AutoCalib

Luis F. Recalde[†]

***Abstract*— Camera calibration is an important process in computer vision that aims to determine a camera's intrinsic and extrinsic parameters. This process estimates lens distortions, focal length, and defines the relationship between 2D image points and real-world 3D coordinates. Accurate camera calibration is essential for various applications, including 3D reconstruction and robotics. This work aims to implement camera calibration algorithms from scratch, as presented in A Flexible New Technique for Camera Calibration by Zhang, providing a detailed approach to obtaining intrinsic and extrinsic camera parameters.**

I. Camera Calibration Algorithm

Accurate estimation of camera parameters is crucial for geometric measurements in computer vision. However, these parameters change based on different factors and are often not provided by manufacturers. Various camera calibration methods exist, ranging from those using a known 3D setup (calibration rig) to techniques such as Zhang's formulation, which rely on multiple views of a structured 3D pattern with known positions.

A. Projection Matrix

The simple and well-known pinhole camera model is used to describe the projection of 3D world points $\mathbf{x} = \begin{bmatrix} x & y & z \end{bmatrix}^T$ onto the camera's sensor plane $\mathbf{s} = \begin{bmatrix} u & v \end{bmatrix}^T$. We assume that the image plane is positioned in front of the optical center, ensuring that the image is not inverted. The perspective transformation can be written as follows:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \hom^{-1} \left(\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \right) \tag{1}$$

this formulation can be expressed in a compact way as:

$$\mathbf{s} = \hom^{-1}(\mathbf{M}_p \hom(\mathbf{x})) \tag{2}$$

where $\hom(\cdot)$ converts Cartesian coordinates into homogeneous coordinates, and $\hom(\cdot)^{-1}$ facilitates the transformation from homogeneous coordinates back to Cartesian. The matrix $\mathbf{M}_p$ encapsulates the internal parameters of the pinhole camera, taking into account that f denotes the focal length. However, the matrix $\mathbf{M}_p$ can be decomposed in the following matrices:

$$\mathbf{M}_p = \mathbf{M}_f \mathbf{M}_0; \quad \mathbf{M}_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

where the matrix $\mathbf{M}_0$ describes transformations between the camera coordinates $\mathbf{s}$ and the world coordinates $\mathbf{x}$.

[†] Luis F. Recalde is PhD student in Robotics Engineering at Worcester Polytechnic Institute lrecalde@wpi.edu

B. Projection and Rigid Body Motions

We can consider the camera as a rigid body with its own transformation; therefore, we should account for both its rotations and translations. The complete imaging transformation for the ideal pinhole camera can be formulated in the following representation:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \hom^{-1} \left(\begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \right) \tag{3}$$

The formulation presented before can be written in a compact form as follows:

$$\mathbf{s} = \hom^{-1}(\mathbf{M}_f \ \mathbf{T}(\mathbf{R}, \mathbf{t}) \ \hom(\mathbf{x})) \tag{4}$$

where $\mathbf{T}(\mathbf{R}, \mathbf{t})$ is a transformation matrix that depends on the translation and rotation.

C. Intrinsic Camera Parameters

This section is essential in order to render the perspective imaging transformation applicable to an actual camera, establishing the correspondence between the coordinates and the image pixels. For this purpose, we consider the following parameters: sensor scales s_x and s_y, which respectively scale the variables in x and y. Additionally, the position of the camera center $s_c = (u_c, v_c)$ and the diagonal distortion s_θ of the image plane must be accounted for.

The complete imaging transformation can be defined as follows:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \hom^{-1} \left(\begin{bmatrix} f s_x & f s_\theta & u_c \\ 0 & f s_y & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \right) \tag{5}$$

The formulation presented before can be written in a compact form as follows:

$$\mathbf{s} = \hom^{-1}(\mathbf{K} \ \mathbf{T}(\mathbf{R}, \mathbf{t}) \ \hom(\mathbf{x})) \tag{6}$$

where:

$$\mathbf{K} = \begin{bmatrix} f s_x & f s_\theta & u_c \\ 0 & f s_y & v_c \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \alpha & \gamma & u_c \\ 0 & \beta & v_c \\ 0 & 0 & 1 \end{bmatrix}$$

captures the intrinsic parameters of the camera.

