

# Line-scan camera calibration: a robust linear approach

R. USAMENTIAGA<sup>1,\*</sup>, D.F. GARCIA<sup>1</sup>, AND F.J. DELACALLE<sup>1</sup>

<sup>1</sup>Department of Computer Science and Engineering, University of Oviedo, 33204 Gijón, Spain

\*Corresponding author: rusamentiaga@uniovi.es

Compiled April 22, 2021

Camera calibration requires three steps: estimation of correspondences between world and image coordinates, computation of a linear solution, and non-linear optimization using the linear estimate as starting point. The resulting accuracy depends mostly on the first and final steps. However, the non-linear optimization method can only achieve an accurate result when given an initial estimate close to the global solution. Therefore, obtaining a good linear estimation is crucial for the performance of the camera calibration procedure. This work proposes a robust method to estimate a linear solution for the calibration of line-scan cameras, resulting in the individual intrinsic and extrinsic parameters by only using a single line-scan. The calculated parameters can then be used by non-linear optimization methods to finely adjust the estimation of all the line-scan camera parameters, including distortions. The proposed procedure does not impose restrictions on particular orientations, always generating a well-conditioned problem than can be solved analytically with no optimization required. Extensive experiments are performed to verify the robustness and accuracy of the proposed method. The comparative results demonstrate that the proposed method provides excellent performance. © 2021 Optical Society of America

**OCIS codes:** (080.0080) Geometric optics; (120.0120) Instrumentation, measurement, and metrology; (150.1488) Calibration; (150.6910) Three-dimensional sensing;

<http://dx.doi.org/10.1364/ao.XX.XXXXXX>

## 1. INTRODUCTION

Automated surface inspection requires fast speed and high resolution. This way, even small flaws can be accurately identified within the production process [1]. In scenarios where the inspection needs to be performed in long moving products or even continuous materials, area-scan cameras are discarded in favor of line-scan cameras, which provide the best solution. Line-scan cameras can acquire images of high resolution at high frame rates [2], presenting a far more cost-efficient solution than area-scan cameras. A wide array of automated surface inspection applications successfully use line-scan cameras, from the inspection of steel strips [3] to the inspection of large aperture optical elements [4].

Camera calibration determines the relationship between 3D world coordinates and the image coordinates, a required step for surface inspection and metrology applications in order to extract the required information to characterize and classify defects. Moreover, calibration enables the determination of the position of regions of interest in the images. It is also required for 3D reconstruction and recognition. Camera calibration is a topic of major interest in computer vision [5, 6]. However, line-scan cameras are different from area-scan cameras, as the second dimension of the image in line-scan cameras is created by the motion of the sensor with respect to the inspected object. This

way, the coordinate in the array of pixels in the line-scan camera follows a central projection, while the other coordinate follows an orthogonal projection. Therefore, methods used for camera calibration in area-scan cameras cannot be directly used for line-scan cameras. Line-scan camera calibration is thus a different problem, with scarce research and almost no tools availability.

Line-scan camera calibration follows two main approaches: scan-based calibration and line-based calibration. Methods based on scan-based calibration [7–9] acquire line-scans while the camera moves relatively from a calibration pattern or calibration object with well-known features. The sequence of line-scans is used to compose a two-dimensional synthetic image considering constant velocity [10]. Then, positions of features with well-known correspondences to world coordinates are detected in the image. Techniques and tools similar to area-scan cameras can then be used to calibrate the line-scan camera. These methods are robust and provide good accuracy. However, they require an elaborated and complex setup [11]. Moreover, these requirements are not suitable for industrial applications that may require periodic calibration [12].

Line-based calibration methods, also known as static calibration methods, are based on line-scan camera imaging models [13–16]. They do not create 2D synthetic images; a single line-scan is enough to calibrate the camera. Thus, they do not require the acquisition of multiple images while the calibration target

is moving, overcoming the limitations of scan-based calibration. However, the estimation of correspondences between world and image coordinates is more difficult. The most common method is derived from the procedure proposed in [13], where a pattern with parallel and diagonal lines is used. The geometry of the pattern and the principle of cross-ratio invariance is used to calculate the world coordinates. The correspondences in the line-scan are obtained using basic signal processing techniques. The procedure to calibrate line-scan cameras using line-based calibration methods is considerably easier, providing the possibility to calibrate the camera in harsh environments with minimal equipment.

Any camera calibration method requires, in general, three steps: estimation of correspondences between points in world and image coordinates, computation of a linear solution, and non-linear optimization using the linear estimate as a starting point. A mathematical model describes the mapping between the world and the image coordinates, such as the pinhole model. In order to estimate the parameters of the model, observations of the scene in world coordinates and the corresponding image need to be estimated. In area-scan images, a flat calibration plate with a printed pattern is commonly used, following the calibration method proposed in [17]. In line-based calibration methods, a common approach is to print parallel and diagonal lines on different planes. An alternative approach is to use a planar pattern observed with different orientations, but this requires a secondary area-scan camera [2, 11]. The result is a set of points in world coordinates that lie on the same plane, but they are not collinear. The next step is the computation of an initial estimation of the parameters models using a linear method. In area-scan cameras, the direct linear transformation is used [18]. Finally, a non-linear optimization method, such as the Newton or the Levenberg–Marquardt methods [19], is used to minimize a particular geometric error. The result is an accurate estimation of the parameter models including non-linear parameters, such as lens distortions.

The accuracy of non-linear optimization methods greatly depends on the starting point used to minimize the cost function. Using an incorrectly calculated starting point can cause the optimization method to converge to a local minimum rather than the global minimum, resulting in a slow suboptimal estimation of the camera model parameters. Moreover, there is a risk the optimization method does not converge, which also requires a difficult to establish stopping criterion. Even when the optimization method reaches the global optimum, the further it starts from the final solution the larger the number of iterations required. Non-linear methods can also avoid using an starting point by sampling the parameter space. Genetic algorithms are examples of such optimization methods [20]. These methods can even avoid premature convergence to local minimum, but requiring a large number of iterations and with no guaranteed convergence to the global minimum. Therefore, an accurate and robust linear solution as starting point is always preferred and crucial for the ability of the calibration procedure to produce the expected results. This work is focused on such linear solution for line-based calibration methods.

This work proposes a robust method for estimating the camera projection matrix from corresponding world and image coordinates in a line-scan camera, providing the detailed mathematical procedure. The robust method estimates a well-conditioned linear solution for the calibration of line-scan cameras that can be solved analytically with no optimization required and minimum constraints. The main contributions of the proposed approach

are the following:

- The individual intrinsic and extrinsic parameters are estimated from a single line-scan, not requiring movements between the camera and the calibration target, and thus suitable for the application in industrial environments.
- The proposed approach does not require optimization methods to estimate the parameters, which represents a major advantage compared with other methods in the literature. The calibration can be solved analytically with no optimization required and minimum constraints. This represents an advantage compared with previous works that have proposed linear methods based on variations of the direct linear transformation, which lead to rank deficiency problems due to the coplanarity of the points in the scene. This is solved using constraints for optimizations that provide unstable results.
- No assumption is established about the orientation or position of the camera. In previous works, the orientation of the camera leads to an ill-conditioned problem where no solution can be found.
- The proposed solution is stable, even when considering Gaussian distributed noise. A robust estimation approach is also proposed to deal with outliers.
- The linear estimation of the projection parameters can be used to initialize the search for non-linear optimizers that guarantee a final accurate estimation of both linear and non-linear camera parameters. This improves the calibration accuracy by also estimating non-linear distortions.
- The proposed method does not depend on the particular approach used to estimate the world and image correspondences. Calibration targets with multiples flat planes are the most common, but a single flat calibration target could also be used with a secondary area scan camera.

