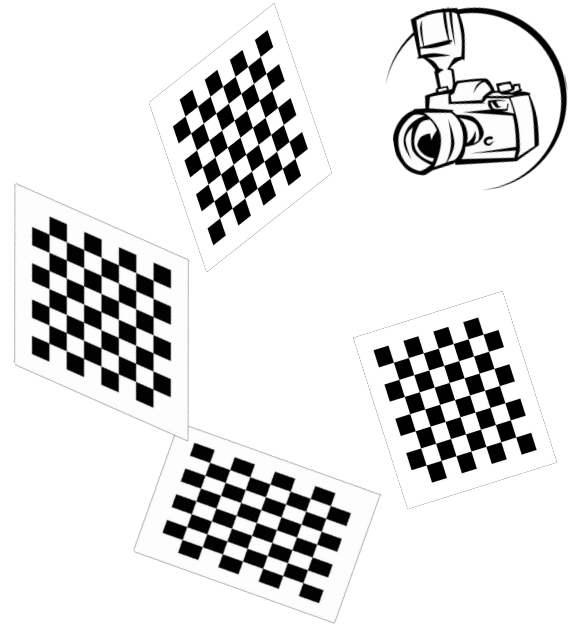


Computer Vision

Class 03



01

Camera Calibration

(Zhang's Algorithm)

What is Calibration?



Calibration tries to recover:

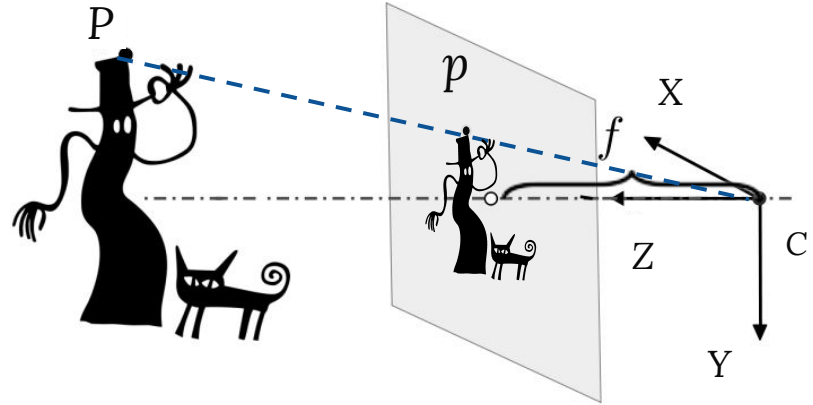
- The intrinsic camera parameters, i.e., the inner transformations of the camera, including focal length, position of the principal point, sensor scale and skew.
- The parameters of the non-linear lens distortion.
- The extrinsic parameters (3D rotation and translation) for each of the given views of the reference pattern.

$$\lambda p' = \underbrace{K}_{\text{intrinsic}} \Pi_0 \underbrace{g(R, T)}_{\text{lens distortion}} P_w$$

$$f(r) = 1 + \underbrace{a_1}_{\text{distortion}} r + \underbrace{a_2}_{\text{distortion}} r^2 + \underbrace{a_3}_{\text{distortion}} r^3 + \underbrace{a_4}_{\text{distortion}} r^4 + \dots$$

Perspective Projection Model

- Frontal Pinhole Model
- Image plane is in front of the optical center, positioned at the distance f from the optical center $C = (0, 0, 0)^T$ and perpendicular to the optical axis
- C is the origin of the camera frame
- The optical axis aligns with the Z-axis and intersects the image plane at $(0, 0, f)^T$



Perspective Projection Model

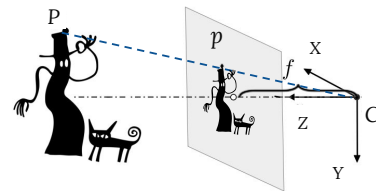
What are we going to estimate?

$$\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} R & T \\ \mathbf{0} & 1 \end{bmatrix}}_{g(\mathbf{R}, T)} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

