AER1515 – Assignment 2 (Feature Point Detection and Correspondences)

1.0 Feature Detection

Figure 1: Key points represented for left image test set (000011)

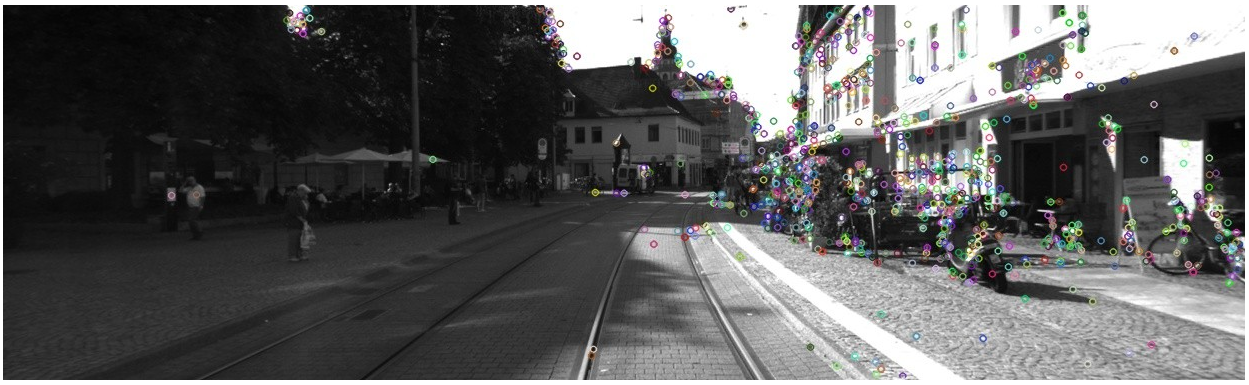

Figure 2: Key points represented for left image test set (000012)

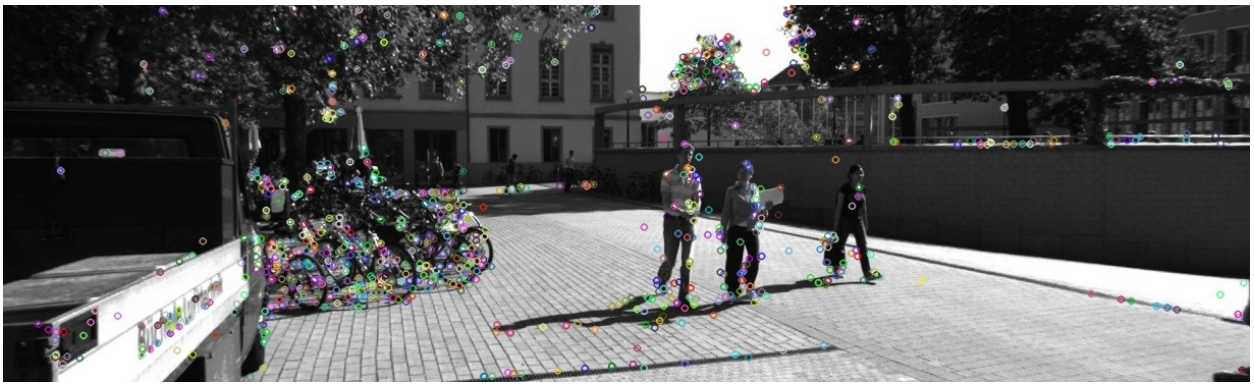

Figure 3: Key points represented for left image test set (000013)

Figure 4: Key points represented for left image test set (000014)

