


MetroVisionLab Toolbox for Camera Calibration and Simulation

METROVISIONLAB package for camera calibration has been developed, designed and implemented by:

David Samper, Jorge Santolaria, Jorge Juan Pastor, Juan José Aguilar in cooperation with other members of [Manufacturing Engineering and Advanced Metrology Group \(GIFMA\)](#) of the [Aragón Institute of Engineering Research \(I3A\)](#).

The current version 1.5 (01/2009) includes TSAI 2D, TSAI 3D, DLT 2D (without distortion), DLT 3D (without distortion), DLT 3D (with distortion), FAUGERAS, ZHANG (without distortion) and ZHANG (with distortion) algorithms for camera calibration.

The **METROVISIONLAB** package runs on MATLAB 7.0 or higher. MATLAB® is a registered trademark of The MathWorks, Inc.

METROVISIONLAB for Camera Calibration and Simulation	
Help online	 Download in pdf help for METROVISIONLAB v. 1
Algorithms review	
Download package and source code	

DISCLAIMER:

This software and the documentations are the property of the authors and **should only be used for scientific and educational purposes**. All software is provided free. Please, send your comments or suggestions to gifma@unizar.es

METROVISIONLAB citations

Please use the following citation format:

- D. Samper, J. Santolaria, J.J. Pastor, J.J. Aguilar et al., METROVISIONLAB Toolbox for Camera Calibration and Simulation, <http://metrovisionlab.unizar.es>

>> GENERAL DESCRIPTION OF METROVISIONLAB

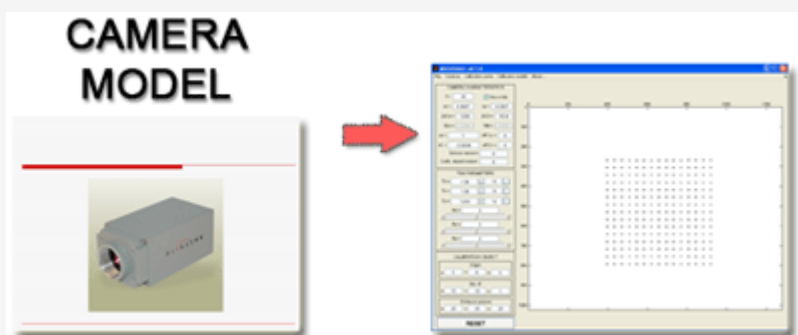
METROVISIONLAB is a toolbox developed for testing, from real or synthetic data, the behaviour of the main camera calibration techniques for use in machine or computer vision and dimensional metrology.

METROVISIONLAB can generate 2D and 3D synthetic calibration points on the basis of a planar gauge object with configurable dimensions. From the standpoint of simulation, it allows represent those points according to the calibration technique and active parameters in the toolbox. From points in X, Y, Z global coordinates (real or generated) that have been loaded from file, it obtains and represents corresponding U, V screen coordinates according to the defined TSAI parameters and characteristics specified for the camera model to simulate. It is possible to introduce Gaussian noise with configurable amplitude both in the global coordinates of the points and in the sensor of the camera. Both the noise, characteristics of the camera and calibration parameters are configurable, updating U, V screen coordinates for each new set of values configured.

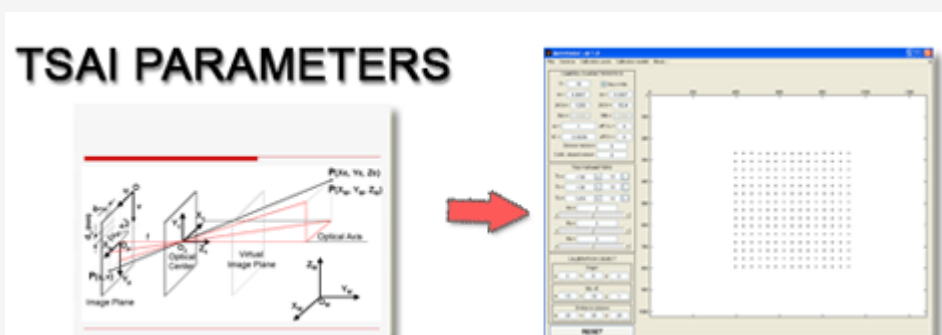
From points in X, Y, Z global coordinates and their corresponding U, V screen coordinates, all of them loaded from ASCII file, **METROVISIONLAB** performs the calibration of the specified camera according to the algorithm selected for calibration, offering the calibration results. These results, in the case of TSAI algorithms, are loaded onto the interface after a calibration, simulating the behaviour of the camera through the representation of the calibration points in screen coordinates. Thus it is possible to compare the accuracy of points reconstruction through various calibration techniques depending on several parameters of influence or error mechanisms. It is also possible to obtain TSAI parameters from calibration results of other algorithms, allowing to compare the results of calibration algorithms for the same calibration conditions.

METROVISIONLAB can manage files for upload and store global and screen coordinates of calibration or reconstructed points, as well as calibration parameters.

The main tasks that you can perform with **METROVISIONLAB** are shown in Fig. 1.



Camera model selection. Predefined camera models and intrinsic parameters configuration.



Extrinsic parameters. Manual or loaded from file extrinsic parameters.

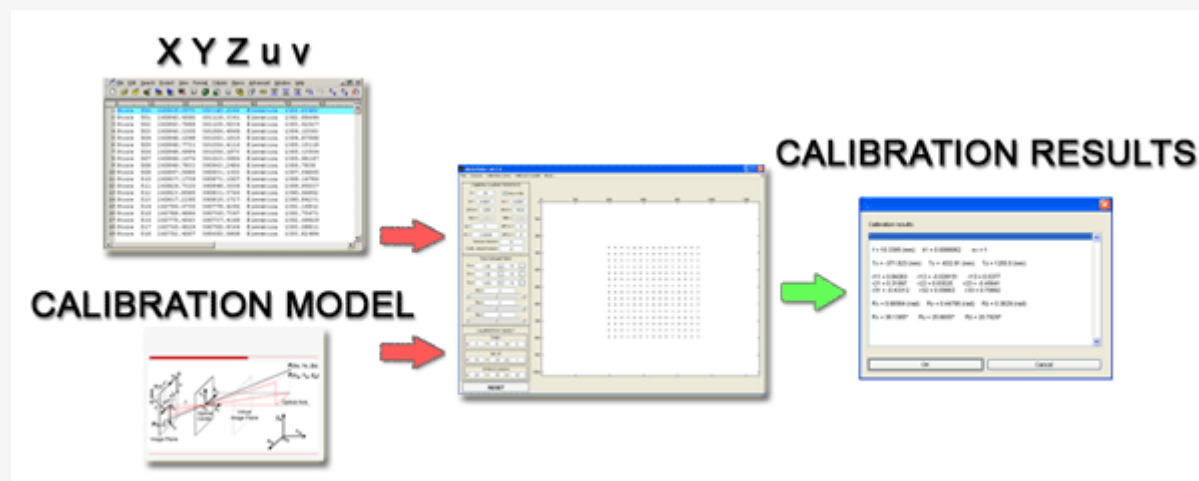
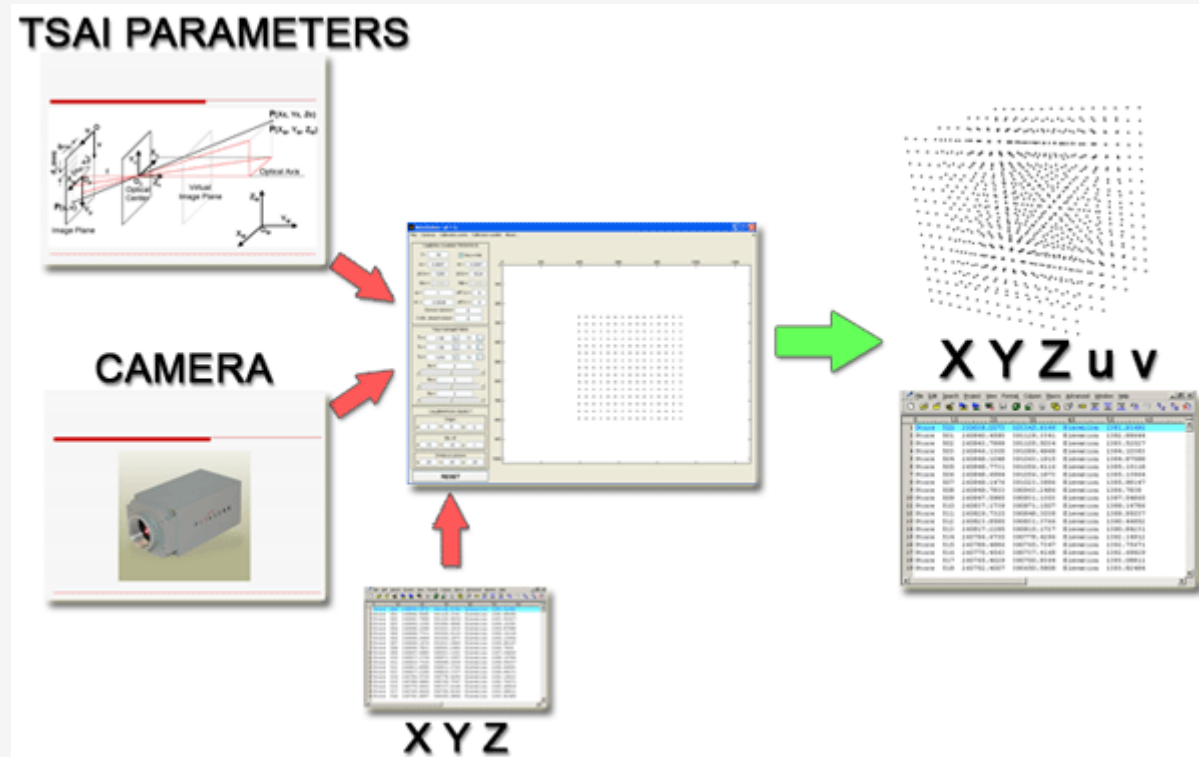
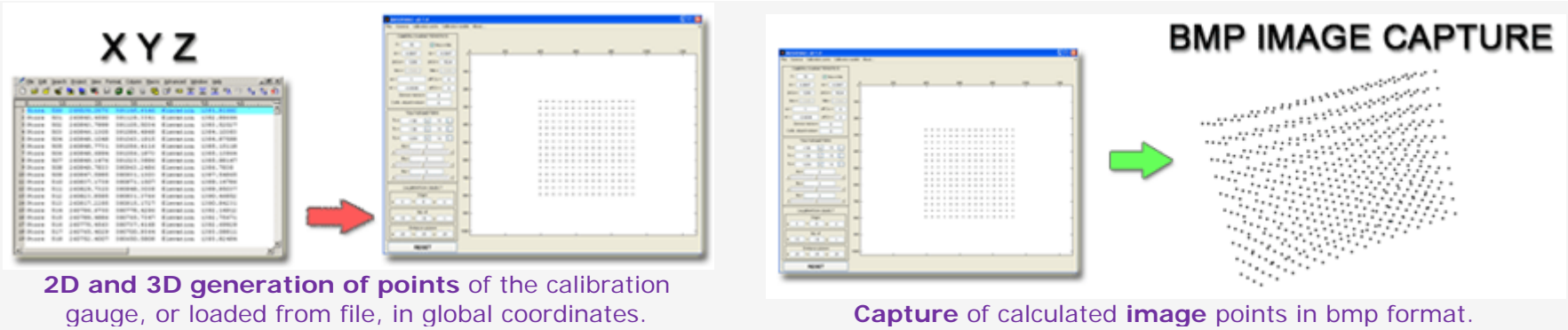


