J Braz. Soc. Mech. Sci. Eng. (2013) 35:93–101
DOI 10.1007/s40430-013-0013-2

TECHNICAL PAPER

Automatic estimation of camera parameters from a solid calibration box

P. R. G. Kurka · J. V. Delgado · C. R. Mingoto · O. E. R. Rojas

Received: 12 August 2011 / Accepted: 14 March 2012 / Published online: 11 April 2013

Abstract The paper presents a simple and automatic procedure to estimate the intrinsic and extrinsic parameters of a camera, based on the image of a standard calibration box. Image processing techniques are used to determine the relevant image points to be used in the calibration procedure. A non-recursive solution scheme is proposed to estimate the intrinsic and extrinsic camera parameters. Simulated and real applications are presented to illustrate the use and performance of the proposed technique.

Keywords Camera calibration · Camera model · Vision geometry

1 Introduction

Calibration is a necessary operation in photogrammetric applications. It consists in finding intrinsic and extrinsic parameters of a pinhole camera model. The extrinsic parameters are used in the conversion of 2-D projected images into real 3-D world objects. Extrinsic parameters are related to the movement of a camera, from the image of a regular object of known dimensions, with respect to a reference frame. Such parameters can also be used to determine the dimension and position of objects in space, with respect to an arbitrary referential, from their simultaneous images in a set of two or more cameras. Dimensional information of objects in space from their camera projections can only be obtained after the intrinsic parameters of the camera have been determined. Intrinsic parameters consist of the camera's focal length, expressed in horizontal and vertical pixel dimensions, skew parameter of the camera's sensing element and the pixel coordinates of the intersection of the camera's optical axis with the image plane. Some popular camera calibration techniques, such as [1, 2], require several images of planar calibration grids, and user actions to indicate special points in the displayed images. The constitutive algebraic equations of such applications require the use of iterative optimization solutions, such as the maximum likelihood scheme, to find homographies between the calibration grid points and its images. The present work proposes a calibration procedure based on the processing of edge points of the single image of an ordinary rectangular box of known dimensions, found in the context of any image processing laboratory. The method uses the concepts of perspective projection and vanishing lines, and is a simple and robust tool to estimate the intrinsic and extrinsic parameters of pinhole cameras, increasing the comprehension and access of researches to the field of 3-D image processing. The direct solution of the non-homographic algebraic problem that leads to the estimates of the intrinsic parameters, proposed in this paper, is a novel alternative solution to the camera calibration problem.

The development of camera calibration methods has grown steadily since the publication of the first works in the late 1970's by El-Aziz and H.M. Abed [3]. It finds

Technical Editor: Glauco Carin.

P. R. G. Kurka (✉) · J. V. Delgado · C. R. Mingoto · O. E. R. Rojas
Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 055, Campinas, São Paulo, Brazil
e-mail: kurka@fem.unicamp.br

J. V. Delgado
e-mail: jaimedelgadovargas@gmail.com

C. R. Mingoto
e-mail: carlosmingoto@gmail.com

O. E. R. Rojas
e-mail: oscar.rojas87@gmail.com

applications in mobile robot navigation, machine vision, biomedical vision, visual surveillance, etc. Camera calibration strategies can be classified as "active" and "passive", according to Li and Zhenzhong [4]. In the first strategy, some sort of marking or pattern is projected onto the scene to allow a structured identification of image points that will be used in the identification scheme. The former camera calibration strategy processes the simple image of known geometrical objects or patterns.

Several approaches which make use of vanishing points and lines have been proposed, using the projective geometry of points in a plane. Reference [5] is an example of such a technique which determines camera calibration parameters using homography between points of a line in space and the projections in a rectification plane. In reference [6], the intrinsic and extrinsic camera parameters are calculated from the image of a calibration box, with the aid of two light points projected onto the box. Lv, Zhao and Nevatia [7] present a methodology to find the intrinsic and extrinsic parameters of a camera from a vertical head-to-feet line segment in the image of a human figure. In this work, a large number of different images is needed to minimize the statistical error. Wenhuan, Zhanwei and Tao [8] present a linear camera calibration procedure, using homography between calibration template with a special rectangular planar shape and its projected image. Nedevschi et al. [9] develop a real-time calibration system for a navigation car system which is able to find the extrinsic parameters from longitudinal lane marks. The work by Wilczkowiak, Boyer and Sturm [10] displays a non-homographic calibration technique using parallelepipeds, which has, in part, the same objective of the present work. The paper by Wilczkowiak however does not propose a strategy to process the projected images, nor does it demonstrate the process used to obtain an estimate solution for the camera's projection matrix. The present paper deals with detailed information on both image processing and analytical solution of the camera calibration problem.

The paper is organized as follows: Sect. 2 presents the image processing techniques to find six contour vertices of the image of the box and uses vanishing lines to find the image location of the two internal vertices. Geometric and algebraic properties of projected points are reviewed in Sect. 3. An analytical solution is proposed in Sect. 4, to assemble and solve the problem of estimation of the camera's projection matrix. The section also shows the calculation of intrinsic and extrinsic parameters from the camera's projection matrix. Section 5 shows applications of the proposed technique. Intrinsic and extrinsic parameters of a virtual camera are extracted, using the proposed technique. The calibration of a web cam, using a simple archive box, is also presented to illustrate the practical usage of the method.

2 Processing the image of a calibration box

The calibration process starts with the image of a solid calibration box of known dimensions, under sufficient light conditions and contrast that enable the segmentation of its visible edges. The image of the box should also be detached from its background, either through a subtraction of two environment images, with and without the box, or by means of a controlled dark background. Convenient techniques to perform edge segmentation of the box image are the algorithms of Canny [11] and Sobel and Prewitt [12].