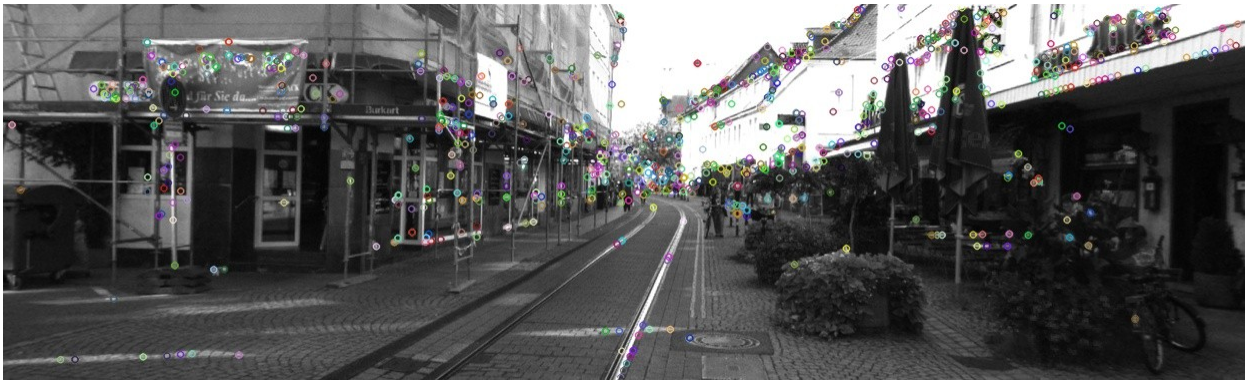

Figure 5: Key points represented for left image test set (000015)

For the feature detector, SIFT detector was used where a 1000 key points were detected. Based on key points detected from the images, it can be noticed that some of the key points chosen are for objects that do not have a unique feature. This is for example, in the case for Figure 1, Certain key points were detected for the sidewalk where the features in that area doesn't necessarily have a unique feature. This would make the matching unreliable.

2.0 Feature Matching

A FLANN based matcher is used for the matching of the extracted features. As per the Lowe's ratio test, only the best matches were kept. The following figures are for the feature matching of the training and test stereo images respectively.

Figure 6: Features Matched for training stereo image pairs (000001)

Figure 7: Features Matched for training stereo image pairs (000002)

Figure 8: Features Matched for training stereo image pairs (000003)

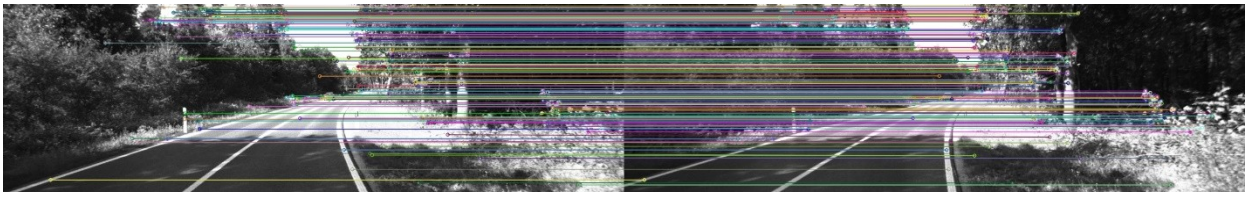

Figure 9: Features Matched for training stereo image pairs (000004)

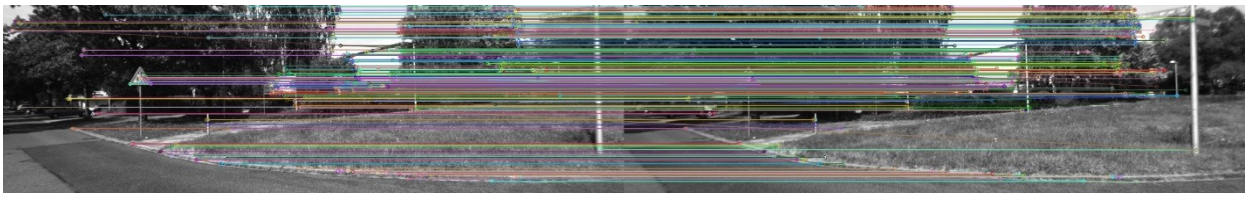

Figure 10: Features Matched for training stereo image pairs (000005)

