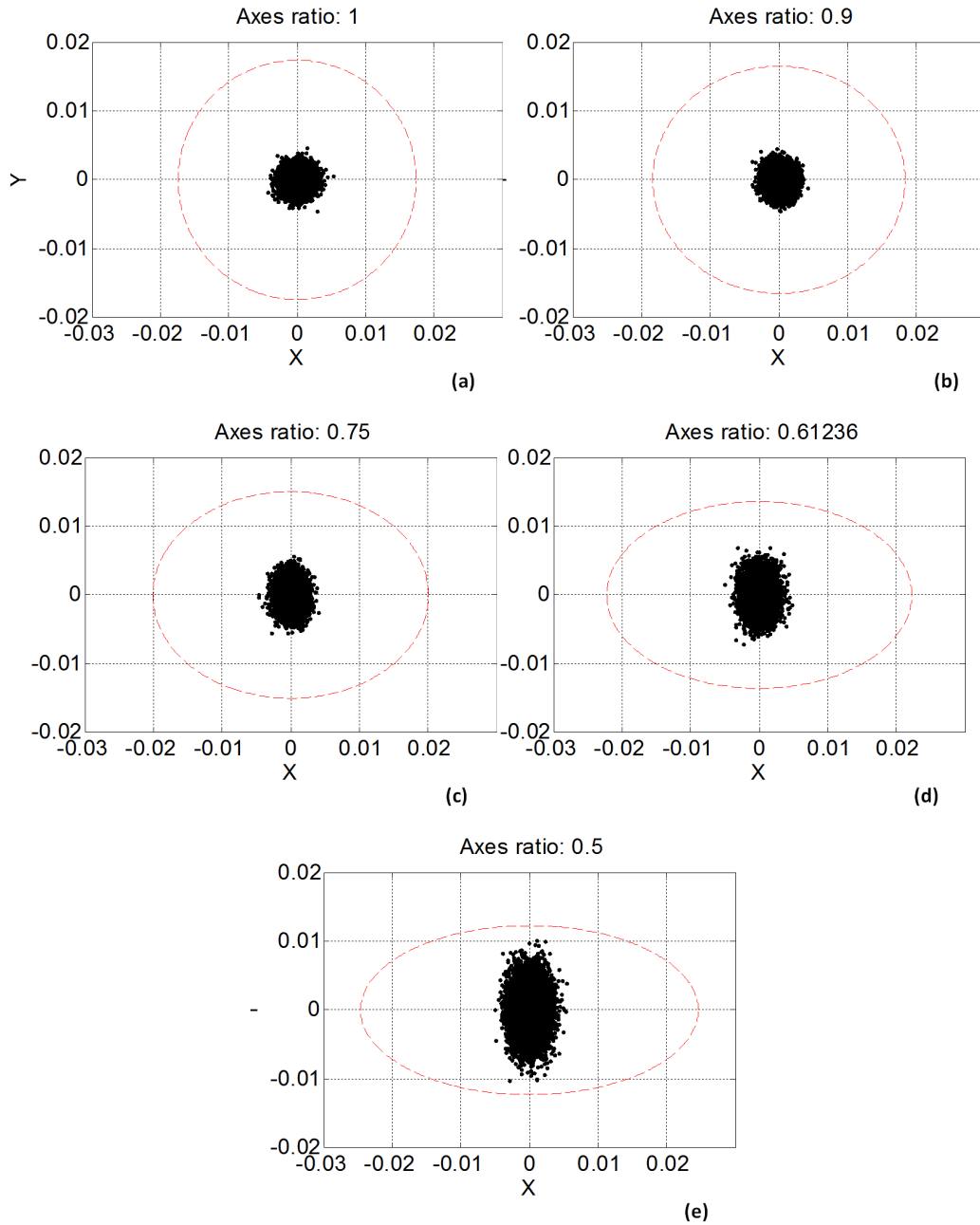
Fig. 5. Panels (a)-(e) show the projection of the estimated normal to $\Lambda_{a,b,c}$ onto the true plane, when the method in Section 3 is applied to ellipses with the same area, but different axes ratios. The true normal projects in [0, 0]. The dash-dotted red line shows the shape of the ellipse onto $\Lambda_{a,b,c}$ (scaled by a factor 1/4000 with respect to its real size).
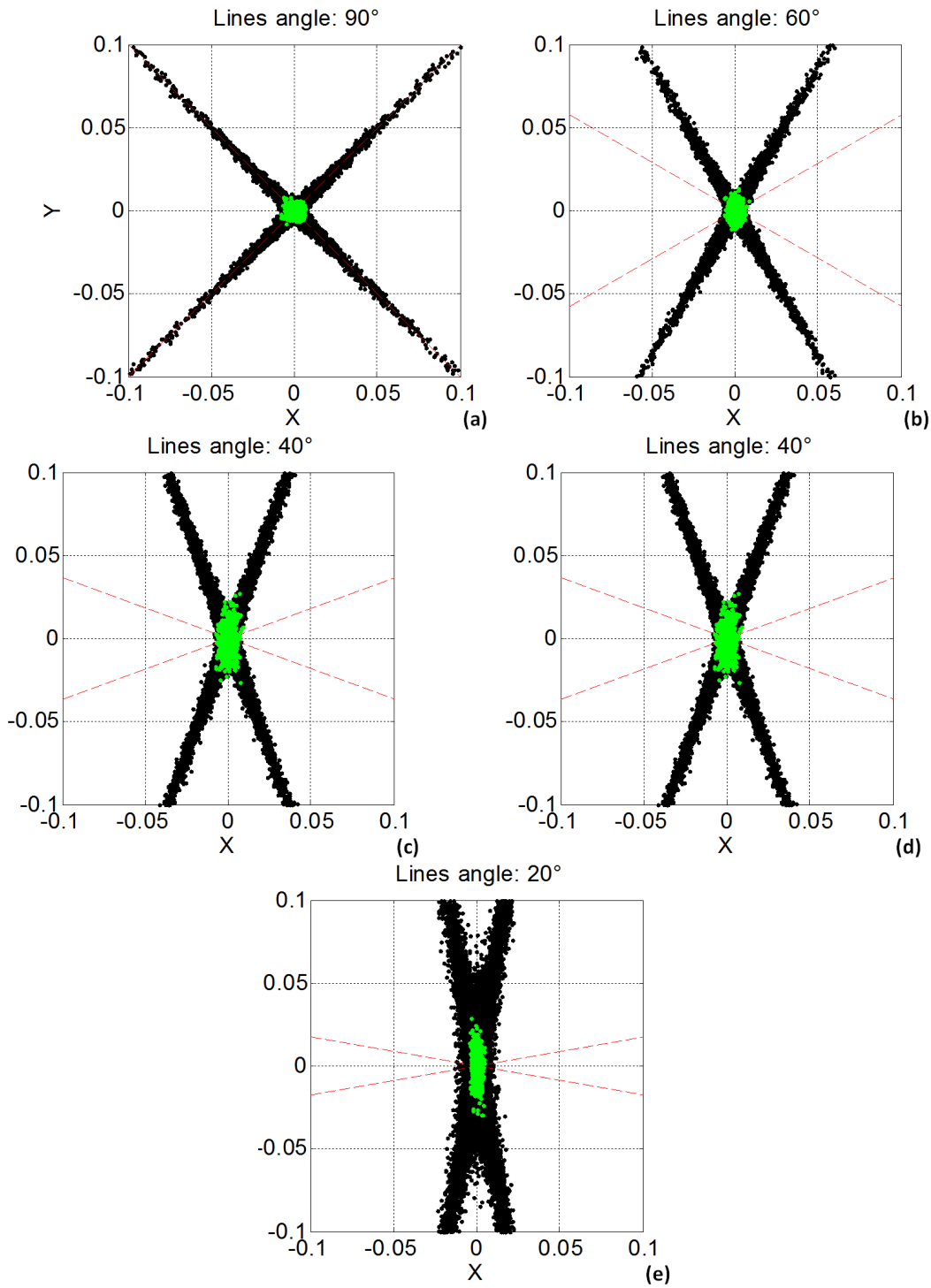