$$p = c + f(r)(p_d - c), \quad \text{with} \quad r = \|p_d - c\|$$

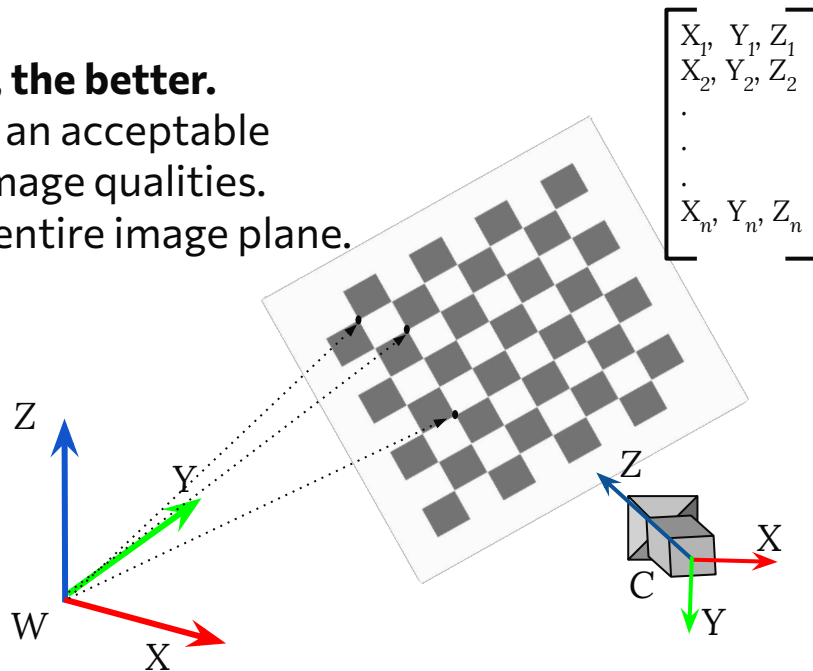
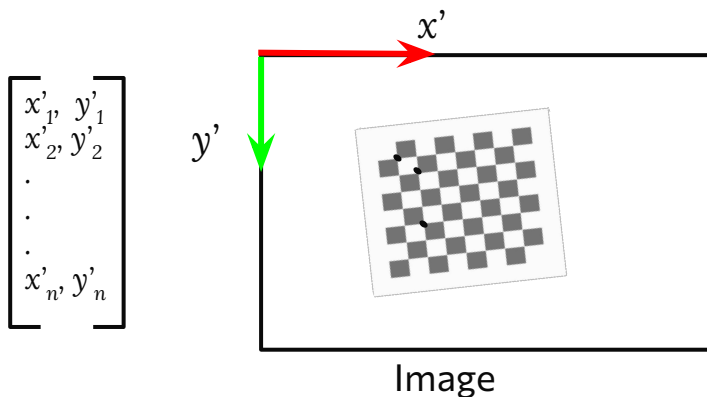
$$f(r) = 1 + a_1 r + a_2 r^2 + a_3 r^3 + a_4 r^4 + \dots$$

$$\mathbf{a} = [a_1, a_2, a_3, \dots]$$



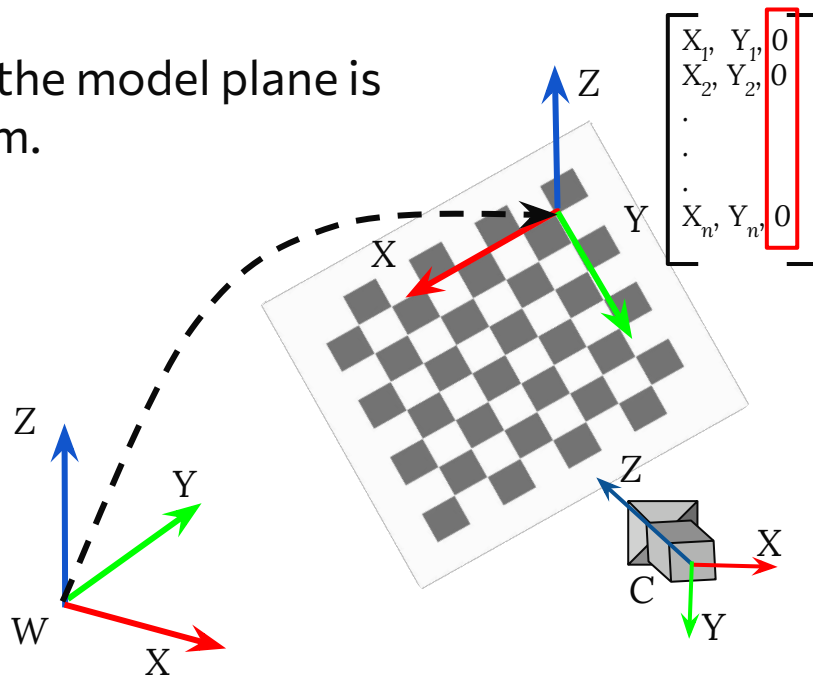
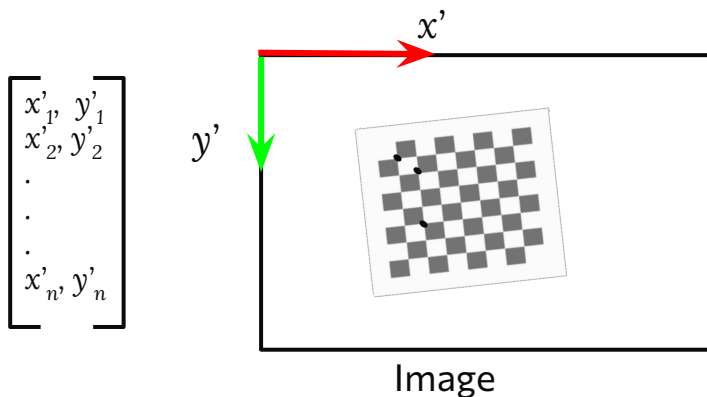
Calibration with a Chessboard

