Homework 1 : AutoCalib

Advait Patole
Masters of Engineering in Robotics
University of Maryland, College Park
Email: apatole@umd.edu

***Abstract*—The field of computer vision has been a great topic of research for many researchers.The problem of camera calibration is a well known problem that can be solved using various image processing algorithms.The paper talks about camera calibration using the method presented by Zhengyou Zhang of Microsoft in his paper, which is regarded as one of the hallmark papers in camera calibration.**

I. INTRODUCTION

In this paper I present my approach of camera calibration as presented in the paper of Zhengyou Zhang of Microsoft in his paper.This paper talks about the implementation of camera calibration by using different images of chessboard taken from different view to calibrate intrinsic, extrinsic parameters and distortion coefficients.

A. Representing the points

For camera calibration image and world points are taken accordingly I have used cv2.findChessboardCorners to find the corners of the chessboard.To refine the pixel coordinates of the corners detected I have made use of cv2.cornerSubPix function.Total 54 corners are detected on the chessboard which are our image points.The world points are considered to be a grid of total 54 points with 9 rows and 6 columns.The size of each square of grid is 21.5mm.The coordinates of the grid start from (1,1) till the total number of rows and columns.The following image shows how the image points on the chessboard image.I have used 13 images of the chessboard taken from different angle for calibration.The above procedure is repeated for all these images and the images points for each image are stored.The homography matrix was calculated for each image.The four corner points were used since only 4 points are required to get the homography.

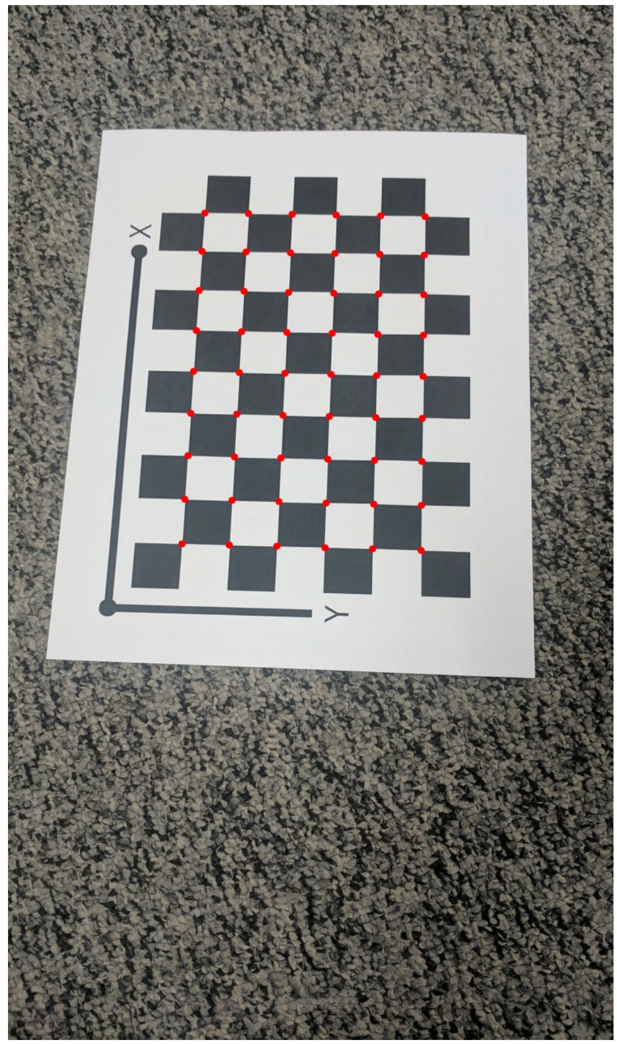

Fig. 1: Representation of points on the chessboard

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

Fig. 2: K matrix

B. Finding approximate K or camera intrinsic matrix

To find the camera intrinsic matrix I have used the homography matrix that was estimated in the previous section.I followed the closed form solution from section 3.1 of Zhang's paper to find out the camera intrinsic matrix.Since the estimated rotation matrix does not satisfy the properties of a rotation matrix as described in the paper I have also used their approach that was presented in appendix C. The initial estimate of intrinsic matrix is

$$\begin{bmatrix} 2037.52 & -0.0004584 & 774.63 \\ 0 & 2022.67 & 1358.88 \\ 0 & 0 & 1 \end{bmatrix}$$

Fig. 3: Initial estimate of K matrix

C. Estimate R and t or camera extrinsics

The camera extrinsic parameters were calculated using the section 3.1 of the paper using the homographies and the camera intrinsic matrix.The camera extrinsic matrix is given by [R—t].

