Calibration

October 1, 2021

1 Introduction

A camera is an optical instrument that captures a visual image. It i composed of a lens that funnel (and distort) the light and a sensor that capture it digitally. This process can be described as a mapping from a 3D space to a 2D one where a point in the image coordinate system can be the projection of infinite locations in the world. This mapping is also known as projective transformation. Because of this transformation, recovering the depth or the position of objects in the scene from a single camera without knowing the characteristics of the object is not possible. If the size of the object is known however, its distance from the camera can be approximated roughly. The only way to recover the object's distance to a camera with good precision is to triangulate its position from multiple cameras. This operation requires complete knowledge of the physics of the cameras (intrinsic parameters) and their positions and orientations in the space (extrinsic parameters).

The intrinsics, also known as internal parameters, are the parameters describing the physics of the camera such as the focal length, optical center, radial and tangential distortion coefficients. Their estimation is a fundamental step in order to remove the nonlinearities introduced by the lens. Removing the distortion from the images allows us to use linear transformations such as homographies and projection matrices. The extrinsics parameters, also called external ones, refer to the parameters describing the position and orientation of the camera in the space. These allow to project a 3D point from the world to the cameras coordinate systems and vice versa.

In this document we describe the procedure to estimate the intrinsic and extrinsic camera parameters for a set of cameras with overlapping field of view. Knowing these parameters for a set of cameras will allow us to recover the position and/or the depth of objects in the scene through triangulation if their projections/locations in the images is known. From now on we assume the camera to be described by the pinhole model.

2 Background

2.1 The camera model

Projection model. A camera is a mapping between the 3D world and a 2D image. The mathematical relationship between these two coordinates (without considering the lens distortion and that the camera is at focus) is, in most cases, described by the pinhole camera model. Under this model, a points in space $\mathbf{X} = (X, Y, Z)^T$ is mapped to the a point on the image plane $\mathbf{x} = (x, y)^T$ where a line, called ray, joining the point $\mathbf{X}$ to the centre of projection meets the image plane.

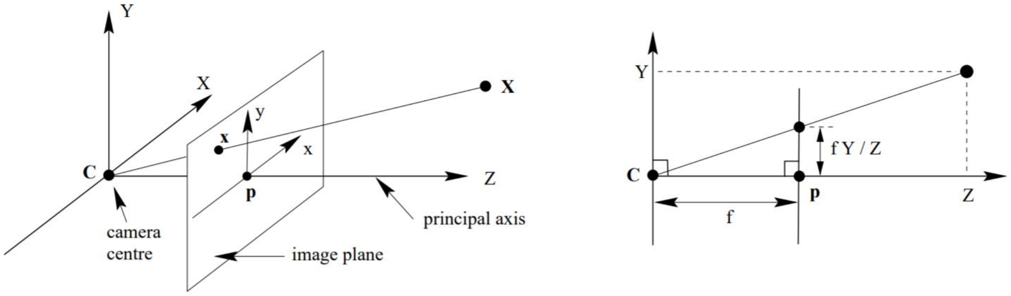

Figure 1: **Pinhole camera geometry** taken from []. **C** is the camera centre and **p** the principal point.

By similar triangles, one can easily find that:

$$(X, Y, Z)^T \mapsto (x, y)^T = (fX/Z, fY/Z)^T \tag{1}$$

where f is the camera focal length (distance from the camera center to the image plane) and Z the depth of the object w.r.t the camera center.