The segmentation process is able to highlight the visible edges and vertices of the box image. Image processing analysis and linear regression interpolations of the highlighted image points determine algebraic expressions for all visible edge lines. Visible edges and vertices are located in the external contour, as well as in the interior of the projected box. The extreme right, left, up and down edge points describe four external vertices of the perspective view of the box, represented in Fig. 1, by $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3$, and, $\boldsymbol{v}_4$. From such extreme vertices and external contour edges, the algorithm is able to determine the position of the two remaining external vertices that are visible in the image. Once the position of the six external vertices is determined, the algorithm estimates via perspective projection properties, the location of the two internal vertices, as well as their visibility condition. Details of the procedure for the determination of vertices of the box image are explained below.

The pixel coordinates of a point of the edge segmented image is given by a generic vector defined as

$$\boldsymbol{E} = \left\{ \begin{bmatrix} i \\ j \end{bmatrix} | i \text{ and } j \text{ are pixel coordinates of an edge point of the image} \right\} \quad (1)$$

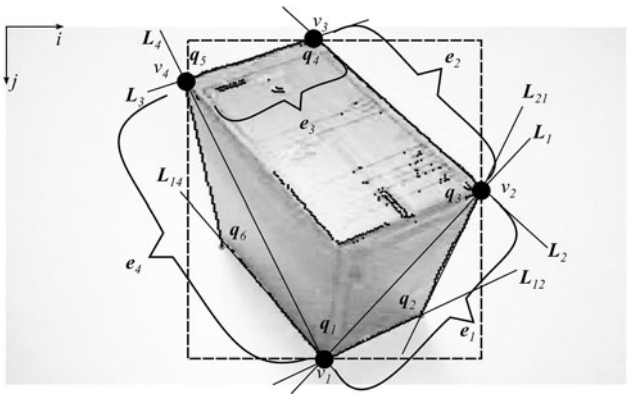

Fig. 1 Segmentation and calculated edges and vertices of the box's contour

Four boundary vertices are located among the edge points, according to the expressions,

$$\begin{aligned}
\boldsymbol{v}_1 &= \left\{ \begin{bmatrix} i_1 \\ j_1 \end{bmatrix} \in \boldsymbol{E} | j_1 = \max(j) \right\}, \\
\boldsymbol{v}_2 &= \left\{ \begin{bmatrix} i_2 \\ j_2 \end{bmatrix} \in \boldsymbol{E} | i_2 = \max(i) \right\}, \\
\boldsymbol{v}_3 &= \left\{ \begin{bmatrix} i_3 \\ j_3 \end{bmatrix} \in \boldsymbol{E} | j_3 = \min(j) \right\} \text{ and} \\
\boldsymbol{v}_4 &= \left\{ \begin{bmatrix} i_4 \\ j_4 \end{bmatrix} \in \boldsymbol{E} | i_4 = \min(i) \right\}
\end{aligned} \tag{2}$$

Vectors containing the coefficients of lines joining pairs of boundary vertices are calculated as,

$$\boldsymbol{L}_1 = \begin{bmatrix} a_1 \\ b_1 \end{bmatrix}, \boldsymbol{L}_2 = \begin{bmatrix} a_2 \\ b_2 \end{bmatrix}, \boldsymbol{L}_3 = \begin{bmatrix} a_3 \\ b_3 \end{bmatrix} \text{and } \boldsymbol{L}_4 = \begin{bmatrix} a_4 \\ b_4 \end{bmatrix} \tag{3}$$

where $a_1, a_2, \ldots, a_4$ and $b_1, b_2, \ldots, b_4$ are respectively the angular and linear coefficients of the lines passing through the pair of points $(\boldsymbol{v}_1, \boldsymbol{v}_2)$, $(\boldsymbol{v}_2, \boldsymbol{v}_3)$ $(\boldsymbol{v}_3, \boldsymbol{v}_4)$ and $(\boldsymbol{v}_4, \boldsymbol{v}_1)$.

Four sets of external boundary edge points between vertices $\boldsymbol{v}_1$ to $\boldsymbol{v}_4$ are found and defined as,

$$\begin{aligned}
\boldsymbol{e}_1 = \Bigg\{ \begin{bmatrix} i \\ j \end{bmatrix} \in \boldsymbol{E} | i \text{ is max. for any given } j, \text{ with } i_1 \le i \le i_2 \\
\text{and } j_2 \le j \le j_1 \Bigg\}, \\
\boldsymbol{e}_2 = \Bigg\{ \begin{bmatrix} i \\ j \end{bmatrix} \in \boldsymbol{E} | i \text{ is max. for any given } j, \text{ with } i_3 \le i \le i_2 \\
\text{and } j_3 \le j \le j_2 \Bigg\}, \\
\boldsymbol{e}_3 = \Bigg\{ \begin{bmatrix} i \\ j \end{bmatrix} \in \boldsymbol{E} | i \text{ is min. for any given } j, \text{ with } i_4 \le i \le i_3 \\
\text{and } j_3 \le j \le j_4 \Bigg\}, \\
\text{and} \\
\boldsymbol{e}_4 = \Bigg\{ \begin{bmatrix} i \\ j \end{bmatrix} \in \boldsymbol{E} | i \text{ is min. for any given } j, \text{with } i_4 \le i \le i_1 \\
\text{and } j_4 \le j \le j_1 \Bigg\}
\end{aligned} \tag{4}$$