D. Initial camera distortion coefficients

The assumption in calculating the initial estimate is that there is no distortion present in the image at all. We thus assume the value of kc to be [0,0].

$$\begin{bmatrix} 0 & 0 \end{bmatrix}$$

Fig. 4: Initial estimate of $k_c matrix$

E. Optimizing the error

So far we have seen from the previous sections that we have the initial estimate of K, R, t, Ks.Now we have to find out the optimal camera parameters such that re projected error between the actual image points and the projected world points is minimum.In order to get these parameters I optimized the error as presented in section 3.3 of the paper.While calculating the optimal camera parameters we have to consider the distortion coefficients as well because initially we have taken the distortion as (0,0) which will change after optimization.The error was calculated using np.linalg.norm which calculates the L2 norm of the projected world points and the image points.For optimizing the error I have used scipy.optimize.least squares is solved with the Levenberg-Marquardt Algorithm.After optimizing I got the optimal camera parameters which I used to plot the projected points on the chessboard.

$$\begin{bmatrix} 2037.52 & -0.0004584 & 774.63 \\ 0 & 2022.674 & 1358.87 \\ 0 & 0 & 1 \end{bmatrix}$$

Fig. 5: K matrix after optimization

$$\begin{bmatrix} 0.0032285 & -0.00322957 \end{bmatrix}$$

Fig. 6: kc matrix after optimization

II. Discussion and Conclusion

The non linear optimization was used to estimate the optimal values of A, R, t and kc such that the re-projection error is reduced.The error that was calculated was the euclidean distance between the re-projected world points and the image points. There is very less difference that is observed between the initial camera intrinsic matrix and the camera intrinsic matrix that was obtained after non-linear optimization.The initial re-projection error was 1.688 and after optimization the re-projection error was reduced to 1.64 for each image.The distortion coefficients initially was considered to be (0,0) implying there was no distortion considered after optimization the distortion was (0.0032285, -0.00322957).This project has helped in better understanding the camera calibration and why it is essential.The outputs of the projected points don't differ much than the original projected points without optimization.