D. Plane-based Calibration

The widely used Zhang camera calibration method utilizes at least two views of a precisely known planar calibration pattern, often referred to as a model with a well-defined layout and dimensions. The calibration model comprises N points as $\{\mathbf{x}_1, ..., \mathbf{x}_N\}$, whose coordinates are known due to the employment of a checkerboard with specific dimensions. Furthermore, these points exist within the plane $x-y$ without of the z component. Additionally, we consider M different images, where features points are extracted as a sensor points:

$$\mathbf{s}_{ij} \in \mathbb{R}^2 \qquad i = \{1, ..., M\} \ \{j = 1, ..., N\}$$

It is essential to ensure that each observed point $\mathbf{s}_{ij}$ corresponds accurately to a specific model point $\mathbf{x}_j$.

1) Estimation of Homography: Considering the formulation presented in (6), we can map the observed point in the model $\mathbf{x}_j$ to the corresponding point image point $\mathbf{s}_{ij}$ as follows:

$$\lambda \begin{bmatrix} u_{i,j} \\ v_{i,j} \\ 1 \end{bmatrix} = \mathbf{KT}(\mathbf{R}_i, \mathbf{t}_i) \begin{bmatrix} x_j \\ y_j \\ z_j \\ 1 \end{bmatrix} \tag{7}$$

where λ is a arbitrary non-zero scale factor. Considering that the model points are assumed to lie in a plane, we have the following representation.

$$\lambda \begin{bmatrix} u_{i,j} \\ v_{i,j} \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} | & | & | & | \\ \mathbf{r}_{i1} & \mathbf{r}_{i2} & \mathbf{r}_{i3} & \mathbf{t}_i \\ | & | & | & | \end{bmatrix} \begin{bmatrix} x_j \\ y_j \\ 0 \\ 1 \end{bmatrix} \tag{8}$$

The formulation presented before can also be written as follows:

$$\lambda \begin{bmatrix} u_{i,j} \\ v_{i,j} \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} | & | & | \\ \mathbf{r}_{i1} & \mathbf{r}_{i2} & \mathbf{t}_i \\ | & | & | \end{bmatrix} \begin{bmatrix} x_j \\ y_j \\ 1 \end{bmatrix} \tag{9}$$

We can notice that the mapping can be written considering an homography matrix as follows:

$$\lambda \begin{bmatrix} u_{i,j} \\ v_{i,j} \\ 1 \end{bmatrix} = \mathbf{H}_i \begin{bmatrix} x_j \\ y_j \\ 1 \end{bmatrix} \tag{10}$$

where

$$\mathbf{H}_i = \begin{bmatrix} | & | & | \\ \mathbf{h}_{i1} & \mathbf{h}_{i2} & \mathbf{h}_{i3} \\ | & | & | \end{bmatrix} = \lambda \mathbf{K} \begin{bmatrix} | & | & | \\ \mathbf{r}_{i1} & \mathbf{r}_{i2} & \mathbf{t}_i \\ | & | & | \end{bmatrix}$$

This work aims to determine the homography matrix $\mathbf{H}_i$ for a set of corresponding points $\mathbf{s}_{ij}$ and $\mathbf{x}_j$.

