SFND CV and Lidar Midterm

Sarah Majors

September 2023

1 Introduction

The intent of this project is to analyze various combinations of feature detectors and descriptors with the goal of identifying the best solution for the continued project of developing a toy advanced driver assistance system. The code, code outputs, and raw data can be found in this repository to support my statements.

2 MP 1: Data Buffer Optimization

Initially to implement the ring buffer, I wanted to utilize the BOOST library ring buffer, but had issues with it compiling. This led me to simply keeping track of how many images are in the buffer and when it reached its maximum, I erased the first image and pushed the new image to the back.

3 MP 2: Keypoint Detection

The Shi-Tomasi detector had been implemented for us and in previous exercises, we had completed the Harris detector so using that information and the example of the Harris detector from the OpenCV API, I was able to implement that one. From there, the rest of the detectors were implemented very similarly to one another.

4 MP 3P: Keypoint Removal

To focus only on points within a rectangle immediately preceding the vehicle, and hopefully focusing on the vehicle preceding the 'driver', a rectangle was defined for us. I used that along with OpenCV's KeyPoint 'inside' function to determine if a point was within the focus window and pushed that point into a vector to then use as the keypoints vector.

5 MP 4: Keypoint Descriptors

Very similar to the keypoint detection task, an example was given of how to use one of the feature descriptors and that pattern was followed to complete the function with the remaining options that would be reviewed.

6 MP 5: Descriptor Matching

Brute Force descriptor matching was already implemented for us. That left implementing FLANN descriptor matching. This was simple except that there is sometimes an issue with type mismatches, so that needs to be rectified first by changing the descSource and descType.

7 MP 6: Descriptor Distance Ratio

The distance ratio is checking the ratio between the distance of the first match to the second is less than 0.8 then the match is likely not being confused with another point in the image and giving us a false match. This helps determine between two good sets of matches.

8 MP 7: Performance Evaluation - Keypoint on Preceeding Vehicle

The empirical record of this assignment can be found in './MP 7 8 9.pdf' but here I will note the neighborhood sizes. Shi-Tomasi had a large spread of points around the image, while the Harris detector had very few points at all. Fast had a lot of points and seemed to catch lines like along the bridge or the running board of the car to the left really well.. The brisk keypoints were very large and I question to what degree those would be useful enough to accurately track the preceeding vehicle. There were fewer orb points, but they were quite large. Akaze keypoints were not as small as Shi-Tomasi or fast, but were much more reasonable than orb and brisk. Sift has a wide range of keypoints with some being quite small to some that are quite large.

—	Img 1	Img 2	Img 3	Img 4	Img 5	Img 6	Img 7	Img 8	Img 9	Img 10
Shi-Tomasi	125	118	123	120	120	113	114	123	111	112
Harris	17	14	18	21	26	14	18	31	26	34
Brisk	254	274	276	275	293	275	289	268	259	250
Fast	419	427	404	423	386	414	418	406	396	401
Orb	91	102	16	113	10	124	129	127	124	125
Akaze	162	157	159	154	162	163	173	175	175	175
Sift	137	131	121	135	134	139	136	147	156	135

9 MP 8: Performance Evaluation - Matching Keypoints on Preceeding Vehicle

This task just involved running every possible combination of keypoint detector and descriptor and checking the number of matches on the preceding vehicle amongst all the frames. Fast had the most matches.

—	BRISK	BRIEF	ORB	AKAZE	SIFT
Shi-Tomasi	944	944	908	—	919
Harris	142	173	162	—	163
Brisk	1544	1675	1479	—	1636
Fast	1544	2831	2768	—	2790
Orb	2183	540	755	—	769
Akaze	—-	—-	—-	1249	—-
Sift	590	693	1078	—	783

10 MP 9: Performance Evaluation - Time

Based on the cumulative information gathered from this analysis (located in './MP 7 8 9.pdf') I would recommend the fast detector with the orb, brief, or brisk descriptors. Fast is the fastest (so it is aptly named), but also offers the most keypoints and keypoint matches in that short amount of time. The below table is in total time for all images in milliseconds.

—	BRISK	BRIEF	ORB	AKAZE	SIFT
Shi-Tomasi	154	154	124	—	187
Harris	300	125	124	—	230
Brisk	3212	3561	3336	—	4018
Fast	50	28	27	—	293
Orb	130	63	88	—	698
Akaze	—-	—-	—-	3328	—-
Sift	1225	1243	1252	—	2116