Fig. 6. Panels (a)-(e) show the projection of the estimated normal to $\Lambda_{a,b,c}$ onto the true plane, when lines intersecting with different angles are used. The dash-dotted red lines shows the pair of lines onto $\Lambda_{a,b,c}$ (not in scale). The black points are associated to geometrical configurations close to the critical condition illustrated in Fig. 10: the angle between the line K_2 and the plane Π shown in Fig. 10, is smaller than 15° for all the black points, whereas it is larger than 15° for the green ones.
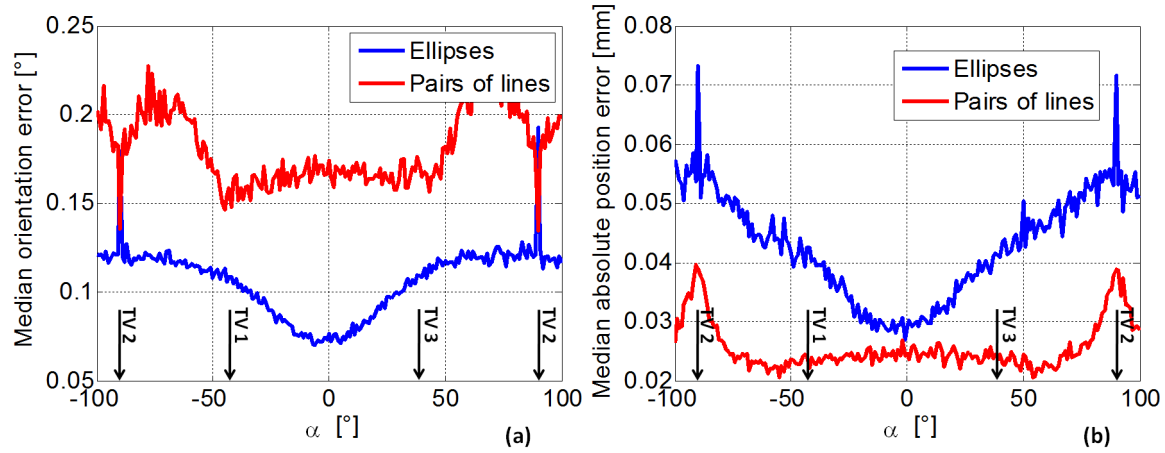
Fig. 7. In panel (a), the median orientation error for ellipses and pairs of lines is shown as a function of the rotation angle, α, around an axis parallel to the Y axis. The vertical arrows indicate the angles for which the plane contains the projection center of one of the cameras. Panel (b) shows the median absolute position error.
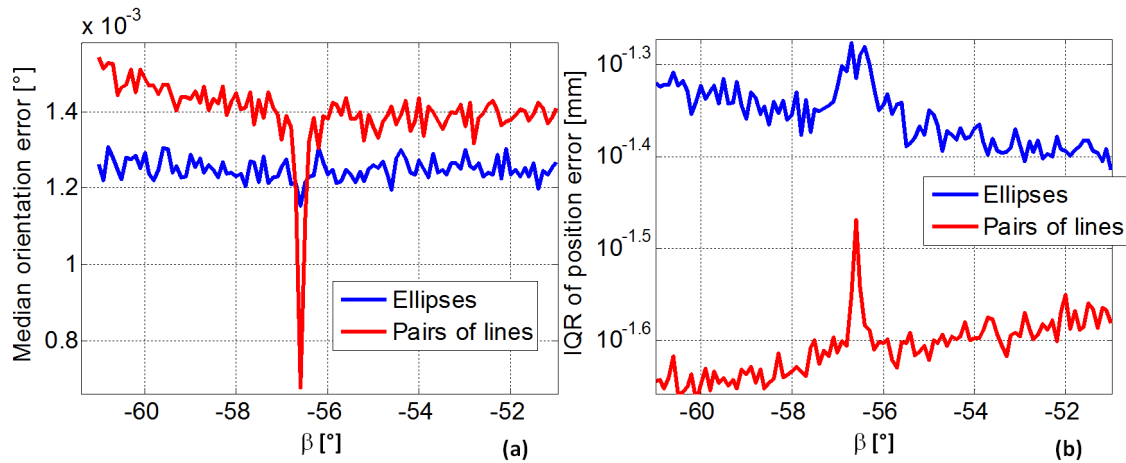