This work implements Direct Linear Transformation (DLT) to estimate the homography. We consider that the model points $\mathcal{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ with $\mathbf{x}_j = \begin{bmatrix} x_j & y_j \end{bmatrix}^T$ and the associated sensor points $\mathcal{S} = \{\mathbf{s}_1, ..., \mathbf{s}_N\}$ with $\mathbf{s}_j = \begin{bmatrix} u_j & v_j \end{bmatrix}^T$ are related by a homography matrix, which is written in homogeneous coorddinates as follows:

$$\begin{bmatrix} u_{i,j} \\ v_{i,j} \\ w_{i,j} \end{bmatrix} = \mathbf{H}_i \begin{bmatrix} x_j \\ y_j \\ 1 \end{bmatrix} \tag{11}$$

The formulation presented before can be also written as:

$$\begin{bmatrix} -\mathbf{x}_j^T & \mathbf{0}_{1\times3} & u_{i,j}\mathbf{x}_j^T \\ \mathbf{0}_{1\times3} & -\mathbf{x}_j^T & v_{i,j}\mathbf{x}_j^T \\ | & | & | \\ -\mathbf{x}_N^T & \mathbf{0}_{1\times3} & u_{i,N}\mathbf{x}_N^T \\ \mathbf{0}_{1\times3} & -\mathbf{x}_N^T & v_{i,N}\mathbf{x}_N^T \end{bmatrix} \mathbf{h}_i = \begin{bmatrix} 0 \\ 0 \\ | \\ 0 \\ 0 \end{bmatrix} \tag{12}$$

where $\mathbf{h}$ is a vector with the elements of the homography matrix, which can be written in a compact form as follows:

$$\mathbf{M}_i \mathbf{h}_i = \mathbf{0} \tag{13}$$

where $\mathbf{M}_i \in \mathbb{R}^{2N\times9}$, $\mathbf{h}_i \in \mathbb{R}^9$ and $\mathbf{0} \in \mathbb{R}^{2N}$. We can estimate each one the of homography matrices $\mathbf{h}_i$ by *Singular Value Decomposition*. This work also incorporates the normalization of data points, a technique that has been shown to enhance numerical stability.

2) Refining Homography by Non-linear Optimization: The homography estimation obtained through the DLT method does not effectively minimize the error within the sensor image. Consequently, this work proposes a non-linear optimization problem designed to enhance the results of the the DLT approach. The optimization aims to minimize the following cost function:

$$\underset{\mathbf{h}_1 \dots \mathbf{h}_M}{\arg\min} \quad \sum_{k=1}^{M} \|\mathbf{s}_k - \mathbf{H}_k \mathbf{x}\|_F^2 \tag{14}$$

This optimization problem was set up and solved using CasADi [1] and the interior point method algorithm IPOPT.

3) Computing Intrinsic Parameters: Until this point, we have computed the homography for M views $\mathcal{H} = \{\mathbf{H}_1, ..., \mathbf{H}_M\}$. However, the homography matrix encodes the intrinsic and extrinsic parameters associated with that view. We can formulate the following definition:

$$\mathbf{H} = \begin{bmatrix} | & | & | \\ \mathbf{h}_1 & \mathbf{h}_2 & \mathbf{h}_3 \\ | & | & | \end{bmatrix} = \lambda \mathbf{K} \begin{bmatrix} | & | & | \\ \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \\ | & | & | \end{bmatrix} \tag{15}$$

Given that $\mathbf{r_1}$ and $\mathbf{r}_2$ must be orthogonal to accurately represent a valid rotational matrix, the following conditions are established:

$$\mathbf{r}_1^T \mathbf{r}_2 = \mathbf{r}_2^T \mathbf{r}_1 = 0$$

$$\mathbf{r}_1^T \mathbf{r}_1 = \mathbf{r}_2^T \mathbf{r}_2 = 1$$

Considering (15), we can have the following definitions:

$$\mathbf{h}_1 = \lambda \mathbf{K} \mathbf{r}_1$$

$$\mathbf{h}_2 = \lambda \mathbf{K} \mathbf{r}_2$$

We can combined the previous definitions to formulate the following:

$$\begin{aligned} \mathbf{h}_1^T \mathbf{B} \mathbf{h}_2 &= 0 \\ \mathbf{h}_1^T \mathbf{B} \mathbf{h}_1 - \mathbf{h}_2^T \mathbf{B} \mathbf{h}_2 &= 0 \end{aligned} \tag{16}$$