The remainder of this paper is organized as follows. Section 2 introduces Line-scan camera imaging model; Section 3 presents the proposed approach for the robust computation of the camera matrix; Section 4 discusses the results obtained including the application of non-linear optimization methods; and finally, Section 5 reports conclusions.

## 2. LINE-SCAN CAMERA IMAGING MODEL

A line-scan camera is modeled by the pinhole model or central projection as shown in Figure 1. Based on this model, the projection of a 3D point in world coordinates denoted by  $M = [X, Y, Z]^T$  onto the 2D point in image coordinates  $m = [u, v]^T$  is given by (1), where  $s$  is an arbitrary scale factor,  $\mathcal{P}$  is the camera projection matrix, and  $M'$  and  $m'$  are the homogeneous coordinates of  $M$  and  $m$ . Figure 1 also represents the most common layout of calibration target with parallel and diagonal lines on different planes.

$$s m' = \mathcal{P} M' \quad (1)$$

A line-scan camera is a special case of camera consisting of only one array of pixels. Thus, a 3D point is projected onto the image point  $[0, v]^T$ , which can be described using (2), where  $Rt$  is the rotation and translation matrix that relates the world and the camera coordinate systems, and  $\mathcal{A}$  is the camera intrinsic matrix

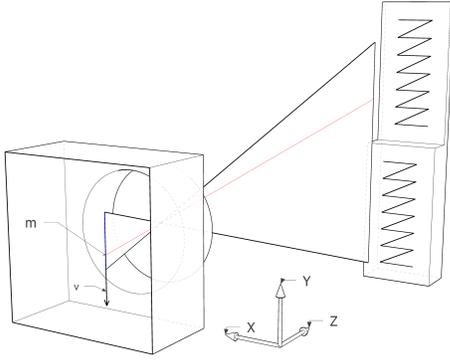


Fig. 1. Line-scan camera imaging model

that describe the camera internal parameters. The composition of  $\mathcal{A}$  and  $Rt$  gives  $\mathcal{P}$ .

$$s \begin{pmatrix} 0 \\ v \\ 1 \end{pmatrix} = \mathcal{A} R t \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \mathcal{P} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2)$$

The camera intrinsic matrix,  $\mathcal{A}$ , combines a perspective projection and a 2D affine transformation expressed as (3), where  $f_y$  is the focal length in the image  $v$  axis expressed in pixel units, and  $c_y$  is the principal point that is usually at the center of the array of pixels. These parameters are often referred to as intrinsic parameters. The imaging model assumes that the sensor line is mounted exactly behind the principal point and the principal distance perpendicular to the sensor line is ignored.

$$\mathcal{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (3)$$

The transformation from world coordinates in the scene to camera coordinates is performed using a 3D rigid-body transformation that involves three rotations ( $\alpha, \beta, \gamma$ ) and three translations ( $t_1, t_2, t_3$ ). These parameters are often referred to as extrinsic parameters. The transformation can be expressed as (4), where  $r_{ij}$  are the coefficients of the rotation matrix using the angles  $\alpha, \beta$ , and  $\gamma$ .

$$Rt = TR = \begin{pmatrix} 1 & 0 & 0 & t_1 \\ 0 & 1 & 0 & t_2 \\ 0 & 0 & 1 & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4)$$

The camera projection matrix,  $\mathcal{P}$ , is the composition of  $\mathcal{A}$  and  $Rt$ , resulting in (5).

$$\mathcal{P} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ c_y r_{31} + f_y r_{21} & c_y r_{32} + f_y r_{22} & c_y r_{33} + f_y r_{23} & c_y t_3 + f_y t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{pmatrix} \quad (5)$$

Consequently, the line-scan camera model, not considering lens distortion, can be represented using (6) and (7). Therefore, eight parameters describe the line-scan camera model, in which two are the intrinsic parameters ( $f_y, c_y$ ) and six are the extrinsic parameters ( $\alpha, \beta, \gamma, t_1, t_2, t_3$ ).

$$0 = X r_{11} + Y r_{12} + Z r_{13} + t_1 \quad (6)$$

$$v = c_y + f_y \frac{X r_{21} + Y r_{22} + Z r_{23} + t_2}{X r_{31} + Y r_{32} + Z r_{33} + t_3} \quad (7)$$

The accuracy of this linear model can be improved by considering lens distortions, which provoke non-linear projections of the scene points onto the image. The most common distortion model is the polynomial model [17]. Other models also exist, such as the division model [21]. In area-scan cameras, 8 coefficients are considered, 6 to model the radial distortion, and 2 to model the tangential distortion. In line-scan cameras, the most common approach is to use only 3 parameters to model radial distortion. Radial distortion is related to light rays bending near the edges of the lens. The radial distortion coefficients are part of the intrinsic parameters, as they do not depend on the scene viewed.

Radial distortion is modeled using (8), where  $v'$  denote the distorted image point,  $r = v - c_y$  is the distortion radius, and  $k_1, k_2$  and  $k_3$  are the radial distortion coefficients.

$$v' = v (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \quad (8)$$

### A. Properties of the rotation matrix

A rotation matrix,  $R$ , is an orthogonal matrix that satisfies  $R R^T = I$ , where  $I$  is the identity matrix [22]. Therefore,  $R^T = R^{-1}$ . A rotation is a linear transformation that preserves angles and lengths. Thus,  $\det(R) = \pm 1$ . The row and column vectors of  $R$  are orthogonal and of unit norm [23].

In  $\mathbb{R}^3$ , a rotation can be represented using rotation vectors as (9).

$$R = \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \quad (9)$$

Given the coefficients of the rotation matrix,  $r_{ij}$ , and a rotation order around the axes, the Euler angles  $\alpha, \beta$  and  $\gamma$  can be calculated [24].

Because the rows and columns of the rotation matrix are orthogonal, the row vectors satisfy (10). This represents the 6 constraints represented in (11). Thus, a rotation matrix in  $\mathbb{R}^3$  has 9 coefficients, but only 3 degrees of freedom ( $\alpha, \beta$  and  $\gamma$ ).

$$r_i r_j^T = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \quad (10)$$

$$\begin{cases} r_{11}^2 + r_{12}^2 + r_{13}^2 = 1 \\ r_{21}^2 + r_{22}^2 + r_{23}^2 = 1 \\ r_{31}^2 + r_{32}^2 + r_{33}^2 = 1 \\ r_{11} r_{21} + r_{12} r_{22} + r_{13} r_{23} = 0 \\ r_{11} r_{31} + r_{12} r_{32} + r_{13} r_{33} = 0 \\ r_{21} r_{31} + r_{22} r_{32} + r_{23} r_{33} = 0 \end{cases} \quad (11)$$

The orthonormality of the rotation matrix can also be used to express the relations in (12), i.e., the cross product of two given rotation vectors is equal to the third.

$$\begin{cases} r_2 \times r_3 = r_1 \\ r_1 \times r_3 = r_2 \\ r_1 \times r_2 = r_3 \end{cases} \quad (12)$$

The coefficients of the rotation matrix,  $r_{ij}$ , represent trigonometric operations with angles  $\alpha$ ,  $\beta$  and  $\gamma$ . These coefficients satisfy the relations in (13), regardless of the rotation order.

