Project point from 3D world to 2D image camera using Camera Calibration

Ali, Mahmoud Ahmed Mohamed
MCV Group, Université de Bourgogne
Le Creusot, Dijon, France
mahmoud_Ali@etu.u-bourgogne.fr

Abstract- A camera, when used as a visual sensor, is an integral part of several domains like robotics, surveillance, space exploration, social media, industrial automation, and even the entertainment industry. For many applications, it is essential to know the parameters of a camera to use it effectively as a visual sensor. In this paper, we will understand the steps involved in camera calibration and their significance. The process of estimating the parameters of a camera is called camera calibration. This means we have all the information (parameters or coefficients) about the camera required to determine an accurate relationship between a 3D point in the real world and its corresponding 2D projection (pixel) in the image captured by that calibrated camera.

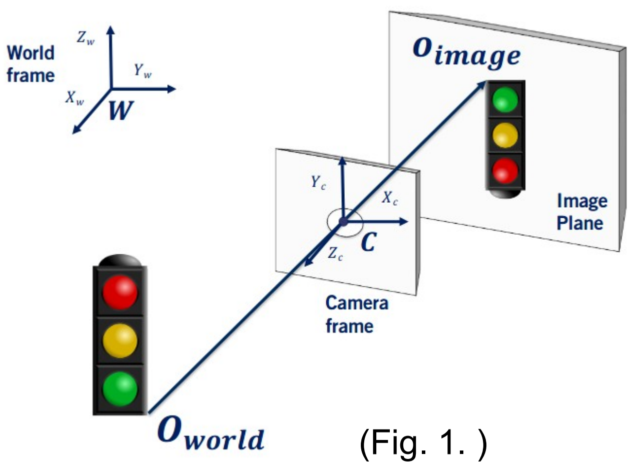
Keywords- Camera calibration; Lens distortion; Parameter estimation; Optimization; Camera modeling; Accuracy evaluation; Object localization; Computer vision.

I. INTRODUCTION

Camera calibration is the first step towards computational computer vision. Although some information concerning the measuring of scenes can be obtained by using uncalibrated cameras [1], calibration is essential when metric information is required. The use of precisely calibrated cameras makes the measurement of distances in areal world from their projections on the image plane possible . Some applications like Object localization: When considering various image points from different objects, the relative position among these objects can be easily determined. This has many possible applications such as in industrial part assembly and obstacle avoidance in robot navigation.

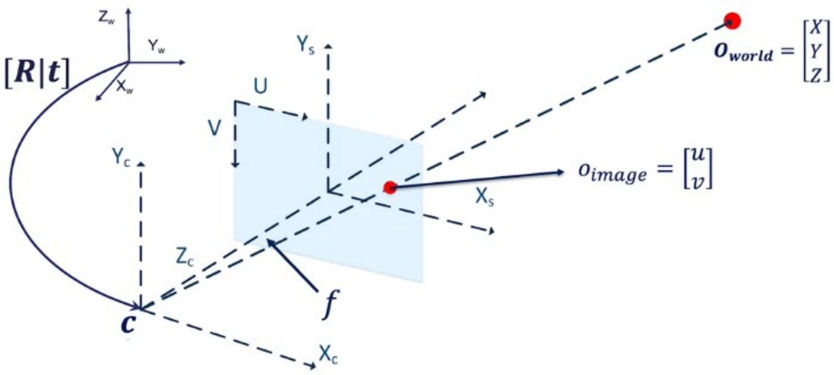
A. Camera Projection Geometry

How model camera geometry through the coordinates system transformation. This transformation used to project points from word frame to image frame. Model transforms using matrix algebra and apply them to 3D points to get 2D projection into image. Light travels from world on object through camera aperture to sensor surface and The result is flipped image of the object in the world (Fig. 1.).



(Fig. 1.)

The problem (Fig. 2.) : project point [X,Y,Z] from world to [u,v] image camera. [World → image (Real Camera)] ,so to avoid Confusion should Define virtual image plane in front of camera center to be convert from [World → image (Simplified Camera)].

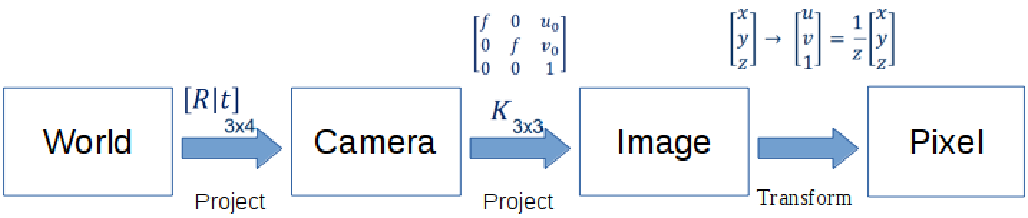


(Fig. 2.)