in which $\mathbf{B} = (\mathbf{K}^{-1})^T \mathbf{K}^{-1}$ is a symmetric matrix. Equation (16) can be expressed in a linear form with respect to the values within matrix $\mathbf{B}$, under the condition $\mathbf{h}_p^T \mathbf{B} \mathbf{h}_q = \boldsymbol{\nu}_{pq}(\mathbf{H})\mathbf{b}$, taking into account that $\mathbf{b} = \begin{bmatrix} b_1 & b_2 & b_3 & b_4 & b_5 & b_6 \end{bmatrix}$.

Therefore, for a homography matrix $\mathbf{H}$ the two equations in (16) can be written as follows:

$$\begin{bmatrix} \boldsymbol{\nu}_{1,2}(\mathbf{H}) \\ \boldsymbol{\nu}_{1,1}(\mathbf{H}) - \boldsymbol{\nu}_{2,2}(\mathbf{H}) \end{bmatrix} \mathbf{b} = \mathbf{0} \tag{17}$$

In order to approximate the values of $\mathbf{b}$, it is necessary to take into account the M views. Subsequently, the homography matrices can be arranged as follows:

$$\begin{bmatrix} \boldsymbol{\nu}_{1,2}(\mathbf{H}) \\ \boldsymbol{\nu}_{1,1}(\mathbf{H}) - \boldsymbol{\nu}_{2,2}(\mathbf{H}) \\ \vdots \\ \boldsymbol{\nu}_{1,2}(\mathbf{H}_M) \\ \boldsymbol{\nu}_{1,1}(\mathbf{H}_M) - \boldsymbol{\nu}_{2,2}(\mathbf{H}_M) \end{bmatrix} \mathbf{b} = \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \tag{18}$$

This formulation can also be written in a compact form as:

$$\mathbf{V}\mathbf{b} = \mathbf{0} \tag{19}$$

where $\mathbf{V} \in \mathbb{R}^{2N \times 6}$ and $\mathbf{b} \in \mathbb{R}^6$ is a vector of the elements of the matrix $\mathbf{B}$. The vector be can be easily estimated by *Singular Value Decomposition*. Once the vector $\mathbf{b}$ is estimated, the camera intrinsic parameters $\mathbf{K}$ can be calculated. Note that the relation $\mathbf{B} = \lambda(\mathbf{K}^{-1})^T \mathbf{K}^{-1}$ holds, and $\mathbf{K}$ can be estimated using *Cholesky Decomposition* since $\mathbf{B}$ is a symmetric matrix.

$$\mathbf{K} = \kappa(\mathbf{L}^T)^{-1}$$

where $\kappa = \mathbf{L}_{3,3}$ and $\mathbf{L} = \mathrm{Chol}(B)$.

4) Computing Extrinsic Parameters: The extrinsic parameters $(\mathbf{R}_i, \mathbf{t}_i)$ can be calculated using the intrinsic parameters $\mathbf{K}$ and the homography matrices along different views $\mathbf{H}_i$.

$$\begin{aligned} \mathbf{r}_{i,1} &= \lambda \mathbf{K}^{-1} \mathbf{h}_{i,1} \\ \mathbf{r}_{i,2} &= \lambda \mathbf{K}^{-1} \mathbf{h}_{i,2} \\ \mathbf{r}_{i,3} &= \mathbf{r}_{i,1} \times \mathbf{r}_{i,2} \\ \mathbf{t}_i &= \lambda \mathbf{K}^{-1} \mathbf{h}_{i,3} \end{aligned} \tag{20}$$

where $\lambda = 1/(\|\mathbf{K}^{-1}\mathbf{h}_{i,1}\|)$

The rotation matrix $\mathbf{R}_i = \begin{bmatrix} \mathbf{r}_{i,1} & \mathbf{r}_{i,2} & \mathbf{r}_{i,3} \end{bmatrix}$ can be defined without guaranteeing the constrains of orthogonality. To address this issue, a new rotation matrix is derived through the application of *Singular Value Decomposition* and the reinterpretation of a $\mathbf{R}_{i,new} = \mathbf{U}\mathbf{V}^T$.

