Spring 2023

MIDTERM

Perception for autonomous robots

March 16 2023

Instructors:
Dr. Samer Charifa

Student:
Nishant Awdeshkumar Pandey

Course code:
ENPM 673

Contents

1 Question 1

- 1.1 The minimum number of matching points needed to get calibrate the image given is ”6”. As the camera projection matrix has 11 degrees of freedom and each matching point gives 2 equations. So to get the value of 11 unknowns one would need at least 6 points that will give 12 equations to solve for the 11 variables.
- 1.2 Pipeline to calibrate the camera for the image given:
  - Image Acquisition: Get the Image. The calibration target should contain a pattern of known geometry such as a chessboard or a circle grid. In this case, the image has such patterns so if needed we can trim it to focus on the desired area.
  - Feature Extraction: Extract features from the target using computer vision algorithms to get the corners of the white quadrilateral in the image.
  - Feature matching: Match the extracted features across the calibration target images.
  - Use the matched features to estimate the intrinsic and extrinsic camera parameters. Intrinsic parameters include the focal length, principal point, and lens distortion coefficients. Extrinsic parameters include the rotation and translation between the camera and the calibration target.
  - Validation: Validate the accuracy of the calibration by projecting a set of known 3D points onto the image plane using the estimated camera parameters and comparing them to their corresponding 2D image points. This can be done using reprojection errors.
  - Refinement: Refine the estimated camera parameters using optimization algorithms.
- 1.3 To find the intrinsic matrix :
  - Normalize and find homogenous coordinates for image and world points.
  - Construct a matrix A that relates the 3D world coordinates to the 2D image coordinates for each calibration point. The matrix A can be constructed as follows:
    For each calibration point i, construct a row of A as follows:
    $$[X_i \ Y_i \ Z_i \ 1 \ 0 \ 0 \ 0 \ 0 \ -x_iX_i \ -x_iY_i \ -x_iZ_i \ -x_i$$
    $$0 \ 0 \ 0 \ 0 \ X_i \ Y_i \ Z_i \ 1 \ -y_iX_i \ -y_iY_i \ -y_iZ_i \ -y_i]$$
    where (X_i, Y_i, Z_i) are the 3D coordinates of the ith calibration point, and (x_i, y_i) are the corresponding 2D image coordinates.
  - Use the SVD (singular value decomposition) algorithm to decompose the matrix A into its singular values and singular vectors. This will give the projection matrix P
  - Then decompose the matrix P using the Gram-Schmidt method. To get K, R and T matrices.
- 1.4 Finding P:
  - Use the SVD (singular value decomposition) algorithm to decompose the matrix A into its singular values and singular vectors. This will give the projection matrix P.
    $$U, s, vh = SVD(P)$$
  - The projection matrix P can be found as the last column of the matrix Vh.
- 1.5 Decompose P using the Gram-Schmidt method and then find the reprojection error. Reprojection error is a measure of the difference between the observed image points and the projected image points using the estimated camera parameters. First using the projection matrix the 3D points are projected onto the image plane to obtain their corresponding 2D image points. Then the reprojection error is calculated by finding the root mean square of the error.

1.1 Results for question 1

These are obtained results of the first question:

The projection matrix:

$$\begin{bmatrix} -0.36699221 & 0.64298462 & -0.15292777 & 0.03538725 \\ 0.01241879 & 0.04701708 & 0.04729546 & 0.02011119 \\ 0.04457098 & -0.17798548 & -0.06557937 & -0.61990009 \end{bmatrix}$$

Normalize

Intrinsic matrix K for question 1:

$$\begin{bmatrix} 28.52667902 & -50.73007229 & 11.21429001 \\ 0. & 9.32313224 & 7.15663313 \\ 0. & 0. & 1. \end{bmatrix}$$

Not Accurate

Rotation matrix R:

$$\begin{bmatrix} -0.99214598 & -0.07983684 & -0.09629351 \\ 0.03357361 & 0.5716076 & -0.81983996 \\ 0.12049554 & -0.81663384 & -0.56443777 \end{bmatrix}$$

Translation vector t:

$$\begin{bmatrix} 25.2135486 & 3.57298587 & 4.17466859 \end{bmatrix}$$

Reprojection error for each point:

$$\begin{bmatrix} 0.99989394 \\ 0.58178868 \\ 0.49409472 \\ 0.89923273 \\ 0.87738258 \\ 1.80608576 \\ 1.08672334 \\ 1.51857191 \end{bmatrix}$$

2 Question 2

Pipeline for finding the reprojection error and intrinsic matrix of the camera:

- Load the image folder.
- Initialize the row and column squares to be 9 and 6 respectively.
- Find corners for each image using in-built functions.
- Display corners.
- Store the corner value and find the reprojection error using these values using the inbuilt function.
- Find the intrinsic matrix by calibrating the camera.
- There are several ways to improve the accuracy of the camera's intrinsic(k) matrix:
  - Increase the number of images used for calibration: The more images used, the more accurate the calibration will be.

