Calibration Tool - User Manual

Lovicourt Léo

May 6, 2022

1 Installation

1.1 Linux distributions

- Extract the archive in any folder.
- Open a terminal in the extracted folder.
- Type **"make"** to build the executable file.
- Type **"./calibrationTool"** to run the program.

1.2 Windows

- Run **"Calibration Tool Setup"**.
- Follow the installer steps.
- To launch the program, the user have 2 options :
  1. Open the folder where the program was installed, and then run **"Calibration Tool"**.
  2. Run the **"Calibration Tool"** shortcut created on the desktop.

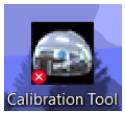

Figure 1: Desktop shortcut of the program.

2 Available models

2.1 Perspective

A perspective camera is designed to mimic the way the human eye sees. It is the most common projection mode used for rendering a 3D scene. It has several parameters available for estimation :

- Focal length : distance from the camera's center to the image plane.
- Principal point : point at the intersection of the optical axis and the image plane.
- Distorsions : distorsions coefficients (k1, k2, k3 are the radial distorsion coefficients, and p1, p2 are the tangential distorsion coefficients).

2.2 Spherical

A spherical camera has a field of view that covers approximately an entire sphere, or at least a full circle in the horizontal plane.

- Generalised focal length : ?
- Principal point : point at the intersection of the optical axis and the image plane.
- Skew : number of pixels per unit length in each direction on the sensor.
- Xi : parameter describing the mirror shape (Christopher Mei's model).
- Distorsions : distorsions coefficients (k1, k2 are the radial distorsion coefficients, and p1, p2 are the tangential distorsion coefficients).

3 Utilisation

3.1 Presentation of the user interface

The user interface is fairly easy to use. At first, 4 buttons are displayed on the screen, each having a specific function.

- **Perspective camera** - Allows a calibration on a perspective camera.
- **Spherical camera** - Allows a calibration on an omnidirectional camera.
- **About** - Displays a short help message.
- **Exit** - Quits the program.

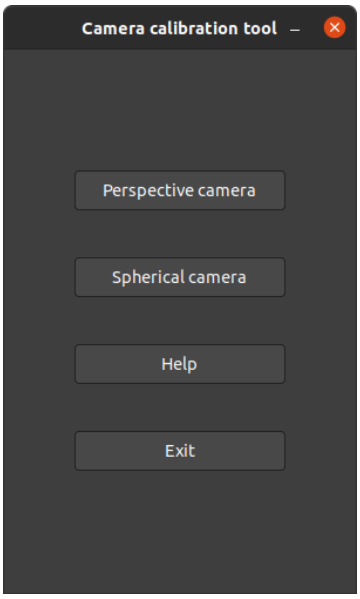

Figure 2: Main menu of the program.

To start any kind of calibration, the user must choose one of the two calibration types available.

3.2 Load images

To perform both calibrations, the user will have to provide at least 3 images of same dimensions. The file formats supported are *.bmp, .tiff, .jpg, .pgm and .png*. Once images are chosen and if they are valid, they will be displayed as a mosaic.