E. Radial Distortion Model

We have not considered any distortion in the lens of the camera; therefore, we can increase the accuracy of the model by including a radial distortion. Radial distortion is generated by radial lines from the center of the image (u_c, v_c) and is only dependent on the radius of a point $\|\mathbf{x}_i\|$. Radial distortion can be modeled as follows:

$$\mathrm{warp}(\mathbf{x}_i, \mathbf{k}) = \mathbf{x}_i(1 + D(\|\mathbf{x}_i\|, \mathbf{k}))$$

where $D(\|\mathbf{x}_i\|, \mathbf{k}) = k_1\|\mathbf{x}_i\|^2 + k_2\|\mathbf{x}_i\|^4$. While a closed-form solution exists for estimating the vector $\mathbf{k}$, this study instead directly formulates a non-linear optimization problem for determining these parameters.

F. Refining all parameters by Non-linear Optimization

We have estimated the intrinsic and extrinsic parameters of the camera considering different views. The last step of the calibration is refining these values by a nonlinear optimization problem, which is written as follows:

$$\underset{\mathbf{K}, \mathbf{k}, \mathbf{t}_1, \mathbf{R}_1 \ldots \mathbf{t}_M, \mathbf{R}_M}{\arg\min} \quad \sum_{k=1}^{M} \|\mathbf{s}_k - \mathbf{K}\mathrm{warp}(\mathbf{T}(\mathbf{R}_k, \mathbf{t}_k)\mathbf{x}, \mathbf{k})\|_F^2 \tag{21}$$

The optimization problem can be complicated to solve since we are not dealing with Euclidean spaces for the rotation matrices. Therefore this work instead of directly optimize the matrices $\mathbf{R}_i$, we use quaternions $\mathbf{q} \in \mathbb{S}^3$ that are simpler

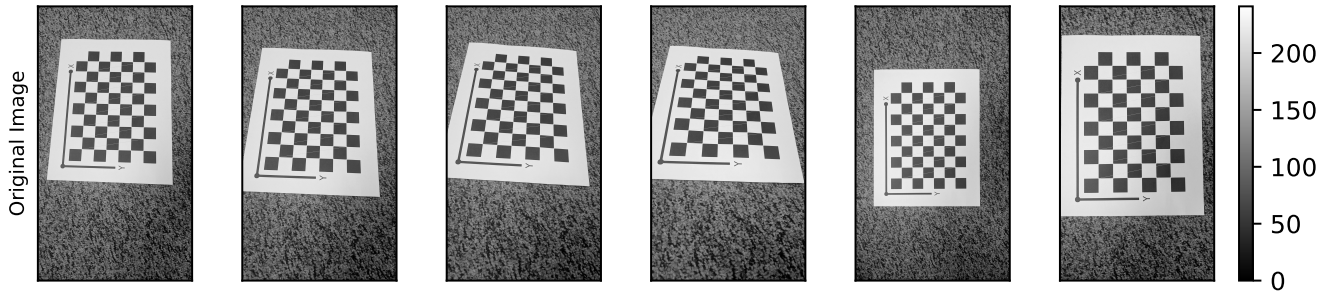

Fig. 1. Corners detection of each view of the cheeseboard

representations of rotations. However, optimizing the four elements of quaternions is still complicated and we need to guarantee one constraint defined as $||\mathbf{q}|| = 1$.

Therefore, this work uses the *Stereographic projection*, which is a standard way to parameterize quaternions $\mathbf{q} \in \mathbb{S}^3$ with an unconstrained vector $\mathbf{x}_q \in \mathbb{R}^3$, so that the optimization algorithm can live purely in an Euclidean space. Considering the *Stereographic projection*, we have the following forward and inverse maps.