Figure 1. General description of METROVISIONLAB

METROVISIONLAB can be useful in the following tasks:

- 1.- Comparison of the performance of various algorithms for camera calibration for the same data set pairs of global and screen point coordinates of the calibration object.
- 2.- Automatic generation of pairs of global and screen point coordinates of the calibration object. Analysis of best calibration positions.
- 3.- Variation of calibration parameters and its influence over camera image capture.
- 4.- Analysis of error mechanisms and parameter influence over camera calibration models. Monte Carlo analysis.
- 5.- Camera calibration from real and simulated data sets.
- 6.- Simulation of Gaussian noise in screen coordinates and in calibration gauge object.
- 7.- Accuracy analysis of calibration models for its use in dimensional metrology.
- 8.- Experimental points data sets verification and validation.
- 9.- Training exercises and computer vision teaching tasks for engineering students.

>> **USER GUIDE**

>> **Starting METROVISIONLAB**

To start **METROVISIONLAB** for Camera Calibration type in MATLAB command window:

Metrovisionlab

>> Main Interface

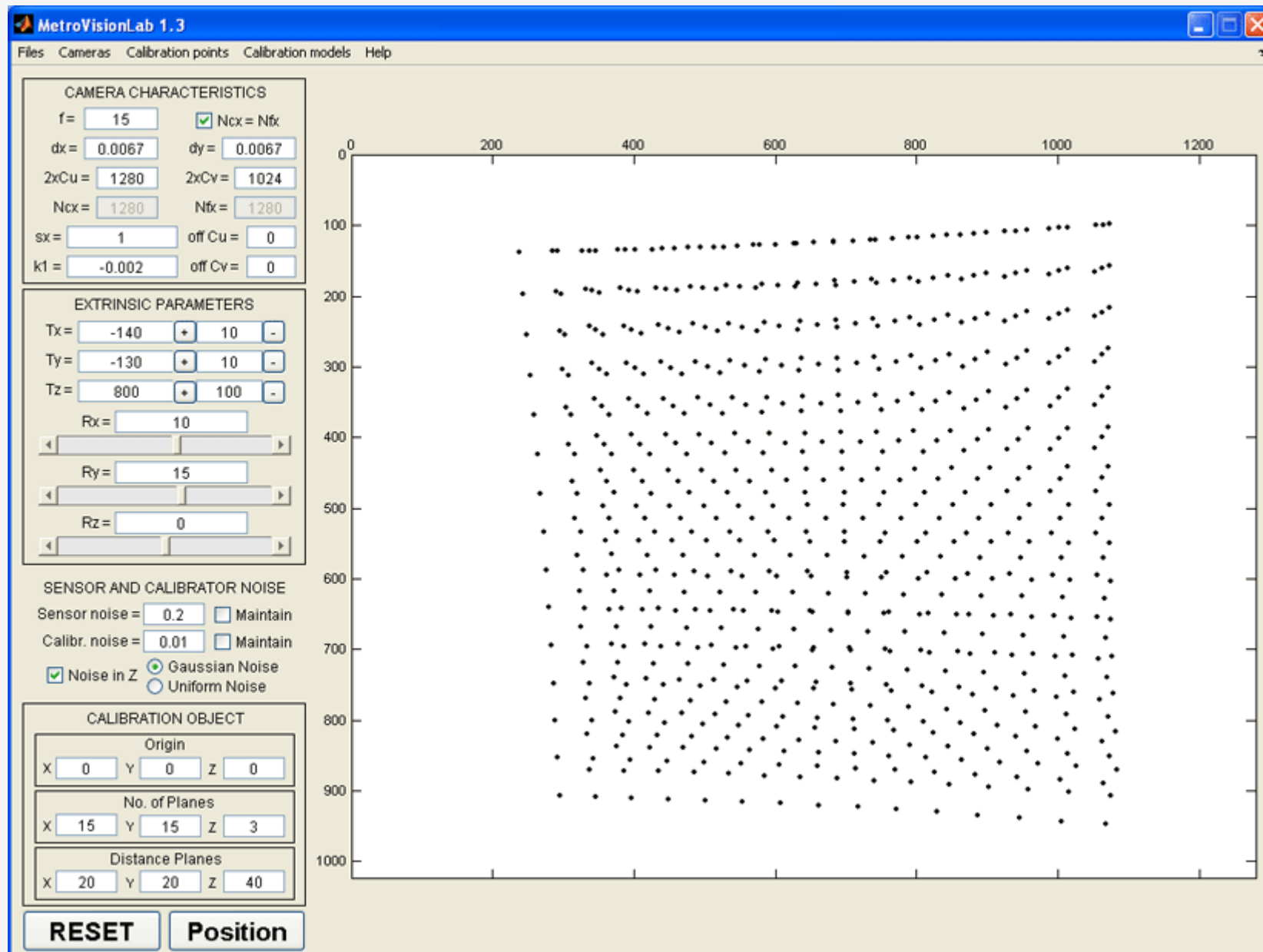


Figure 2. Main interface of METROVISIONLAB

>> Configuration tools

Over the left of the main screen (Fig. 2) are the configuration tools. When you open the toolbox, the tools are initialized with predefined values. Any change in those parameters will update the representations of the points in the view window.

Those tools allow you to:

- Define camera characteristic parameters (**CAMERA CHARACTERISTICS**). In this frame it is possible to modify camera intrinsic parameters and Tsai's model intrinsic camera constants:
 - **f**: Effective focal length of the pin-hole camera (mm).
 - **dx, dy**: Center to center distance between adjacent sensor elements in X and Y (mm).
 - **2xCu, 2xCv**: Twice the coordinates u, v of principal point or twice the coordinates u, v of center of radial lens distortion (pixel).
 - **Ncx**: number of sensor elements in x direction of the camera (sels).
 - **Nfx**: number of pixels in x direction of the frame grabber (pixel). (Checkbox **Ncx=Nfx** is predefined as active, since it must be the starting point in modelling digital cameras. If you are modelling an analog camera that uses framegrabber, uncheck this to introduce appropriate values for Ncx and Nfx).
 - **sx**: scale factor for uncertainty in the resampling of the horizontal scanline in framegrabber.
 - **k1**: 1st order radial lens distortion coefficient.
 - **Off Cu, Off Cv**: Offsets in coordinates u, v for the coordinates of principal point (pixel).
 - **Sensor noise, Calibration object noise**: Standard deviation to generate random noise according to a normal distribution with mean 0 in sensor u, v coordinates and global X, Y, Z coordinates.
- Modify extrinsic parameters (**EXTRINSIC PARAMETERS**):
 - **Rx, Ry, Rz**: Rotation angles for the transform between the global coordinates and the camera coordinates (degrees).
 - **Tx, Ty, Tz**: Translation terms for the transform between the global and camera reference systems. (The global reference system is established according to X, Y, Z values loaded from file or according to the coordinates of the upper left corner point of the first plane of generated, configured in Origin)
- Define noise parameters for the camera sensor and the calibration gauge object (**SENSOR AND CALIBRATOR NOISE**). In this frame it is possible to assign **random gaussian or uniform noise** to the coordinates of the points in the sensor or gauge object. **'Maintain'** checkboxes specify whether the noise generated the first time will be maintained or not. If this checkbox is unmarked, each time that the camera is moved, a new random noise will be added to the coordinates. **'Noise in Z coord.'** checkbox specifies whether the random noise will be generated also in Z direction or not. This is useful to avoid noise generation in Z coordinate in 2D calibration tests.