Orthogonal distance vectors $\boldsymbol{d}_k$, between the external edge points ($\boldsymbol{e}_1$ $\boldsymbol{e}_2$ $\boldsymbol{e}_3$ and $\boldsymbol{e}_4$) and the vertices joining lines ($\boldsymbol{L}_1$, $\boldsymbol{L}_2$, $\boldsymbol{L}_3$, $\boldsymbol{L}_4$), are calculated as

$$\begin{aligned}
\boldsymbol{d}_1 &= \left\{ \begin{bmatrix} \frac{a_1(a_1 i - j + b_1)}{a_1^2+1} \\ \frac{j - a_1 i - b_1}{a_1^2+1} \end{bmatrix} \middle| \begin{bmatrix} i \\ j \end{bmatrix} \in \boldsymbol{e}_1 \right\}, \\
\boldsymbol{d}_2 &= \left\{ \begin{bmatrix} \frac{a_2(a_2 i - j + b_2)}{a_1^2+1} \\ \frac{j - a_2 i - b_2}{a_2^2+1} \end{bmatrix} \middle| \begin{bmatrix} i \\ j \end{bmatrix} \in \boldsymbol{e}_2 \right\}, \\
\boldsymbol{d}_3 &= \left\{ \begin{bmatrix} \frac{a_3(a_3 i - j + b_3)}{a_1^2+1} \\ \frac{j - a_3 i - b_3}{a_3^2+1} \end{bmatrix} \middle| \begin{bmatrix} i \\ j \end{bmatrix} \in \boldsymbol{e}_3 \right\} \text{and} \\
\boldsymbol{d}_4 &= \left\{ \begin{bmatrix} \frac{a_4(a_4 i - j + b_4)}{a_4^2+1} \\ \frac{j - a_4 i - b_4}{a_4^2+1} \end{bmatrix} \middle| \begin{bmatrix} i \\ j \end{bmatrix} \in \boldsymbol{e}_4 \right\}
\end{aligned} \tag{5}$$

Such distance vectors are used to verify the existence of additional vertices between the pairs of vertices listed above, based on the criteria of a minimum average distance. The existence of an additional vertex between any of the pairs of extreme contour vertices will yield a high mean value for the corresponding orthogonal distance between the vertices joining line and the external edge points. If no vertices are found between pairs of extreme contour vertices, then the external edge points will coincide with the vertices joining line, yielding a low mean value for the edge-line distances.

Orthogonal linear regressions are made to fit lines to all edge segment points between each pair of identified external vertices. A refined estimation for the box's external vertices is made by calculating the crossings of such regression lines.

Figure 1 shows a typical box image with the calculated edge points and identified entities. Edge points are shown as black dots. External edge segment sets are indicated in the figure as $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, $\boldsymbol{e}_3$ and $\boldsymbol{e}_4$. The four boundary vertices are identified as $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, $\boldsymbol{v}_3$, and $\boldsymbol{v}_4$. Lines between two adjacent boundary vertices are represented by $\boldsymbol{L}_1$, $\boldsymbol{L}_2$, $\boldsymbol{L}_3$ and $\boldsymbol{L}_4$, and are generally calculated through a linear regression involving the external edge segments. Whenever the edge segments do not form a single straight line, this indicates the existence of an intermediary boundary vertex of the box. In such cases, additional lines are fit to the intermediary sets of edge segments. Such lines are represented in the figure by $\boldsymbol{L}_{12}$ and $\boldsymbol{L}_{21}$, between vertices $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, and $\boldsymbol{L}_{14}$ and $\boldsymbol{L}_{41}$, between vertices $\boldsymbol{v}_4$ and $\boldsymbol{v}_1$. The crossings of the edge interpolating lines define the six vertices, q_1 to q_6 in the contour of the box. The coordinates of such points are given as,

$$\boldsymbol{q}_k = \begin{bmatrix} i_k \\ j_k \end{bmatrix}, k = 1, \ldots, 6 \tag{6}$$

Three perspective vanishing points, calculated from pairs of vertices joining lines, are used in the process of finding the internal vertices of the box, as shown in Fig. 2.

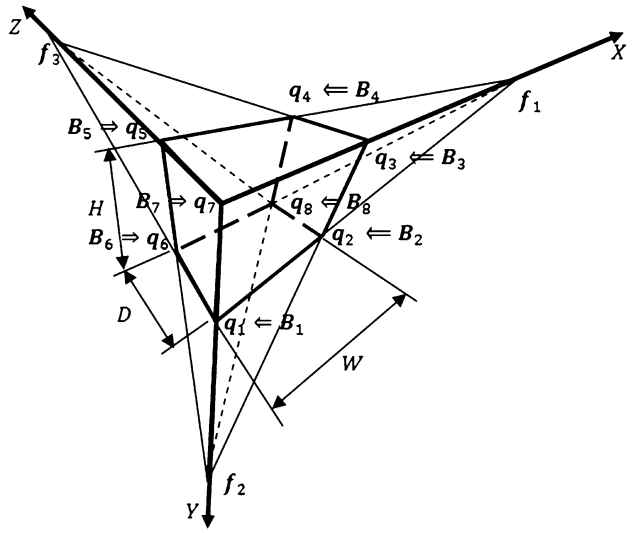

Fig. 2 Vanishing points, perspective lines and reference axis

Vanishing point f_1 is the intersection of lines $\overline{q_1q_2}$ and $\overline{q_4q_5}$, vanishing point f_3 is the intersection of lines $\overline{q_2q_3}$ and $\overline{q_5q_6}$ and vanishing point f_3 is the intersection of lines $\overline{q_3q_4}$ and $\overline{q_6q_1}$. Lines $\overline{f_1q_3}$, $\overline{f_2q_1}$ and $\overline{f_3q_5}$, as well as lines $\overline{f_1q_6}$, $\overline{f_2q_4}$ and $\overline{f_3q_2}$ meet at one of the two internal box vertices q_7 or q_8. In order to determine which lines meet at the front vertex, q_7, or at the rear vertex, q_8, it is necessary to count the number of internal segmented points that coincide with any one of such lines. The lines that have the largest number of coinciding segment points join at the front vertex q_7. The intersection of all remaining vanishing point and external vertices lines define vertex q_8. Representation of the coordinate reference directions X, Y and Z is then arbitrarily fixed at point q_7, and parallel to the box's edges. Directions X, Y and Z coincide with lines $\overline{q_7q_3}$, $\overline{q_7q_1}$ and $\overline{q_7q_5}$ respectively. Vertices q_2, q_4 and q_6 are representations of the spatial vertices that are in planes XY, XZ and YZ, respectively. Vertex q_8 represents the location of the camera hidden spatial vertex.