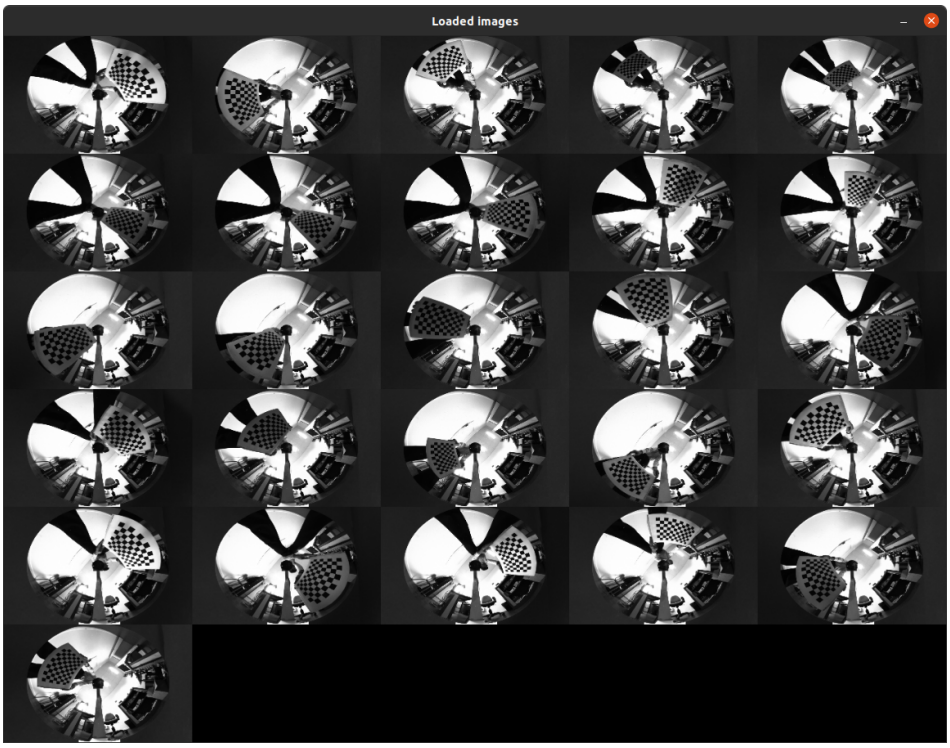

Figure 3: Mosaic containing images loaded.

3.3 Extract grid corners

This operation allows the user to extract the grid corners from each selected image. All images will be displayed one by one in a window, and corners will have a red cross on them if the board was found. The board size must be at least 4×4, and for a $n \times m$ grid, there should be $(n-1) \times (m-1)$ corners found. The user will have multiple ways to go through all selected images.

- **Pressing Enter, Y, or O** to set this image as valid and go to the next image.
- **Pressing N** to specify that this image shouldn't be used for calibration and go to the next image.

The user also has the possibility to zoom on an image. To do so, he has to hold the **left-click** on the image and draw a rectangle around the desired area. Once the selection is done, press **Enter** to confirm selection and perform a zoom, or press **Escape** to undo the selection. A **left-click** on the image will have the same effect as pressing Escape. Once a zoom has been performed, the user can return to the initial view by pressing **Escape**, or go to to the next image using the keys presented earlier.

A valid image is one whose corners have all been detected and colored.

Note : the program will itself set an image as not valid if not any corner has been found.