III. Output

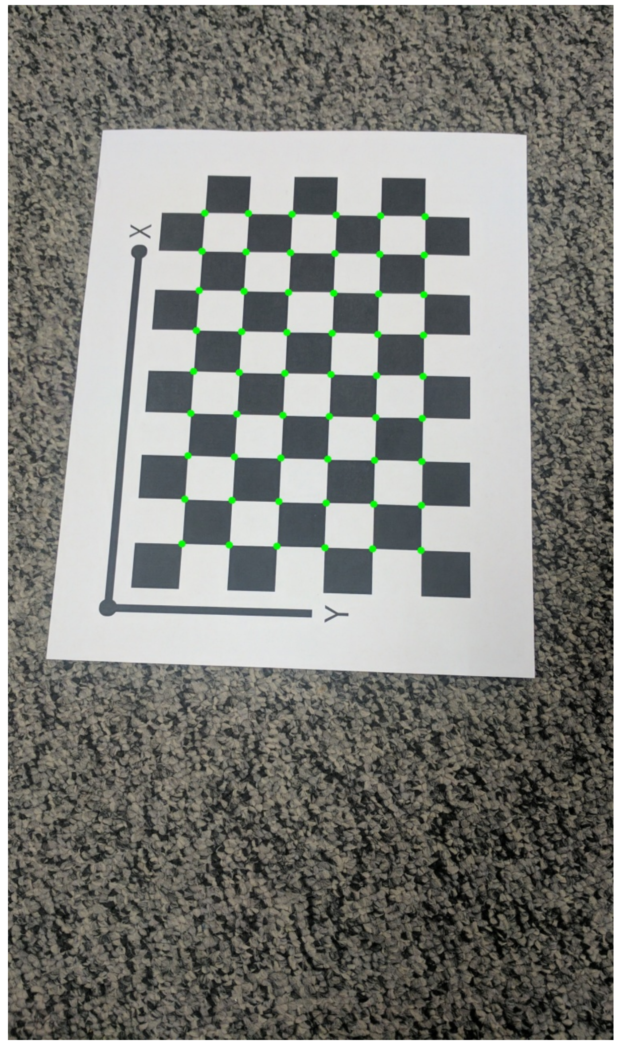

Fig. 7: Output for image 1

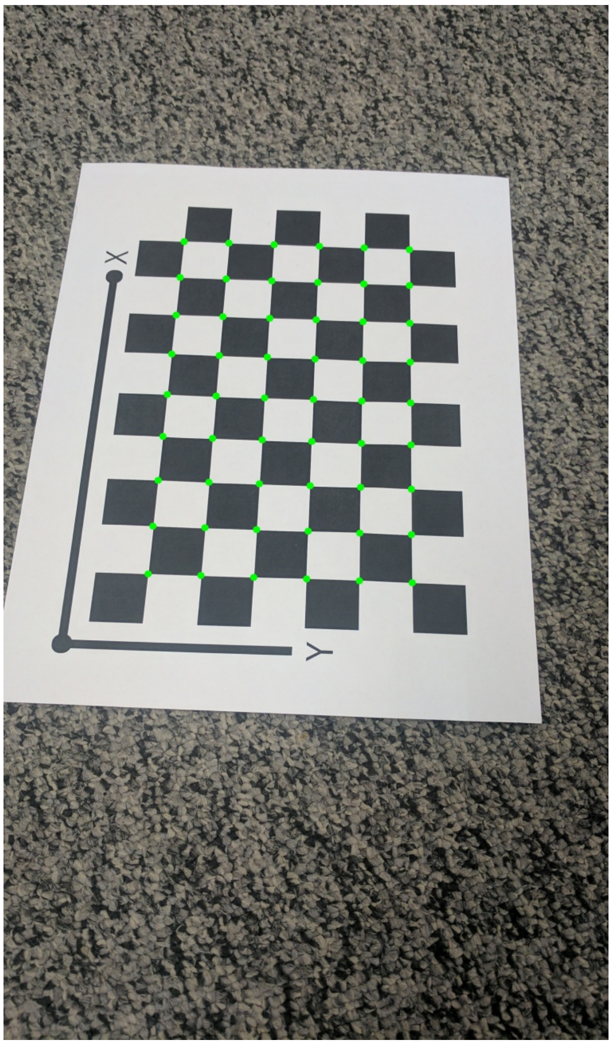

Fig. 8: Output for image 2

