ENPM 673: Perception for Autonomous Robots

Project 3 Report

Adithya Gaurav Singh

Calibration

Calibration aspect of this project was to compute fundamental and essential matrices given the correspondence points.

Following steps were followed:

- Compute the keypoints in both images and their descriptors.This was done using OpenCV's built in SIFT feature detector.
- Match the keypoints between both the images using the descriptors. For this OpenCV provides the FLANN matcher method
- Once the features are matched, inliers need to be filtered since not all matches are good matches.
- Inlier filtering can be done following the RANSAC technique using Fundamental Matrix coplanarity constraint as the criteria for threshold.
- At random select 8 match pairs and compute the Fundamental matrix using the SVD decomposition. For every such fundamental matrix calculated, iterate through every match pair and compute $X_2^TFX_1$. If the matrix multiplication values to a less than a set threshold, if so increment score by 1. Keep record of the maximum score and save the inliers for the highest score.
- With the highest score inliers, compute the final fundamental matrix.
- Essential matrix can be calculated from the fundamental matrix if the camera instrinsics are known. Use the relation $E = K^TFK$ to solve for Essential matrix where K is the intrinsic matrix.
- To find out camera, pose essential matrix needs to be decomposed to obtain R and t matrices that specify the relative camera poses.
- Perform SVD decomposition on the Essential matrix and obtain U and V matrices. The obtain following 4 solutions

  $(C_1, R_1) = U(:,:2)\ ,\ UWV^T$

  $(C_2,R_2) = -U(:,:2)\ ,\ UWV^T$

  $(C_3,R_3) = -U(:,:2)\ ,\ UW^TV^T$

  $(C_4,R_4) = U(:,:2)\ ,\ UW^TV^T$

  Where W = [0 -1 0]
  [1 0 0]
  [0 0 1]
- Using 3D triangulation and chierality condition identify the correct solution of the camera pose. The key is to note that traingualted points in 3D should be in front of both the cameras.

Rectification

Once the Fundamental and Essential matrices are computed the calibration process is completed. Now it is to be noticed that the image pair can be shifted in vertical direction which would make it harder to compute depth for each image row. To optimize the process stereo rectification is performed which warps both images to have same inflection. Hence, the matched keypoints are aligned row-wise. Following steps for Stereo Rectification:

- Find epipolar lines for both the images using the fundamental matrix. This can be performed by using OpenCV built in function cv2.computeCorrespondEpilines. The basic idea is to use fundamental matrix coplanarity constraint $X_2^T F X_1 = 0$. In this expression, FX_1 is substituted for epipole line equation. On plotting these lines it can be seen that they are not aligned horizontally
- Using OpenCV stereoRectifyUncalibrated method, a homography can be calculated for each image pair using the fundamental matrix and inlier keypoints. This Homography warps each image to a perspective where the epilines of each of them area aligned.

Correspondence

Once the image pair is aligned, each row contains corresponding matched features. Now it becomes easier to compute disparity map for the pair as all keypoints are now aligned row wise. This implies every image point can be searched along corresponding 1D row on the paired image.

Following steps for computing disparity:

- For each row in left image matrix section out a window of 30x30
- In same row in the right image slide this block at every pixel location and compute sum of absolute differences in both the blocks. Keep track of the least score and return its index.
- Pixel value for each coordinate in the disparity map is the difference between the current index and index of best match found in right image

Compute Depth Image

Depth image computation from the disparity map is fairly straight forward if baseline and focal length is known.

Follwoing steps were followed to compute the depth image:

- Iterate over each pixel value in the disparity map
- Create an empty 1 channel depth map.
- Each pixel value of depth map can be calculated as follows:

$$Depth[x][y] \ = \ \frac{Baseline * focal\ length}{disparity[x][y]}$$

Problems faced:

- Calculation of camera pose from essential matrix is not trivial. Extra steps for triangulation of 2D points to 3D points needed to be performed to get correct estimate of Camera pose.
- Correspondence is an extremely time consuming process, hence the 1D search for every image block of left image was limited to a horizontal window of 112 pixels centered around the pixel value. The underlying assumption is that features in image pairs are likely to be located in close vicinity of each other in a stereo pair
- Sum of Squared Differences is an expensive process hence Sum of Absolute differences was used to facilitate the 1D search.

Output:

Output can be found at the following link :
https://drive.google.com/drive/folders/1FzxYAbuwSNCZAntNGFIf1o-xmDj--eIh?usp=sharing