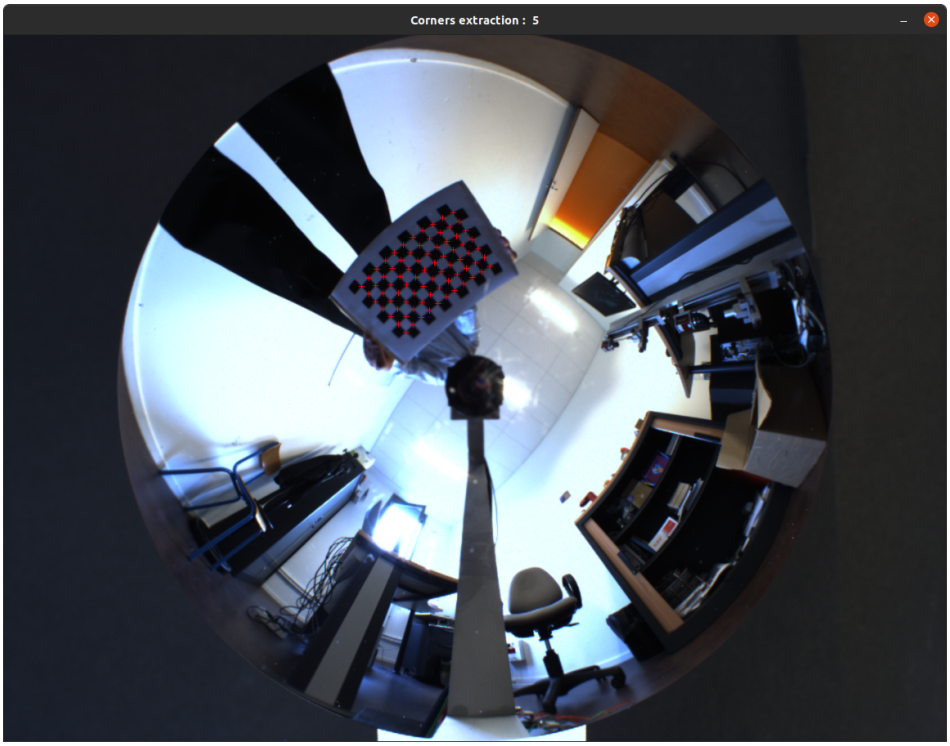

Figure 4: Corners extraction of an image.

After going through all the images, a popup will ask the user if the extraction was successful. The user can select **"No"** to do another extraction with eventually other parameters and other images, or **"Yes"** to then perform calibration.

3.4 Calibration

As the program has now all the required data, the user can now perform calibration. Once the operation is done, a popup will appear to tell the user the calibration was successful. However, it may fail if the user has set images as valid even though they are not.

3.5 Show corners reprojection

By clicking on this button, the user will go again through all the **valid** images to view the reprojection of the corners calculated for each frame. For each corner, it will have a blue cross displayed were the corner was found during extraction, and a red cross at the recalculated position of this corner. There will finally be a legend at the bottom-left of the image, including these two crosses and the error rate (in pixel) of the image.