Fig. 8. In panel (a), the median orientation error for ellipses and pairs of lines is shown as a function of the rotation angle, β, around the X axis. For β ~ -56.5° the plane contains the center of the first and third camera. In panel (b), the IQR of the absolute position error is shown for the same geometrical configuration.
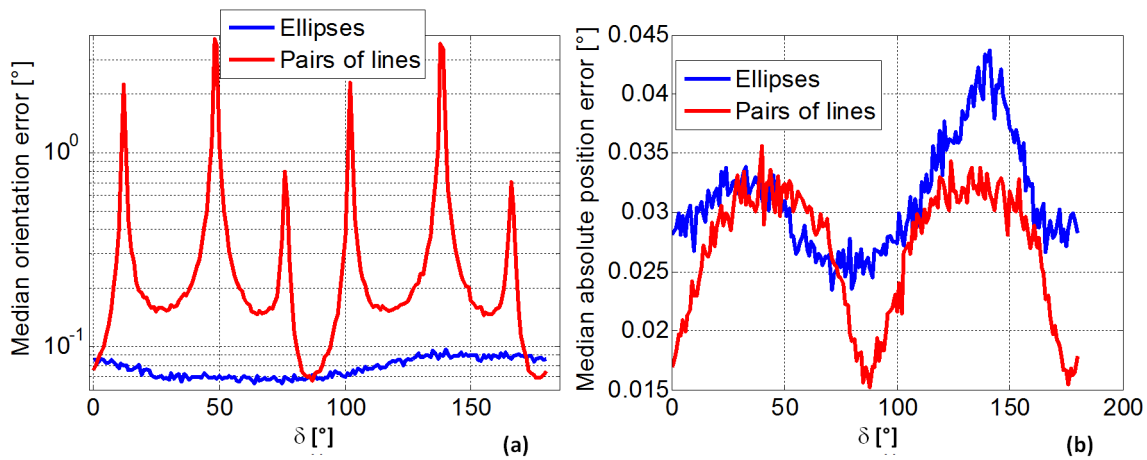
Fig. 9. In panel (a), the median orientation error is shown as a function of the rotation, δ, around the conics center. Panel (b) shows the median absolute position error.
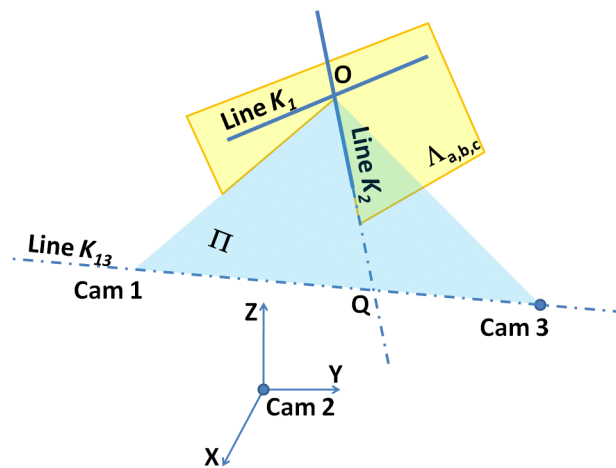