The Box's dimensions W, H and D are parallel to the coordinate reference axis X, Y and Z, respectively. The global coordinates of the box's vertices, which coincide with the projected vertices q_1 to q_8 are

$$\boldsymbol{B}_1 = \begin{bmatrix} 0 \\ H \\ 0 \end{bmatrix}, \boldsymbol{B}_2 = \begin{bmatrix} W \\ H \\ 0 \end{bmatrix}, \boldsymbol{B}_3 = \begin{bmatrix} W \\ 0 \\ 0 \end{bmatrix}, \boldsymbol{B}_4 = \begin{bmatrix} W \\ 0 \\ D \end{bmatrix},$$
$$\boldsymbol{B}_5 = \begin{bmatrix} 0 \\ 0 \\ D \end{bmatrix}, \boldsymbol{B}_6 = \begin{bmatrix} 0 \\ H \\ D \end{bmatrix}, \boldsymbol{B}_7 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \text{ and } \boldsymbol{B}_8 = \begin{bmatrix} W \\ H \\ D \end{bmatrix}. \tag{7}$$

3 Geometric and algebraic properties of projected points

3.1 A generic point in space with global coordinates is defined as

$$\boldsymbol{B}_k = \begin{bmatrix} X_k \\ Y_k \\ Z_k \end{bmatrix} \tag{8}$$

The same point, seen from a reference frame whose axes are rotated by a rotation matrix $\mathbf{R}$ and whose origin is translated from the origin of the global coordinates reference frame by a translation $\boldsymbol{T}$, is expressed as

$$\boldsymbol{P}_k \equiv \begin{bmatrix} X'_k \\ Y'_k \\ Z'_k \end{bmatrix} = \mathbf{R}\boldsymbol{B}_k + \boldsymbol{T}. \tag{9}$$

Point $\boldsymbol{P}_k$ is projected in the image plane of a pinhole camera, with focal distance f, located at the origin of the displaced reference frame. The metric coordinates of the projected point are defined as

$$\boldsymbol{p}_k = \begin{bmatrix} x'_k \\ y'_k \\ f \end{bmatrix} \tag{10}$$

The geometry of the projected point [13, 14] is represented in Fig. 3.

Point $\boldsymbol{P}_k$ of the displaced reference frame relates to the coordinates of the projected point $\boldsymbol{p}_k$ as

$$\boldsymbol{P}_k = \frac{Z'_k}{f}\boldsymbol{p}_k. \tag{11}$$

The projected image vertex point $\boldsymbol{q}_k$, given in homogeneous coordinates $\hat{\boldsymbol{p}}_k$ is related to its camera metric coordinates as

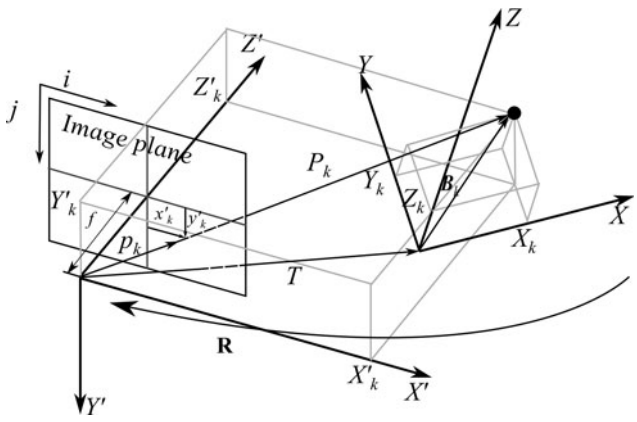

Fig. 3 Geometry of projection of a point

$$\hat{\boldsymbol{p}}_k = \frac{1}{f}\boldsymbol{A}\boldsymbol{p}_k \tag{12}$$

where,

$$\mathbf{A} = \begin{bmatrix} S_x f & 0 & u_0 \\ 0 & S_y f & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{13}$$

and

$$\hat{\boldsymbol{p}}_k = \begin{bmatrix} \boldsymbol{q}_k \\ 1 \end{bmatrix}. \tag{14}$$

A is the intrinsic calibration matrix that comprises the pixel density parameters S_x, S_y, given in units of pixel by metric distance, the focal distance f, u_0 and (u_0, v_0) the pixel coordinates of the optical center of the image.

The homogeneous pixel coordinates vector can be expressed directly from the vector of global coordinates of point $\boldsymbol{B}_k$, by combining expressions (9), (11) and (12):

$$Z_k'\hat{\boldsymbol{p}}_k = \mathbf{A}(\mathbf{R}\boldsymbol{B}_k + \boldsymbol{T}) \tag{15}$$

Next section uses Eq. (15) to establish a non-recursive, analytical approach to estimate the intrinsic and extrinsic camera calibration parameters.

