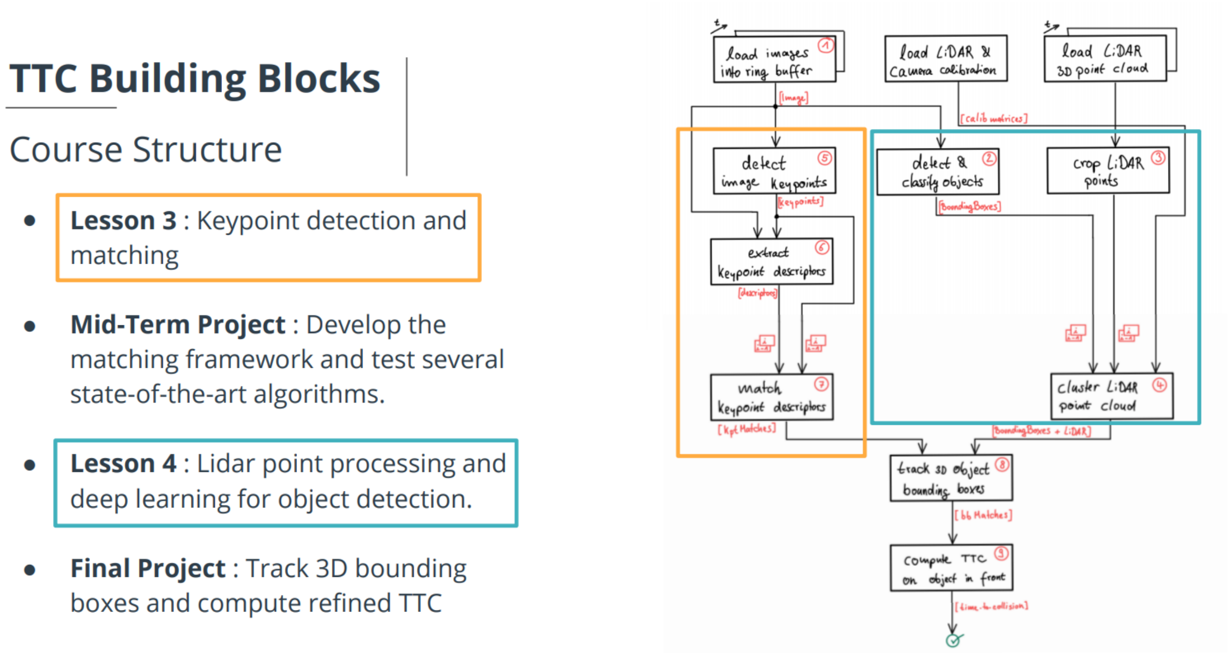
Tracking of Object in 3D Space

Project Report

-Anurag Kankanala



TTC Building Blocks

Course Structure

- **Lesson 3** : Keypoint detection and matching
- **Mid-Term Project** : Develop the matching framework and test several state-of-the-art algorithms.
- **Lesson 4** : Lidar point processing and deep learning for object detection.
- **Final Project** : Track 3D bounding boxes and compute refined TTC

load images into ring buffer 1
load LiDAR & Camera calibration
load LiDAR 3D point cloud
t
[Image]
[calib matrices]
detect image Keypoints 5
detect & classify objects 2
crop LiDAR points 3
[keypoints]
[BoundingBoxes]
extract keypoint descriptors 6
[descriptors]
match keypoint descriptors 7
cluster LiDAR point cloud 4
[KptMatches]
[BoundingBoxes + LiDAR]
track 3D object bounding boxes 8
[bb Matches]
compute TTC on object in front 9
[time-to-collision]

FP.1 Match 3D Objects

Implement the method "matchBoundingBoxes", which takes as input both the previous and the current data frames and provides as output the ids of the matched regions of interest (i.e. the boxID property). Matches must be the ones with the highest number of keypoint correspondences.

Code can be found in the file : camFusion_Student.cpp | line : 286-340.