For example, the demonstration of the first relation in (13) can be seen in (14).

$$\begin{cases} r_{11} r_{22} - r_{12} r_{21} = r_{33} \\ r_{22} r_{33} - r_{23} r_{32} = r_{11} \\ r_{11} r_{23} - r_{13} r_{21} = -r_{32} \\ r_{11} r_{32} - r_{12} r_{31} = -r_{23} \\ r_{11} r_{22} - r_{12} r_{21} = r_{33} \\ r_{22} r_{33} - r_{23} r_{32} = r_{11} \\ r_{11} r_{33} - r_{13} r_{31} = r_{22} \\ r_{12} r_{33} - r_{13} r_{32} = -r_{21} \\ r_{12} r_{23} - r_{13} r_{22} = r_{31} \\ r_{12} r_{33} - r_{13} r_{32} = -r_{21} \end{cases} \quad (13)$$

$$\begin{aligned} r_{11} r_{22} - r_{12} r_{21} &= \\ \cos(\beta) \cos(\gamma) \left( \cos(\alpha) \cos(\gamma) - \sin(\alpha) \sin(\beta) \sin(\gamma) \right) \\ + \cos(\beta) \sin(\gamma) \left( \cos(\alpha) \sin(\gamma) + \cos(\gamma) \sin(\alpha) \sin(\beta) \right) &= \\ \cos(\alpha) \cos(\beta) \cos(\gamma)^2 + \cos(\alpha) \cos(\beta) \sin(\gamma)^2 &= \\ \cos(\alpha) \cos(\beta) &= r_{33} \end{aligned} \quad (14)$$

### 3. COMPUTATION OF THE LINE-SCAN CAMERA MATRIX

The procedure to calculate the camera projection matrix requires a set of points  $M_i$  in world coordinates and the set of corresponding points  $m_i$  in the image. The points  $M_i$  are constrained to lie on the viewing plane. The objective is to compute the projective transformation,  $\mathcal{P}$ , that maps  $M_i$  to  $m_i$ .

The development of (2) produces two equations: (6) and (7). The first equation, (6), represents the viewing plane. The variables of this equation are the first row of  $\mathcal{P}$ , described in (5). The second equation, (7), represent the central projection in the camera. The variables of this equation are the second and third row of  $\mathcal{P}$ . These equations have different unknown variables. Thus, they can be solved independently. The proposed approach for the calculation of the viewing plane and the central projection is described next.

### A. Calculation of the viewing plane

The calculation of the viewing plane can be seen as fitting a plane from a set of points. Many algorithms make assumptions about the orientation. These methods can fail when the plane is (nearly) parallel to a specific axis, such as  $Z = 0$ . This work uses a method based on singular value decomposition that minimizes the orthogonal distance making no assumptions about the orientation.

The viewing plane can be defined using (15). In order to calculate the coefficients of the plane, at least 3 non-collinear points are required. However, due to the influence of noise more points are required to robustly calculate the coefficients of the plane.

$$X r_{11} + Y r_{12} + Z r_{13} + t_1 = 0 \quad (15)$$

Given  $n$  points on the plane, the coefficients of the plane can be obtained by solving the system of homogeneous equation in (16), where  $A_J$  and  $J$  can be expressed as (17) and (18). The obvious solution to this over-determined set of equations is  $J = 0$ , but this solution is not of interest. Moreover, if  $J$  is a solution of this set of equations, then  $\rho J$  is also a solution for any scalar  $\rho$ . Thus, a constraint is established:  $\|J\| = 1$ . Then, the least-squares solution can be estimated.

$$A_J J = 0 \quad (16)$$

$$A_J = \begin{pmatrix} X_1 & Y_1 & Z_1 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ X_n & Y_n & Z_n & 1 \end{pmatrix} \quad (17)$$

$$J = \begin{pmatrix} r_{11} \\ r_{12} \\ r_{13} \\ t_1 \end{pmatrix} \quad (18)$$

The matrix  $A_J$  can be decomposed using singular value decomposition as (19).

$$A_J = USV^T \quad (19)$$

The solution of the system of homogeneous equation is the last column of  $V$ ,  $V^{last}$ , which is the eigenvector of  $A_J^T A_J$  corresponding to the smallest eigenvalue [18]. The solution,  $J'$ , is worked out up to a scale factor, i.e.,  $J' = V^{last} = \rho J$ , where the value of  $\rho$  is unknown. However, considering the constraint established by  $r_1$  in (11), the coefficients of the plane can be finally calculated using (20).

$$\begin{pmatrix} r_{11} \\ r_{12} \\ r_{13} \\ t_1 \end{pmatrix} = \frac{\pm 1}{\sqrt{(J'_1)^2 + (J'_2)^2 + (J'_3)^2}} \begin{pmatrix} J'_1 \\ J'_2 \\ J'_3 \\ J'_4 \end{pmatrix} \quad (20)$$

### B. Calculation of the projection

A possible solution for the estimation of the projection represented by the last two rows of  $\mathcal{P}$  is to extract  $X$  from (6) and substitute it into (7). This is the procedure proposed in [13]

and followed in more recent works [9, 11, 14]. However, this method is numerically unstable depending on the orientation of the camera. Extracting and substituting  $X$  when the viewing plane is close to  $X = 0$  will fail. Moreover, when the plane is nearly parallel to  $X = 0$  the method is very sensitive to changes or errors in the input, leading to and ill-conditioned problem. Therefore, this is not a completely general solution. The robust approach is to pick the path with best conditioning: extracting the variable with the largest coefficient in the viewing plane. Only when  $|r_{11}| > \max(|r_{12}|, |r_{13}|)$  shall  $X$  be used. When  $|r_{21}| > \max(|r_{11}|, |r_{13}|)$ ,  $Y$  is used; and when  $|r_{13}| > \max(|r_{11}|, |r_{12}|)$ ,  $Z$  is used. The procedure for each variable is similar, but the resulting equations are different. The result is a well-conditioned problem than can be solved analytically very robustly with no optimization required.

When  $r_{11} > \max(r_{12}, r_{13})$ ,  $X$  is extracted from (6) resulting in (21).

$$X = -\frac{t_1 + Y r_{12} + Z r_{13}}{r_{11}} \quad (21)$$

Substituting  $X$  into (7) results in (22), where the numerator,  $N$ , can be represented as (23) and the denominator,  $D$ , as (24).

$$v = \frac{N}{D} \quad (22)$$

$$N = (r_{11} r_{22} - r_{12} r_{21}) Y f_y + (r_{11} r_{32} - r_{12} r_{31}) Y c_y \\ + (r_{11} r_{23} - r_{13} r_{21}) Z f_y + (r_{11} r_{33} - r_{13} r_{31}) Z c_y \\ + (r_{11} t_2 - r_{21} t_1) f_y + (r_{11} t_3 - r_{31} t_1) c_y \quad (23)$$

$$D = (r_{11} r_{32} - r_{12} r_{31}) Y + (r_{11} r_{33} - r_{13} r_{31}) Z + r_{11} t_3 - r_{31} t_1 \quad (24)$$

Considering the relations between the coefficients of the rotation matrix in (13), (23) and (24) can be further simplified as (25) and (26).

$$N = (f_y r_{33} - c_y r_{23}) Y + (c_y r_{22} - f_y r_{32}) Z \\ + c_y (r_{11} t_3 - r_{31} t_1) + f_y (r_{11} t_2 - r_{21} t_1) \quad (25)$$

$$D = -Y r_{23} + Z r_{22} + r_{11} t_3 - r_{31} t_1 \quad (26)$$

Renaming variables according to (27) results in (28), which represents a projection. This equation can be expressed as (29).