For any quaternion $\mathbf{q} = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix} \in \mathbb{S}^3$ with $q_0 \neq -1$, the *forward* mapping is defined by

$$\text{forward} : \{\mathbf{q} \in \mathbb{S}^3 : q_0 \neq -1\} \to \mathbb{R}^3,$$

$$\mathbf{x}_q = \text{forward}(\mathbf{q}) = \frac{\begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}}{1 + q_0}$$

This mapping is the stereographic projection of $\mathbf{q}$ onto $\mathbb{R}^3$.

For any vector $\mathbf{x}_q \in \mathbb{R}^3$, the *inverse* mapping is defined by:

$$\text{inverse} : \mathbb{R}^3 \to \mathbb{S}^3 : q_0 \neq -1,$$

$$\mathbf{q} = \text{inverse}(\mathbf{x}_q) = \begin{bmatrix} \frac{1-||\mathbf{x}_q||}{1+||\mathbf{x}_q||} & \frac{2\mathbf{x}_q}{1+||\mathbf{x}_q||} \end{bmatrix}$$

Based on *Stereographic projection*, we are able to formulate a new optimization problem within the Euclidean space.

$$\underset{\mathbf{K}, \mathbf{k}, \mathbf{t}_1, \mathbf{x}_{q1} \ldots \mathbf{t}_M, \mathbf{x}_{qM}}{\arg\min} \sum_{k=1}^{M} \|\mathbf{s}_k - \mathbf{K}\text{warp}(\mathbf{T}(\mathbf{x}_{qk}, \mathbf{t}_k)\mathbf{x}, \mathbf{k})\|_F^2 \tag{22}$$

This optimization problem was set up and solved using CasADi and the IPOPT. Additionally, the cost function was structured to ensure that its Hessian remains sparse. Formulating the problem in this manner can significantly reduce the time required to solve it. The sparsity of the matrix is shown in Fig. 2.

II. Estimation of the Intrinsic and Extrinsic Parameters

This section presents the results of the previously described algorithm used to estimate the intrinsic and extrinsic parameters of a camera. In particular, this formulation was tested on 13 images of a chessboard taken from different viewpoints.

The model points were defined under the assumption that the chessboard consists of a 7×10 grid of squares. Consequently, only the internal corners were considered, resulting in a grid 6×9 with each corner equally spaced by $0.0215[m]$.

To detect these corners in the image plane, the OpenCV function 'cv2.findChessboardCorners' was used. The points detected across all images are shown in Fig. 1.

The total projection error, computed using the cost function presented in Eq. (21), is shown in Table I. These values demonstrate the advantages of using non-linear optimization to improve the accuracy of the estimated parameters, resulting in noticeably lower errors across the images. Finally, the time required to optimize the intrinsic and extrinsic parameters for all images was $0.02[s]$.

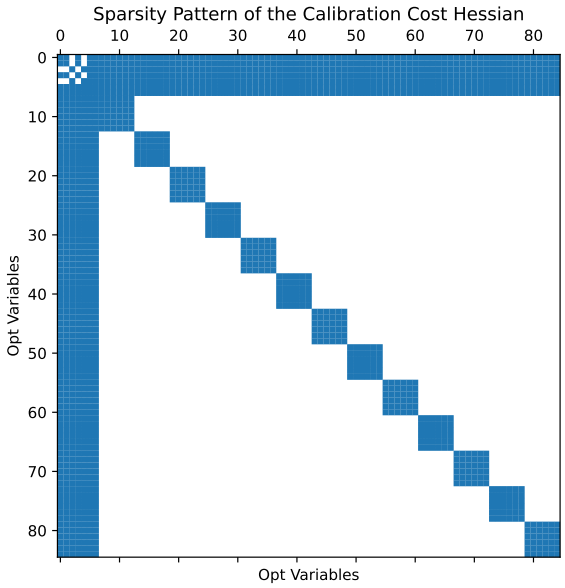

Fig. 2. Hessian of the cost function showing its sparsity

The intrinsic parameters are presented as follows: For the linear estimation:

$$\mathbf{K} = \begin{pmatrix} 2.053 \times 10^3 & -4.67 \times 10^{-1} & 7.62 \times 10^2 \\ 2.83 \times 10^{-14} & 2.036 \times 10^3 & 1.352 \times 10^3 \\ -6.40 \times 10^{-17} & 3.54 \times 10^{-17} & 1.00 \end{pmatrix}$$