Fig. 10. Schematic representation of a critical geometrical configuration for the pairs of lines, K_1 and K_2, that belong to the plane $\Lambda_{a,b,c}$. The line K_2 intersects in Q the line K_{13}, through the center of the first and third camera. K_{13}, and Q belong to Π while K_1, in general, does not.

6.4 Critical geometrical configurations for Quan's algorithm

   The critical configurations alredy identified for the proposed algorithm in case of ellipses are also critical for Quan's algorithm. These occur each time $\Lambda_{a,b,c}$ includes the center of one or more cameras. In this case, the projection of the conic onto the camera image plane degenerates into a segment and the plane orientation cannot be derived. With respect to the proposed method, Quan's algorithm

suffers from an additional critical configuration, that occurs when the center of one camera lies on the surface of the cone associated to the observed conic on the other camera. In this case, a zero by zero division occurs and the estimate of the conic plane becomes unfeasible.
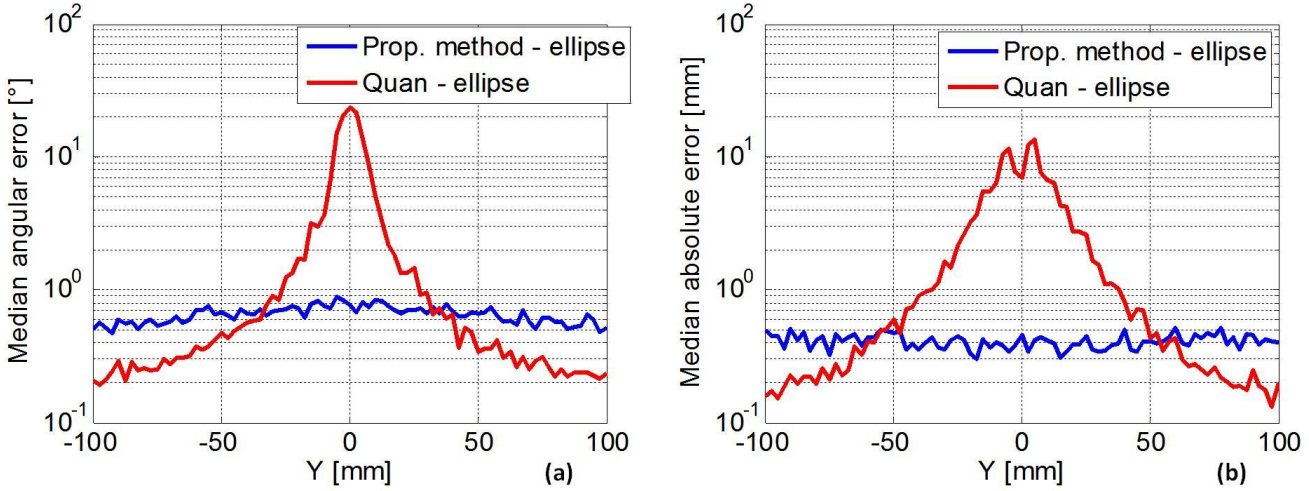

Fig. 11. In panel (a), the median orientation error for the proposed method and for the Quan procedure[1], is shown as a function of the Y coordinate of the third camera. Panel (b) shows the median absolute position error.

We have verified this with an ad-hoc simulation. We moved the third camera to the position [-100mm -100mm 1400mm] in the set-up described in section 6.2 and translate the camera by steps of 2.5mm along the Y direction. For each position, we have generated 100 ellipses, γ, in 3D space passing through the point [0mm 0mm 700mm] with a random orientation. The critical geometrical configuration described above occurs when Y = 0. For each position, γ was analytically projected through (9) onto the image plane of the three cameras, to get γ_1, γ_2 and γ_3. Noise was added to their parameters, coherently with the fitting errors reported in Table II. The conics γ_1, γ_2 and γ_3 were finally used to estimate $\Lambda_{a,b,c}$ through the method described in Section 3 and through [1].

Fig. 11 reports the median orientation error and the median absolute position error as a function of the Y coordinate of the third camera. It can be observed that Quan's method looses accuracy in the range -30mm < Y < +30mm (in this range the median orientation error is higher than 1° and the

median absolute position error is higher than 1mm). The proposed method, on the contrary, maintains the same good accuracy also for values of Y inside this range.

6.5 Critical geometrical configurations for the linear reconstruction from lines pairs

Also the linear procedure in Section 3.4 presents a critical geometrical configuration that is not shared by our method. This occurs when the planes π_{ki} and π_{kj}, associated to the lines l_{ki} and l_{kj} observed by the i-th and j-th cameras, are almost parallel. In this situation, the intersection between the two planes varies significantly for small errors in the estimate of l_1 or l_2, and the estimate of the orientation of the 3D plane becomes consequently inaccurate. In practice, when two cameras observe a 3D line l_k and the center of the two cameras are almost coplanar to it, the orientation of l_k cannot be estimated accurately. On the other side, using linear pairs of lines, produces a smaller error in all the other configurations reported in Section 6. However, correct matching between the lines should be available in advance.

7. DISCUSSION

The approach presented here constitutes a unified framework to estimate the pose of a plane in 3D space from a set of projections of a conic or of a pair of intersecting lines, that lie on the plane. No a-priori knowledge on the conics parameters or any initialization is required. Although the method can be appliedwithouth modifications to any kind of conics, including hyperbola and parabola, we have considered here only ellipses and intersecting lines, that are frequently encountered in real scenarios.

The method is based on solving a linear system composed of six equations in five unknowns. This could be further simplified to six equations in four unknowns, substituting the term $d_3\rho_1 - d_1\rho_3$ with a single unknown, or even further reduced to three equations in three unknowns.