$$\begin{cases} K_1 = f_y r_{33} - c_y r_{23} \\ K_2 = c_y r_{22} - f_y r_{32} \\ K_3 = c_y (r_{11} t_3 - r_{31} t_1) + f_y (r_{11} t_2 - r_{21} t_1) \\ K_4 = -r_{23} \\ K_5 = r_{22} \\ K_6 = r_{11} t_3 - r_{31} t_1 \end{cases} \quad (27)$$

$$v = \frac{K_1 Y + K_2 Z + K_3}{K_4 Y + K_5 Z + K_6} \quad (28)$$

$$Y K_1 + Z K_2 + 1 K_3 - v Y K_4 - v Z K_5 - v K_6 = 0 \quad (29)$$

Given  $n$  points, the coefficients of  $K = [K_1, \dots, K_6]^T$  can be obtained by the system of homogeneous equation in (30), where

$A_K$  can be expressed as (31). A minimum of 6 points are required to solve the system of equations with 6 unknowns.

$$A_K K = 0 \quad (30)$$

$$A_K = \begin{pmatrix} Y_1 & Z_1 & 1 & -v Y_1 & -v Z_1 & -v \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_n & Z_n & 1 & -v Y_n & -v Z_n & -v \end{pmatrix} \quad (31)$$

The matrix  $A_K$  can be decomposed using singular value decomposition as before. The solution of the system of homogeneous equation is the eigenvector of  $A_K$  corresponding to the smallest eigenvalue. This provides a solution up to a scale factor, i.e., the obtained solution is  $K' = \lambda K$ , with the value of  $\lambda$  unknown. Thus, this solution provides the values of  $\lambda r_{23} = -K'_4$  and  $\lambda r_{22} = K'_5$  considering the variables in (27).

Because  $r_1$  and  $r_2$  are perpendicular, the dot product of these vectors is null, as indicated in (11). Thus, (32) can be obtained by multiplying both sides by  $\lambda$ . Solving this equation for  $\lambda r_{21}$  results in (33). This division is always safe as  $r_1$  is a unit vector and  $|r_{11}| > \max(|r_{12}|, |r_{13}|)$ , i.e.,  $|r_{11}| > 0$ .

$$0 = \lambda r_{11} r_{21} + \lambda r_{12} r_{22} + \lambda r_{13} r_{23} \quad (32)$$

$$\lambda r_{21} = -\frac{r_{12} \lambda r_{22} + r_{13} \lambda r_{23}}{r_{11}} \quad (33)$$

The value of  $\lambda$  can finally be obtained from (34) using the constraint about the unit length of the rotation vector.

Because  $r_2$  is a rotation vector, the length is 1 as indicated in (11). This can be also expressed as (34). Thus, (35) can be obtained by multiplying both sides by  $\lambda$ . This gives the solution for  $\lambda$ .

$$1 = \pm \sqrt{(r_{21})^2 + (r_{22})^2 + (r_{23})^2} \quad (34)$$

$$\lambda = \pm \sqrt{(\lambda r_{21})^2 + (\lambda r_{22})^2 + (\lambda r_{23})^2} \quad (35)$$

Using the calculated value of  $\lambda$ , the value of  $r_2$  can be obtained from (36).

$$\begin{pmatrix} r_{21} \\ r_{22} \\ r_{23} \end{pmatrix} = \frac{1}{\lambda} \begin{pmatrix} \lambda r_{21} \\ \lambda r_{22} \\ \lambda r_{23} \end{pmatrix} \quad (36)$$

The orthonormality properties of the rotation matrix described in (12) can be used to calculate  $r_3$  using (37).

$$\begin{pmatrix} r_{31} \\ r_{32} \\ r_{33} \end{pmatrix} = \begin{pmatrix} r_{11} \\ r_{12} \\ r_{13} \end{pmatrix} \times \begin{pmatrix} r_{21} \\ r_{22} \\ r_{23} \end{pmatrix} \quad (37)$$

The coefficients of  $K$  can be obtained from (38). Finally, the rest of the required variables to calculate the coefficients of the camera projection matrix  $\mathcal{P}$  can be calculated from (39).

$$K = \frac{1}{\lambda} K', \quad (38)$$

$$\begin{cases} f_y = \frac{K_1 r_{22} + K_2 r_{23}}{r_{22} r_{33} - r_{23} r_{32}} \\ c_y = \frac{K_1 r_{32} + K_2 r_{33}}{r_{22} r_{33} - r_{23} r_{32}} \\ t_2 = -\frac{K_1 K_6 r_{32} + K_2 K_6 r_{33} - K_3 r_{22} r_{33} + K_3 r_{23} r_{32} - K_1 r_{21} r_{22} t_1 - K_2 r_{21} r_{23} t_1}{r_{11} (K_1 r_{22} + K_2 r_{23})} \\ t_3 = \frac{K_6 + r_{31} t_1}{r_{11}} \end{cases} \quad (39)$$

When  $|r_{12}| > \max(|r_{11}|, |r_{13}|)$  the procedure is similar but it all starts by extracting  $Y$  from (6), which results in (40).

$$Y = -\frac{t_1 + X r_{11} + Z r_{13}}{r_{12}} \quad (40)$$

Substituting  $Y$  into (7) and renaming variables according to (41), the solution in this case is obtained from the singular value decomposition of (42).

$$\begin{cases} K_1 = c_y r_{23} - f_y r_{33} \\ K_2 = f_y r_{31} - c_y r_{21} \\ K_3 = c_y (r_{12} t_3 - r_{32} t_1) + f_y (r_{12} t_2 - r_{22} t_1) \\ K_4 = r_{23} \\ K_5 = -r_{21} \\ K_6 = r_{12} t_3 - r_{32} t_1 \end{cases} \quad (41)$$

$$A_K = \begin{pmatrix} X_1 & Z_1 & 1 & -v X_1 & -v Z_1 & -v \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_n & Z_n & 1 & -v X_n & -v Z_n & -v \end{pmatrix} \quad (42)$$

Using a similar approach, the unknown variables required for the projection are obtained from (43) and (44), with  $|r_{12}|$  always guaranteed  $> 0$ .

$$\lambda r_{22} = -\frac{r_{11} \lambda r_{21} + r_{13} \lambda r_{23}}{r_{12}} \quad (43)$$

$$\begin{cases} f_y = -\frac{K_1 r_{21} + K_2 r_{23}}{r_{21} r_{33} - r_{23} r_{31}} \\ c_y = -\frac{K_1 r_{31} + K_2 r_{33}}{r_{21} r_{33} - r_{23} r_{31}} \\ t_2 = -\frac{K_1 K_6 r_{31} + K_2 K_6 r_{33} + K_3 r_{21} r_{33} - K_3 r_{23} r_{31} - K_1 r_{21} r_{22} t_1 - K_2 r_{22} r_{23} t_1}{r_{12} (K_1 r_{21} + K_2 r_{23})} \\ t_3 = \frac{K_6 + r_{32} t_1}{r_{12}} \end{cases} \quad (44)$$

When  $|r_{13}| > \max(|r_{11}|, |r_{12}|)$ ,  $Z$  is extracted from (6) resulting in (45).

$$Z = -\frac{t_1 + X r_{11} + Y r_{12}}{r_{13}} \quad (45)$$

Following a similar procedure,  $Z$  is substituted into (7) and variables are renamed according to (46). The solution is now obtained from the singular value decomposition of (47).