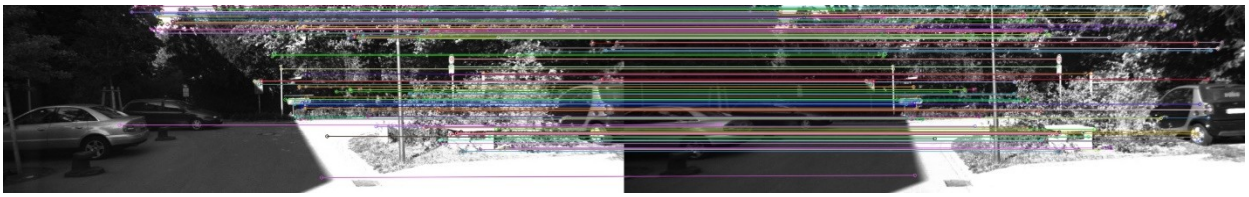

Figure 11: Features Matched for training stereo image pairs (000006)

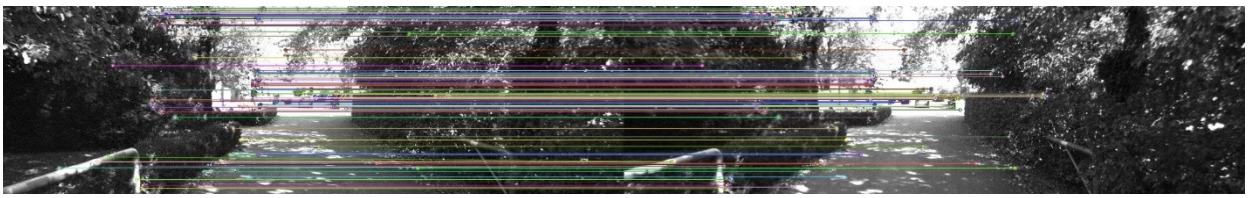

Figure 12: Features Matched for training stereo image pairs (000007)

Figure 13: Features Matched for training stereo image pairs (000008)

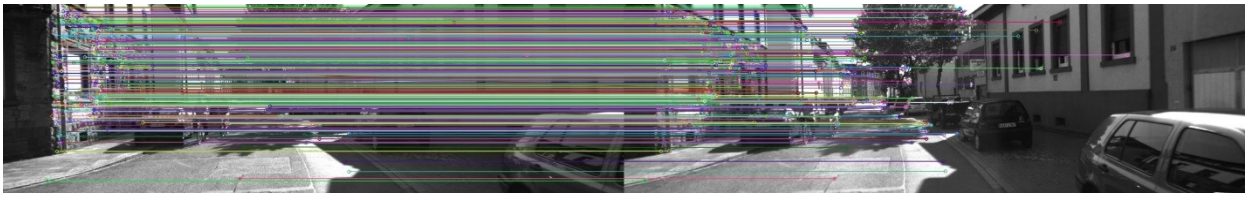

Figure 14: Features Matched for training stereo image pairs (000009)

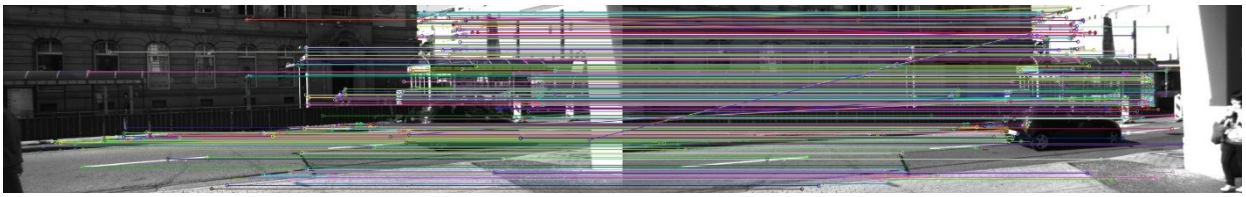

Figure 15: Features Matched for training stereo image pairs (000010)