- Define the parameters for calibration gauge points generation (**CALIBRATION OBJECT**):
 - **Origin**: Define the global X, Y, Z coordinates of the upper left corner point of the first plane of points. Here you can define the position and orientation of the global reference system for your tests.
 - **Number of planes**: Define the number of planes in X, Y and Z direction that will be used in the automatic generation of calibration gauge object.
 - **Distance planes**: Define the distance between the point's generation planes in X, Y and Z directions.
- Check the relative position between the calibration points and the camera (**POSITION BUTTON**). This button opens a new window where camera frame, calibration gauge object frame, and calibration points are represented according to the options specified in the configuration tools.

>> View window

In the right zone of the main window (Fig. 2) is showed a view of the calibration gauge object point. Those points are generated or loaded from file and represented according to the values set in configuration parameters. Any change in those parameters will update the view window, so that this view always show the points from the point of view of the camera reference system, with the characteristics of the camera and Tsai parameters defined (Fig. 3).

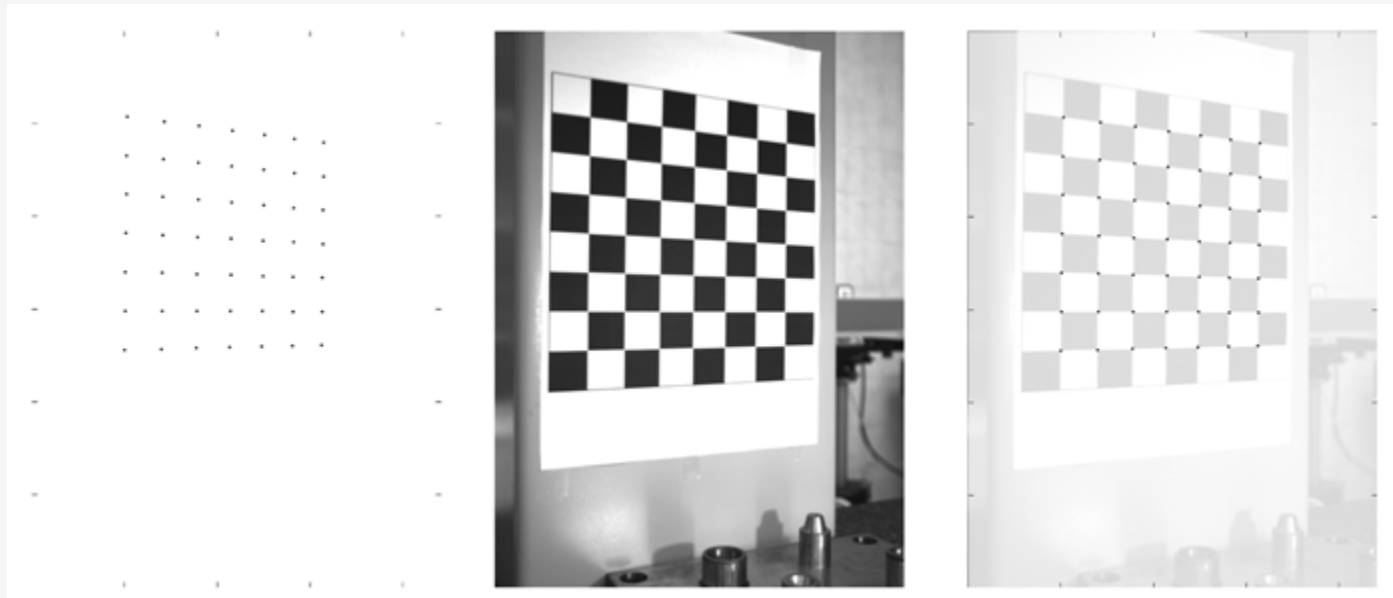


Figure 3. Comparison between real and generated calibration points.

>> Menu structure and commands

>> Files menu

-New test-

Establishes default values for camera characteristics, Tsai parameters and calibration object. This command is also executed when you press **RESET** button at the bottom of the main windows.

-Open test-

Opens a MetroVision Lab file (*.gen) and loads all configuration parameters to the interface.

-Save test-

Saves all configuration parameters of the current test in a MetroVision Lab file (*.gen) to be recovered executing the **Open test** command.

-Load Tsai parameters-

Loads a Tsai calibration file (*.cal). This file contains Tsai parameters that are loaded in the interface. Points in the view window are updated according to the loaded Tsai parameters.

-Save Tsai parameters-

Saves current Tsai parameters of the interface to a Tsai calibration file (*.cal).

-Save synthetic points-

Saves X, Y, Z global coordinates and corresponding u, v screen coordinates for the points obtained with current configuration parameters and showed in the view window. This command generates a Calibration points (*.pto) file.

-Capture image-

Captures and saves a image (*.bmp) of the view window to a file.

>> Cameras menu

This menu allows selecting the three sample cameras implemented in the toolbox: PixelINK PL-741 (1280x1024), PixelINK PL-761 (752x480) and PixelINK PL-781 (2208x3000). You can easily implement new camera models modifying the provided source code. When you check a camera, their intrinsic camera constants are loaded in the **CAMERA CHARACTERISTICS** frame. You must manually specify the values for the camera constants to make tests with other camera models different from the implemented camera samples.

>> Calibration points menu

This menu has the commands that control the mode of the generation of points:

-Load points (X, Y, Z)-

When checked, this options loads the calibration points from a Calibration points (*.pto) ascii file and updates the view window according to configuration parameters. Each row of that file represents the global coordinates of the point in the format:

```
X Y Z
X Y Z
...
X Y Z
```

If you use this generation mode in a file with more than three columns (e.g. X Y Z u v), only the first three columns are loaded according to the specified format.

-Load points (X, Y, Z, u, v)-

When checked, this options loads the calibration points from a Calibration points (*.pto) ascii file and updates the view window according to configuration parameters. Each row of that file represents the global and screen coordinates of the point in the format:

```
X Y Z u v
X Y Z u v
...
X Y Z u v
```

This mode allows to load global and screen coordinates pairs of calibration points to test the calibration algorithms.

-Chessboard (only coplanar)-

When checked, a chessboard is used as gauge calibration object, providing the calibration points according with the values specifiend in the **CALIBRATION OBJECT** frame. When this generation technique is used, some options in this frame are disabled, since it is only allow to use the chessboard as gauge object with coplanar calibration points.

- Generate calibration points-

It is the default generation mode for the calibration points. When checked, the calibration object points showed in the view window are controlled by the parameters specified in the **CALIBRATION OBJECT** frame.

>> Calibration models menu

This menu allows executing camera calibration according to any of the listed calibration algorithms. You can make a calibration with synthetically generated or loaded from file points. From currently loaded global and corresponding screen coordinates of the calibration object those commands perform the specified calibration. Available algorithms in version 1 are Tsai 2D, Tsai 3D, DLT 2D (without distortion), DLT 3D (without distortion), DLT 3D (with distortion), FAUGUERAS (3D without distortion), ZHANG (without distortion) and ZHANG (with distortion). You can review the general characteristics of the implemented algorithms and references [here](#).

The results of intrinsic and extrinsic parameters and transformation matrix terms obtained are shown after each calibration (Fig 4.). Also, characteristic parameters depending on the calibration method chosen are shown. You can **save the calibration results** to a file for later analysis.

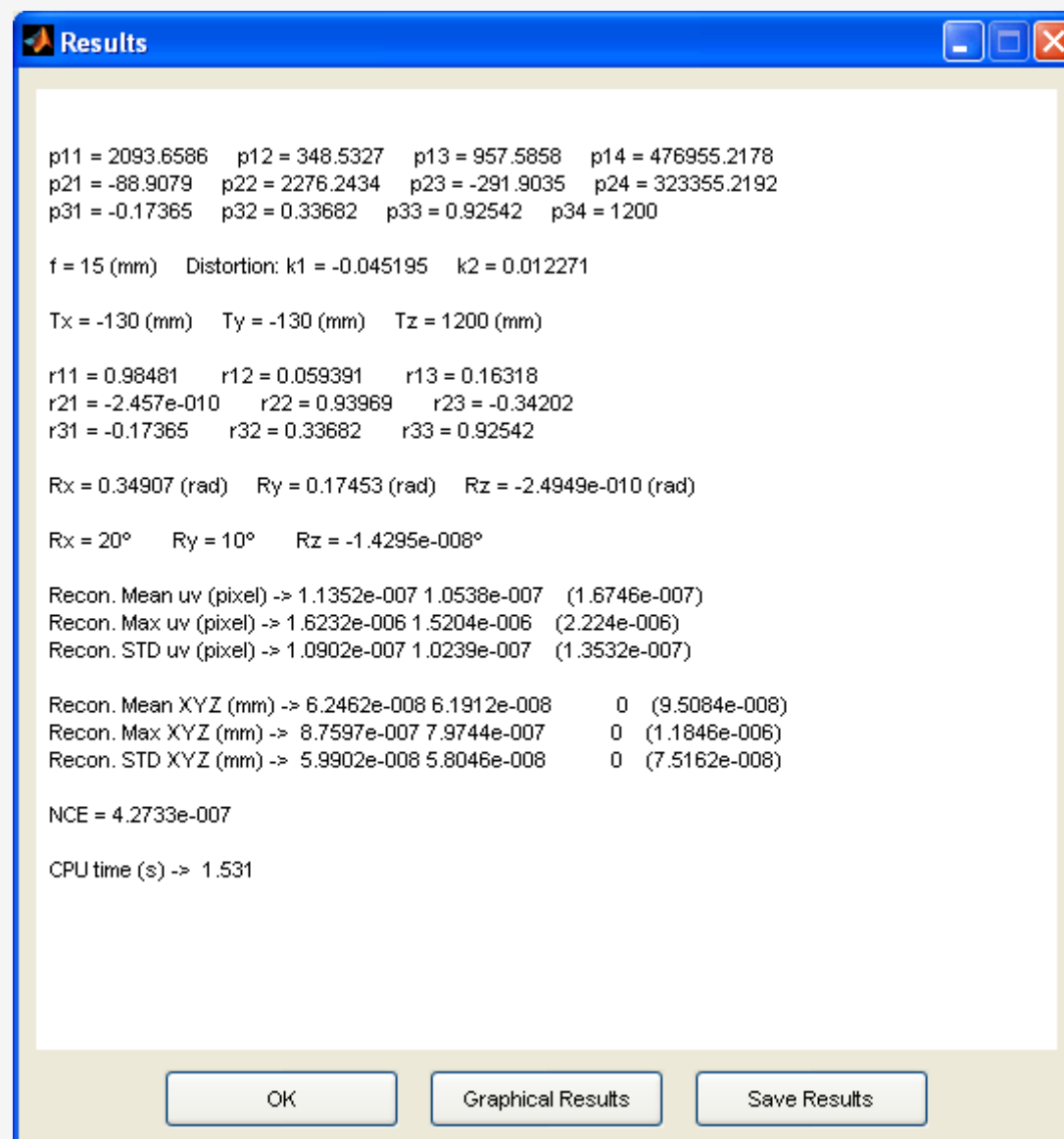


Figure 4. Calibration results sample (FAUGERAS with distortion).

