

Machine Vision Assignment 5: Feature Detect

Dan Blanchette

October 22, 2024

1 ORB and Brute Force

This implementation did fairly well matching one of the three images to the master. There were some outliers, like my photo frame in the background and the vandal flag. These descriptors and feature matching implementation were not the top performers in my tests.

2 FAST, ORB, and Brute Force

In a couple of words, not great. This implementation was subject to FAST finding many features but missing many critical points for the master image. Using orb to help generate descriptors did help, but overall, it found more background features in some instances than SIFT and brute force did.

3 SIFT and Brute Force

This pairing did much better compared to ORB and BF. I'm sure lighting was a critical factor in predicting the master image and the various objects in the image background, which change based on the angle of the Blu-ray sleeve. The evaluation of key points and descriptor values included my Christmas tree, printer, picture frames, and T-shirt as part of the master image, but they weren't. However, there were fewer instances of this happening.

4 SIFT, FLANN, and Homography

This implementation is the most accurate and impressive detection of the bunch. No outliers were detected using homography paired with SIFT and FLANN. This is due to thresholding and identifying pixel values like those with a mask and the KNN algorithm. The mask helps refine the KNN output by eliminating outlying pixel values.

5 Conclusions

If I had to rank the tested implementations, it would be as follows:

1. SIFT FLANN Homography
2. SIFT Brute Force
3. ORB Brute Force
4. FAST ORB Brute Force

I also must consider that brute force will not be a top performer in real-time image detection. In that case, the SIFT FLANN Homography code would be my starting point for anything I implement in the future.