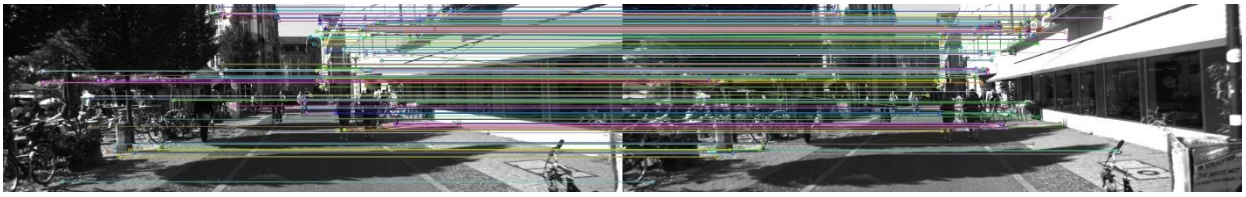

Figure 16: Features Matched for test stereo image pairs (000011)

Figure 17: Features Matched for test stereo image pairs (000012)

Figure 18: Features Matched for test stereo image pairs (000013)

Figure 19: Features Matched for test stereo image pairs (000014)

Figure 20: Features Matched for test stereo image pairs (000015)

2.1 Observations

It is assumed that the images have been rectified and thus the stereo images would follow the epipolar constraint. During the analysis, it was observed that some of the feature points gave a disparity value of 0. This in practice would not make sense as this would mean that the feature is positioned at infinity from the camera. For these cases, the depth is ignored. Depth is calculated using the following equation, Depth = focal length * baseline / disparity. As depth gets closer to infinity, disparity goes to 0, thus being same pixel.

3.0 Outlier Rejection

It is noted that the number of matches are reduced after the outlier rejection. Most of the matches are now horizontal and follow the epipolar constraint. The diagonal matches are now removed since they are classified as outliers. Also key points that didn't have unique features are no kept as inliers since it is hard to match them between the image pairs.

The tuned RANSAC parameters are:

- maxIters = 5000
- Confidence = 0.925

The following are the correspondences between the training stereo images:

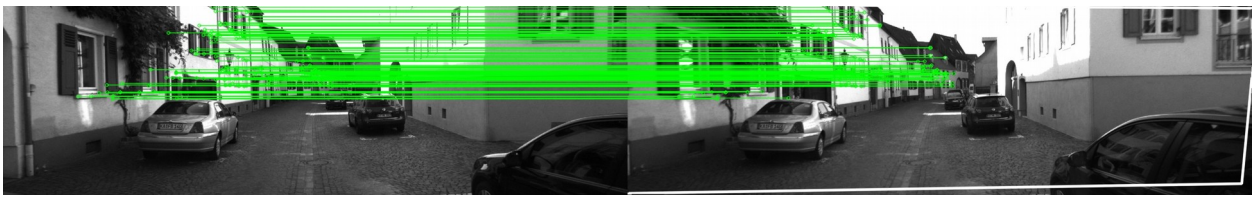

Figure 21: Features Matched for train stereo image pairs (000001)

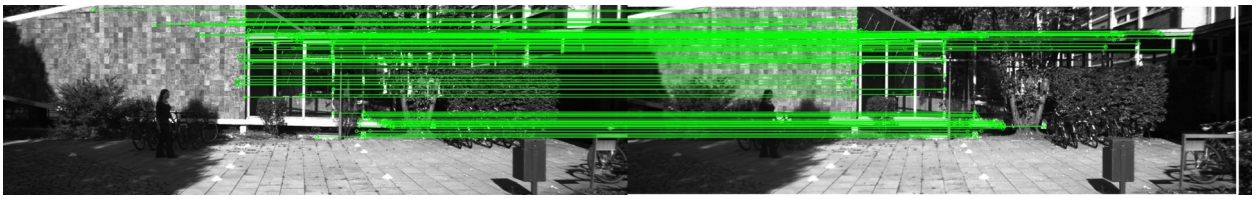

Figure 22: Features Matched for train stereo image pairs (000002)