- We have to associate the 3D points of the chessboard with the 2D points in the image.
- Use at least three **GOOD** images. **The more, the better.** Usually with 20 images or more you can get an acceptable calibration, depending on the camera and image qualities.
- Vary position and orientation, to cover the entire image plane.



Calibration with a Chessboard

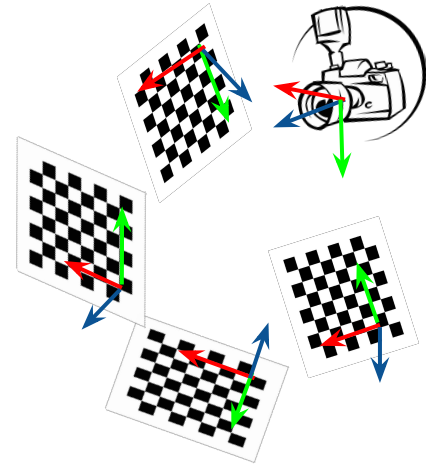
- To make our life easier...
- Without loss of generality, we assume the model plane is on $Z = 0$ of the world coordinate system.



Calibration Process

The calibration algorithm works as follows:

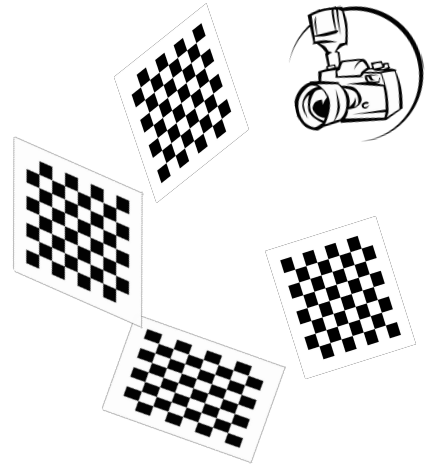
- Detect the corners in the images of the chessboard.
- Estimate the intrinsic parameters by calculating the homographies between the chessboard and the image plane.
- Once the intrinsic parameters are known, estimate the extrinsic parameters for each calibration image.
- Using the intrinsic and extrinsic parameters, reproject the 3D chessboard corners onto the image plane and compare them with the detected points in order to estimate the radial distortion coefficients.
- Finally, refine all parameters by treating calibration as a nonlinear minimization problem, solved using the Levenberg–Marquardt algorithm.



Calibration Procedure

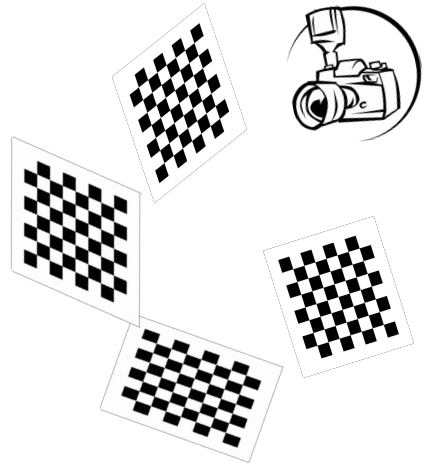
The recommended calibration procedure is as follows:

- a. Print a pattern and attach it to a planar surface;
- b. Take several images of the chessboard from different orientations by moving either the plane or the camera, making sure to cover the entire image plane without losing squares of the board;
- c. Specify the 3D coordinates of the chessboard corners;
- d. Detect the feature points (corners) in all the images;
- e. Run the calibration to estimate the intrinsic and extrinsic parameters of the camera, and also the coefficients of the radial distortion;
- f. If needed, refine all parameters through an optimization method.



Calibration Pattern Generator

<https://calib.io/pages/camera-calibration-pattern-generator?authuser=0>



Credits



- Zhengyou Zhang,
A Flexible New Technique for Camera Calibration,
Technical Report MSR-TR-98-71



- Zhengyou Zhang, A Flexible New Technique for Camera Calibration, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 11, November 2000