This, also called central projection, can be expressed, in homogeneous coordinates, in matrix form as follow:

$$\begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{2}$$

The perspective projection of Eq. 2 describes a model where the image coordinate system is orthogonal (no skew), the principal point is in the center of the image and that it ranges between -1 and 1. To account for the fact that (i) the image is defined in pixels, (ii) that the origin of the image coordinate system is at the top left corner, (iii) that the principal point which is the projection of the center of the lens in the image is not necessarily in the middle of the image and (iv) that the image coordinate might be skewed we have:

$$\begin{pmatrix} uw \\ vw \\ w \end{pmatrix} = \begin{bmatrix} F_x & s & c_x \\ 0 & F_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{3}$$

$$\mathbf{x} = K\mathbf{X_{cam}} \tag{4}$$

where $F_x = fm_x$ and $F_y = fm_y$ represent the focal length in pixels, m the number of pixels per unit of distance and s the skew parameter. For modern cameras the image coordinate system can be considered orthogonal, hence $s = 0$. This matrix K is called *camera calibration matrix* or *intrinsic matrix.*

Eq. 4 describes a mapping between a 3D space (called camera coordinate system) whose coordinate system is at the camera center to a 2D space defined in pixels.

As the points in 3D are in general defined in another reference system which is common to other cameras, we have to account for the rotation R and translation t of the camera w.r.t this reference system. $[R|t]$ is called the pose of the camera and are the *extrinsic parameters.* R is orthonormal. The extrinsic parameters allows us to project 3D points defined in the world coordinate system for example, to the 3D coordinate systems of the cameras and vice versa. From the camera coordinate system to the image we then use Eq. 4. Combining the two give us the 3×4 projection matrix in Eq. 5.

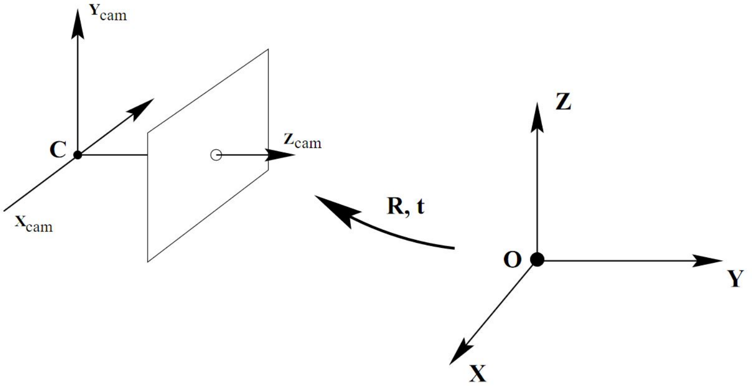

Figure 2: The Euclidean transformation between the world and the camera coordinate frames.

$$\mathbf{x} = K[R|t]\mathbf{X} \tag{5}$$

Distortion model. A good approximation of the distortion created by the lens on the image can be expressed as a combination of radial and tangential distortion functions. These are nonlinear and are expressed in polynomial form.

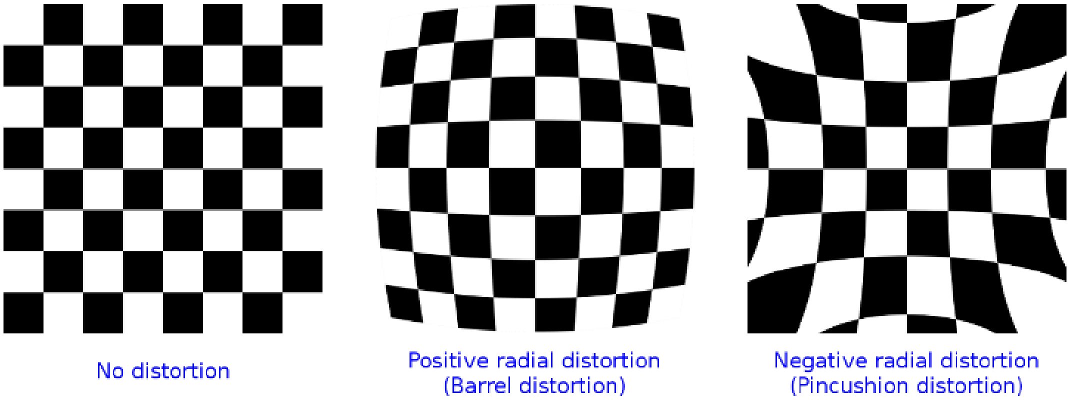

Figure 3: Radial distortions.