In the Results window, it is possible to represent **graphical results** of the calibration (Fig. 5). This option shows graphs for the reconstruction errors of the calibration points, both in image coordinates and world coordinates. This tool is suitable to obtain characteristic parameters of the calibration carried out, and to have an overall idea of the later measurement error with the current calibration. The button "**Save Reconstruction Errors**" saves to file the reconstruction error for the calibration points used in each coordinate.

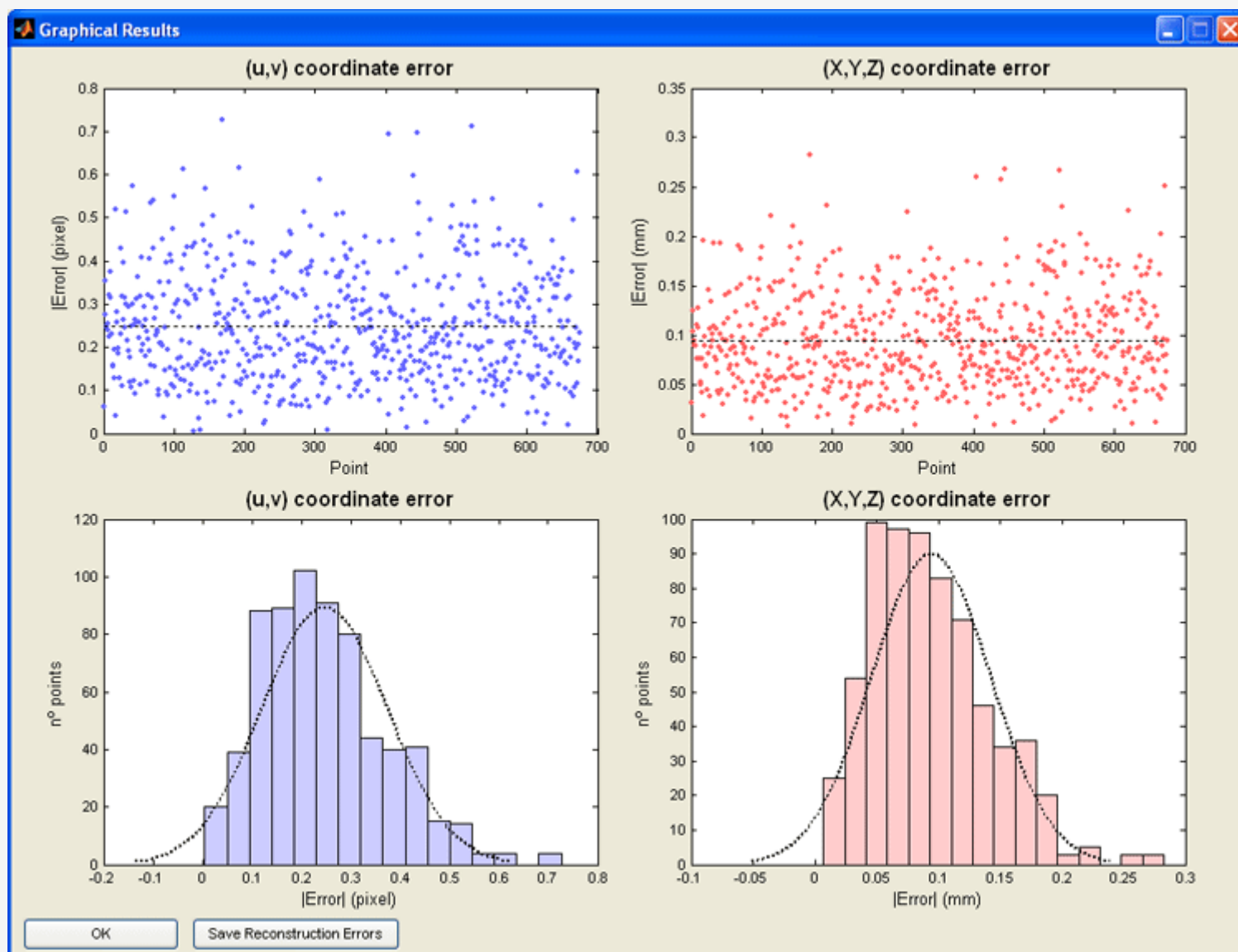


Figure 5. Reconstruction errors for the calibration points in image and world coordinates.

Those commands allow to study the influence of error sources over calibration model chosen and the behaviour of the different calibration models.

>> **Help menu**

-About...-

Shows general information about the Toolbox.

>> CALIBRATION ALGORITHMS

- | | |
|---|-------------------------------|
| >> DLT Model (Direct Linear Transformation) | >> DLT3D |
| | >> DLT 2D |
| >> TSAI Model | >> TSAI 2D |
| | >> TSAI 3D |
| >> FAUGERAS Model | >> FAUGUERAS |
| | >> CORRECTION |
| >> ZHANG Model | |
| >> ANNEXES | >> NCE |

>> DLT Model (Direct Linear Transformation)

It is based on the colineality between a point expressed in world frame (x, y, z) , its equivalent in image frame coordinates (u, v) and the central projection point of the camera.

>> DLT 3D

Eleven coefficients L_1, \dots, L_{11} are needed to establish the relationship between the points in the world reference system and their equivalents in the image reference system, according to DLT method.

$$u = \frac{L_1x + L_2y + L_3z + L_4}{L_9x + L_{10}y + L_{11}z + 1}$$

$$v = \frac{L_5x + L_6y + L_7z + L_8}{L_9x + L_{10}y + L_{11}z + 1}$$

DLT calibration consist of calculate the eleven parameters, and since each point provides two equations, a minimum of six points to calibrate is necessary.

$$\begin{bmatrix} \frac{x}{R} & \frac{y}{R} & \frac{z}{R} & \frac{1}{R} & 0 & 0 & 0 & 0 & \frac{-ux}{R} & \frac{-uy}{R} & \frac{-uz}{R} \\ 0 & 0 & 0 & 0 & \frac{x}{R} & \frac{y}{R} & \frac{z}{R} & \frac{1}{R} & \frac{-vx}{R} & \frac{-vy}{R} & \frac{-vz}{R} \end{bmatrix} \cdot \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \\ L_9 \\ L_{10} \\ L_{11} \end{bmatrix} = \begin{bmatrix} \frac{u}{R} \\ \frac{v}{R} \end{bmatrix}$$

$$R = L_9x + L_{10}y + L_{11}z + 1$$

To improve the results obtained by this method is necessary to include in the above equations the correction of errors caused by optical lenses distortion and deviation of the optical center.

$$u = \frac{L_1x + L_2y + L_3z + L_4}{L_9x + L_{10}y + L_{11}z + 1} + \Delta u$$

$$v = \frac{L_5x + L_6y + L_7z + L_8}{L_9x + L_{10}y + L_{11}z + 1} + \Delta v$$

$$\Delta u = \xi(L_{12}r^2 + L_{13}r^4 + L_{14}r^6) + L_{15}(r^2 + 2\xi^2) + L_{16}\xi\eta$$

$$\Delta v = \eta(L_{12}r^2 + L_{13}r^4 + L_{14}r^6) + L_{15}\xi\eta + L_{16}(r^2 + 2\eta^2)$$

$$\xi = u - u_0$$

$$\eta = v - v_0$$

$$r^2 = \xi^2 + \eta^2$$

$$\begin{bmatrix} \frac{x}{R} & \frac{y}{R} & \frac{z}{R} & \frac{1}{R} & 0 & 0 & 0 & 0 & \frac{-ux}{R} & \frac{-uy}{R} & \frac{-uz}{R} & \xi r^2 & \xi r^4 & \xi r^6 & (r^2 + 2\xi^2) & \xi\eta \\ 0 & 0 & 0 & 0 & \frac{x}{R} & \frac{y}{R} & \frac{z}{R} & \frac{1}{R} & \frac{-vx}{R} & \frac{-vy}{R} & \frac{-vz}{R} & \eta r^2 & \eta r^4 & \eta r^6 & \xi\eta & (r^2 + 2\eta^2) \end{bmatrix} \cdot \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \\ L_9 \\ L_{10} \\ L_{11} \\ L_{12} \\ L_{13} \\ L_{14} \\ L_{15} \\ L_{16} \end{bmatrix} = \begin{bmatrix} \frac{u}{R} \\ \frac{v}{R} \end{bmatrix}$$

Where L_{12} , L_{13} and L_{14} correspond to the distortion correction, and L_{15} and L_{16} to the deviation of the optical center. By increasing the number of parameters in order to solve the equations system, it is necessary to increase the minimum calibration points up to eight.