3.2 Assembly and solution of the calibration parameters problem

The homogeneous pixel coordinate vectors of the projection of three distinct vertices, $\hat{\boldsymbol{p}}_{ka}$, $\hat{\boldsymbol{p}}_{kb}$, $\hat{\boldsymbol{p}}_{kc}$, together with their corresponding spatial position vectors $\boldsymbol{B}_{k_a}$, $\boldsymbol{B}_{k_b}$, $\boldsymbol{B}_{k_c}$, are manipulated to establish the basic relations of the parameter estimation problem. Two different weighted additions of the vectors are made, eliminating $\boldsymbol{T}$ in the algebraic equation:

$$\hat{\mathbf{P}}_{k_I}\boldsymbol{\lambda} = \mathbf{A}\mathbf{R}\boldsymbol{B}_{k_I} \tag{16}$$

and

$$\hat{\mathbf{P}}_{k_{II}}\boldsymbol{\lambda} = \mathbf{A}\mathbf{R}\boldsymbol{B}_{k_{II}}. \tag{17}$$

where,

$$\boldsymbol{\lambda} = \begin{bmatrix} Z_{k_a}' \\ Z_{k_b}' \\ Z_{k_c}' \end{bmatrix}, \tag{18}$$

$$\hat{\mathbf{P}}_{k_I} = \begin{bmatrix} 0.5\hat{\boldsymbol{p}}_{k_a} & -\hat{\boldsymbol{p}}_{k_b} & 0.5\hat{\boldsymbol{p}}_{k_c} \end{bmatrix} \tag{19}$$

$$\hat{\mathbf{P}}_{k_{II}} = \begin{bmatrix} -\hat{\boldsymbol{p}}_{k_a} & 0.5\hat{\boldsymbol{p}}_{k_b} & 0.5\hat{\boldsymbol{p}}_{k_c} \end{bmatrix} \tag{20}$$

$$\boldsymbol{B}_{k_I} = (0.5\boldsymbol{B}_{k_a} - \boldsymbol{B}_{k_b} + 0.5\boldsymbol{B}_{k_c}) \tag{21}$$

and

$$\boldsymbol{B}_{k_{II}} = (-\boldsymbol{B}_{k_a} + 0.5\boldsymbol{B}_{k_b} + 0.5\boldsymbol{B}_{k_c}) \tag{22}$$

Matrices $\hat{\mathbf{P}}_{k_I}$ and $\hat{\mathbf{P}}_{k_{II}}$ are non-singular, leading to the following combination of Eqs. (16) and (17):

$$\mathbf{A}_\mathbf{R}\boldsymbol{B}_{k_I} - \hat{\mathbf{P}}_{k_{III}}\mathbf{A}_\mathbf{R}\boldsymbol{B}_{k_{II}} = 0 \tag{23}$$

where,

$$\hat{\mathbf{P}}_{k_{III}} = \hat{\mathbf{P}}_{k_I}\hat{\mathbf{P}}_{k_{II}}^{-1}$$

and

$$\mathbf{A}_\mathbf{R} = \mathbf{A}\mathbf{R}. \tag{24}$$

Expression (23) can be rewritten, using the Kronecker product, denoted by $\otimes$ and the matrix to vector operator, denoted by $vec()$:

$$\left[\boldsymbol{B}_{k_I}^T \otimes \mathbf{I} - \boldsymbol{B}_{k_{II}}^T \otimes \hat{\mathbf{P}}_{k_{III}}\right] vec(\mathbf{A}_\mathbf{R}) = \mathbf{0} \tag{25}$$

Term I of Eq. (25) is the identity matrix of dimension 3×3, and the vector operator stacks the columns of matrix $\mathbf{A}_\mathbf{R}$, turning it into a vector of dimension 9×1.

Equation (25) can be expanded, in the presence of $n \geq 3$ different sets of coefficient matrices of the type $\hat{\mathbf{P}}_{k_{III}}$, and vectors of the type $\boldsymbol{B}_{k_I}$ and $\boldsymbol{B}_{k_{II}}$, each set made from 3 spatial vectors and their corresponding homogeneous pixel coordinates, yielding

$$\mathbf{P}\bar{\boldsymbol{e}} = 0 \tag{26}$$

where,

$$\mathbf{P} = \begin{bmatrix} \left(\boldsymbol{B}_{1_I}^T \otimes \mathbf{I} - \boldsymbol{B}_{1_{II}}^T \otimes \hat{\boldsymbol{P}}_{1_{II}}\right) \\ \left(\boldsymbol{B}_{2_I}^T \otimes \mathbf{I} - \boldsymbol{B}_{2_{II}}^T \otimes \hat{\boldsymbol{P}}_{2_{III}}\right) \\ \vdots \\ \left(\boldsymbol{B}_{n_I}^T \otimes \mathbf{I} - \boldsymbol{B}_{n}^T \otimes \hat{\boldsymbol{P}}_{n_{III}}\right) \end{bmatrix}. \tag{27}$$

and $\bar{e}$ is the representation for $vec(\mathbf{A}_\mathbf{R})$

Equation (26) is a system of $3n \times 9$ homogeneous equations, which has solutions in the null space of **P**. In practical cases, where the image projections of the spatial points are approximated by discrete pixel positions, matrix **P** will not have a null space, and $\bar{\boldsymbol{e}}$ will be estimated as the right singular vector of **P** associated to its smallest singular value.