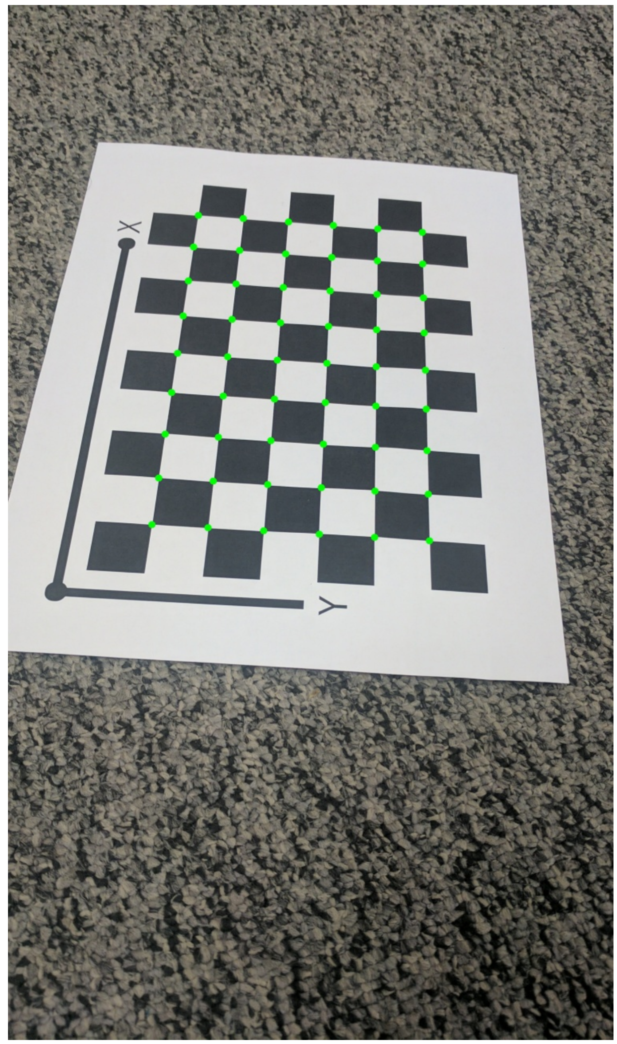

Fig. 9: Output for image 3

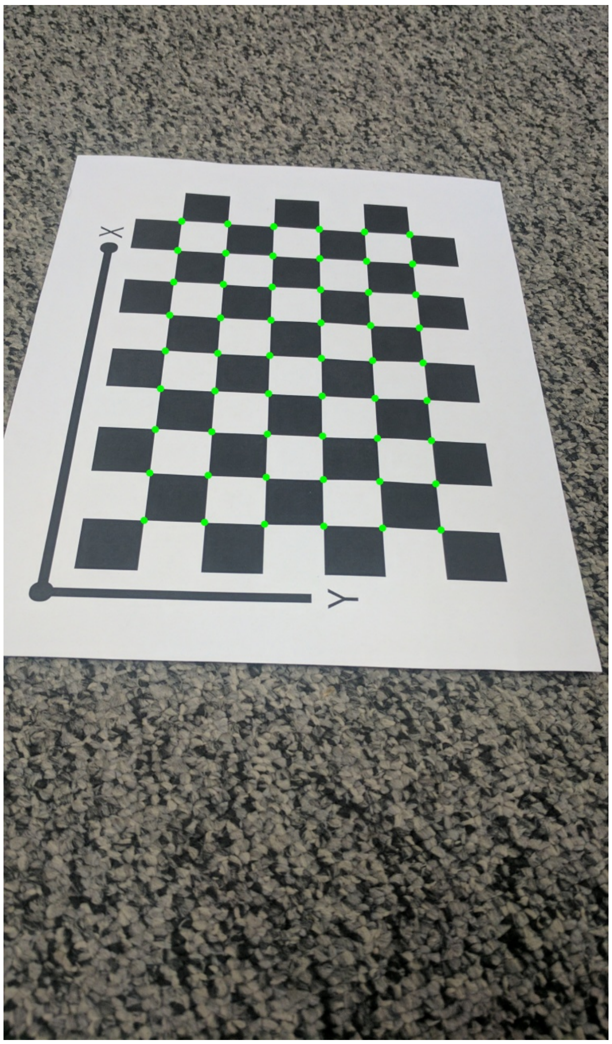

Fig. 10: Output for image 4