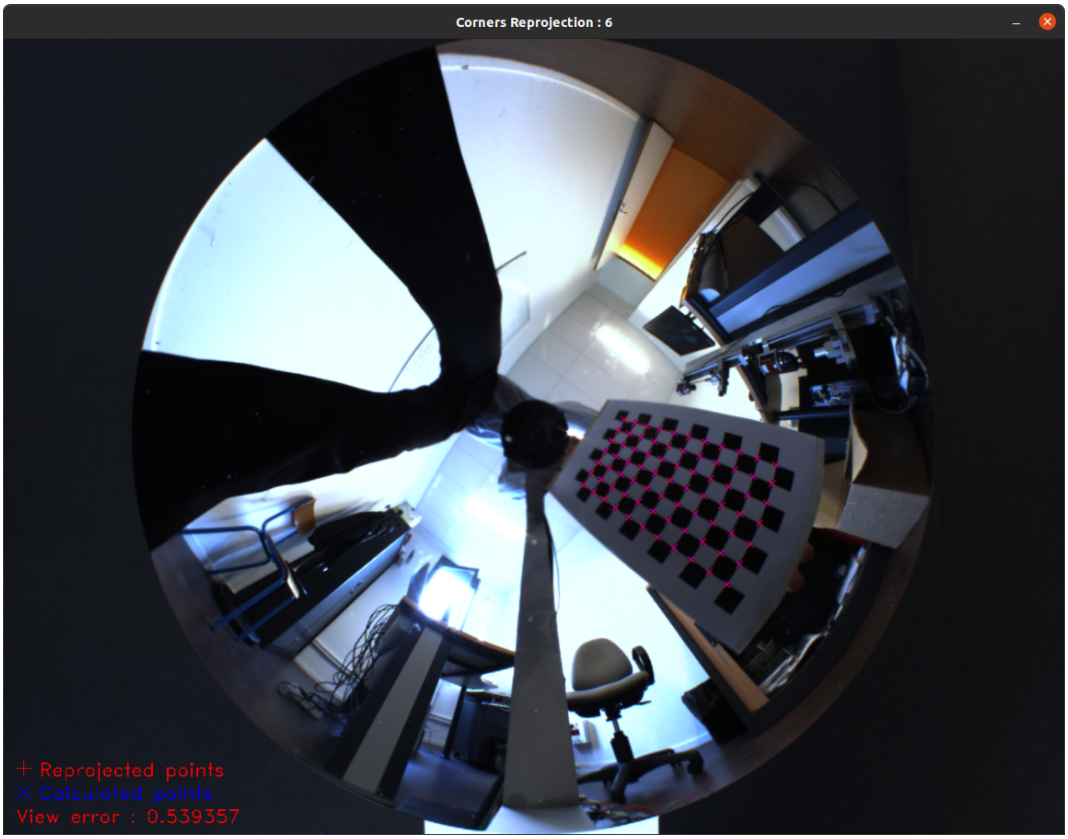

Figure 5: Corners reprojection of an image after calibration.

3.6 Calibration results

A frame will appear on the screen, with the results of the calibration. It will contain various informations, depending on the type of calibration performed, but also the informations about the images used for calibration.

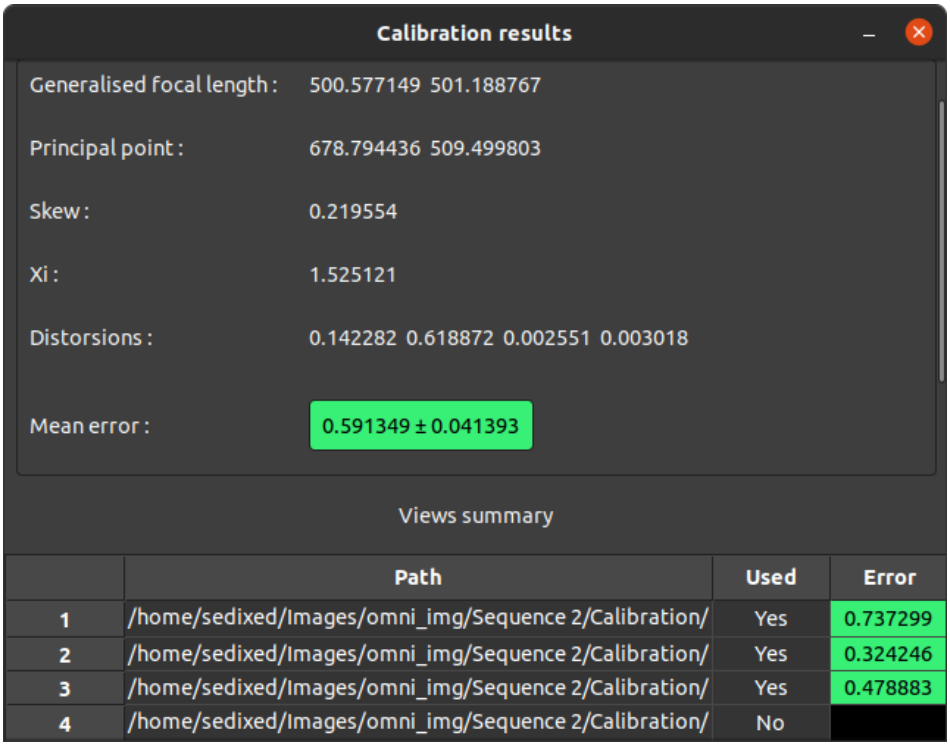

Figure 6: Spherical calibration results.

3.7 Save

The user can save the results of the calibration after clicking on **"Calibration results"** as an YML file. It will contain all the data calculated through the calibration, and more precisely :

- Type of calibration performed (perspective / spherical)
- Number of images provided
- Properties of the mire used
- Mean error
- Method used *(Perspective only)*
- Render size and search size
- Intrinsics parameters
- Distorsions coefficients
- Skew *(Spherical only)*
- Flags (integer form)
- Images data (validity, path, error rate, extrinsics parameters)

3.8 Load YML file

The user can load a YML file previously saved through this program. By doing so, it will set all the parameters according to the values readed in the file chosen. However, this operation may fail if the file has been modified externally with incorrect data.