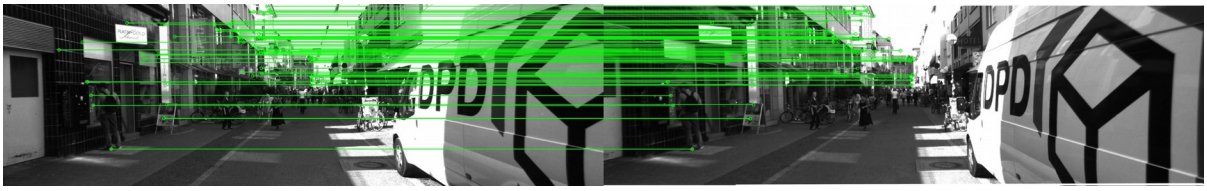

Figure 23: Features Matched for train stereo image pairs (000003)

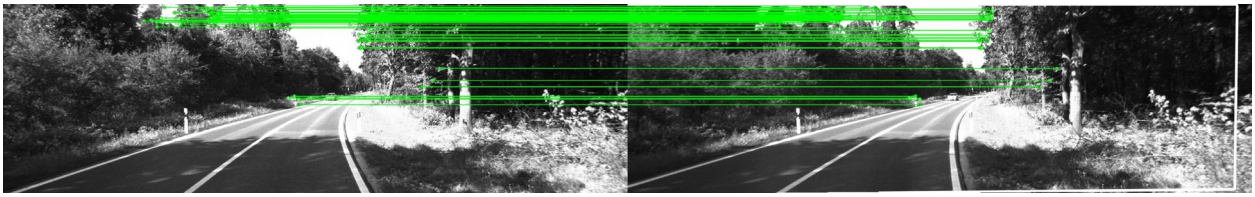

Figure 24: Features Matched for train stereo image pairs (000004)

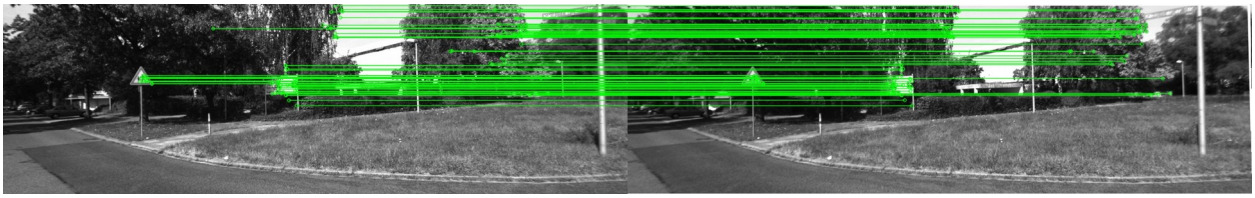

Figure 25: Features Matched for train stereo image pairs (000005)

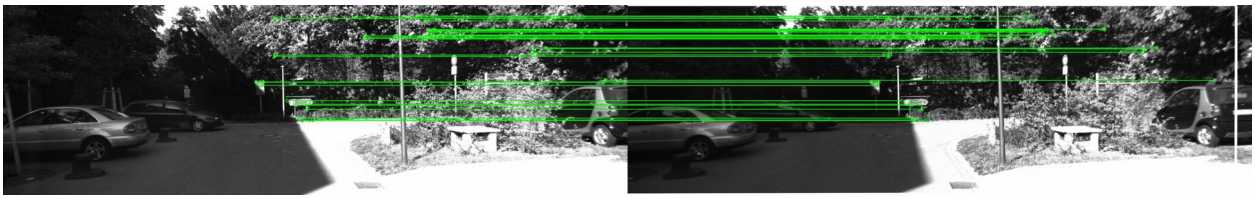

Figure 26: Features Matched for train stereo image pairs (000006)

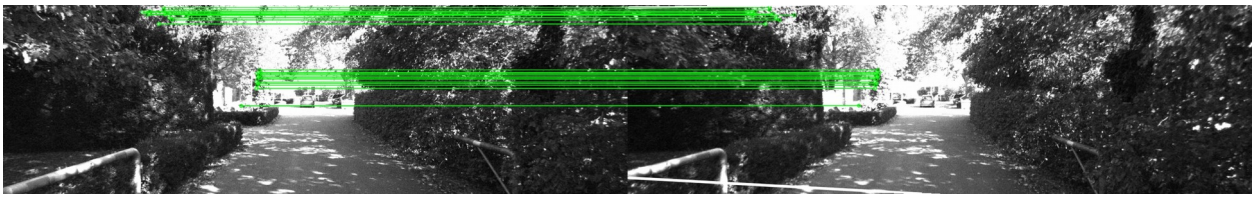

Figure 27: Features Matched for train stereo image pairs (000007)