Define Translation vector and Rotation matrix to model any transformation world coordinate frame to another (world → camera) and we should not that the scale information is lost when point project from 3D world onto the 2D image plan.

B. Compute The Projection

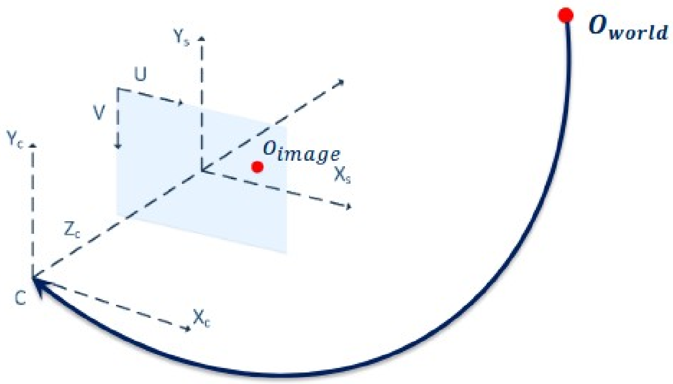
The homogeneous coordinates of point O in 3D space can be transformed to the camera plane, with the camera projection matrix P, which includes both extrinsic and extrinsic parameters. So the process of Projecting 3D point in world coordinate to pixel coordinate is done in 3 steps:(Fig.3)



(Fig. 3.)

First: World to Camera

we begin with the transformation from the world to the camera coordinate frame. This is performed using the rigid body transformation matrix T, which has R and little t in it. (Fig. 4.).

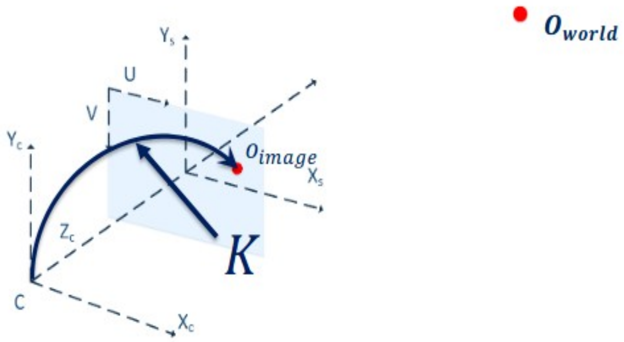


$$o_{camera} = [R|t]O_{world}$$

(Fig. 4.)

Second: Camera to image

transform camera coordinates to image coordinates. To perform this transformation, we define the matrix K as a three-by-three matrix. This matrix depends on camera intrinsic parameters, which means it depends on components internal to the camera such as the camera geometry and the camera lens characteristics(Fig. 5.).



(Fig. 5.)

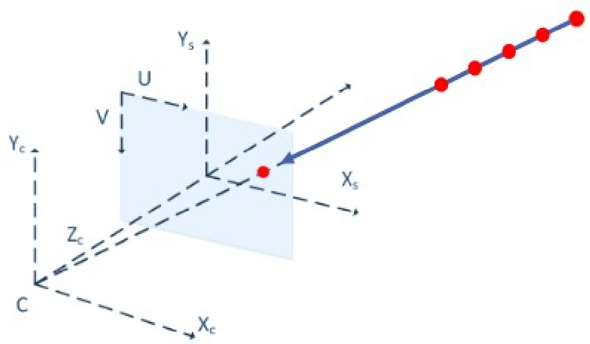
$$o_{image} = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix} o_{camera} = K o_{camera}$$

Third: Image to Pixel

The third step we convert from image plan to pixel coordinates to get the homogeneous coordinates in the image plan and multiply by Arbitrary scale(s) is useful when formulate the calibration problem(Fig. 6.).

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} su \\ sv \\ s \end{bmatrix}$$

Scale: s



(Fig. 6.)