The estimation vector is reshaped into a matrix of dimension 3×3, which is an approximation for matrix $\mathbf{A}_\mathbf{R}$, that is,

$$\bar{\mathbf{A}}_\mathbf{R} = Reshape(\bar{\mathbf{e}}, 3 \times 3) \tag{28}$$

Matrix $\overline{\mathbf{A}}_{\mathbf{R}}$ is also an estimation of the product of the intrinsic calibration and rotation matrix, according to Eq. (24). The orthogonal property of the rotation matrix allows the manipulation of Eq. (24), to yield an estimation of the self-transpose product of the calibration matrix, that is,

$$\mathbf{A}\,\mathbf{R}\,\mathbf{R}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}} = \mathbf{A}\mathbf{A}^{\mathrm{T}} \simeq \tilde{\mathbf{A}}_{\mathbf{R}}\,\tilde{\mathbf{A}}_{\mathbf{R}}^{\mathrm{T}}. \tag{29}$$

Matrix $\tilde{\mathbf{A}}_{\mathbf{R}}$ of Eq. (29) is the normalized version of $\bar{\mathbf{A}}_{\mathbf{R}}$, forcing that its third row, third column element is 1, in order to satisfy the condition of the likewise element of $\mathbf{A}\mathbf{A}^{\mathrm{T}}$.

Estimations of the intrinsic parameters of the camera are derived from the self-transpose product of matrix $\tilde{\mathbf{A}}_{\mathbf{R}}$, according to Eqs. (29) and (13):

$$\tilde{u}_0 \simeq \text{element of row 1, column 3 of } \tilde{\mathbf{A}}_{\mathbf{R}}\tilde{\mathbf{A}}_{\mathbf{R}}^{T} \tag{30}$$

$$\tilde{v}_0 \simeq \text{element of row 2, column 3 of} \tilde{\mathbf{A}}_{\mathbf{R}}\tilde{\mathbf{A}}_{\mathbf{R}}^{T} \tag{31}$$

$$\tilde{s}_x f \simeq \sqrt{\text{element of row 1, column 1 of} \tilde{\mathbf{A}}_{\mathbf{R}}\tilde{\mathbf{A}}_{\mathbf{R}}^{T} - \tilde{u}_0^2} \tag{32}$$

$$\tilde{S}_y f \simeq \sqrt{\text{element of row 2, column 2 of } \tilde{\mathbf{A}}_{\mathbf{R}}\tilde{\mathbf{A}}_{\mathbf{R}}^{T} - \tilde{v}_0^2}. \tag{33}$$

A preliminary estimate for the rotation matrix can be obtained from Eq. (24) as

$$\tilde{\boldsymbol{R}}' = \tilde{\mathbf{A}}^{-1}\overline{\mathbf{A}}_{\mathbf{R}} \tag{34}$$

where,

$$\tilde{\mathbf{A}} = \begin{bmatrix} \tilde{S}_x f & 0 & \tilde{u}_0 \\ & \tilde{S}_y f & \tilde{v}_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{35}$$

The final estimate for the rotation matrix, $\tilde{\mathbf{R}}$, is obtained by normalizing the singular values of $\tilde{\mathbf{R}}'$, that is,

$$\tilde{\mathbf{R}} = \mathbf{U}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^{T} \tag{36}$$

where $\mathbf{U}_{\mathrm{R}}$ and $\mathbf{V}_{\mathrm{R}}$ are matrices with the left and right singular vectors of $\tilde{\mathbf{R}}'$.

The translation vector $\boldsymbol{T}$ can be estimated, together with a vector of depth projection values, by applying Eq. (15) to a number $n \geq 2$ of pairs of spatial and projected points, via the least squares solution of a linear system of equations:

$$\begin{bmatrix} \hat{\boldsymbol{p}}_1 & 0 & \dots & 0 & -\tilde{\mathbf{A}} \\ 0 & \hat{\boldsymbol{p}}_2 & \dots & 0 & -\tilde{\mathbf{A}} \\ \vdots & \vdots & \ddots & \vdots & -\tilde{\mathbf{A}} \\ 0 & 0 & \dots & \hat{\boldsymbol{p}}_n & -\tilde{\mathbf{A}} \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{\lambda}} \\ \tilde{\boldsymbol{T}} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{A}}_{\mathbf{R}}\boldsymbol{B}_1 \\ \tilde{\mathbf{A}}_{\mathbf{R}}\boldsymbol{B}_2 \\ \vdots \\ \tilde{\mathbf{A}}_{\mathbf{R}}\boldsymbol{B}_n \end{bmatrix}. \tag{37}$$

where $\tilde{\boldsymbol{\lambda}}$ is the vector of estimated depth projections defined as

$$\tilde{\boldsymbol{\lambda}} = \begin{bmatrix} Z_1' \\ Z_2' \\ \vdots \\ Z_n' \end{bmatrix} \tag{38}$$

3.3 Simulated and real applications

The complete camera calibration procedure can be applied in the following sequence:

(a) obtain an image of a box of known dimensions, with sufficient light conditions and contrast that enable segmentation of its visible edges.
(b) Find all the edges in the image and calculate four vertices from extreme edge points, according to Eqs. (1) and (2).
(c) Calculate the linear coefficients of the lines that join such vertices, according to Eq. (3) and the distance between the extreme edge points and the vertices joining lines, according to Eqs. (4) and (5). The norm of such distance vectors is used to find the two remaining external vertices, that are located between some of the four extreme contour vertices.
(d) Find the equations of six lines that best fit the external boundary edge points of the projected box. With such lines, calculate the perspective vanishing points and estimate the coordinates of internal vertices of the box projection, as shown in Fig. 2.
(e) Associate the dimensions of the box with the vectors of space position of the vertices, according to Eq. (7).
(f) Build a number $n \geq 3$ of different sets of matrices, each set using the spatial coordinates of coordinates of three different vertices of the box, and their corresponding homogeneous pixel vectors, according to Eqs. (19)–(22).
(g) Assemble the homogeneous system problem matrix, using the sets of coefficient matrices, according to Eqs. (25) and (27). Solve the system via singular value decomposition to obtain an estimate of the camera parameters matrix, according to Eq. (28).
(h) Estimate the intrinsic and extrinsic camera parameters according to Eqs. (29)–(37).