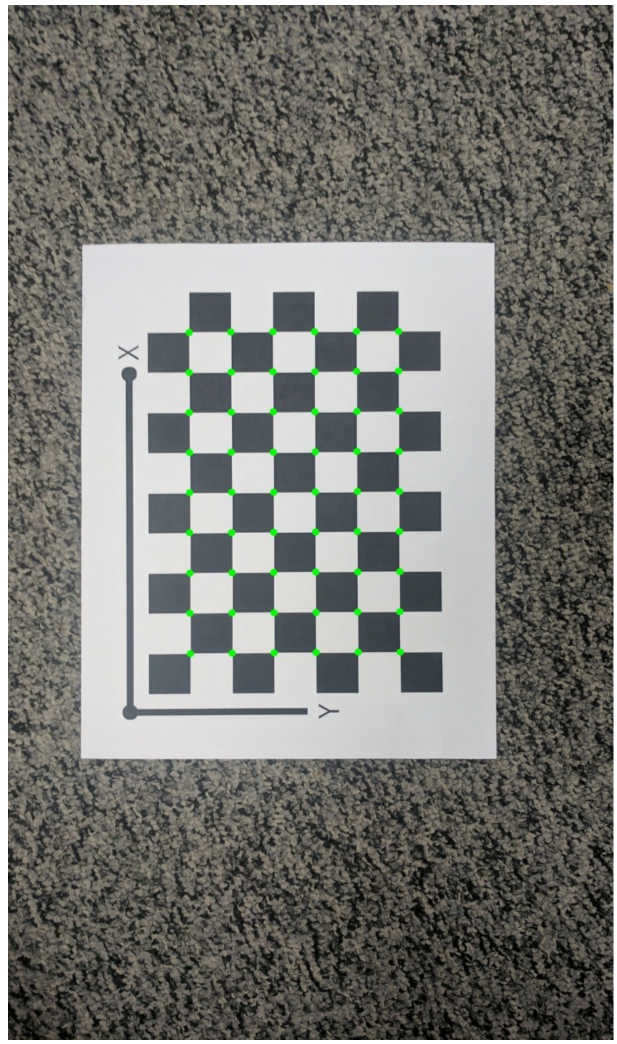

Fig. 11: Output for image 5

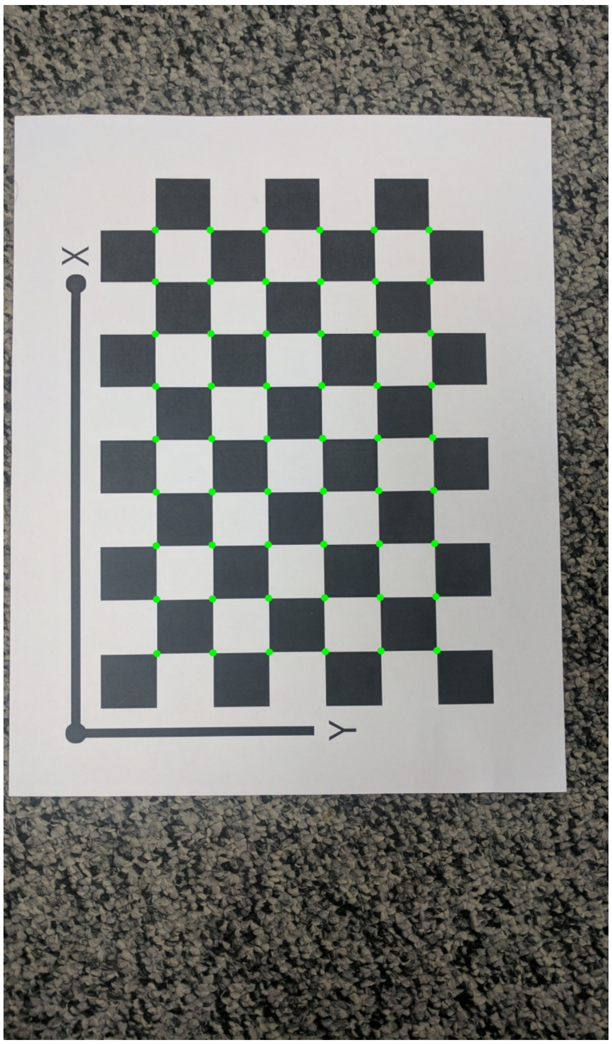

Fig. 12: Output for image 6

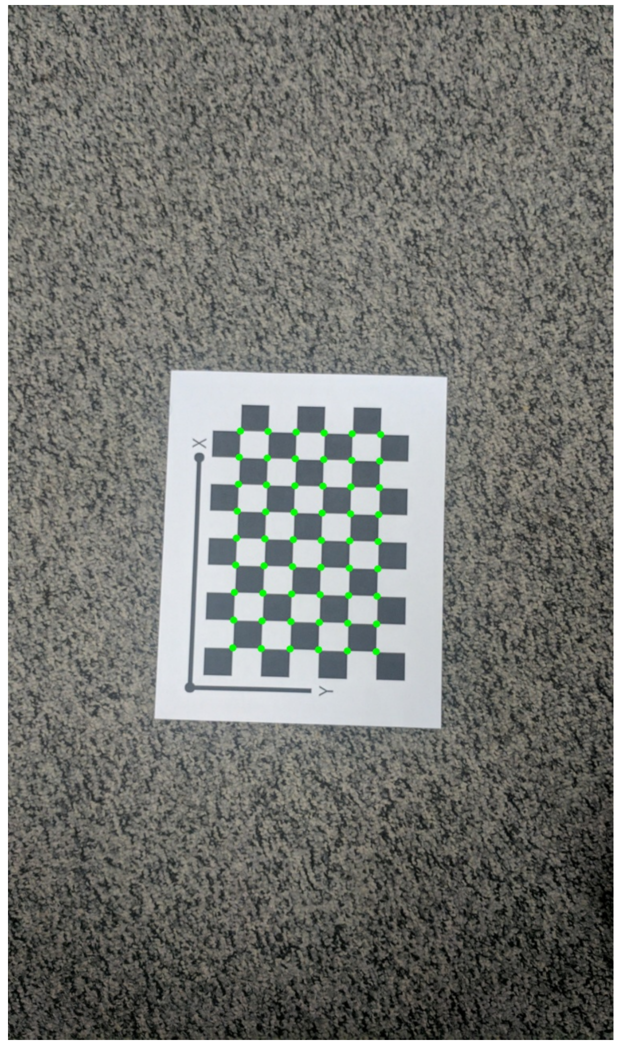