Radial distortion occurs when light rays bend more near the edges of a lens than they do at its optical center. The smaller the lens, the greater the distortion.

$$u_{dist} = x \cdot (1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6)$$
$$v_{dist} = y \cdot (1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6)$$

(x, y) is the position of a pixel/point in the normalized image coordinate system (undistorted) while (u_{dist}, v_{dist}) of the distorted image again in normalized coordinate. $r = \sqrt{x^2 + y^2}$ is the normalized distance of the pixel/point from the principal point which is (roughly) in the center of the image and k_1, k_2 and K_3 are the distortion coefficients.

Tangential distortion instead occurs when the lens and the image plane are not parallel.

$$u_{dist} = x + (2 \cdot p_1 \cdot x \cdot y + p_2 \cdot (r^2 + 2 \cdot x^2))$$
$$v_{dist} = y + (2 \cdot p_2 \cdot x \cdot y + p_1 \cdot (r^2 + 2 \cdot y^2))$$

where p_2 and p_2 are the distortion coefficients.

In OpenCV, the intrinsic parameters are always in this order $[k_1\ k_2\ p_1\ p_2\ k_3]$.

3 Intrinsics estimation

To estimate the intrinsic parameters such that matrix K and the distortion coefficients we rely on OpenCV implementation of the Zhang's camera calibration approach [4].

The procedure requires imaging the same planar pattern with known calibration points from different viewpoints. As the position of the calibration points is known w.r.t a reference system on the pattern itself, correspondences between them and their projection in the images can be established. An estimate of the projection matrix can be computed for each image which is subsequently decomposed into rotation R, translation t and the intrinsic matrix K. A minimum of 3 viewpoints is required to estimate the coefficients of K. As the estimated position of the calibration points in the images are noisy, the solution is typically found in the least-squares sense using as many viewpoints as possible. The distortion coefficients are as well estimated using linear least-squares. Finally, all parameters are refined iteratively.

The typical calibration pattern is a black and white checkerboard where the calibration points are the inner corners (where two black or two white corners meet). An example checkerboard with 6 × 8 inner corners is shown in Fig. 4. Other patterns such as a grid of circles can be used. In this case, the calibration points are the center of the circles which have to be detected in the image using a different approach then the one used for corners.

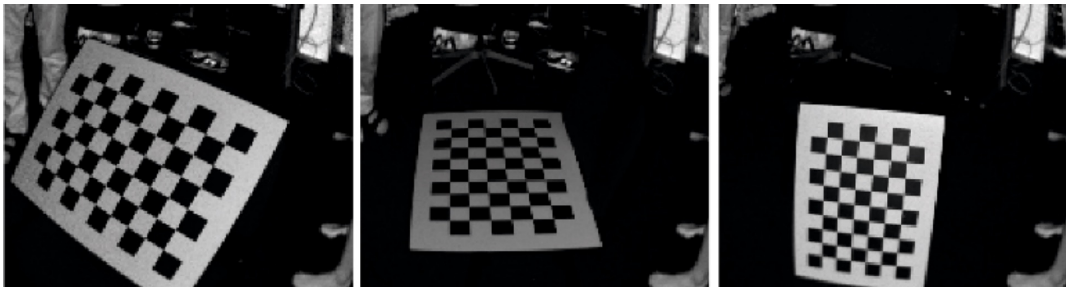

Figure 4: Typical calibration images for Zhang's approach. The checkerboard is printed and glued on a flat surface, it is at an angle w.r.t the camera sensor and it occupies roughly half of the image. This calibration pattern has 6 inner corners along the height and 9 inner corners along the width.

3.1 Practical steps

We list here the steps required to obtain a proper calibration. More details with dos and don'ts are given in the tutorial.