$$\begin{cases} K_1 = f_y r_{32} - c_y r_{22} \\ K_2 = c_y r_{21} - f_y r_{31} \\ K_3 = c_y (r_{13} t_3 - r_{33} t_1) + f_y (r_{13} t_2 - r_{23} t_1) \\ K_4 = -r_{22} \\ K_5 = r_{21} \\ K_6 = r_{13} t_3 - r_{33} t_1 \end{cases} \quad (46)$$

$$A_K = \begin{pmatrix} X_1 & Y_1 & 1 & -v X_1 & -v Y_1 & -v \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_n & Y_n & 1 & -v X_n & -v Y_n & -v \end{pmatrix} \quad (47)$$

The unknown variables required for the projection are obtained from (48) and (49), with  $|r_{13}|$  always guaranteed  $> 0$ .

$$\lambda r_{23} = -\frac{r_{12} \lambda r_{22} + r_{13} \lambda r_{23}}{r_{13}} \quad (48)$$

$$\begin{cases} f_y = \frac{K_1 r_{21} + K_2 r_{22}}{r_{21} r_{32} - r_{22} r_{31}} \\ c_y = \frac{K_1 r_{31} + K_2 r_{32}}{r_{21} r_{32} - r_{22} r_{31}} \\ t_2 = -\frac{K_1 K_6 r_{31} + K_2 K_6 r_{32} - K_3 r_{21} r_{32} + K_3 r_{22} r_{31} - K_1 r_{21} r_{23} t_1 - K_2 r_{22} r_{23} t_1}{r_{13} (K_1 r_{21} + K_2 r_{22})} \\ t_3 = \frac{K_6 + r_{33} t_1}{r_{13}} \end{cases} \quad (49)$$

### C. Non-linear optimization

The linear estimation of the projection matrix can be used as a robust starting point in the final step of the calibration: the non-linear optimization. This step greatly improves the calibration accuracy by also estimating non-linear distortions. The goal is to find the intrinsic and extrinsic parameters that minimize (50), where  $M'_i$  represents the homogeneous coordinates of world point  $M_i$ ,  $m'_i$  is its correspondence in the image,  $n$  is the number of points, and  $\mathcal{P}'$  is a function that projects the world points into the image considering lens distortions.

$$\min_{f_y, c_y, \alpha, \beta, \gamma, t_1, t_2, t_3, k_1, k_2, k_3} \sum_{i=1}^n [m'_i - \mathcal{P}'(M'_i)]^2 \quad (50)$$

### D. Robust estimation

The proposed approach provides an optimal solution when considering Gaussian distributed noise. However, it would fail if outliers were present in the data. This issue can be solved using a variation of RANSAC [25], a simple yet extremely robust method to fit models to noisy data. This method can provide a solution even when data is corrupted with a large number of outliers. The proposed approach is to divide all the data points in subsets of 6 point correspondences, the minimum number of points required to estimate the linear solution. These subsets are obtained from all the possible permutations of the points taken 6 at a time without repetition. Non-valid solutions are removed (collinear points). For each of these valid subsets, an estimation of the calibration is computed and the median reprojection distance for all the points is calculated. The subset that minimizes the median distance is the candidate for solution. This solution is then re-estimated for all points within a threshold distance (inliers). The processes is repeated until the number of inliers

converges, providing a robust solution that is not affected by the outliers. The method only needs to be applied when outliers are known to be present in the data. However, it is also safe to be applied when there are no outliers, providing the same results than then linear approach with an increased computational cost. In case it is needed, the procedure could be optimized to reduce this cost.

### E. Decomposition of the camera matrix

The previous procedure can be used to estimate the matrix  $\mathcal{P}$  given a set of 3D points in the scene and the corresponding image coordinates. For testing purposes, a given camera matrix,  $\mathcal{P}$ , can also be decomposed into the intrinsic and extrinsic parameters. In general, the camera matrix decomposition is performed using QR factorization method [26]. However, according to (5), the parameters can be estimated from (51), where  $\mathcal{P}_{ij}$  are the coefficients of the camera projection matrix. All operations are guaranteed to provide valid results.

$$\left\{ \begin{array}{l} [r_{11}, r_{12}, r_{13}] = [\mathcal{P}_{11}, \mathcal{P}_{12}, \mathcal{P}_{13}] \\ [r_{31}, r_{32}, r_{33}] = [\mathcal{P}_{31}, \mathcal{P}_{32}, \mathcal{P}_{33}] \\ [r_{21}, r_{22}, r_{23}] = [r_{11}, r_{12}, r_{13}] \times [r_{31}, r_{32}, r_{33}] \\ t_1 = \mathcal{P}_{14} \\ t_3 = \mathcal{P}_{34} \\ c_y = [\mathcal{P}_{21}, \mathcal{P}_{22}, \mathcal{P}_{23}] \cdot [\mathcal{P}_{31}, \mathcal{P}_{32}, \mathcal{P}_{33}] \\ f_y = \|[ [\mathcal{P}_{21}, \mathcal{P}_{22}, \mathcal{P}_{23}] \times [\mathcal{P}_{31}, \mathcal{P}_{32}, \mathcal{P}_{33}] \| \\ t_2 = (\mathcal{P}_{24} - c_y t_3) / f_y \end{array} \right. \quad (51)$$

The position of the camera in world coordinates,  $O^C$ , can be calculated using (52), which represents the transformation of the camera origin,  $[0, 0, 0, 1]$  in homogeneous coordinates, from the camera coordinates system to the world coordinates system.

$$O^C = (Rt)^{-1} [0, 0, 0, 1]^T \quad (52)$$

Given the homogeneous coordinate of a point in the image,  $m'_i$ , the set of points in space that map to this point constitutes a ray in space passing through the camera center. This is known as the back-projection of points to rays. This ray can be defined by  $O^C$  and  $\mathcal{P}^+ m'_i$ , where  $\mathcal{P}^+$  is the pseudo-inverse of  $\mathcal{P}$  that can be calculated using (53).

$$\mathcal{P}^+ = \mathcal{P}^T (\mathcal{P} \mathcal{P}^T)^{-1} \quad (53)$$

## 4. RESULTS AND DISCUSSION

### A. Application of the procedure

The calibration procedure requires a set of points in the scene in world coordinates and the correspondences in the line-scan. Figure 2a shows an example of points in world coordinates. In this case, the points are distributed in the viewing plane in ideal conditions.

In a real calibration, the experimental setup would include a calibration target with multiples planes from where these points could have been obtained. Another possible approach is a flat target moved around the scene. This latter approach usually requires a secondary camera to transform all coordinates into a single coordinate system. The line-scan camera observes the calibration target, producing line-scans with the coordinates of the control points.

The objective of the experiments is to estimate the camera intrinsic and extrinsic parameters from these points and the corresponding projections in the line-scan. In this first experiment, points are distributed in 10 planes, using 5 points per plane.

The first step of the procedure is the calculation of the viewing plane. The result can be seen in Figure 2b. The viewing plane where the world points lie on is estimated by minimizing the sum of the squares of the residuals using the proposed procedure. Given the coefficients of the viewing plane, the rest of the parameters of the projection matrix are calculated using the described approach. The result robustly identifies the intrinsic and extrinsic camera parameters. Figure 2c shows the resulting position and orientation of the line-scan camera.

The result of the camera calibration procedure allows for the projection of any point in the scene onto image coordinates. The inverse projection is an ill-defined problem, but the projection ray in world coordinates from any pixel in the image can also be calculated. Moreover, given the extrinsic parameters, transformations between coordinates systems can be easily performed. All these are advantages of having a calibrated system.