Fig. 13: Output for image 7

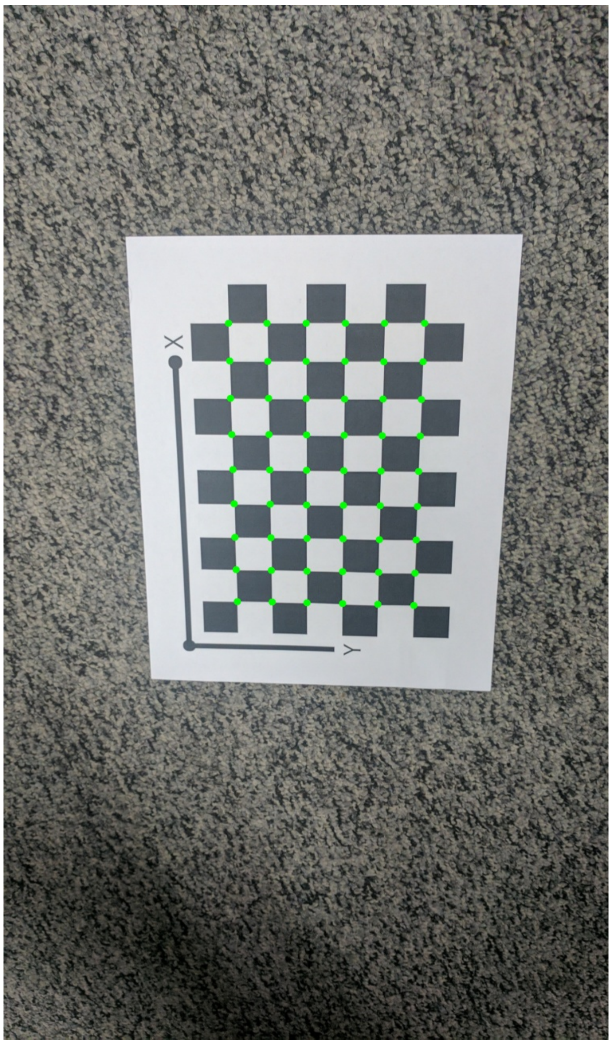

Fig. 14: Output for image 8

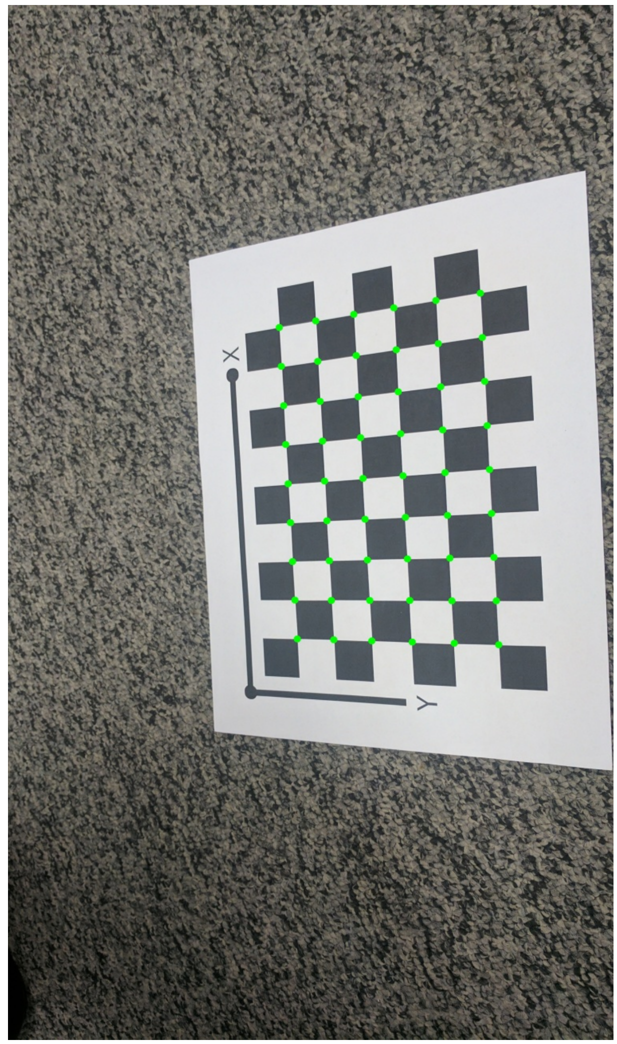