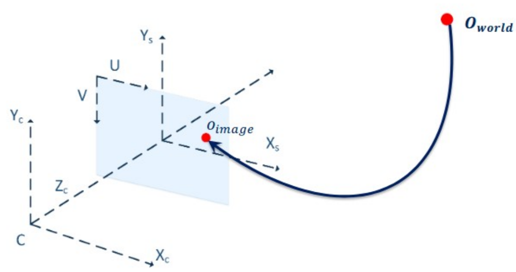
C. Camera Calibration:

Find unknown Intrinsic and Extrinsic camera parameter, Intrinsic Matrix are Parameters component internal to the camera such as (focal length, geometry and radial distortion coefficients of the lens) and use to project Point from camera coordinate to image plan. Extrinsic Matrix are Parameters of camera position [R| t].External to camera and specific to location camera in world coordinate frame. Used to transform 3D point from world to camera coordinate.(Fig. 7.)

$$o = PO = K[R|t]O$$

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$P = K\,[R|t]$$



(Fig. 7.)

For each unknown in this equation will get different problem in computer vision see the blew table.

Unknown in Equation	Problem
Oim ~ K ***[R| t]*** Ow	Relative Poss
Oim ~ K [R| t] ***Ow***	3D Reconstruction
Oim ~ K [R| t] Ow	Image Rendering
Oim ~ K ***[R| t] Ow***	Calibration (SFM)
Oim ~ ***K [R| t] Ow***	Self-Calibration + (SFM)

II. LITERATURE REVIEW

These different approaches can also be classified regarding the calibration method used to estimate the parameters of the camera model:

1-**Non-linear optimization techniques.** A calibrating technique becomes non-linear when any kind of lens imperfection is included in the camera model.

2- **Linear techniques** which compute the transformation matrix. These techniques use the least squares method to obtain a transformation matrix which relates 3Dpoints with their 2D projections. The advantage here is the simplicity of the model which consists in a
simple and rapid calibration.

3- **Two-step techniques.** These techniques use a linear optimization to compute some of the parameters and,as a second step, the rest of the parameters are computed iteratively.
The calibrating method depends on the model used to approximate the behavior of the camera. The linear mod-els, i.e. Hall and Faugeras–Toscani, use a least-squares technique to obtain the parameters of the model. How-ever, non-linear calibrating methods, as with Faugeras–Toscani with distortion, Tsai and Weng, use a two-stage technique. As a first stage, they carry out a linear approximation with the aim of obtaining an initial guess and then a further iterative algorithm is used to optimize the parameters. In this section, each calibrating method is explained detailing the equations and the algorithm used to calibrate the camera parameters.

III. METHODOLOGY

we will used Zhang's calibration method this calibration process is explained by a flowchart (Fig. 11.).

Step 1: Define real world coordinates with checkerboard pattern

World Coordinate System : Our world coordinates are fixed by this checkerboard pattern that is attached to a wall in the room. Our 3D points are the corners of the squares in the checkerboard. Any corner of the above board can be chosen to the origin of the world coordinate system. We use it because Checkerboard patterns are distinct and easy to detect in an image. Not only that, the corners of squares on the checkerboard are ideal for localizing them because they have sharp gradients in two directions. In addition, these corners are also related by the fact that they are at the intersection of checkerboard lines. All these facts are used to robustly locate the corners of the squares in a checkerboard pattern.

Step 2 : Capture multiple images of the checkerboard from different viewpoints

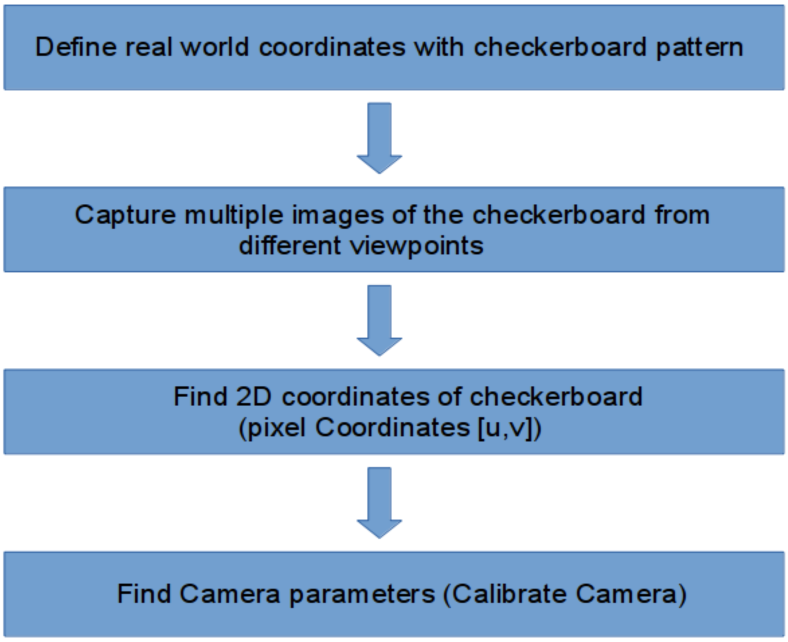
we keep the checkerboard static and take multiple images of the checkerboard by moving the camera. Alternatively, we can also keep the camera constant and photograph the checkerboard pattern at different orientations. The two situations are similar mathematically.