For the nonlinear optimization:

$$\mathbf{K}_{opt} = \begin{pmatrix} 2.041 \times 10^3 & -4.67 \times 10^{-1} & 7.61 \times 10^2 \\ 0.00 & 2.033 \times 10^3 & 1.346 \times 10^3 \\ 0.00 & 0.00 & 1.00 \end{pmatrix}$$

Here, $\mathbf{K}$ and $\mathbf{K}_{opt}$ represent the values obtained using linear estimation and the nonlinear optimization problem, respectively.

TABLE I
Projection Error Image Plane

	Linear Estimation	Non-linear Estimation
Projection Error	16.09	5.62

On the other hand, the intrinsic parameters related to radial distortion during the linear and nonlinear estimation are presented as follows:

$$\mathbf{k} = \begin{bmatrix} 1.12 \times 10^{-8} & -1.83 \times 10^{-14} \end{bmatrix}$$

$$\mathbf{k}_{opt} = \begin{bmatrix} 0.1705 & -0.723 \end{bmatrix}$$

The estimation of the extrinsic parameters is presented in Fig. 3, which shows the accuracy of the estimation and demonstrates that we were able to obtain an approximation of the chessboard relative to the camera.

The final results associated to the rectification of the image and the detection of the corners are presented in Fig. 4.