Fig. 15: Output for image 9

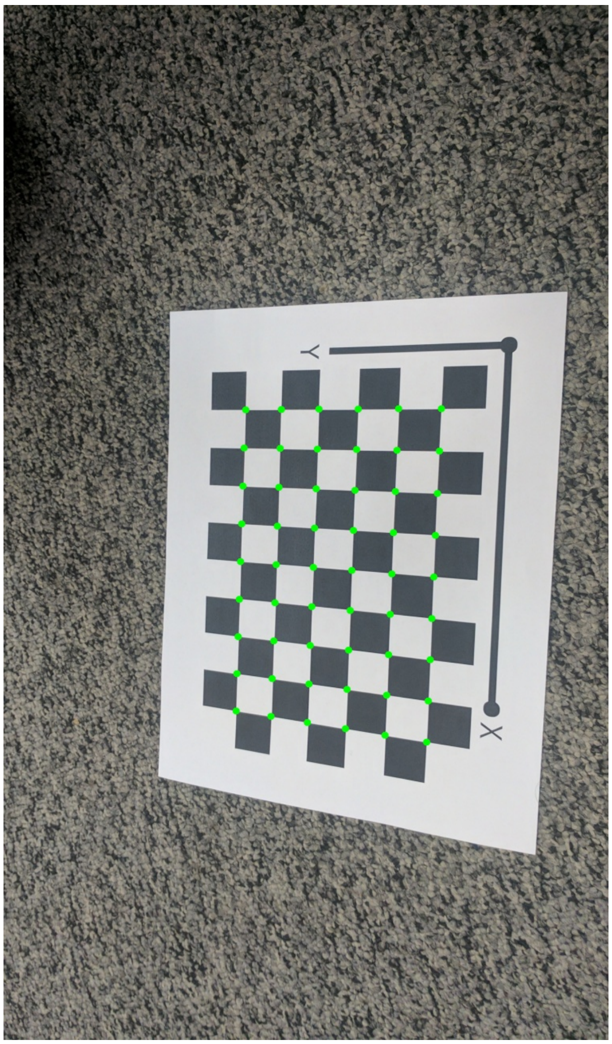

Fig. 16: Output for image 10

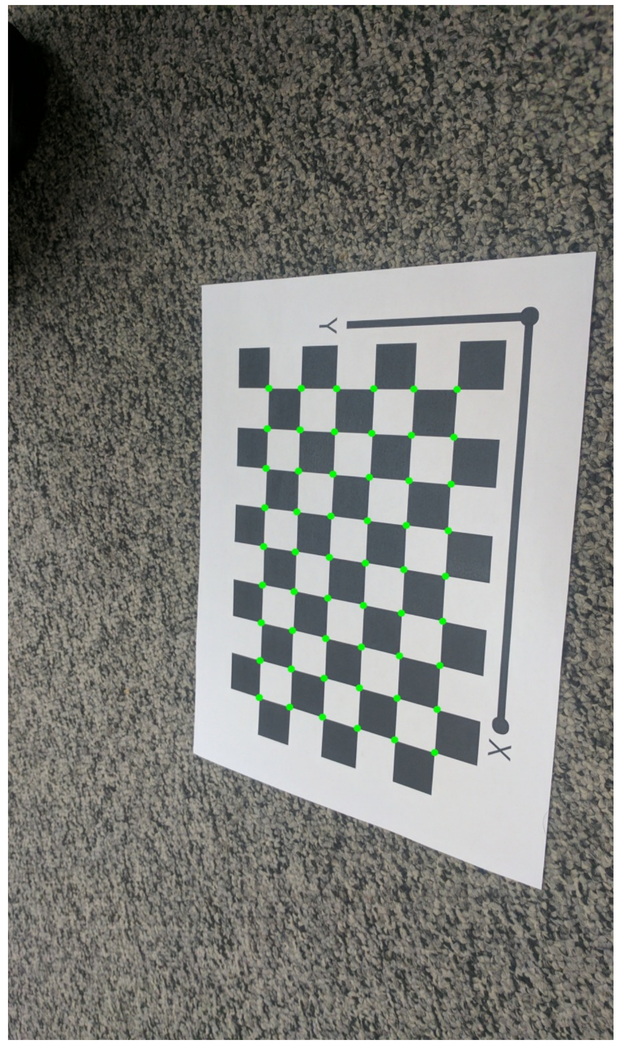

Fig. 17: Output for image 11