Step 3 : Find 2D coordinates of checkerboard

We now have multiple of images of the checkerboard. We also know the 3D location of points on the checkerboard in world coordinates. The last thing we need are the 2D pixel locations of these checkerboard corners in the images.

Step 4: Calibrate Camera

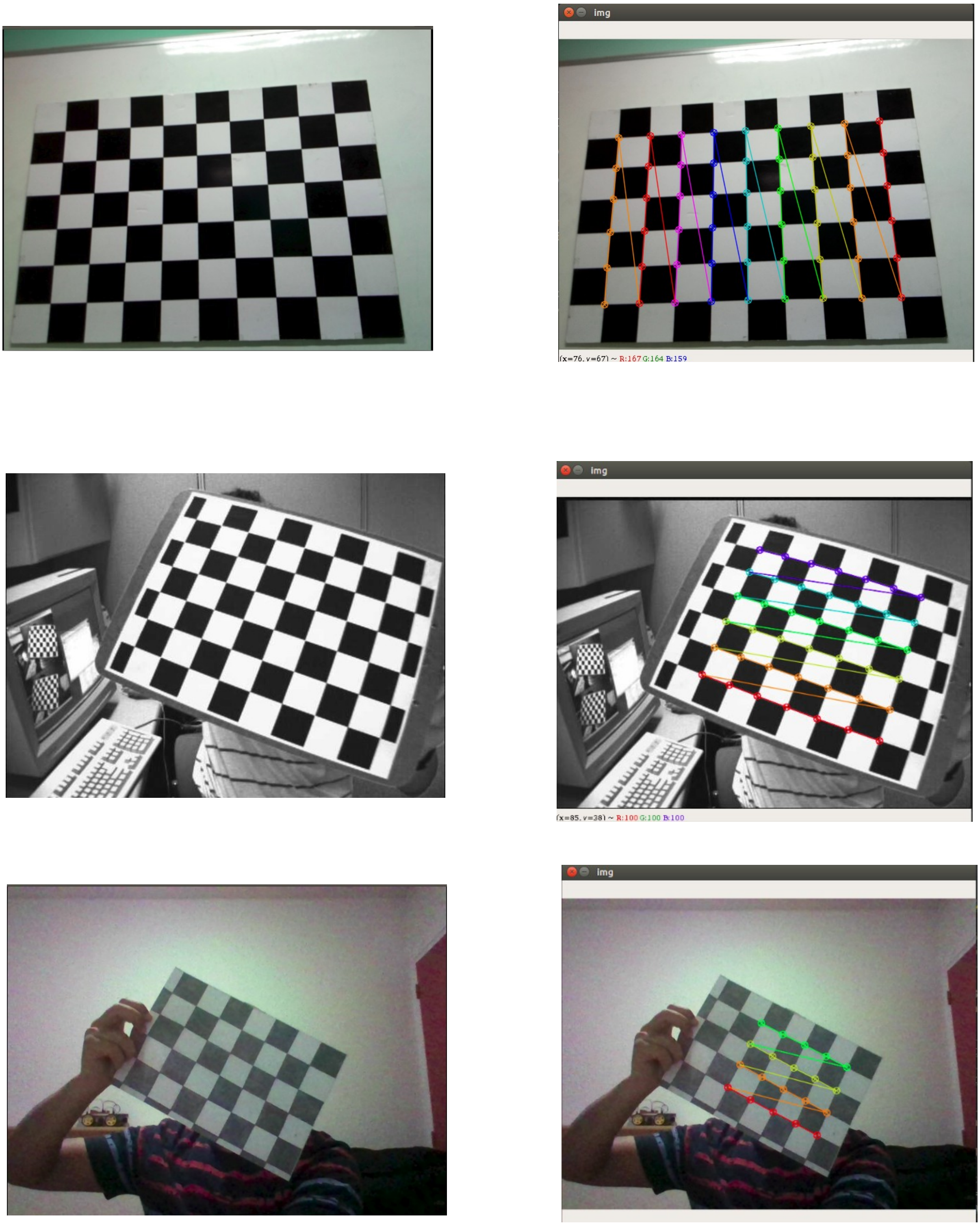
The final step of calibration is to pass the 3D points in world coordinates and their 2D locations in all images to OpenCV's `calibrateCamera` method. The implementation is based on a paper by Zhengyou Zhang. The math is a bit involved and requires a background in linear algebra.