1. Setting up the camera:
   (a) The camera has to be in the same mode/configuration as in the final system. That is, same zoom, focus, aperture, resolution, etc.. If the camera will be used in video mode in the final system you should used the same mode here as well.
   (b) Auto-focus, auto-stabilization or other such features that dynamically alter the geometry of the image must be disabled.
2. Preparing the calibration pattern:
   (a) Print the calibration pattern with the highest resolution and by making sure the squares have the correct aspect ratio.
   (b) Glue the pattern on a solid flat surface. Keep in mind that the quality of the pattern influences the calibration.
3. Acquisition of the calibration images:
   (a) Move either the pattern or the camera in order to acquire a set of images from different viewpoints. (typical range 40-200 images) If the camera is in video mode, move it slowly to avoid blur.
   (b) Ideally, the pattern should be completely visible in the image but is is not a strict requirement as the algorithm discard the images that are not usable. The pattern should be at an angle w.r.t the image sensor and take roughly half of the size of the image.
   (c) Ideally, the pattern depicted in the image set should cover the whole image frame with variations in the pose/viewpoint. The corners of the image should have special care.
4. Calibration:
   (a) At this stage we have a set of calibration images (typical range 40-200) that can be readily used with our tool.
   (b) After calibration, the various errors and the visual outputs indicates you if the algorithm converged to a proper solution. It this is not the case, there are several actions that can be undertook to solve it. We described them in detail in the tutorial .

4 Extrinsics estimation

To estimate the extrinsics parameters, which are the rotation R and the translation t, of each camera in the multiview setup with overlapping field of view, we rely on a method called Bundle Adjustment (BA) [3].

BA is a technique that simultaneously recover a 3D structure (i.e. a set of 3D locations) and viewing parameters (i.e. camera pose and possibly intrinsic matrix and radial distortion) from only a set of observations in images. More precisely, it amounts to jointly refining a set of initial camera and structure parameter estimates for finding the set of parameters that most accurately predict the locations of the observed points in the set of available images. Due to the nature of the problem, the initial parameters have to be set from a relative good starting point in order to avoid poor local minimum. We will show in section 4.2 how an approximate solution can be found using linear methods.

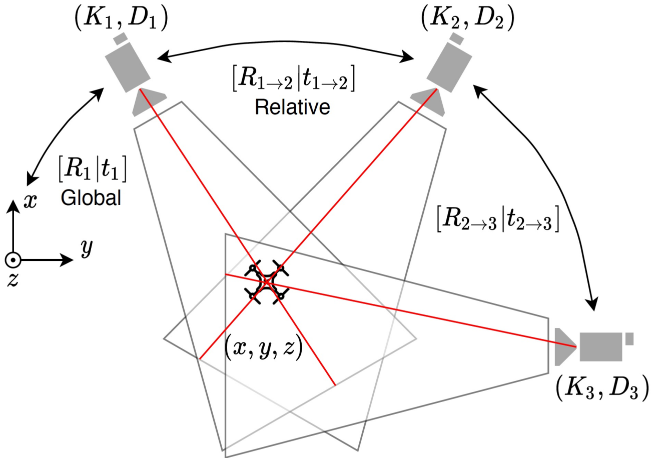

Figure 5:

4.1 Bundle adjustment

BA is a nonlinear optimization method that jointly refines a set of initial camera parameters $\{R_i\}_{i=1}^N$, $\{t_i\}_{i=1}^N$ with optional intrinsics $\{K_i\}_{i=1}^N$ and the distortion parameters $\{D_i\}_{i=1}^N$ and some 3D object locations $\{X_j\}_{j=1}^M$ by minimizing the distance to their corresponding locations $\{x_j\}_{j=1}^M$ in the images.

The objective function is defined as

$$\min_{C_i, X_i} \sum_i^N \sum_j^M m_{ij} \cdot d(\mathbf{Q}(C_i, X_j) - x_j)^2 \tag{6}$$

where $C_i = \{R_i, t_i, K_i, D_i\}$ refers to the camera parameters of view i, $\mathbf{Q}(.)$ the projection function, m_{ij} is a binary variable that is 1 when the point is visible in the view and 0 otherwise and $d(.)$ is the Euclidean distance.