Once obtained the coefficients of DLT method, it is possible to calculate intrinsic and extrinsic parameters of the calibrated camera. With the above coefficients, whether the optical defects have been corrected or not, you can create the following projection matrix:

$$P = \begin{bmatrix} L_1 & L_2 & L_3 & L_4 \\ L_5 & L_6 & L_7 & L_8 \\ L_9 & L_{10} & L_{11} & 1 \end{bmatrix}$$

Terms (L_9, L_{10}, L_{11}) of P matrix have correspondence with the terms of the rotation matrix R , (r_{31}, r_{32}, r_{33}) , except an scale factor. It must meet that:

$$\sqrt{r_{31}^2 + r_{32}^2 + r_{33}^2} = 1$$

Therefore, one can calculate the scale factor as:

$$\sqrt{L_9^2 + L_{10}^2 + L_{11}^2} = |\lambda| \sqrt{r_{31}^2 + r_{32}^2 + r_{33}^2}$$

$$|\lambda| = \sqrt{L_9^2 + L_{10}^2 + L_{11}^2}$$

$$\begin{bmatrix} p'_{11} & p'_{12} & p'_{13} & p'_{14} \\ p'_{21} & p'_{22} & p'_{23} & p'_{24} \\ p'_{31} & p'_{32} & p'_{33} & p'_{34} \end{bmatrix} = \frac{1}{|\lambda|} \begin{bmatrix} L_1 & L_2 & L_3 & L_4 \\ L_5 & L_6 & L_7 & L_8 \\ L_9 & L_{10} & L_{11} & 1 \end{bmatrix}$$

$$\begin{bmatrix} -f_x & 0 & u_0 \\ 0 & -f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} = \begin{bmatrix} p'_{11} & p'_{12} & p'_{13} & p'_{14} \\ p'_{21} & p'_{22} & p'_{23} & p'_{24} \\ p'_{31} & p'_{32} & p'_{33} & p'_{34} \end{bmatrix}$$

Once normalized P , matrix, the parameters of the camera can be calculated.

$$t_z = \text{sgn} \cdot p'_{34}$$

Where sgn is a sign to determine according to the position of the camera regarding the world referente frame in Z axis.

$$u_0 = p'_{11}p'_{31} + p'_{12}p'_{32} + p'_{13}p'_{33}$$

$$v_0 = p'_{21}p'_{31} + p'_{22}p'_{32} + p'_{23}p'_{33}$$

$$f_x = \sqrt{(p'_{11}^2 + p'_{12}^2 + p'_{13}^2) - u_0^2}$$

$$f_y = \sqrt{(p'_{21}^2 + p'_{22}^2 + p'_{23}^2) - v_0^2}$$

$$f = f_x d_x = f_y d_y$$

Where d_x and d_y are the distances between the centres of the camera sensor elements.

$$r_{1t} = \text{sgn} \cdot \frac{p'_{1t} - u_0 p'_{3t}}{f_x} \quad t = 1,2,3$$

$$r_{2t} = \text{sgn} \cdot \frac{p'_{2t} - v_0 p'_{3t}}{f_y} \quad t = 1,2,3$$

$$r_{3t} = \text{sgn} \cdot p'_{3t} \quad t = 1,2,3$$

$$t_x = \text{sgn} \cdot \frac{p'_{14} - u_0 t_z}{f_x}$$

$$t_y = \text{sgn} \cdot \frac{p'_{24} - v_0 t_z}{f_y}$$

The decomposition of projection matrix P , does not guarantee that rotation matrix R is orthogonal, therefore its orthogonality must be ensured. An easy way to do this is through SVD decomposition.

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = U \cdot D \cdot V^T$$

It is only necessary to replace D matrix by the identity matrix $I_{3 \times 3}$ to ensure that R is orthogonal.

>> DLT 2D

It is identical to the DLT 3D method except for the fact that z coordinates are constant. Then, the number of parameters is reduced from eleven to eight, so only four calibration points are needed.

$$u = \frac{L_1 x + L_2 y + L_3}{L_7 x + L_8 y + 1}$$

$$v = \frac{L_4 x + L_5 y + L_6}{L_7 x + L_8 y + 1}$$

$$\begin{bmatrix} \frac{x}{R} & \frac{y}{R} & \frac{1}{R} & 0 & 0 & 0 & \frac{-ux}{R} & \frac{-vy}{R} \\ 0 & 0 & 0 & \frac{x}{R} & \frac{y}{R} & \frac{1}{R} & \frac{-vx}{R} & \frac{-vy}{R} \end{bmatrix} \cdot \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \end{bmatrix} = \begin{bmatrix} u \\ v \\ R \end{bmatrix}$$

$$R = L_7x + L_8y + 1$$

>> DLT REFERENCES

- Abdel-Aziz, Y.I., Karara, H.M. (1971). **Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry**. *Proceedings of the Symposium on Close-Range Photogrammetry* (pp. 1-18). Falls Church, VA: American Society of Photogrammetry.
- Shapiro, R. (1978). **Direct linear transformation method for three-dimensional cinematography**. *Res. Quart.*49, 197-205.
- Liu, H.T. (2001). **Video-Based Stereoscopic Imaging and Measurement System (SIMS) for Undersea Applications**. *MTI/IEEE Proc. Oceans 2001 Conf.*, Honolulu, Hawaii.
- Hatze, H. (1988). **High-precision three-dimensional photogrammetric calibration and object space reconstruction using a modified DLT-approach**. *J. Biomech* 21, 533-538.

>> TSAI Model

Tsai method for calibration is based on the pin-hole camera model. This model transforms points in world reference frame (X_w, Y_w, Z_w) to points in image reference frame (u, v) . To simplify his calibration model compared to other models, Tsai supposes known some of the camera parameters like d_x and d_y , and if they are known with enough accuracy, it is also possible to introduce the central projection point (C_x, C_y) as input data. Another simplification of the method consists of considering only radial distortion of the image, simplifying the model because most of the calculations are performed over linear equations.

The transformation between world coordinates (X_w, Y_w, Z_w) and image coordinates (u, v) is carried out in four basic steps:

1.-Firstly, world coordinates (X_w, Y_w, Z_w) are transformed into coordinates in camera reference frame (X_c, Y_c, Z_c) using the rotation matrix R and the translation vector T , which are obtained during camera calibration.

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + T$$

2.-Next step consist of converting the coordinates of the points from camera reference frame (X_c, Y_c, Z_c) to the sensor reference frame, without considering the distortion (x_u, y_u) . To make this coordinate transformation, the perspective projection of the pin-hole camera geometry is used, so it is necessary to calibrate the focal length f of the camera.

$$x_u = f \frac{X_c}{Z_c}$$

$$y_u = f \frac{Y_c}{Z_c}$$

3.-In the third step undistorted sensor coordinates (x_u, y_u) are transformed to distorted sensor coordinates (x_d, y_d) . In this step is necessary to have calibrated the radial distortion factor k_1 .

$$x_d + x_d k_1 r^2 = x_u$$

$$y_d + y_d k_1 r^2 = y_u$$

$$r = \sqrt{x_d^2 + y_d^2}$$

4.-In the last step the coordinates of the points are transformed from the sensor reference frame with distortion (x_d, y_d) to image reference frame (u, v) . In this step is necessary to have calibrated the camera scale factor s_x . If manufacturer's data related to the optical center (C_x, C_y) are good enough, the calibration of these parameters can be obviated and the parameters given by the manufacturer can be used instead.

$$u = \frac{s_x x_d}{d_x} + C_x$$

$$v = \frac{y_d}{d_y} + C_y$$

>> TSAI 2D

From the point of view of the Tsai calibration method, two different steps can be distinguished. In the first step, orientation (R) of the camera is calculated, and also its position in x and y axis (T_x, T_y) . In the second step is calculated an approximation to some of the parameters that will be used as initial values in the optimization. Those parameters are: effective focal length (f) , distortion coefficient (k_1) and the position of the camera in z axis (T_z) .

From the image captured by the camera, the screen coordinates of calibration points (u_i, v_i) are extracted, with $i = 1, \dots, n$, and with n the number of calibration points considered. When coplanar calibrations are used, it is not possible to determine the scale factor s_x using Tsai method. If the camera used for calibration is digital, and relatively modern, it can be assured that this parameter will be one. If it is not the case, the camera must be calibrated using non-coplanar calibration gauge objects to obtain the correct value of s_x . In 2D Tsai calibration method $s_x = 1$ is established (if the real value is unknown) and depending on the camera type, calibration results will be or not the correct ones.

$$x_{dt} = \frac{d'_x}{s_x}(u_t - C_x)$$

$$y_{dt} = d'_y(v_t - C_y)$$

$$[y_{dt}X_{wt} \quad y_{dt}Y_{wt} \quad y_{dt} \quad -x_{dt}X_{wt} \quad -x_{dt}Y_{wt}] \begin{bmatrix} T_y^{-1}r_1 \\ T_y^{-1}r_2 \\ T_y^{-1}T_x \\ T_y^{-1}r_4 \\ T_y^{-1}r_5 \end{bmatrix} = x_{dt}$$

Where $T_y^{-1}r_1$, $T_y^{-1}r_2$, $T_y^{-1}T_x$, $T_y^{-1}r_4$ and $T_y^{-1}r_5$ are the unknown terms of the equation, so at least five calibration points are needed to calibrate the camera.

$$C = \begin{bmatrix} r'_1 & r'_2 \\ r'_4 & r'_5 \end{bmatrix} = \begin{bmatrix} \frac{r_1}{T_y} & \frac{r_2}{T_y} \\ \frac{r_4}{T_y} & \frac{r_5}{T_y} \end{bmatrix}$$