3.9 Preferences

The user can set some preferences for the calibration. Except the parameters to estimate that have to be selected before calibration, they have to be selected before grid corners extraction, otherwise the user will have to extract the corners again.
A parameter is fixed if its checkbox is unchecked, otherwise it will be estimated by the program.

- **Parameters to estimate** : these are parameters the user can choose to fix to a default value that will be used for calibration.

  1. **Focal length** *(Perspective only)* : can be fixed using two values.
  2. **Principal point** : can be fixed to the center of an image loaded.
  3. **Distorsions** : each of the distorsion coefficients can be fixed to 0.
  4. **Method** *(Perspective only)* : the frame provides a button to select the calibration method used. The **RO** (**R**eleasing **O**bjects) method is an extension of the default one. In many common cases with inaccurate, unmeasured, roughly planar targets (calibration plates), this method can dramatically improve the precision of the estimated camera parameters.
  5. **Skew** *(Spherical only)* : can be fixed to 0.
  6. **Xi** *(Spherical only)* : can be fixed to 1.

- **Size of the window for the corners detection** : When the corners are being extracted, an automatic procedure is used to refine its coordinates. The size of the window used can be specified here.

- **Render window size** : Preferred size of the window in which the images are displayed.

- **Calibration pattern properties** : The pattern properties are very important to obtain good results. The size of the squares has to be set according to the pattern you are using. All measurements are in millimeters.

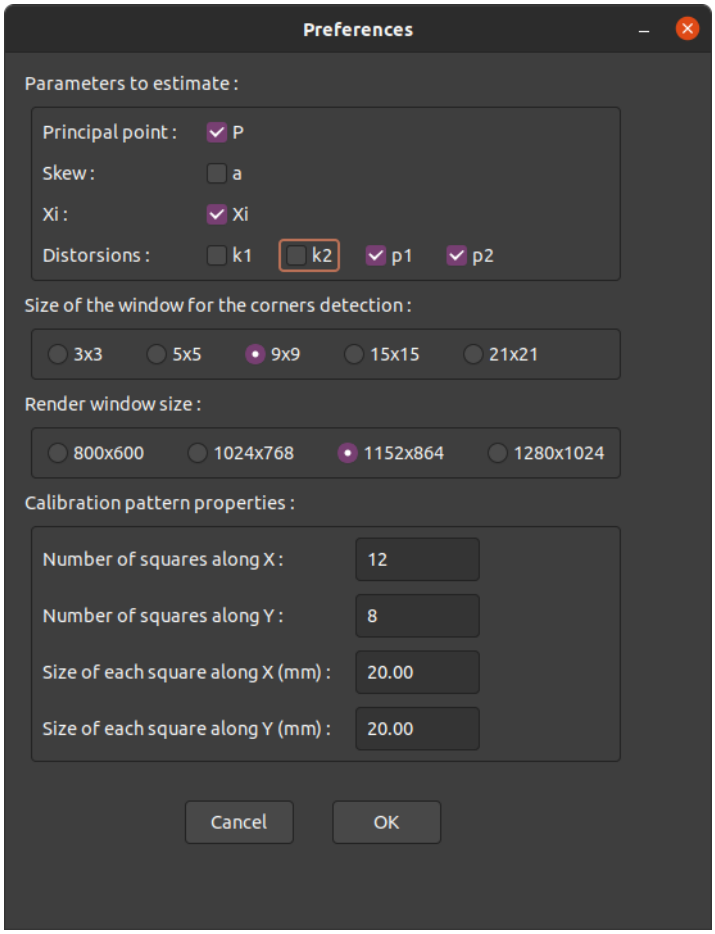

Figure 7: Preferences frame for a spherical calibration.

Note : the parameters to estimate won't be editable before loading images as it requires the size of an image for the principal point.

3.10 Exit

Quits the program.