However, in these cases, the norm of the coefficient matrix becomes close to singularity and the estimate becomes unstable. This has been verified through simulations and therefore further simplification of (16) was avoided.

Experimental results (Table I) show that ellipses provide a more accurate estimate of the orientation of $\Lambda_{a,b,c}$ than intersecting lines, in spite of a more accurate 2D fitting (Table II). Ellipses incorporate more information about plane orientation than pairs of lines, possibly because all directions are represented by the points on the ellipse perimeter while only two directions are represented by pairs of lines. Figs. 4a, 4b and 6 demonstrate that less eccentric ellipses produce a more accurate estimate of $\Lambda_{a,b,c}$ orientation than highly eccentric ones. This can be explained by two concurrent factors. First, projecting a highly eccentric ellipse at small angles produces an ellipse, that is generally harder to fit [15], [16]. Moreover, the 2D area in which the 2D points are distributed shrinks in one direction making more difficult to estimate the orientation around an axis orthogonal to that direction. This is clearly shown in Figs. 5, where the orientation error is mainly distributed along the minor axis of the ellipse, whereas the maximum accuracy is achieved for rotations around its minor axis. Similar considerations apply also to intersecting lines: when the lines are orthogonal each other, the error distribution is isotropic (green points in Fig. 6a), whereas it becomes anisotropic for small intersection angles (green points in Fig. 6e).

The method presented here has the advantage of linearity with respect to Quan's method. Moreover, it does not suffer from the criticial configuration identified in Section 6.4.

A few geometrical configurations have to be avoided when intersecting lines are considered to derive the plane orientation through the proposed method. These are characterized by one of the two intersecting lines cutting the line through the projection centers of two of the three

cameras (as in Fig. 10). Although by a geometrical point of view such configuration is not critical, we have verified that, in this case, the left term of the linear combination in (16) is a matrix, whose associated conic is broken into the projections of K_1 and K_2 onto the plane of camera 2, but the factor corresponding to K_1 does not depend on a,b,c. As a consequence, the proposed method can reconstruct only K_2; in fact the matrix of the linear system (16), in this case, has rank 3, and the solutions correspond to a pencil of planes passing through K_2. This explains also why the estimate of the plane position is correct in this case (see Fig. 9b): as each plane of the pencil passes through K_2, it also necessarily passes through **O** and the position error is consequently low. Similarly, when the line K_2 is coplanar with the center of the cameras 1 and 2 (or 2 and 3) the solution of (16) corresponds to a pencil of planes through K_1 and the conclusion is similar.

We observe also that, when $\Lambda_{a,b,c}$ contains the center of the first and third camera, the rank of the matrix associated to the linear system (16) is reduced to 3 for ellipses and 1 for pairs of lines. The reduced rank is explained with the fact that two of the three projections of γ degenerate into a segment, and therefore they include little information about the conic shape. Despite of this, simulations (Fig. 8) demonstrate that the algorithm provides a reliable estimate of the plane parameters. This can be explained observing that, when the plane of the conic passes through the center of one camera, the conic projection degenerates into a segment and that camera should be sufficient by itself to correctly identify $\Lambda_{a,b,c}$. By an algebraic point of view, this means that cancellation of significant terms in (15) does not occur in this case.

The method achieves a slightly lower accuracy with respect to linear reconstruction, but it has the advantage of not requiring to solve the correspondence problem before.

We remark here that a derivation similar to that of the proposed method is reported in [13].

However, it was devoted to the estimate of only 3D ellipsoids and no analysis of critical configurations was reported.

To derive optimal pose estimate of the plane, some effort has been made to make conics fitting most accurate. Although conic fitting and line fitting have received great attention [15], [16], no algorithm was developed specifically tailored for fitting intersecting lines. The two steps algorithm described here, thanks to the specific representation of the lines, allows using an algebraic approach to implement a geometrical minimization, that is known to be more accurate with respect to the algebraic one [15]. This is clearly shown by the fitting error reported in Table II that is lower for pairs of lines in spite of the lower number of 2D points used in the estimate (about a half).

The proposed method was applied to the case of three cameras. However it can be extended, without any significant modification, to any arbitrary number of cameras higher that three. Each additional camera adds six equations and only one unknown to (16). This increases the accuracy and reduces the possibility of critical configurations.