If none of the rows and columns in C matrix is null, T_y can be calculated with:

$$T_y = \sqrt{\frac{S_p - [S_p^2 - 4(r'_1r'_5 - r'_4r'_2)^2]^{1/2}}{2(r'_1r'_5 - r'_4r'_2)^2}}$$

$$S_p = r'^2_1 + r'^2_2 + r'^2_4 + r'^2_5$$

On the other hand, if any of the rows or columns of C matrix is null, T_y will be calculated using the equation:

$$T_y = \sqrt{(r'^2_i + r'^2_j)^{-1}}$$

Where r'_i and r'_j made reference to the not null row or column elements of C matrix. In the calculation of T_y , it exists ambiguity in its sign, reason why it must be determined. For that, any calibration point t is selected, being its coordinates in image reference frame (u_t, v_t) far from the central point (C_x, C_y) . Usually, the farthest calibration point is selected. Once determined point t , its (X_{wt}, Y_{wt}, Z_{wt}) coordinates in world reference frame are considered, and the following calculations are performed:

$$r_1 = (T_y^{-1}r_1)T_y$$

$$r_2 = (T_y^{-1}r_2)T_y$$

$$r_4 = (T_y^{-1}r_4)T_y$$

$$r_5 = (T_y^{-1}r_5)T_y$$

$$T_x = (T_y^{-1}T_x)T_y$$

$$x = r_1X_{wt} + r_2Y_{wt} + T_x$$

$$y = r_4X_{wt} + r_5Y_{wt} + T_y$$

$$x_{dt} = \frac{d'_x}{s_x}(u_t - C_x)$$

$$y_{dt} = d'_y(v_t - C_y)$$

If x and x_{dt} have the same sign, and the same happens with y and y_{dt} , then the sign of term T_y is positive. Otherwise the sign for T_y is negative. Once known the sign of T_y , rotation matrix R can be calculated, and also the position in x axis T_x .

$$r_1 = (T_y^{-1}r_1)T_y$$

$$r_2 = (T_y^{-1}r_2)T_y$$

$$r_3 = (1 - r_1^2 - r_2^2)^{1/2}$$

$$r_4 = (T_y^{-1}r_4)T_y$$

$$r_5 = (T_y^{-1}r_5)T_y$$

$$r_6 = \text{sgn} \cdot (1 - r_4^2 - r_5^2)^{1/2}$$

$$r_7 = (r_2r_6) - (r_3r_5)$$

$$r_8 = (r_3r_4) - (r_1r_6)$$

$$r_9 = (r_1r_5) - (r_2r_4)$$

$$T_x = (T_y^{-1}T_z)T_y$$

$$\alpha = (r_1 r_4 + r_2 r_5)$$

In the term r_6 of the rotation matrix, the sign (sgn) is still undefined. If $\alpha < 0$ the sign of r_6 is positive, while if $\alpha > 0$, the sign of r_6 is negative.

$$R = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix}$$

Once obtained R matrix, its orthogonality must be assured by the transformation to the associated Euler angles. After that, this transformation is undone. This way, the orthogonality of R matrix is guaranteed.

With the calculated values of R , T_x and T_y can start the next step of the Tsai's method. As it is not possible to know a priori the radial distortion factor k_1 , it is supposed to be zero and approximated values for f and T_z are calculated. Those approximated values are used as initial values in the optimisation, allowing calculate the exact values of k_1 , f and T_z .

$$y_t = r_4 X_{wt} + r_5 Y_{wt} + T_y$$

$$w_t = r_7 X_{wt} + r_8 Y_{wt}$$

$$[y_t \quad -d_y y_{dt}] \begin{bmatrix} f \\ T_z \end{bmatrix} = w_t d_y y_{dt}$$

With the approximation calculated for f in the above equation, it is possible to check if the orientation extracted from R matrix is correct. If $f < 0$ it is necessary to change the sign in some of the terms in R matrix, and recalculate the approximations of f and T_z .

$$R = \begin{bmatrix} r_1 & r_2 & -r_3 \\ r_4 & r_5 & -r_6 \\ -r_7 & -r_8 & r_9 \end{bmatrix}$$

Once obtained all the calibration parameters, the following equation is optimized to obtain definitive values of the intrinsic and extrinsic parameters.

$$d_y y_{dt} + d_y y_{dt} k_1 r_i^2 = f \frac{r_4 X_{wt} + r_5 Y_{wt} + T_y}{r_7 X_{wt} + r_8 Y_{wt} + T_z}$$

$$r_i = \sqrt{\left(\frac{d_x^t x_{dt}}{s_x}\right)^2 + (d_y y_{dt})^2}$$

>> TSAI 3D

From the point of view of the Tsai calibration method, two different steps can be distinguished. In the first step, orientation (R) of the camera is calculated, and also its position in x and y axis (T_x, T_y). In the second step is calculated an approximation to some of the parameters that will be used as initial values in the optimization. Those parameters are: effective focal length (f), distortion coefficient (k_1) and the position of the camera in z axis (T_z).

From the image captured by the camera, the screen coordinates of calibration points (u_t, v_t) are extracted, with $t = 1, \dots, n$, and with n the number of calibration points considered. To make the calculations of this step the approximation $s_x = 1$ is established. The real value for the scale factor will be obtained in the optimization.

$$x_{dt} = \frac{d_x^t}{s_x} (u_t - C_x)$$

$$y_{dt} = d_y (v_t - C_y)$$

$$[y_{dt} X_{wt} \quad y_{dt} Y_{wt} \quad y_{dt} Z_{wt} \quad y_{dt} \quad -x_{dt} X_{wt} \quad -x_{dt} Y_{wt} \quad -x_{dt} Z_{wt}] \begin{bmatrix} T_y^{-1} s_x r_1 \\ T_y^{-1} s_x r_2 \\ T_y^{-1} s_x r_3 \\ T_y^{-1} s_x T_x \\ T_y^{-1} r_4 \\ T_y^{-1} r_5 \\ T_y^{-1} r_6 \end{bmatrix} = x_{dt}$$

Where $T_y^{-1} s_x r_1$, $T_y^{-1} s_x r_2$, $T_y^{-1} s_x r_3$, $T_y^{-1} T_x$, $T_y^{-1} r_4$, $T_y^{-1} r_5$ y $T_y^{-1} r_6$ are the unknown terms of the equation, so at least seven calibration points are needed to calibrate the camera.

$$T_y = \frac{1}{\sqrt{(T_y^{-1} r_4)^2 + (T_y^{-1} r_5)^2 + (T_y^{-1} r_6)^2}}$$

In the calculation of T_y , it exists ambiguity in its sign, reason why it must be determined. For that, any calibration point t is selected, being its coordinates in image reference frame (u_t, v_t) far from the central point (C_x, C_y). Usually, the farthest calibration point is selected. Once determined point t , its (X_{wt}, Y_{wt}, Z_{wt}) coordinates in world reference frame are considered, and the following

calculations are performed:

$$r_1 = (T_y^{-1}r_1)T_y$$

$$r_2 = (T_y^{-1}r_2)T_y$$

$$r_4 = (T_y^{-1}r_4)T_y$$

$$r_5 = (T_y^{-1}r_5)T_y$$

$$T_x = (T_y^{-1}T_x)T_y$$

$$x = r_1X_{wt} + r_2Y_{wt} + T_x$$

$$y = r_4X_{wt} + r_5Y_{wt} + T_y$$

$$x_{dt} = \frac{d^t x}{s_x}(u_t - C_x)$$

$$y_{dt} = d_y(v_t - C_y)$$

If x and x_{dt} have the same sign, and the same happens with y and y_{dt} , then the sign of term T_y is positive. Otherwise the sign for T_y is negative. The next step is to calculate the scale factor s_x .

$$s_x = |T_y| \sqrt{(T_y^{-1}r_1)^2 + (T_y^{-1}r_2)^2 + (T_y^{-1}r_3)^2}$$

At this point, the rotation matrix R can be obtained, and also the position in x axis T_x .

$$r_1 = \frac{(T_y^{-1}r_1)T_y}{s_x}$$

$$r_2 = \frac{(T_y^{-1}r_2)T_y}{s_x}$$

$$r_3 = \frac{(T_y^{-1}r_3)T_y}{s_x}$$

$$r_4 = (T_y^{-1}r_4)T_y$$

$$r_5 = (T_y^{-1}r_5)T_y$$

$$r_6 = (T_y^{-1}r_6)T_y$$

$$r_7 = (r_2r_6) - (r_3r_5)$$

$$r_8 = (r_3r_4) - (r_1r_6)$$

$$r_9 = (r_1r_5) - (r_2r_4)$$

$$T_x = (T_y^{-1}T_x)T_y$$

$$R = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix}$$

Once obtained R matrix, its ortogonality must be assured by the transformation to the associated Euler angles. After that, this transformation is undone. This way, the ortogonality of R matrix is guaranteed.

