Pablo Medrano
A20410758

cs512

Homework 4

1. Problem statement

The problem chosen to solve in this assignment is non-coplanar calibration.

1.1. Write a program to extract feature points from the calibration target and show them on the image using openCV functions.

1.2. Write a program that does the calibration process from a text file with 3D points and its correspondent 2D points. The program should display the intrinsic and extrinsic parameters as well as the mean square error between the known and computed position of the image points.

1.3. Implement the RANSAC algorithm for robust estimation.

2. Proposed solution

2.1.

Using the functions suggested by the professor it is possible to extract the points and after that they are written in a file.

2.2.

The camera calibration and the RANSAC algorithm are both implemented in the same program.

For the camera calibration process, the matrix M is computed getting the last column of the vector V after computing the SVD of the matrix AtA.

The matrix A corresponds to the matrix of equations given by the 3DH points. I am using all the points so the matrix A shape is (328*2, 12) since there are 328 points and each one gives two equations.

Once we have the matrix M, all the parameters can be computed following the non-planar calibration equations

2.3.

For the RANSAC algorithm, we have to do the next k times:

1. Compute the matrix M with n random points.
2. Compute the estimated image points using the matrix M.
3. Compute the distance between the 2D estimated points and the real ones.
4. Compute the t as 1.5*median of the values in the step 3.
5. Find all inliers in the data with distances smaller than t.
6. Recompute matrix M with all the inliers.
7. Compute MSE for all the points using the matrix M in step 6.

To compute the matrix M and the distance in all these loops I use the functions that I created for the second part.

After doing this k times we have some matrices M with the corresponding MSE of each of them. We choose the one with the smaller MSE to be the best one and its corresponding matrix M and its parameters.

3. Implementation details

The implementation of the first part was easier but with the rest I encountered some difficulties that I explain next.

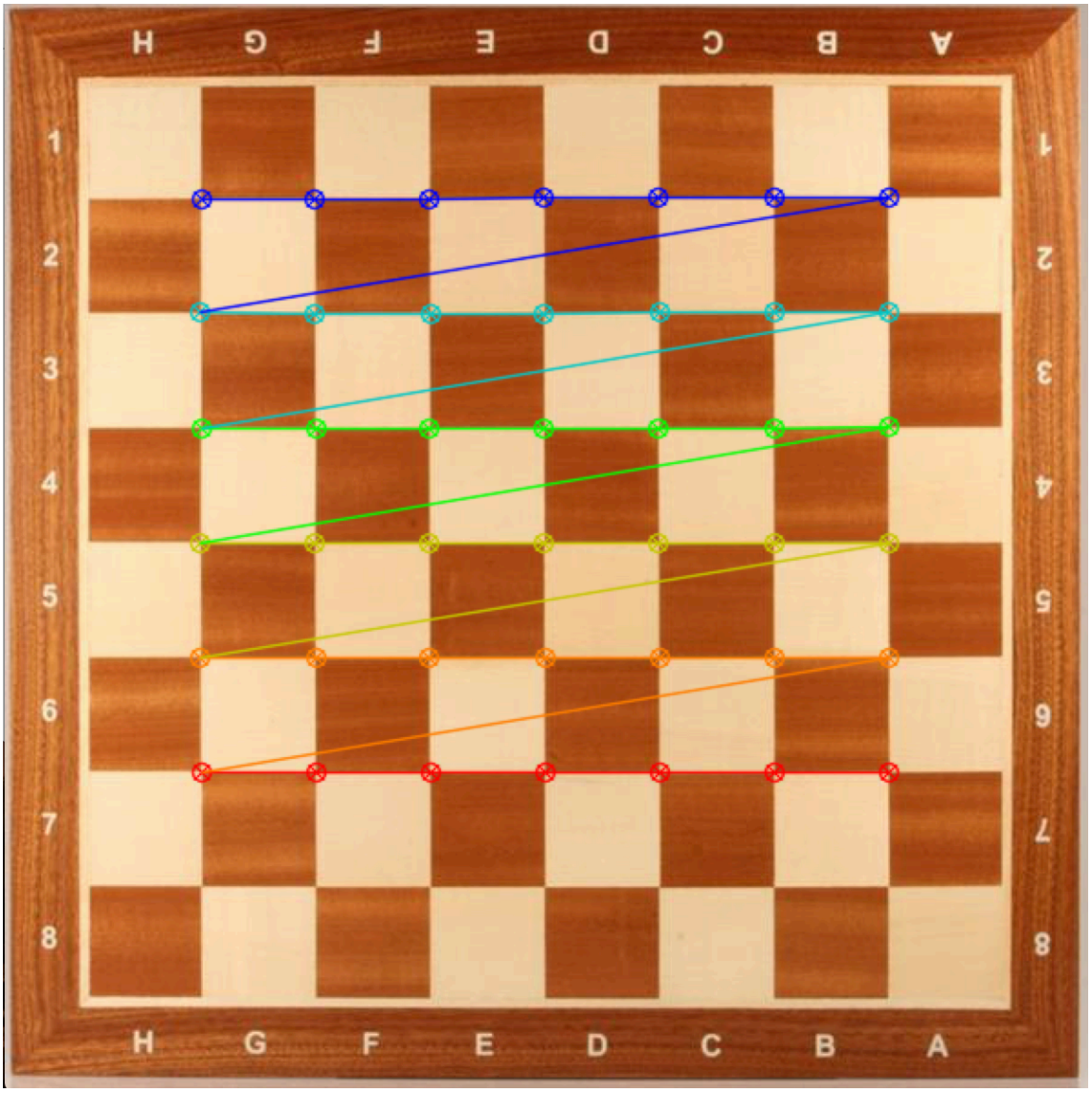
I programmed the equations and everything as I have explained in the previous but I do not know if I am doing something wrong because using the data provided by the professor I do not get the same values for the parameters and the MSE is big.

As I say I cannot find my error if there is one because I computed the equations just as the professor explained in class.

Since I use the functions from the part two in the RANSAC algorithm the parameters I get are not the same as the ones the professor posted online but the algorithm itself should be right.
As parameter I used n = 9, just some more then the 6 needed, and for the loops I realized that doing more did not improve the results so I just left 3. However, this could be wrong in part because of the parameters.

4. Results and discussion

The points marked in the image in the first question are these:



A text file with the points is also obtained with the function.

In the second and third parts the result is the matrix M and the intrinsic and extrinsic parameters and the MSE associated to them.
As I said in the implementation part the parameter I obtained are different from the ones the professor posted and so I get a big MSE of about 150000.

I have not tried the program with other images because I do not think my program is completely right but I have not being able to fix in time to try something else

5. References

https://docs.opencv.org/3.0-beta/doc/tutorials/calib3d/camera_calibration/camera_calibration.html

https://docs.opencv.org/3.0-beta/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html

https://docs.opencv.org/3.0-beta/doc/tutorials/features2d/trackingmotion/corner_subpixeles/corner_subpixeles.html

https://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html