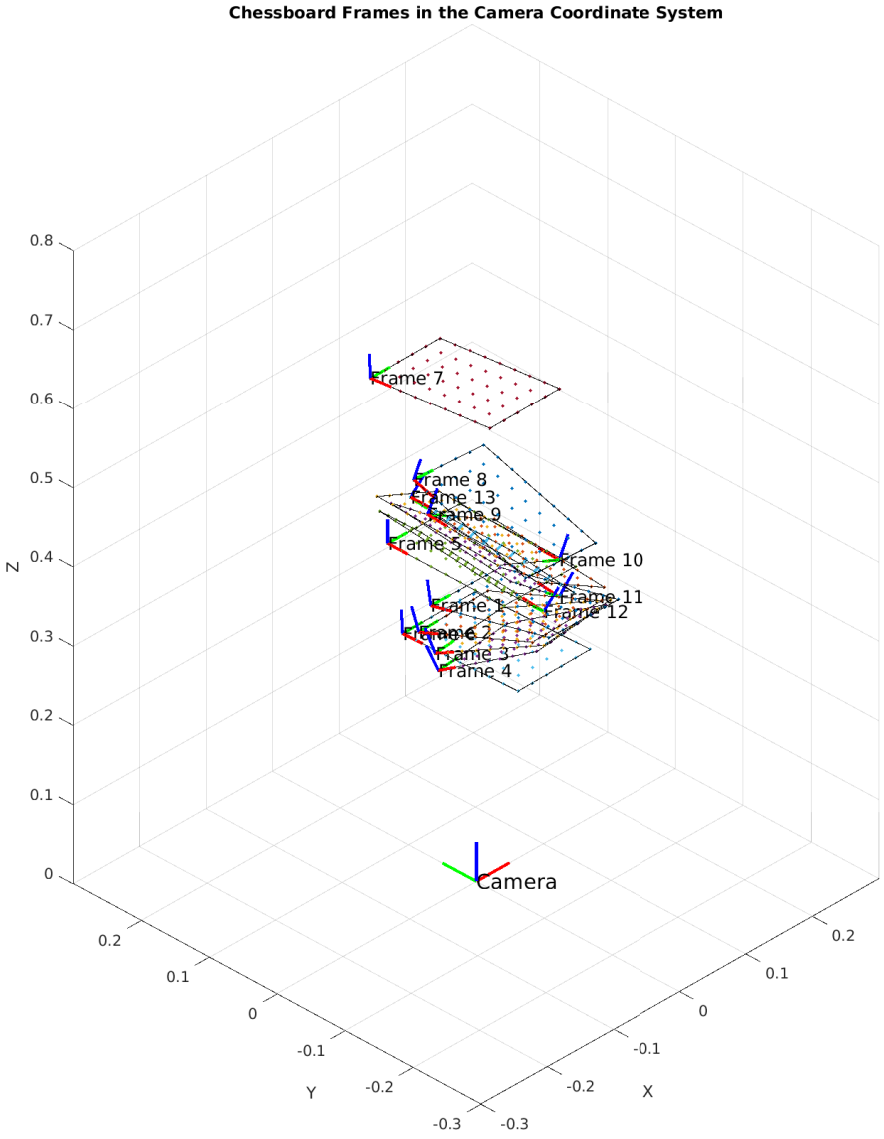

Fig. 3. Extrinsic parameters approximation

REFERENCES

[1] J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "Casadi—a software framework for nonlinear optimization and optimal control," *Math. Programm. Comput.*, vol. 11, no. 1, pp. 1–36, 3 2019.

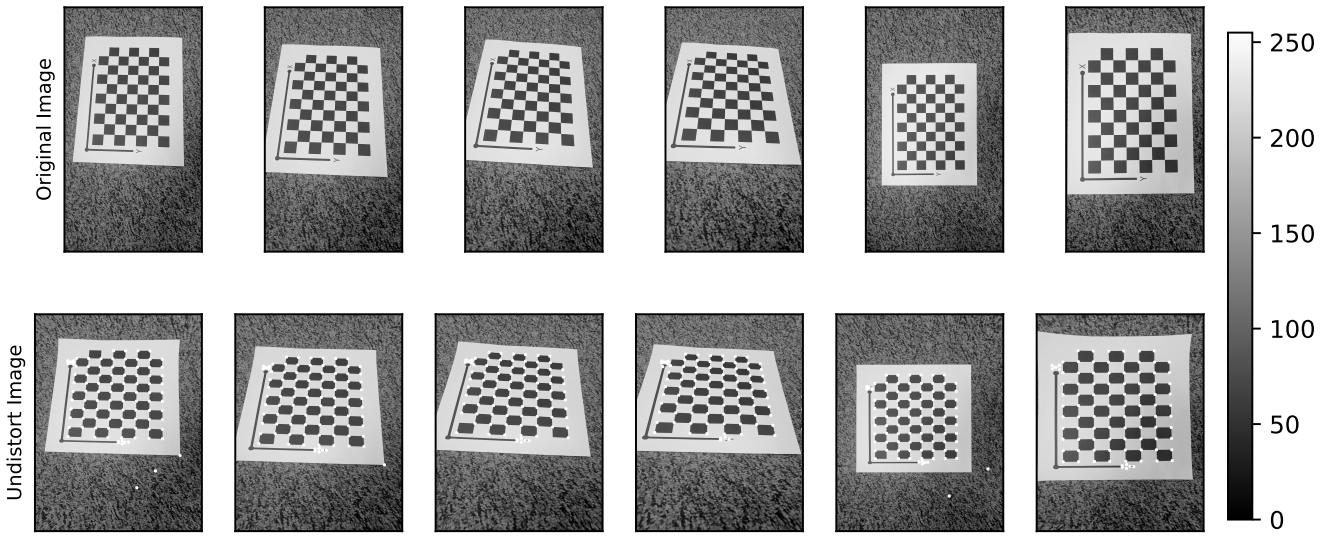

Fig. 4. Rectification and re-projection of the corners