With the calculated values of R , T_x and T_y can start the next step of the Tsai's method. As it is not possible to know a priori the radial distortion factor k_1 , it is supposed to be zero and approximated values for f and T_z are calculated. Those approximated values are used as initial values in the optimization, allowing calculate the exact values of k_1 , f and T_z .

$$y_t = r_4X_{wt} + r_5Y_{wt} + r_6Z_{wt} + T_y$$

$$w_t = r_7X_{wt} + r_8Y_{wt} + r_9Z_{wt}$$

$$[y_t \quad -d_y y_{dt}] \begin{bmatrix} f \\ T_z \end{bmatrix} = w_t d_y y_{dt}$$

With the approximation calculated for f in the above equation, it is possible to check if the orientation extracted from R matrix is correct. If $f < 0$ it is necessary to change the sign in some of the terms in R matrix, and recalculate the approximations of f and T_z .

$$R = \begin{bmatrix} r_1 & r_2 & -r_3 \\ r_4 & r_5 & -r_6 \\ -r_7 & -r_8 & r_9 \end{bmatrix}$$

Once obtained all the calibration parameters, the following equation is optimized to obtain definitive values of the intrinsic and extrinsic parameters.

$$d_y y_{dt} + d_y y_{dt} k_1 r_t^2 = f \frac{r_4 X_{wt} + r_5 Y_{wt} + r_6 Z_{wt} + T_y}{r_7 X_{wt} + r_8 Y_{wt} + r_9 Z_{wt} + T_x}$$

$$r_t = \sqrt{\left(\frac{d_x}{s_x} x_{dt}\right)^2 + (d_y y_{dt})^2}$$

>> TSAI REFERENCES

- Tsai, R.Y. (1986). **An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision.** *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Miami Beach, FL, pp. 364-374.
- Tsai, R.Y. (1987). **A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses.** *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 4, pp. 323-344.
- Willson, R.G., & Shafer, S.A. (1994). **What is the Center of the Image?** *Journal of the Optical Society of America A*, Vol. 11, No. 11, pp.2946-2955.
- Tapper, M., McKerrow, J., Abrantes, J. (2002). **Problems Encountered in the Implementation of Tsai's Algorithm for Camera Calibration.** *Proc. 2002 Australasian Conference on Robotics and Automation*, Auckland.
- Kim, D. (2006). **Camera Calibration.** 2006-SN-003-EN Unpublished. Intelligent Systems Research Center. Sungkyunkwan University. Korea.

>> FAUGERAS Model

The calibration method proposed by Faugeras is very similar to DLT 3D model where no optical defects are corrected.

>> FAUGERAS

This calibration method is to find a projection matrix that relates image coordinates (u_t, v_t) with world coordinates (X_t, Y_t, Z_t) .

$$\lambda_t m_t = P M_t$$

$$m_t = (u_t, v_t)$$

$$M_t = (X_t, Y_t, Z_t)$$

Where λ_t is a scale factor. Being the projection matrix P :

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}$$

$$\begin{bmatrix} \lambda_t u_t \\ \lambda_t v_t \\ \lambda_t \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X_t \\ Y_t \\ Z_t \\ 1 \end{bmatrix}$$

$$u_t = \frac{p_{11}X_t + p_{12}Y_t + p_{13}Z_t + p_{14}}{p_{31}X_t + p_{32}Y_t + p_{33}Z_t + p_{34}}$$

$$v_t = \frac{p_{21}X_t + p_{22}Y_t + p_{23}Z_t + p_{24}}{p_{31}X_t + p_{32}Y_t + p_{33}Z_t + p_{34}}$$

To obtain the twelve terms of the projection matrix it is enough to solve the following equation. Since each calibration point provides two equations, at least six calibration points are needed.

$$\begin{bmatrix} X_t & Y_t & Z_t & 1 & 0 & 0 & 0 & 0 & -u_t X_t & -u_t Y_t & -u_t Z_t & -u_t \\ 0 & 0 & 0 & 0 & X_t & Y_t & Z_t & 1 & -v_t X_t & -v_t Y_t & -v_t Z_t & -v_t \end{bmatrix} \cdot \begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \\ p_{31} \\ p_{32} \\ p_{33} \\ p_{34} \end{bmatrix} = 0$$

One way to solve the above equation, is to apply SVD decomposition to the matrix L (another way could be through the calculation of the pseudoinverse of the matrix formed from the world coordinates of the points).

$$L = \begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -u_1 X_1 & -u_1 Y_1 & -u_1 Z_1 & -u_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1 X_1 & -v_1 Y_1 & -v_1 Z_1 & -v_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & -u_n X_n & -u_n Y_n & -u_n Z_n & -u_n \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & -v_n X_n & -v_n Y_n & -v_n Z_n & -v_n \end{bmatrix}$$

$$L = U \cdot D \cdot V^T$$

The twelve terms of the projection matrix P_{12} would be the twelve elements of the last column of the matrix V , obtained in the SVD decomposition. Once obtained the projection matrix P , intrinsic and extrinsic parameters can be extracted from this matrix. The parameters can be easily calculated directly through decomposition of the matrix P (it is also possible to calculate using QR decomposition).

$$\begin{bmatrix} -f_x & 0 & u_0 \\ 0 & -f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} = \frac{1}{|\lambda|} \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}$$

The first step is to calculate the scale factor λ . The elements of the last row of the rotation matrix R must meet:

$$\sqrt{r_{31}^2 + r_{32}^2 + r_{33}^2} = 1$$

Therefore:

$$\sqrt{p_{31}^2 + p_{32}^2 + p_{33}^2} = |\lambda| \sqrt{r_{31}^2 + r_{32}^2 + r_{33}^2}$$

$$|\lambda| = \sqrt{p_{31}^2 + p_{32}^2 + p_{33}^2}$$

$$\begin{bmatrix} p'_{11} & p'_{12} & p'_{13} & p'_{14} \\ p'_{21} & p'_{22} & p'_{23} & p'_{24} \\ p'_{31} & p'_{32} & p'_{33} & p'_{34} \end{bmatrix} = \frac{1}{|\lambda|} \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}$$

$$\begin{bmatrix} -f_x & 0 & u_0 \\ 0 & -f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} = \begin{bmatrix} p'_{11} & p'_{12} & p'_{13} & p'_{14} \\ p'_{21} & p'_{22} & p'_{23} & p'_{24} \\ p'_{31} & p'_{32} & p'_{33} & p'_{34} \end{bmatrix}$$

Once normalized P matrix, the camera parameters can be obtained.

$$t_z = \text{sgn} \cdot p'_{34}$$

Where sgn is a sign to determine according to the position of the camera regarding the world referent frame in Z axis.

$$u_0 = p'_{11}p'_{31} + p'_{12}p'_{32} + p'_{13}p'_{33}$$

$$v_0 = p'_{21}p'_{31} + p'_{22}p'_{32} + p'_{23}p'_{33}$$

$$f_x = \sqrt{(p'_{11}^2 + p'_{12}^2 + p'_{13}^2) - u_0^2}$$

$$f_y = \sqrt{(p'_{21}^2 + p'_{22}^2 + p'_{23}^2) - v_0^2}$$

$$f = f_x d_x = f_y d_y$$

Where d_x and d_y are the distances between the centres of the camera sensor elements.

$$r_{1t} = \text{sgn} \cdot \frac{p'_{1t} - u_0 p'_{3t}}{f_x} \quad t = 1,2,3$$

$$r_{2t} = \text{sgn} \cdot \frac{p'_{2t} - v_0 p'_{3t}}{f_y} \quad t = 1,2,3$$

$$r_{3t} = \text{sgn} \cdot p'_{3t} \quad t = 1,2,3$$

$$t_x = \text{sgn} \cdot \frac{p'_{14} - u_0 t_z}{f_x}$$

$$t_y = \text{sgn} \cdot \frac{p'_{24} - v_0 t_z}{f_y}$$

The decomposition of projection matrix P , does not guarantee that rotation matrix R is orthogonal, therefore its orthogonality must be ensured. An easy way to do this is through SVD decomposition.

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = U \cdot D \cdot V^T$$

It is only necessary to replace D matrix by the identity matrix $I_{3 \times 3}$ to ensure that R is orthogonal.

>> CORRECTION OF OPTICAL DISTORTION IN FAUGERAS METHOD

To eliminate optical distortion during Faugeras calibration, correction polynomials are used, considering two coefficients of radial distortion correction k_1 and k_2 .

$$u_d = u_u + k_1(u_u - u_o)(x^2 + y^2) + k_2(u_u - u_o)(x^2 + y^2)^2$$

$$v_d = v_u + k_1(v_u - v_o)(x^2 + y^2) + k_2(v_u - v_o)(x^2 + y^2)^2$$

Where: (u_d, v_d) are the distorted image coordinates, (u_u, v_u) are the image coordinates without distortion and (x, y) are the distorted sensor coordinates.

First of all, an initial calibration according to Faugeras method is carried out to achieve an approximation to the projection matrix P . Thus, from this approximation, it is possible to calculate the distortion coefficients k_1 and k_2 . To this end, the image coordinates without distortion (u_u, v_u) are obtained using the world coordinates of the points (X_w, Y_w, Z_w) and the projection matrix P .

$$u_u = \frac{p_{11}X_w + p_{12}Y_w + p_{13}Z_w + p_{14}}{p_{31}X_w + p_{32}Y_w + p_{33}Z_w + p_{34}}$$

$$v_u = \frac{p_{21}X_w + p_{22}Y_w + p_{23}Z_w + p_{24}}{p_{31}X_w + p_{32}Y_w + p_{33}Z_w + p_{34}}$$

The distorted sensor coordinates (x, y) are obtained from:

$$x = d_x(u_d - u_o)$$

$$y = d_y(v_d - v_o)$$

Once known the image coordinates without distortion and the distorted sensor coordinates, the equations of distortion correction can be written in matrix form:

$$\begin{bmatrix} u_d - u_u \\ v_d - v_u \end{bmatrix} = \begin{bmatrix} (u_u - u_o)(x^2 + y^2) & (u_u - u_o)(x^2 + y^2)^2 \\ (v_u - v_o)(x^2 + y^2) & (v_u - v_o)(x^2 + y^2)^2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$

$$d = D \cdot k$$

Where (u_d, v_d) are the distorted image coordinates obtained during the extraction of image points captured by the camera. Thus, it is possible to calculate an initial value to the distortion coefficients k_1 and k_2 .