### B. Comparison with previous methods

In order to demonstrate the robustness of the proposed procedure, it is applied to a line-scan camera with different orientations. Also, the results are compared with the method proposed by Li [11, 14, 16], which is a common procedure to estimate a linear solution. The main principle of this method is also used in [13]. Thus, similar results are expected.

This experiment assumes a set of points distributed in the viewing plane and the correspondences in the line-scan are known. Points are distributed in different planes, as can be seen in Figure 2, with 10 planes and 5 points per plane. The camera is rotated following the order ZYX.

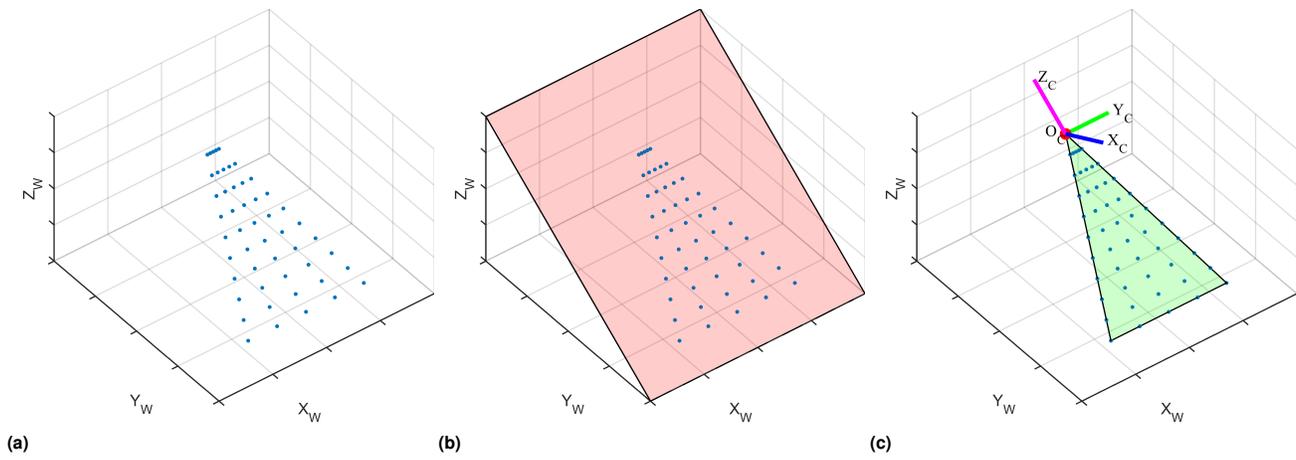
The metric used to assess the calibration error is the Root Mean Square Error (*RMSE*) defined in (54), which measures the differences between the image coordinates  $m'_i$  in homogeneous coordinates and the projection of the world points  $m'_i$  considering the estimated camera projection matrix  $\mathcal{P}$ , where  $n$  is the number of points.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (m'_i - \mathcal{P} M'_i)^2}{n}} \quad (54)$$

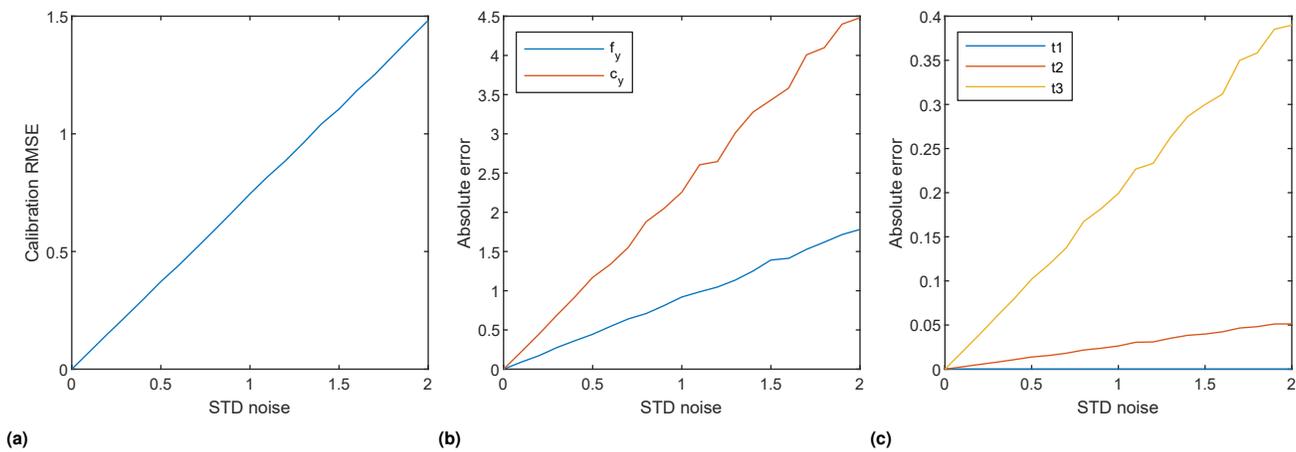
The results of the calibration are shown in Table 1. As can be seen, the proposed method provides similar results regardless of the orientation of the camera. On the other hand, the method proposed by Li fails when  $\gamma$  is 90° or very close. In the vicinity of this angle, the method provides a solution that is not optimal, greatly increasing the error compared with the proposed method. In all the considered cases, the proposed method provides a negligible error, thus, robust under different orientations. The reason why the compared method fails is the value of  $r_{11}$ . When this value zero or close, the mathematical operations are unstable.

### C. Performance with Gaussian noise

Gaussian noise is added to the image points to test the performance of the procedure with noisy data. The calibration (*RMSE*) is estimated considering the corrupted data. In addition, the calculated camera parameters are compared with the real values. The noise added to the image points has zero mean and *STD* standard deviation, varied from 0 to 2 in 0.1 increments. For



**Fig. 2.** Line-scan camera calibration. (a) Points in the scene in world coordinates, (b) Calculation of the viewing plane. (c) Computation of the projection: rotation and orientation



**Fig. 3.** Performance with added noise. (a) Calibration error, (b) Errors in intrinsic parameters. (c) Errors in translations parameters

Orientation (°)			Calibration error ( <i>RMSE</i> )	
$\alpha$	$\beta$	$\gamma$	Li	Proposed method
0	0	0	6.61e-07	6.61e-07
0	0	90	-	6.61e-07
70	0	85	6.11e-06	8.07e-07
70	0	90.001	2.79e-02	5.74e-07
70	0	89.999	1.11e-01	4.04e-07

**Table 1.** Results with different orientations

each experiment, 1000 repetitions are performed and the results are averaged. The results can be seen in Figure 3.

The results indicate that errors increase with the noise level, as expected. As can be seen in Figure 3a there is a linear relationship between the noise level and the resulting calibration error. Figures 3b and 3c indicate that  $c_y$  and  $t_3$  are the most sensitive parameters to noise.  $t_1$  is not affected by noise in the image points, as it is directly calculated from the world points. These results demonstrate that the procedure remains stable despite the added noise in the data.

#### D. Performance with outliers

This experiment evaluates the performance of the proposed procedure when data is corrupted with outliers, which can be caused by reflections or other motives. The experiments in Table 1 are repeated adding a varying percentage of outliers, from 10% to 40%. The results are similar to those obtained without the outliers. Thus, the proposed approach is able to cope with a large proportion of outliers producing the same accurate results. Figure 4 shows the results in one of the test. As can be seen, the calibration parameters are estimated correctly even when there is a large number of outliers in the data.

#### E. Performance with different number of planes

In order to calibrate the line-scan camera points must not be collinear, as it is not possible to estimate a plane with only from points on a line. Thus, multiples points must be located in different planes, for example using a calibration target designed this way. As the number of planes increases, the calibration should provide better results due to the increased information available. This experiment tests the performance of the calibration procedure with a different number of planes. The results can be seen in Figure 5.

