Answer Sheet 3

Topic: Feature Detectors, Descriptors, Epipolar Geometry, RANSAC

Yihao Wang 03740227

November 22, 2022

Part 2: Epipolar constraint

Prove:
Assume the translation between camera 1 and camera 2 is rotation R and translation T followed by a projection. Then we have

$$\lambda_1 x_1 = X, \lambda_2 x_2 = RX + T$$

Merge these equation, we have

$$\lambda_2 x_2 = R(\lambda_1 x_1) + T$$

Then multiply the equation with $\hat{T}$, we have

$$\lambda_2 \hat{T} x_2 = \lambda_1 \hat{T} R x_1$$

Finally, we project it onto x_2, and have

$$x_2^T \hat{T} R x_1 = 0$$

Compare with the form of epipolar constraint, we know that

$$E = \hat{T} R$$

Part 4: Bag-of-Words for Place Recognition

- The main difference between match_all and match_bow functions:
  Match_all uses Brute-force matching. To be specific, it uses RANSAC to match features and filter inliers. Match_bow uses Bag-of-Words method, which first extracts the words of each frame, and then compares their similarity.

- num_bow_candidates corresponds to num_results in query function. It helps us to find the top n highest rated (most similar) frames.

- We compare the number of candidate pairs and inliers when using the match_all and match_bow functions:
  For match_all function:
  -Brute-force matching 13284 image pairs...
  -Successfully matched 1007 out of 13284 image pairs with a total of 44350 inlier feature matches (109579 total). New total of matched image pairs is 13366.

We compute the percentage:

$1007/13284 = 7.58\%, 44350/109579 = 40.47\%$

For match_bow function:

-Matching 3649 image pairs using BoW...

-Successfully matched 456 out of 3649 image pairs with a total of 23974 inlier feature matches (43848 total). New total of matched image pairs is 13284.

We compute the percentage:

$456/3649 = 12.5\%, 23974/43848 = 54.68\%$

We can see that the BoW could use relative more image pairs and have higher matching rate. However, they have similar new total of matched image pairs.

- If the number of candidate pairs increases to 2×1000 image, brute-force matching will work similar, while Bag-of-Words may have better result.

  That is because more images means larger dataset for BoW to train. Therefore, it is expected to have more accurate dictionary.