The image processing and calibration techniques described above are employed in the identification of the intrinsic and extrinsic parameters of cameras, from simulated and real images of a solid box.

3.4 Simulation data

A solid box of dimensions 360 × 245 × 135 × (mm) is generated in the 3-D modeling program Blender, and ten poses of the box at different positions and rotations are acquired by a virtual camera with 800 × 600 pixels, and

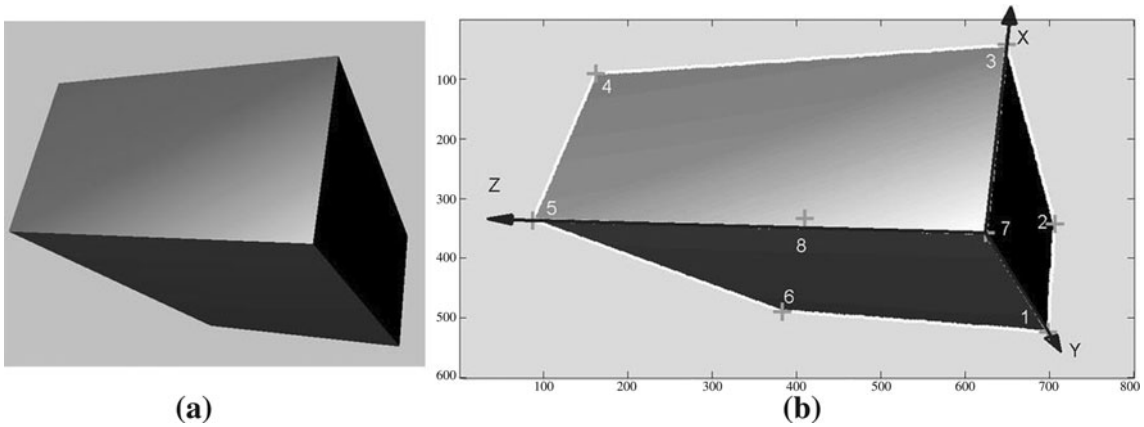

Fig. 4 Blender rendered images: **a** raw image, **b** image with identified axis and vertices

Table 1 Information setup in the 3D modeling software

IMAGE	BLENDER Translation vector (mm)	IDENTIFIED Translation vector (mm)	BLENDER x, y, z rotation angles (degrees)	IDENTIFIED x, y, z rotation angles (degrees)	IDENTIFIED Intrinsic parameters $s_x.f$, $s_y.f$, u_0, v_0)
#01	[−114.19	[−114.33	197.6	197.18	[961.36; 961.59; 399.95; 308.41]
	−166.05	−174.29	200.29	200.26	
	729.26]	720.02]	−190.55	−190.58	
#02	[−114.19	[−107.030	−203.95	−204.04	[971.94; 972.21; 390.56; 302.13]
	−166.05	−169.590	147.89	148.49	
	729.26]	729.501]	190.09	190.16	
#03	[−51.02	[−48.94	−225.6	−225.35	[963.37; 961.57; 397.19; 288.44]
	−83.28	−78.098	211.3	211.9	
	637.82]	630.97]	−239.7	−239.56	
#04	[138.17	[133.46	−186.38	−186.49	[963.81; 960.65; 407.67; 302.09]
	−94.37	−97.657	151.65	151.12	
	629.17]	620.87]	−194.04	−194.02	
#05	[58.35	[59.627	−199.76	−199.31	[971.88; 969.38; 397.80; 293.15]
	−131.88	−129.82	149.84	149.99	
	592.76]	590.83]	191.88	191.58	
#06	[58.35	[58.196	190.37	190.29	[960.71; 962.45; 400.87; 299.28]
	−131.88	−133.5	137.19	137.08	
	592.76]	585.91]	−186.41	−186.3	
#07	[−49.02	[−46.266	184.18	183.48	[966.84; 968.38; 396.69; 309.64]
	−144.44	−153.18	201.47	201.25	
	678.98]	673.72]	−221.08	−221.41	
#08	[−98.87	[−113.82	−209.08	−208.49	[974.2; 977.94; 421.00; 295.14]
	−55.83	−54.965	211.61	210.6	
	697]	699.25]	−195.15	−194.9	
#09	[34.38	[31.205	−194.36	−194.86	[958.71; 957.41; 407.56; 305.38]
	−97.17	−101.29	149.43	148.99	
	447.52]	438.25]	−184.74	−184.46	
#10	[−09.94	[−09.8313	−209.73	−209.93	[961; 961.55; 400.48; 301.07]
	−84.67	−86.792	141.19	141.23	
	445.34]	437.12]	204.44	204.54	

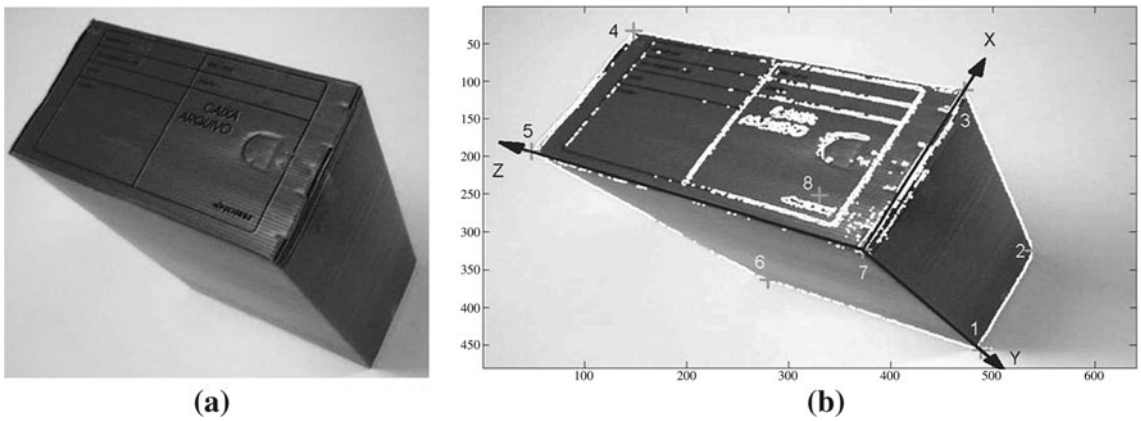

Fig. 5 VX-100 LifeCam images: a raw image, b image with identified axis and vertices