Calibration errors decrease when the number of planes increases. The relation is not linear, errors decrease faster when using a low number of planes. From 10 planes onwards, the results are almost negligible. The intrinsic parameter  $c_y$  and the translation  $t_3$  are also the most sensitive parameters to the variation in the number of planes. In the rotation parameters,  $\beta$  is more sensitive than the others. Calibrations errors are not given in this experiment, as decreasing the number of points can also decrease the overall calibration error due to overfitting.

#### F. Performance with lens distortion

This work presents a robust linear approach to calibrate line-scan cameras, which provides the starting point for the non-linear optimization required to estimate lens distortion. Calibration

only using a linear approach will not provide good results when the lens in the camera include aberrations, as they provoke deviations from the linear projection. This is why non-linear optimization is a required step to obtain a highly accurate calibration. This experiment shows the calibration error when the camera lens presents radial distortion.

The results can be seen in Figure 6, varying  $k_1$  from 0 to 0.1 in 0.01 increments. The results show large errors in the calibration (*RMSE*) as well as the intrinsic parameters. When optical distortions are present, the linear estimation deviates from the real camera parameters. This demonstrates that linear calibration is not a final procedure. It could only be applied when no lens distortions are present. In general, non-linear optimization is required for better accuracy.

#### G. Performance with non-linear optimization

Linear calibration methods cannot deal with optical distortions. Thus, large errors are generated when lens aberrations are present. The solution is to use a non-linear optimization method as the final step of the calibration. This step not only estimates non-linear camera parameters, such as radial distortions, it also finely adjusts the estimation about the rest of the parameters.

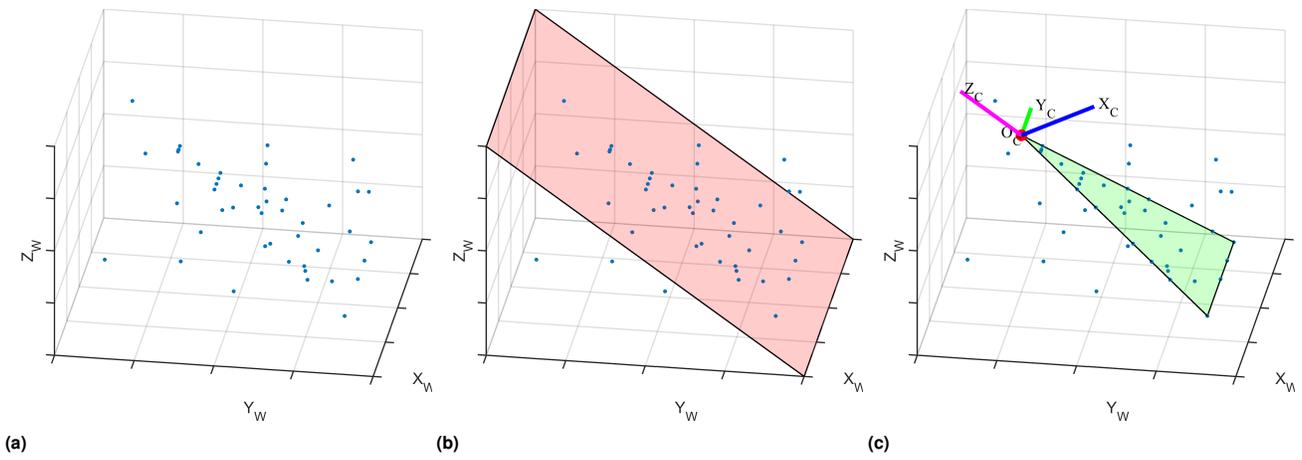
A comparison between the linear and non-linear calibration is shown in Table 2, where the non-linear optimization uses the linear estimate as a starting point as given in (50). As can be seen, the performance of the linear calibration degrades rapidly as radial distortion increases. However, the estimation of the camera parameters by the linear calibration can be used effectively by the non-linear method as a robust starting point. Regardless of the radial distortion, the non-linear optimization method provides a very low calibration error. Thus, the linear calibration provides the required solution to enable the last calibration step to produce the expected results. Moreover, similar calibration errors are obtained regardless of camera orientation and other parameters.

$k_1$	Linear ( <i>RMSE</i> )	Non-linear ( <i>RMSE</i> )
0.00	6.61e-07	1.15e-12
0.01	5.21e-01	8.84e-12
0.04	2.08e+00	5.37e-07
0.05	2.60e+00	1.94e-07
0.08	4.17e+00	6.14e-07
0.10	5.21e+00	6.66e-06

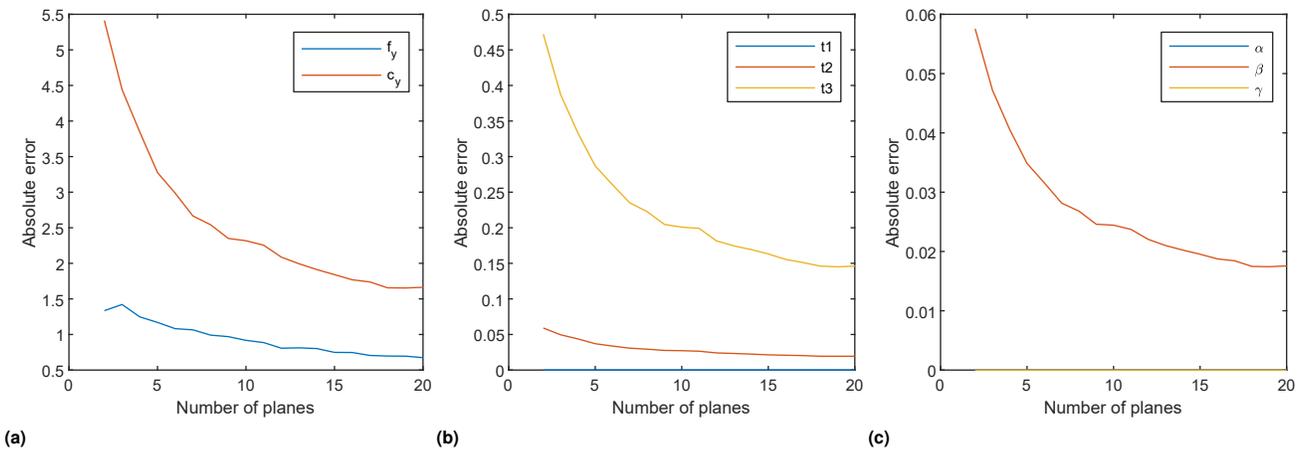
**Table 2.** Results with non-linear optimization