(Fig. 8.)

IV. EXPERIMENTS & RESULTS

In this experiment we try to work on different dataset (Fig. 9.) and note the result in each one based on the number of images used each time and in The calculation is done by projecting the 3D chessboard points into the image plane using the final set of calibration parameters like camera matrix, distortion coefficients, rotation and translation vectors which return from *calibrate Camera* method (Fig. 10) where we used python and opencv as a tool to solve this problem.



(Fig. 9.)

Dataset	
A (13 images)	
camera matrix	**[534.070886226348, 0.0, 341.53407107253565] [0.0, 534.1191479816231, 232.94565221109772] [0.0, 0.0, 1.0]**
distortion coefficients	**[-0.2929716213439848, 0.10770688662177534, 0.0013103848957804632, -3.110229774446826e-05, 0.043479914222931494]**
B(14 images)	
camera matrix	**[504.1018772671398, 0.0, 313.0928084691982] [0.0, 503.47888150855147, 243.49793518485268] [0.0, 0.0, 1.0]**
distortion coefficients	**[0.21523588287645243, -0.49167016141860237, 0.0009599699172590149, -0.002511878037236217, 0.3004766601232317]**
C(14 images)	
camera matrix	**[753.6653999277385, 0.0, 234.73666665774672] [0.0, 770.4060989880483, 102.10265598230573] [0.0, 0.0, 1.0]**
distortion coefficients	**[0.030827764337020478, 0.8525224676052454, - 0.0565775602239213, -0.020622171565229858, - 3.2246910362470955]**

(Fig. 10.)

CONCLUSION

Computer vision is a key branch of artificial intelligence, aiming at understanding the surrounding environment from the visual information captured by cameras. 3D computer vision is a subfield of computer vision that focuses on extracting 3D metric information from 2D images. This is a very challenging task since what we have in an image is the projection of a 3D scene, whose depth information is lost during image formation (image projection), this paper work to get calibrated camera to have all the information (parameters or coefficients) about the camera required to determine an accurate relationship between a 3D point in the real world and its corresponding 2D projection (pixel) in the image captured by that. And test the algorithm on 3 different data set and extract the camera matrix ,distortion coefficients and rotation and translation vectors.

REFERENCES

[1]https://d3c33hcgiwev3.cloudfront.net/cDumlEngEemplgpfqc6zSA_62a8e5331ed94b4180ec28eebe430ad2_C3M1L2---Camera-Calibration.pdf?Expires=1585180800&Signature=iFoxTgyLSM0U16X66dJyPgfBrblu4Tp6~t2kfhIG5TBgGTFVaLNy2vMKm4Ja8zQfCUwGnGN86gh2~nbNRk0gZ6xLSbKcA6jAFReEnwQ-qEBcbniJ7qke9VwN5z2~~IIQSJhjlJYc5J8lsxS66hLj2qnSJJcpyz~CKrdILRBD3C8_&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A

[2]https://d3c33hcgiwev3.cloudfront.net/YIQ6_kngEemo6BKcywtCxg_521e910db1a7419caea7e19c275a87b7_C3M1L1P2---Camera-Projective-Geometry.pdf?Expires=1585180800&Signature=kUetJrwsjhC8zLl4VcsJ8qGnCGQQwuU0pF8oqVKqUH4E3f4pdpGYuOW3PRAm6q9WWd6XJW2LjBAxxpj001haIhU7eMBYAPj7R1thARnLS35~-IK-SNtMl9B8oldMBfRj7MIhpo3SJMd0jdntFUVqLaF-iuiNoLR9~n5nTeqU3c_&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A

[3] https://www.learnopencv.com/camera-calibration-using-opencv/

[4]http://citeseerx.ist.psu.edu/viewdoc/downloaddoi=10.1.1.142.4676&rep=rep1&type=pdf

[5] https://docs.opencv.org/3.4.3/dc/dbb/tutorial_py_calibration.html