Table 2 Identified parameters of VX-100 LifeCam camera

Parameters	
Intrinsic	
Focal distance in horizontal pixel distance:	$S_x f = 790.24\ \left(\frac{\text{pix}}{\text{mm}}.\text{mm}\right)$
Focal distance in vertical pixel distance:	$S_y f = 792.34\ \left(\frac{\text{pix}}{\text{mm}}.\text{mm}\right)$
Camera's optical axis in horizontal pixel distance:	$u_0 = 325.72$ (pixel)
Camera's optical axis in vertical pixel distance:	$v_0 = 228.72$ (pixel)
Extrinsic	
Translation vector	$\boldsymbol{T} = \begin{bmatrix} 22.5 \\ 50.2 \\ 408.6 \end{bmatrix}$ (mm)
Rotation angles	$\theta_x = -60.3°, \theta_y = 14.9°, \theta_z = 63.1°$

equal horizontal and vertical digital resolutions of $960\frac{\text{pix}}{\text{mm}} \cdot \text{mm}$.. The ten images are processed to find the position of the vertices of the box. Figure 4 shows one of the rendered images of the box with identified vertices. Table 1 shows a comparison between the exact translation vectors and rotation angles of the camera, as set up in the 3D modeling program and those identified with the proposed calibration technique. The rotation angles are calculated from the rotation matrices identified in the calibration procedure, according to Rodrigues Rotation Formula (RRF). It is seen in Table 1 that the identified extrinsic and intrinsic parameters agree with the values provided by the virtual modeling program.

4 Experimental data

Image of a real box, of dimensions 360 × 245 × 135 (mm) is obtained with a Microsoft VX-1000 LifeCam with 640 × 480 pixels and equal horizontal and vertical digital resolutions of $800\frac{\text{pix}}{\text{mm}}.\text{mm}$. The optical axes are located approximately at the center of the image. Figure 5 shows the camera image of the box and the identified vertices and axis.

The spatial width, depth and height of the box are associated to the positions of the vertices identified in the image, yielding the calibration results shown in Table 2 below.

5 Conclusions

The work presented the development of a practical technique to determine the intrinsic and extrinsic parameters of a camera, using a simple calibration box. Detailed information is provided regarding the image processing steps that enable the determination of the position of the box's vertices. The algebraic operations that are used to calculate the intrinsic and extrinsic parameters are presented in a simple and direct way, helping the understanding and application of the camera calibration procedure. The simulation data example shows that the algorithm is able to estimate, with a good numerical precision, the original intrinsic camera parameters and camera movement. The experimental data example, likewise, yields consistent quantitative and qualitative results, showing the potential application of the proposed technique.

Acknowledgments The authors wish to thank UNICAMP and the Brazilian research funding agencies, FAPESP, CNPq and CAPES, sponsors of the present work.

References

1. Zhang Z (1999) Flexible camera calibration by viewing a plane from unknown orientations. International conference on computer vision (ICCV'99), Corfu, pp 666–673
2. Sturm PF, Maybank SJ (1999) On plane-based camera calibration: a general algorithm, singularities, applications. In: Proceedings of the CVPR, vol 1. Fort Collins, CO, pp 432–437

3. Abdel-Aziz YI, Karara HM (1978) Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In: Proceedings of the ASP/UI symposium on close-range photogrammetry, pp 420–475
4. Wang Q, Fu L, Liu Z (2010) Review on camera calibration. In: Proceedings Chinese control and decision conference (CCDC), pp 3354–3358
5. Liebowitz D, Zisserman A (1998) Metric rectification for perspective images of planes. In: IEEE computer society conference on computer vision and pattern recognition, pp 482–488
6. Chu CW, Hwang S, Jung SK (2001) Calibration-free approach to 3D reconstruction using light stripe projections on a cube frame. In: Proceedings of the third international 3-D digital imaging and modeling conference, pp 13–19
7. Lv F, Zhao T, Nevatia R (2006) Camera calibration from video of a walking human. IEEE Trans Pattern Anal Mach Intell 28: 1513–1518
8. Wenhuan W, Zhanwei C, Tao JZ (2010) A new camera calibration method based on rectangle constraint. In: Proceedings of the 2nd international intelligent systems and applications (ISA) Workshop, pp 1–4
9. Nedevschi S, Vancea C, Marita T, Graf T (2007) Online extrinsic parameters calibration for stereovision systems used in far-range detection vehicle applications. IEEE Trans Intell Transp Syst 8:651–660
10. Wilczkowiak M, Boyer E, Sturm P (2001) Camera calibration and 3D reconstruction from single images using parallelepipeds. In: Proceedings of the eighth IEEE international conference on computer vision ICCV 2001, vol 1, pp 142–148
11. Canny JA (1986) Computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 8(6):679–698
12. Parker JR (1997) Algorithms for image processing and computer vision. John Wiley & Sons Inc., New York, pp 23–29
13. Faugeras OD (1993) Three-dimensional computer vision: a geometric viewpoint. MIT press, Cambridge
14. Trucco E, Verri A (1998) Introductory techniques for 3-D computer vision. Prentice Hall, Upper Saddle River