## 5. CONCLUSIONS

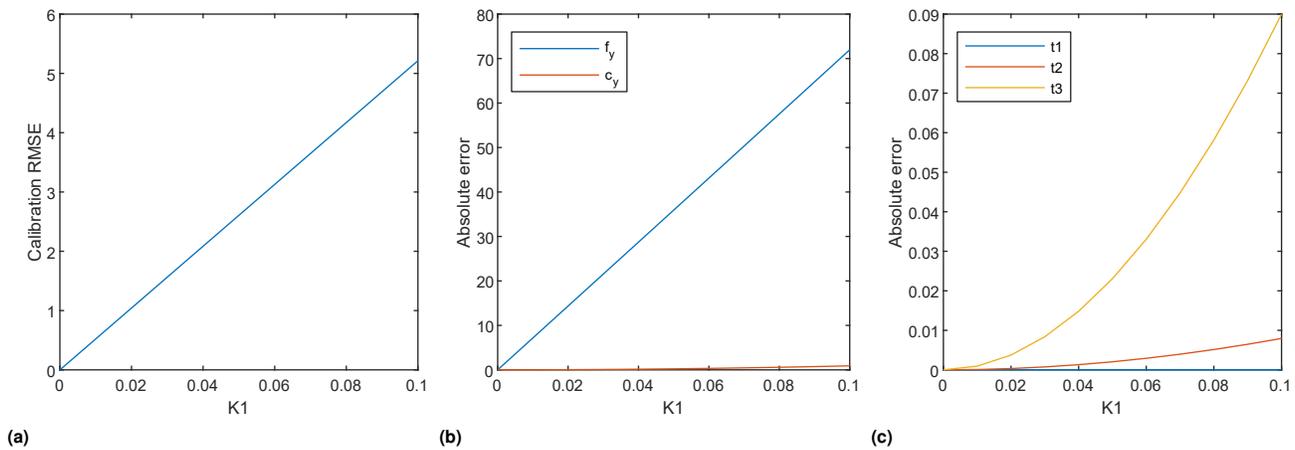
This work proposed a static camera calibration method for line-scan cameras that provides an accurate solution with no assumption about the orientation or position. The proposed method is robust and accurate, only requiring a set of non-collinear points in world coordinates and the correspondences in the image. These correspondences can be easily obtained using the principle of cross-ratio invariance, not requiring movements between the camera and the calibration target. Thus, suitable for the application in industrial environments. The estimation of the intrinsic and extrinsic camera parameters can be used effectively



**Fig. 4.** Line-scan camera calibration with outliers. (a) Points in the scene in world coordinates, (b) Calculation of the viewing plane. (c) Computation of the projection: rotation and orientation



**Fig. 5.** Performance with different number of planes. (a) Errors in intrinsic parameters. (b) Errors in translation parameters, (c) Errors in rotation parameters



**Fig. 6.** Performance with lens distortion. (a) Calibration error, (b) Errors in intrinsic parameters. (c) Errors in translations parameters

by non-linear optimization methods to finely adjust the camera parameters, including non-linear distortions.

Tests demonstrate the accuracy and robustness of the proposed procedure. Unlike previous methods, it provides very accurate results regardless of the orientation of the camera. In addition, it shows a stable behavior with noisy data, linearly increasing the calibration error as the noise level increases. Tests also indicate it can be applied with a different number of planes. As expected, the performance when lens distortions are present is not good. However, tests verify the linear estimation of the parameters can be used effectively to obtain a very accurate calibration even when high distortions are present. Overall, the proposed method provides a robust and accurate solution for a critical step in camera calibration.

## FUNDING

Project RTI2018-094849-B-I00 of the Spanish National Plan for Research, Development and Innovation.

## DISCLOSURES

The authors declare no conflicts of interest.

## REFERENCES

1. D.-M. Tsai and T.-Y. Huang, "Automated surface inspection for statistical textures," *Image and Vision computing* **21**, 307–323 (2003).
2. B. Sun, J. Zhu, L. Yang, S. Yang, and Z. Niu, "Calibration of line-scan cameras for precision measurement," *Applied optics* **55**, 6836–6843 (2016).
3. D. F. Garcia, R. Usamentiaga, I. Marin, J. A. Gonzalez, and N. de Abajo, "Shape inspection system for variable-luminance steel plates with real-time adaptation capabilities to luminance variations," *Real-Time Imaging* **8**, 303–315 (2002).
4. X. Tao, Z. Zhang, F. Zhang, and D. Xu, "A novel and effective surface flaw inspection instrument for large-aperture optical elements," *IEEE Transactions on Instrumentation and Measurement* **64**, 2530–2540 (2015).
5. J. Salvi, X. Armangué, and J. Batlle, "A comparative review of camera calibrating methods with accuracy evaluation," *Pattern recognition* **35**, 1617–1635 (2002).
6. C. S. Fraser, "Automatic camera calibration in close range photogrammetry," *Photogrammetric Engineering & Remote Sensing* **79**, 381–388 (2013).
7. R. Gupta and R. I. Hartley, "Linear pushbroom cameras," *IEEE Transactions on pattern analysis and machine intelligence* **19**, 963–975 (1997).
8. J. Draréni, S. Roy, and P. Sturm, "Plane-based calibration for linear cameras," *International Journal of Computer Vision* **91**, 146–156 (2011).
9. B. Hui, G. Wen, Z. Zhao, and D. Li, "Line-scan camera calibration in close-range photogrammetry," *Optical Engineering* **51**, 053602 (2012).
10. C. Steger and M. Ulrich, "A camera model for line-scan cameras with telecentric lenses," *International Journal of Computer Vision* pp. 1–20 (2020).
11. D. D. Li, G. Wen, and S. Qiu, "Cross-ratio-based line scan camera calibration using a planar pattern," *Optical Engineering* **55**, 014104 (2016).
12. R. Usamentiaga and D. Garcia, "Multi-camera calibration for accurate geometric measurements in industrial environments," *Measurement* **134**, 345–358 (2019).
13. R. Horaud, R. Mohr, and B. Lorecki, "On single-scanline camera calibration," *IEEE transactions on robotics and automation* **9**, 71–75 (1993).
14. D. Li, G. Wen, B. W. Hui, S. Qiu, and W. Wang, "Cross-ratio invariant based line scan camera geometric calibration with static linear data," *Optics and Lasers in Engineering* **62**, 119–125 (2014).
15. R. Liao, J. Zhu, L. Yang, J. Lin, B. Sun, and J. Yang, "Flexible calibration method for line-scan cameras using a stereo target with hollow stripes," *Optics and Lasers in Engineering* **113**, 6–13 (2019).
16. K. Song, B. Hou, M. Niu, X. Wen, and Y. Yan, "Flexible line-scan camera calibration method using a coded eight trigrams pattern," *Optics and Lasers in Engineering* **110**, 296–307 (2018).
17. Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence* **22**, 1330–1334 (2000).
18. R. Hartley and A. Zisserman, *Multiple view geometry in computer vision* (Cambridge university press, 2003).
19. H. Yu and B. M. Wilamowski, "Levenberg-marquardt training," *Industrial electronics handbook* **5**, 1 (2011).
20. N. Das, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, "A genetic algorithm based region sampling for selection of local features in handwritten digit recognition application," *Applied Soft Computing* **12**, 1592–1606 (2012).
21. A. W. Fitzgibbon, "Simultaneous linear estimation of multiple view geometry and lens distortion," in "Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001," , vol. 1 (IEEE, 2001), vol. 1, pp. 1–1.
22. D. Gruber, "The mathematics of the 3d rotation matrix," in "Xtreme Game Developers Conference," (2000), pp. 1–14.
23. K. Shoemake, "Animating rotation with quaternion curves," in "Proceedings of the 12th annual conference on Computer graphics and

- interactive techniques," (1985), pp. 245–254.
24. A. L. Schwab and J. P. Meijaard, "How to draw euler angles and utilize euler parameters," in "International Design Engineering Technical Conferences and Computers and Information in Engineering Conference," , vol. 42568 (2006), vol. 42568, pp. 259–265.
  25. P. H. Torr and A. Zisserman, "Mlesac: A new robust estimator with application to estimating image geometry," *Computer vision and image understanding* **78**, 138–156 (2000).
  26. D. S. Watkins, *Fundamentals of matrix computations*, vol. 64 (John Wiley & Sons, 2004).