8 CONCLUSION

In this paper, we proposed a method for determining the pose of a plane containing a conic, observed by at least three calibrated cameras, solving a linear system of six equations in five unknowns. Experimental and theoretical analysis has highlighted few geometrical degenerate configurations that do not allow estimating correctly the pose.

APPENDIX A

The (ρ, θ) parameters of a line in polar form can be estimated, from a set of points $\{\mathbf{p_j} = [x_j,$

$y_j]^T\}_{j=1..N}$, minimizing the squared algebraic error, which is defined as:

$$F = \tfrac{1}{2} \sum_{j=1..N} \left(x_j \cos\theta + y_j \cos\theta - \rho\right)^2 \tag{A1}.$$

Notice that the distance of the point $\mathbf{p_j}$ from the line $x\cdot\cos(\theta)+y\cdot\sin(\theta)=\rho$ is given by the projection of the vector $[x_j, y_j]^T-[\rho\cdot\cos(\theta), \rho\cdot\sin(\theta)]^T$ over the direction $[\cos(\theta), \sin(\theta)]^T$, which leads to:

$$\left[x_j - \rho\cos\theta \quad y_j - \rho\sin\theta\right]\left[\cos\theta \quad \sin\theta\right]^T = x_j \cos\theta + y_j \sin\theta - \rho \tag{A2}.$$

Comparing (A1) and (A2), we notice that the squared algebraic error corresponds exactly to the sum of the squared distances of the points from the line. Minimizing (A1) leads therefore to the *total least square* estimate of the line.

The stationary points of F are identified posing $\delta F/\delta\rho = 0$ and $\delta F/\delta\theta = 0$; the first condition leads to:

$$\partial F/\partial\rho = 0 \Rightarrow \rho = x^* \cos\theta + y^* \sin\theta \tag{A3},$$

where $\mathbf{p}^* = [x^*, y^*]^T$ is the barycenter of the set of points. In practice, (A3) highlights that the fitted line will pass through $\mathbf{p}^*$. Substituting (A3) into $\delta F/\delta\theta = 0$ we obtain:

$$\cos^2\theta \sum_{j=1..N}\left(x_j - x^*\right)y_j - \sin^2\theta \sum_{j=1..N}\left(y_j - y^*\right)x_j +$$
$$+ \cos\theta\sin\theta \sum_{j=1..N}\left[\left(y_j - y^*\right)y_j - \left(x_j - x^*\right)x_j\right] \tag{A4};$$

dividing then by $\cos^2(\theta)$, we obtain a second degree equation in $\tan(\theta)$:

$$\tan^2\theta \sum_{j=1..N}\left(y_j - y^*\right)x_j - \sum_{j=1..N}\left(x_j - x^*\right)y_j +$$
$$- \tan\theta \sum_{j=1..N}\left[\left(y_j - y^*\right)y_j - \left(x_j - x^*\right)x_j\right] = 0 \tag{A5}.$$

The values assumed by θ in the two stationary points of F are given by:

$$
\begin{cases}
\Delta = \left\{ \sum_{j=1..N}\left[(y_j - y)y_j - (x_j - x)x_j\right] \right\}^2 - 4\left[\sum_{j=1..N}(y_j - y)x_j \right]\left[-\sum_{j=1..N}(x_j - x)y_j \right] \\
\theta = \operatorname{atan}_2\left(\sum_{j=1..N}\left[(y_j - y)y_j - (x_j - x)x_j\right] \pm \sqrt{\Delta},\, 2\sum_{j=1..N}(y_j - y)x_j \right)
\end{cases}
\qquad (A6).
$$

Note that, using the atan$_2$ function, it is not necessary to suppose that $\cos^2(\theta)$ differs from zero: the estimate is in fact correct even when the fitted line is horizontal (that is, $\theta = \pm 90°$).

Once the two values of θ have been estimated though (A6), the corresponding values of ρ are computed from (A3). The two lines identified by these parameters are orthogonal each other; one of them minimizes F, whereas the other one maximizes it. The total least squares line is found computing directly F as in (A1) for both the lines, and selecting the one associated to the smallest value of F.

REFERENCES

[1] L. Quan, "Conic Reconstruction and Correspondence from Two Views," IEEE Trans. PAMI, 18(2), pp.151-60, Feb. 1996.