4.2 Initial solution

We describe here the approach used to compute the initial solution, that is a pose for each camera, which serves the purpose of initializing the Bundle Adjustment parameters.

Relative poses. We assume here that the intrinsic matrix K_i and the distortion coefficients D_i are known for each one of the cameras and that the images are undistorted. Given two views with overlapping field of view and a set of image keypoints that are in common in both views, a *fundamental matrix* from view 1 to view 2 $F_{1\to2}$ can be computed using the 8-point method [2] or with a robust estimator such as RANSAC. As K_1 and K_2 are known, $F_{1\to2}$ can be used to compute the *essential matrix* $E_{1\to2} = K_2^T F_{1\to2} K_1$ which can then be decomposed to find the relative rotation $R_{1\to2}$ and translation $t_{1\to2}$ between the two cameras up to a scale. There is a theoretical ambiguity when reconstructing/decomposing the relative euclidean poses of two cameras from their essential matrix. This ambiguity is linked to the fact that, given a 2D point in an image, we cannot tell whether the corresponding 3D point is in front of the camera or behind the camera. For this reason, the decomposition of E leads to four solutions shown in Fig. 6. Since in our setup we orient all the cameras toward the center of the scene, the good solution can be found by verifying that the majority of the triangulate points are in front of both cameras. It is important to note also that the fundamental and essential matrices so as the relative poses are defined up to scale. This term means that the same relative pose or fundamental matrix is valid for infinite setup where only the scale varies. More formally, if $s \in \mathrm{R}$ is a factor, $s \cdot F$ is as well a valid solution that minimizes the error on the image points. For this reason, concatenating relative poses require taking care of the scale. The estimation of the scale usually requires external information such as the distance between points in the world.

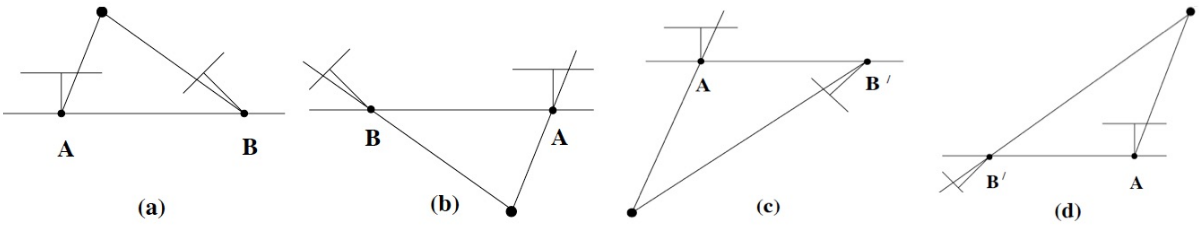

Figure 6: The four possible solution for the essential matrix decomposition. Only in a) the point is in front of both cameras.

Now that we have retrieved the relative pose between two cameras, if the pose of the first camera is known we can readily compute the pose of the second using $R_2 = R_{1\to 2} \cdot R_1$ and $t_2 = R_{1\to 2} \cdot R_1 + t_{1\to 2}$.

Concatenation of the relative poses. The reason behind the retrieval of the relative camera poses is to being able to concatenate them and hence to describe the pose of each camera in the setup. If we consider the cameras to be the nodes of a graph where the edges encode the relative poses among them, if a spanning tree like the one in Fig. 7 is given, all the poses can be computed w.r.t. one of the cameras. These poses can then be used as the initial solution for Bundle Adjustment.

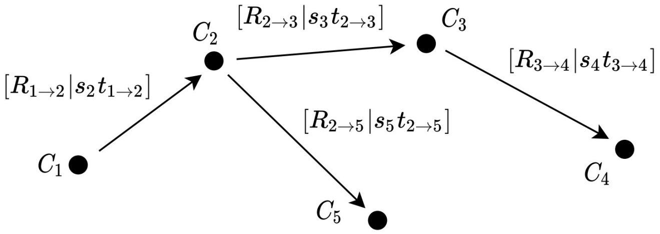
C_2
$[R_{2\to 3}|s_3t_{2\to 3}]$
C_3
$[R_{1\to 2}|s_2t_{1\to 2}]$
$[R_{3\to 4}|s_4t_{3\to 4}]$
$[R_{2\to 5}|s_5t_{2\to 5}]$
C_1
C_4
C_5

Figure 7: Example a spanning tree (minimal tree) connecting every camera through relative poses. As the relative poses are up to scale, each one of them has a different scale s_i which has to be equalized before concatenation.