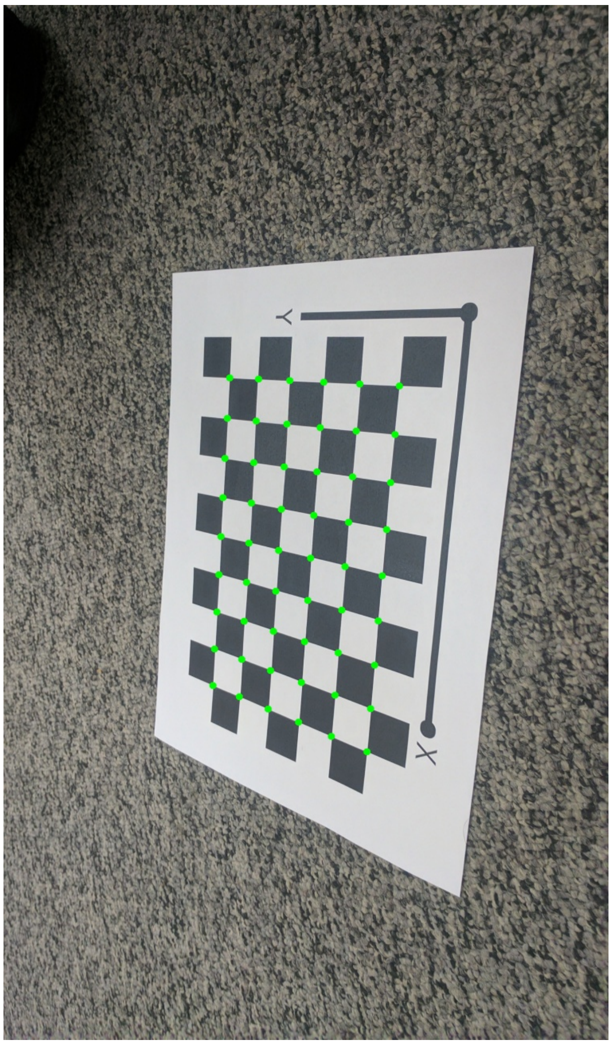

Fig. 18: Output for image 12

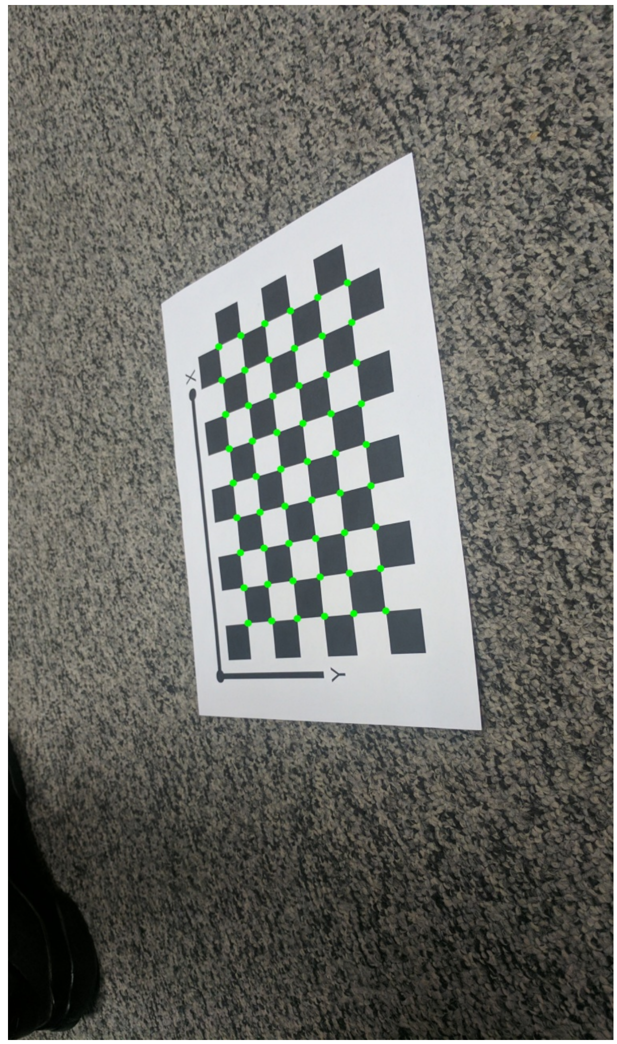

Fig. 19: Output for image 13