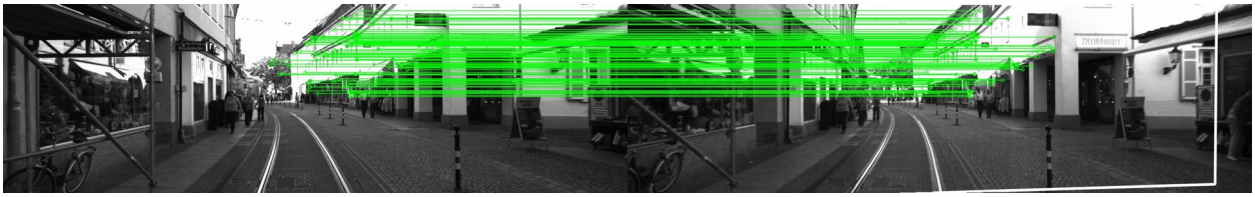

Figure 28: Features Matched for train stereo image pairs (000008)

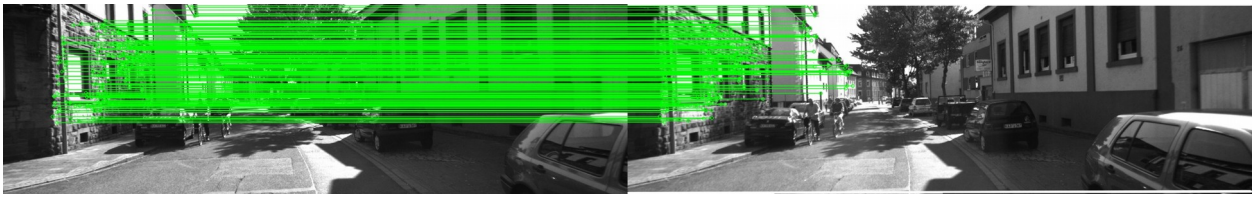

Figure 29: Features Matched for train stereo image pairs (000009)

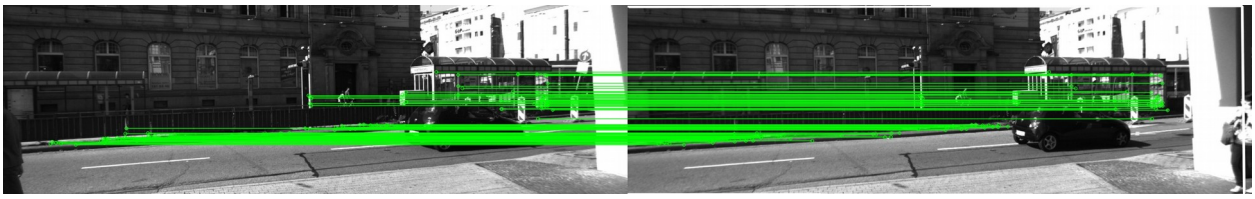

Figure 30: Features Matched for train stereo image pairs (000010)

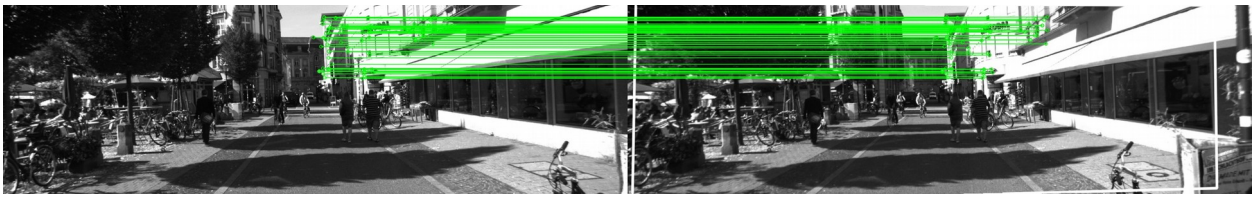

Figure 31: Features Matched for test stereo image pairs (000011)