If the pose for camera C_1 is given so as the relative poses $C_{1\to 2}$ and $C_{2\to 3}$, the pose of C_3 w.r.t. C_1 can be found with:

$$\begin{aligned}
R_2 &= R_{1\to 2} \cdot R_1 \\
t_2 &= R_{1\to 2} \cdot R_1 + t_{1\to 2} \\
R_3 &= R_{2\to 3} \cdot R_2 \\
t_3 &= R_{2\to 3} \cdot R_2 + s \cdot t_{2\to 3}
\end{aligned}$$

where s is the relative scale between the two relative poses that can be computed as the scale difference between common triangulate points.

4.3 Global registration

As we saw in the previous section, the camera poses obtained as initial solution so as the solution found by Bundle Adjustment are all defined w.r.t. the pose of the first camera. In order to move the poses to a global reference system with the correct scale we estimate the rigid transformation (rotation, scale, and translation) between the triangulated points and their true position in the world. Once the transformation is found, the poses can be projected to this new reference system. If a drone has been used to create the image landmarks, the true positions in the world used to compute this transformation can be GPS coordinates.

To find the rigid transformation (rotation, scale, and translation) between two point clouds with known correspondences, we compute an initial estimate using Procustes regsitration [1] then, perform a optimization to minimize the distance between the two point sets in the least squares sense.

4.4 Practical steps

We list here the steps required to perform the Bundle Adjustment. More details with dos and don'ts are given in the tutorial.

1. Setting up the cameras:
   (a) The positions of the cameras during the calibration have to be the same as in the final system. A new calibration of the extrinsics is required every time a camera move.
   (b) The cameras have to have the same mode/configuration as in the final system. That is, same zoom, focus, aperture, resolution, etc..
   (c) Auto-focus, auto-stabilization or other such features that dynamically alter the geometry of the image must be disabled.
2. Data acquisition:
   (a) The procedure requires a set of synchronized images/videos depicting one or multiple objects such as a drone or plane.
   (b) The images have to be annotated with the positions of the objects. The more precise the annotations the better the calibration.
3. Calibration:
   (a) At this stage we have a set of image points for each calibration object in the images that will be used to calibrate the cameras.
   (b) The steps to perform are: estimation of the relative poses, concatenation, bundle adjustment and finally global registration.
   (c) The various errors and the visual outputs indicates you if the algorithm converged to a good solution. It this is not the case, there are several actions that can be undertook to solve it. We described them in detail in the tutorial.

5 Conclusion

In this document we have explained the main concept and mathematics used in vision to model image formation. We have then described the methods used to estimate intrinsics and extrinsics parameters so as the practical steps required. The tool we developed implements these concepts and methods in a user-friendly interface that allows to measure the accuracy of the estimations and to find possible errors along the way.

Future Work. For what it concerns the methods used to estimate the various parameters, they have demonstrated to be reliable and effective in different situations. In term of outputs/messages, the tool could provide more insights on the accuracy in 3D rather than simply re-projection error and recognizing when the bundle adjustment is capped by the bounds would be useful as well.

References

[1] J.C. Gower and G.B. Dijksterhuis. Procrustes problems. 2005.

[2] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. In *Nature*, 1981.

[3] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle adjustment — a modern synthesis. In Bill Triggs, Andrew Zisserman, and Richard Szeliski, editors, *Vision Algorithms: Theory and Practice*, pages 298–372, 2000.

[4] Zhengyou Zhang. A flexible new technique for camera calibration. 1998.