[2] J. Usabiaga, A. Erol, G. Bebis, R. Boyle, X. Twombly, "Global hand pose estimation by multiple camera ellipse tracking," Machine Vision and Applications, 21(1), pp. 1-15, 2009.

[3] J. I. Ronda, A. Valdes, "Conic Geometry and Autocalibration from two Images," J. Math. Imaging Vis., 28, pp. 135-9, 2007.

[4] G. Jiang, H. Tsui, L. Quan, A. Zisserman, "Geometry of Single Axis Motions Using Conic Fitting," IEEE Trans. PAMI, 25(10), pp. 1343-8, Oct. 2003.

[5] C. Doignon, M. De Matelin, "A Degenerate Conic-Basec Method for a Direct Fitting and 3-D Pose of Cylinders with a Single Perspective View", in proc. IEEE Int. Conf. Rob. And Automat., Rome, Italy, 10-14 April 2007.

[6] B. Rosenhahn, G. Sommer, "Pose Estimation in Conformal Geometric Algebra Part II: Real-time Pose Estimation Using Extended Feautre Concpets," Journ. Of Meth. Imag. And Vision, 22, pp.49-70, 2005.

[7] T. L. Sun, "Conics enhanced vision approach for easy and low-cost 3D tracking," Pattern Recognition, 37, pp. 1441-50, 2004.

[8] J.Y.Kaminski, A.Shashua, "Multiple View Geometry of General Algebric Curves", Int. Jour. of Comp. Vision, 56(3), pp. 195-219, 2004.

[9] R. Safaee-Rad, I. Tchoukanov, B. Benhabib, K. C. Smith, "3D-Pose Estimation from a Quadratic-Curved Feature in Two Perspective Views," 11th IAPR International Conference on Pattern Recognition, Vol.I. Conference A: Computer Vision and Applications, Proceedings., pp.341-4, 1992.

[10] L. Li, S. De Ma, "3D Pose Estimation from an n-degree Planar Curved Feature in Two Perspective Views", in Proc. 13th Int. Conf. on Pattern Recognition, pp. 374-7, 1996.

[11] M. Fryers, J.Y. Kaminski, M. Teicher, "Some applications of algebraic curves to computational vision," in "Applications of Algebraic Geometry To Physics, Computations And Coding Theory", NATO Science Series, p.121-137, 2001.

[12] M. Fryers, J. Y. Kaminski, M. Teicher, "Recovering an Algebraic Curve Using its Projections From Different Points: Applications to Static and Dynamic Computational Vision," arXiv:math/0208099v1, available in: http://arxiv.org/abs/math.AG/0208099, 2002.

[13] S.D. Ma, L. Li, "Ellipsoid Reconstruction from Three Perspective Views," icpr, vol. 1, pp.344, 13th International Conference on Pattern Recognition (ICPR'96) - Volume 1, 1996.

[14] R. Hartley, A. Zissermann, Multiple View Geometry, pp. 190-192. Cambdridge Univ. Press, 2000.

[15] A. W. Fitzgibbon, R. B. Fisher, "A Buyer's Guide to Conic Fitting," in Proc. British Machine Vision Conference, 1995.

[16] M. Harker, P. O'Leary, P. Zsombor-Murray, "Direct Type-Specific Conic Fitting and Eigenvalue Bias Correction, " Image and Vision Computing, 26, pp. 372-81, 2008.

[17] L. A. F. Fernandes, M. M. Oliveira, "Real-time line detection through an improved Hough transform voting scheme," Pattern Recognition, 41(1), pp. 299–314, Jan. 2008.

[18] Jean-Yves Bouguet, Camera Calibration Toolbox for Matlab®, available in: http://www.vision.caltech.edu/bouguetj/calib_doc/.

[19] R. C. Gonzalez, R. E. Wood, Digital Image Processing, Addison-Wesley, 1992.

[20] W. Liu and K. Kanatani, Interpretation of conic motion and its applications , International Journal of Computer Vision, Vol. 10, No. 1 (1993), pp. 67-84.

[21] K. Kanatani and W. Liu, 3-D interpretation of conics and orthogonality, CVGIP: Image Understanding, Vol. 58, No. 3 (1993), pp. 286-301.