Once obtained the approximations to P , k_1 and k_2 , an iterative method is used to minimize the objective function:

$$\sum_{i=1}^n \|(u_{u+d}, v_{u+d})_i - (u_u, v_u)_i\|^2$$

$$(u_{u+d}, v_{u+d})_i = f(P, k_1, k_2, d_i, (X_w, Y_w, Z_w)_i)$$

Where $(u_{u+d}, v_{u+d})_i$ are the distorted image coordinates obtained adding deviation d_i , due to distortion, to the image coordinates without distortion $(u_u, v_u)_i$.

>> FAUGERAS REFERENCES

- Kim, D. (2006). **Camera Calibration**. 2006-SN-003-EN Unpublished. Intelligent Systems Research Center. SungKyunKwan University. Korea.
- Faugeras, O. (1993). **Three-Dimensional Computer Vision: A Geometric Viewpoint**, The MIT Press.

>> ZHANG Model

Zhang model is a camera calibration method that uses traditional calibration techniques (known calibration points) and self-calibration techniques (correspondence between the calibration points when they are in different positions). To perform a full calibration by the Zhang method at least three different images of the calibration gauge are required, either by moving the gauge or the camera itself. If some of the intrinsic parameters are given as data (orthogonality of the image or optical center coordinates) the number of images required can be reduced to two.

In a first step, an approximation of the estimated projection matrix with one of the classical calibration methods is obtained, usually DLT or Faugeras. Subsequently, applying self-calibration techniques, it is obtained the absolute conic matrix that allows us to calculate the intrinsic parameters matrix, and from this, the rotation and translation matrices for each of the images. Finally, radial distortion is modeled by means of two coefficients, and all the parameters obtained above are optimized, but taking into account the distortion.

>> ZHANG

First of all it is necessary to obtain an approximation of the projection matrix H , which, by homography, permits relate the world coordinates (X_w, Y_w, Z_w) to image coordinates (u, v) of calibration points. the approximation to H can be obtained using the DLT or Faugeras methods.

$$s \cdot m = A[R \cdot T] \cdot M$$

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \begin{bmatrix} r_1 & r_2 & r_3 & t \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

Where s is a scale factor. As calibration points are coplanar, $Z_w = 0$ can be taken.

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix}$$

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = H \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix}$$

Once obtained the projection matrix H an optimization is performed to improve it. The image coordinates are calculated (u', v') using H matrix, and also the world coordinates of the calibration points (X_w, Y_w, Z_w) .

$$\begin{bmatrix} s \cdot u' \\ s \cdot v' \\ s \end{bmatrix} = H \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix}$$

$$u' = \frac{s \cdot u'}{s}$$

$$v' = \frac{s \cdot v'}{s}$$

$$error = \sqrt{(u' - u)^2 + (v' - v)^2}$$

By minimizing $error$ function, we get a better approximation to matrix H . When H matrix has been already improved, we can proceed with the absolute conic matrix calculation B .

$$H = A \begin{bmatrix} R & T \end{bmatrix}$$

$$\begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix} = \lambda \cdot A \begin{bmatrix} r_1 & r_2 & t \end{bmatrix}$$

Where λ is a scale factor. r_1 y r_2 are orthonormal vectors.

$$h_1^{-T} A^{-T} A^{-1} h_2 = 0$$

$$h_1^{-T} A^{-T} A^{-1} h_1 = h_2^{-T} A^{-T} A^{-1} h_2$$

$$B = A^{-T} A^{-1} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}$$

Being B a symmetric matrix, it is possible to define the vector b that would describe the absolute conic image as:

$$b = [b_{11} \quad b_{12} \quad b_{22} \quad b_{13} \quad b_{23} \quad b_{33}]^T$$

Using b it is possible to define:

$$\begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix} = H$$

$$h_i^T B h_j = v_{ij}^T b \quad i = 1, 2, 3 \quad j = 1, 2, 3$$

$$v_{ij}^T b = 0$$

$$v_{ij} = [h_{i1}h_{j1} \quad h_{i1}h_{j2} + h_{i2}h_{j1} \quad h_{i2}h_{j2} \quad h_{i3}h_{j1} + h_{i1}h_{j3} \quad h_{i3}h_{j2} + h_{i2}h_{j3} \quad h_{i3}h_{j3}]^T$$

Where h_{ij} is the i th row and j th column element of the matrix H . For each of the planes we can define the homogeneous system of equations (each plane provides two equations related with its associated H):

$$\begin{bmatrix} v_{12}^T \\ (v_{11} - v_{22})^T \end{bmatrix} b = 0$$

By solving the above system (SVD decomposition) we obtain the vector b , which allows us to calculate the intrinsic parameters matrix A .

$$A = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$v_0 = \frac{(b_2 b_4 - b_1 b_3)}{(b_1 b_3 - b_2^2)}$$

$$\lambda = b_3 - \frac{b_4^2 + v_0 (b_2 b_4 - b_1 b_3)}{b_1}$$

$$\alpha = \sqrt{\frac{\lambda}{b_1}}$$

$$\beta = \sqrt{\frac{\lambda b_1}{(b_1 b_2 - b_3^2)}}$$

$$\gamma = \frac{-b_2 \alpha^2 \beta}{\lambda}$$

$$u_o = \gamma \frac{v_o}{\alpha} - b_4 \frac{\alpha^2}{\lambda}$$

Starting from A matrix and H matrices of each plane it is possible to calculate the rotation matrices R and translation vectors T associated to each image.

$$\lambda = \frac{1}{\|A^{-1}h_1\|} = \frac{1}{\|A^{-1}h_2\|}$$

$$r_1 = \lambda A^{-1}h_1$$

$$r_2 = \lambda A^{-1}h_2$$

$$r_3 = r_1 \times r_2$$

$$T = \lambda A^{-1}h_3$$

$$R = [r_1 \ r_2 \ r_3]$$

Due to possible errors, both in the extraction of points as in the manufacture and gauge measurement, rotation matrix R may not be orthogonal, therefore we must ensure their orthogonality. An easy way to do this is by SVD decomposition.

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = U \cdot D \cdot V^T$$

It is only necessary to replace D matrix by identity matrix $I_{3 \times 3}$ to ensure that R is orthogonal. Once ensured the orthogonality of matrix R , through a non linear optimization (Levenberg-Marquardt) the problem variables are refined by minimizing the equation:

$$\sum_{i=1}^n \sum_{j=1}^m \|(u, v)_{ij} - (u_p, v_p)_{ij}\|^2$$

Where n is the number of captured images of the gauge, m is the number of gauge points and $(u_p, v_p)_{ij}$ are the image coordinates obtained by:

$$\begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix}_i = A [R_i \ T_i] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}_j$$

The last step of the model is to correct the radial distortion caused by lenses in the image coordinates of the points. Being (u, v) the ideal image coordinates (without distortion), (u_d, v_d) the distorted image coordinates and (x, y) the ideal screen coordinates, we have:

$$u_d = u + (u - u_o)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2]$$

$$v_d = v + (v - v_o)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2]$$

First it will be needed to calculate an approximation of the distortion coefficients k_1 and k_2 using the following equation and the pseudoinverse matrix method.

$$\begin{bmatrix} u_d - u \\ v_d - v \end{bmatrix} = \begin{bmatrix} (u - u_o)(x^2 + y^2) & (u - u_o)(x^2 + y^2)^2 \\ (v - v_o)(x^2 + y^2) & (v - v_o)(x^2 + y^2)^2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$

Once obtained the estimation of the coefficients, it is performed an optimization similar to that used to refine the parameter matrices of the camera, but this time the below equation is minimized:

$$\sum_{i=1}^n \sum_{j=1}^m \|(u, v)_{ij} - (u_p, v_p)_{ij}\|^2$$

$$(u_p, v_p)_{ij} = f(A, R_i, T_i, k_1, k_2, (X_w, Y_w, Z_w)_j)$$

>> ZHANG REFERENCES

- Zhang, Z. (2000). **A flexible new technique for camera calibration**. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11):1330-1334.
- Zhang, Z. (1999). **Flexible Camera Calibration By Viewing a Plane From Unknown Orientations**. International

>> ANNEXES

>> NCE Coefficient (Normalized Calibration Error)

NCE coefficient provides a value that gives an idea of the accuracy reached in a calibration test. This coefficient is independent of the conditions of the test (camera position and orientation, calibration gauge object position and orientation, camera field of view, camera resolution,...). Thus, not only several calibration test can be compared, but also it is possible to compare calibrations carried out with the different camera calibration methods without the influence of the peculiarities of each one.

$$NCE = \frac{1}{n} \sum_{i=1}^n \left[\frac{(x_i - X_i)^2 + (y_i - Y_i)^2}{(a_i^2 + b_i^2)/12} \right]$$

Where: (X_i, Y_i) are the world coordinates of the calibration points; (x_i, y_i) are the world coordinates of the calibration points calculated from the image coordinates and from the parameters obtained in calibration; and a_i y b_i are the dimensions in u and v directions respectively, of the projection of a pixel in an orthogonal plane to the camera optical axis and is placed at distance z_i from the camera optical center. In case of digital cameras with square pixels $a = b$.

$$a_i = b_i = \frac{z_i}{f}$$

being z_i the distance between the optical center of the camera and the plane that contains the calibration point i .

If $NCE < 1$ the reconstruction error of the calibration points is lower, in mean value, than the digitalization noise of one pixel for the calibration distance. For a value of $NCE \approx 1$ the residual distortion is negligible compared to the noise of digitalization. And if $NCE \gg 1$, the camera calibration is inadequate.

>> NCE REFERENCES

- Juyang, W., Cohen, P., Herniou, M., (1992). **Camera Calibration with Distortion Models and Accuracy Evaluation**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 4, pp. 965-980.