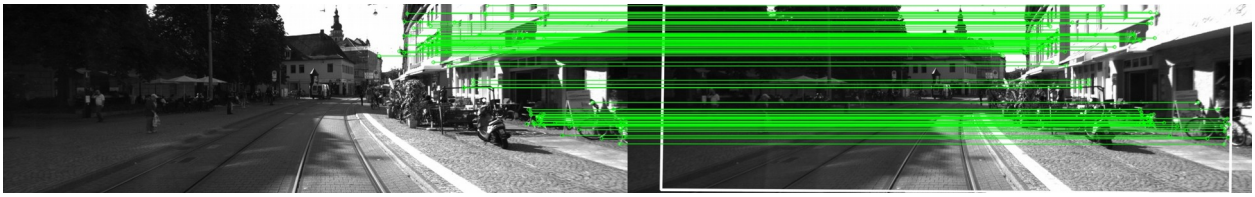

Figure 32: Features Matched for test stereo image pairs (000012)

Figure 33: Features Matched for test stereo image pairs (000013)

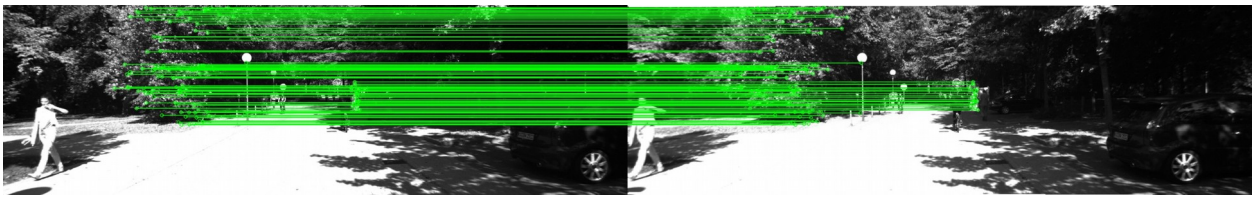

Figure 34: Features Matched for test stereo image pairs (000014)

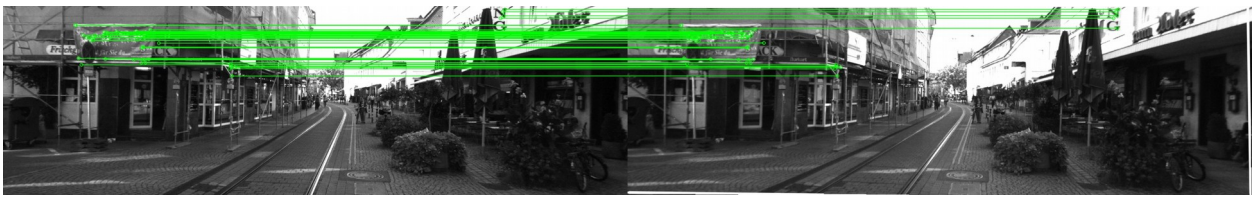

Figure 35: Features Matched for test stereo image pairs (000015)

The outlier rejection algorithm used is RANSAC implemented via OpenCV. The findHomography() function is used for this. The RANSAC algorithm has the following steps:

1. The fewest number of samples needed to represent the model are randomly selected. The model is a 3x3 matrix consisting of a rotation and translation component.
2. Compute the model parameters using this randomly selected points.
3. Determine how many matches fit the model
4. If the number of inliers is greater than a threshold or we have reached a preset number of iterations, go to the next step.
5. Recompute the model parameters from the iteration that has the most inliers.

RMSE Result:

When RANSAC is implemented for outlier rejection, the following Root Mean Squared Errors are calculated for the training image set.

RMSE	
Training Image	RANSAC Implemented
000001	11.2665
000002	5.6927
000003	2.2866
000004	32.1054
000005	39.2464
000006	7.4711
000007	3.3492
000008	23.8952
000009	8.8125
000010	10.5825