- Use a larger board: A larger board with more squares can provide more calibration information and improve the accuracy of the calibration.
- Use different orientations and positions: Capturing images of the board from different orientations and positions can help to capture a more diverse set of calibration information.
- Ensure that the board is flat and level: The board should be flat and level when capturing images, as any warping or distortion can lead to errors in the calibration.
- Use more accurate measurements for the size of the squares: If the size of the squares is measured inaccurately, it can lead to errors in the calibration.

2.1 Result for question 2

Image 0: Reprojection error: 0.11981903660519842
Image 1: Reprojection error: 0.26101283783489354
Image 2: Reprojection error: 0.40943910148025753
Image 3: Reprojection error: 0.5417584763336714
Image 4: Reprojection error: 0.22194391542138783
Image 5: Reprojection error: 0.3536628262209409
Image 6: Reprojection error: 0.05196775866308302
Image 7: Reprojection error: 0.22466896403009337
Image 8: Reprojection error: 0.48103441090042875
Image 9: Reprojection error: 0.40418099118168593
Image 10: Reprojection error: 0.48104652963205696
Image 11: Reprojection error: 0.5136874552439844
Image 12: Reprojection error: 0.4297445332479452
Intrinsic Camera matrix:

$$\begin{bmatrix} 2.23165807e+03 & 0.00000000e+00 & 7.78116041e+02 \\ 0.00000000e+00 & 2.45420808e+03 & 1.32346726e+03 \\ 0.00000000e+00 & 0.00000000e+00 & 1.00000000e+00 \end{bmatrix}$$

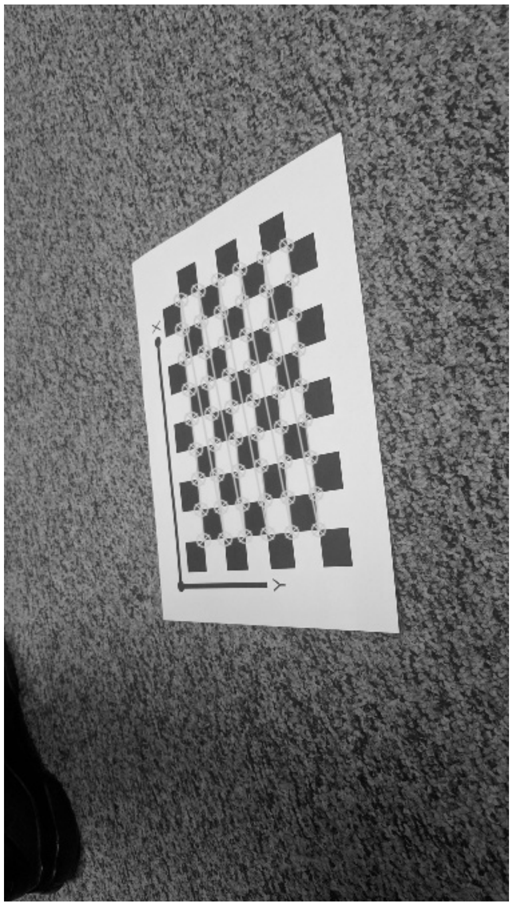

Figure 1: Corners for one of the images

3 Problems Encountered

- I was not aware of the normalization concept for question one. Which I included later on to get better output.
- I was confused between approaches because there are many ways to K matrix, however, the sequence of the question gives a slight hint which helps.
- In the second question no problems were encountered.

4 References

- Normalization- `https://www.cc.gatech.edu/classes/AY2016/cs4476_fall/results/proj3/html/agartia3/index.html` `https://intensity-normalization.readthedocs.io/en/latest/algorithm.html`
- A matrix- `https://towardsdatascience.com/camera-calibration-with-example-in-python-5147e945cdeb`
- gram-schmidt - `http://homepages.math.uic.edu/~jan/mcs507f13/gramschmidt.py`
- for second question open cv documentation was referred to.