Algorithm used :

1. Check if the match is present in the ROI of both currFrame and prevFrame.
2. For such matches , populate a map that has the prevBoundingBox ID as the key.
3. Value is a vector of counts, indexed by the currBoundingBox ID.
4. For every prevID, iterate over the vectors and find the best currID.

FP.2 Compute Lidar based TTC

Compute the time-to-collision in second for all matched 3D objects using only Lidar measurements from the matched bounding boxes between current and previous frame.

Code can be found in the file : camFusion_Student.cpp | line : 220-284

Interquartile Range (IQR) has been used to identify outliers and use the x_min from the resultant lidar points.

https://www.geeksforgeeks.org/interquartile-range-iqr/

FP.3 Associate Keypoint Correspondences with Bounding Boxes

Prepare the TTC computation based on camera measurements by associating keypoint correspondences to the bounding boxes which enclose them. All matches which satisfy this condition must be added to a vector in the respective bounding box.

Code can be found in the file camFusion_Student.cpp | line 131-176

Interquartile Range (IQR) has been used to identify outliers and use euclidean distances within the suggested IQR range to identify kptMatches for a bounding box.

FP.4 Compute Camera-based TTC

Compute the time-to-collision in second for all matched 3D objects using only keypoint correspondences from the matched bounding boxes between current and previous frame.

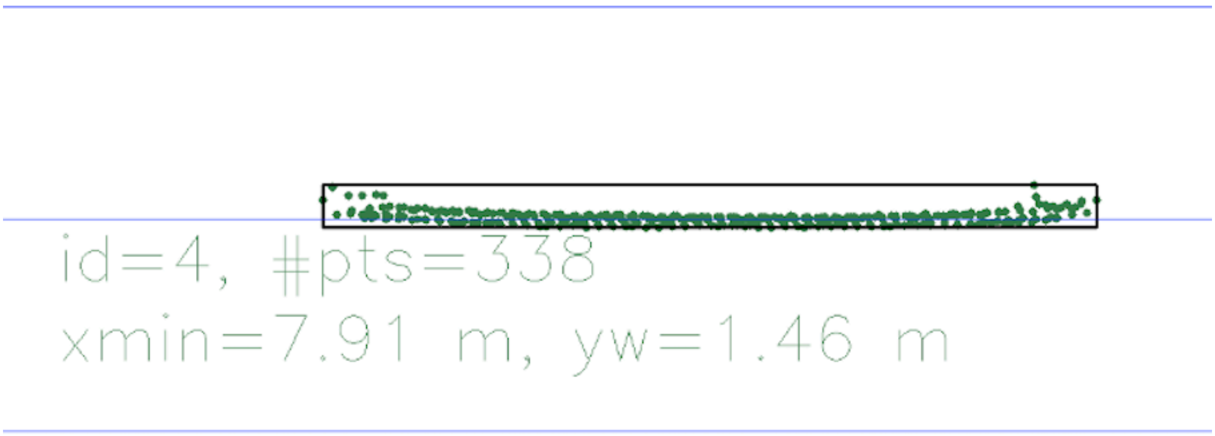
Code can be found in the file camFusion_Student.cpp | line 179-218

To deal with outlier correspondences, a min distance of 100 and median distance ratio has been used to calculate the TTC.

FP.5 Performance Evaluation 1

Find examples where the TTC estimate of the Lidar sensor does not seem plausible. Describe your observations and provide a sound argumentation why you think this happened.

Lidar based TTC did not show any outliers as IQR was used to discard any outliers. Average of the Q3 and Q1 was used to find the minX value to use for estimating the distance to the preceding vehicle. This ensured, the lidar point used was always actually on the preceding vehicle and not an erroneous detection.



FP.6 Performance Evaluation 2

Run several detector / descriptor combinations and look at the differences in TTC estimation. Find out which methods perform best and also include several examples where camera-based TTC estimation is way off. As with Lidar, describe your observations again and also look into potential reasons.

Camera TTC and Lidar TTC values were recorded for possible detector/descriptor combinations. Results can be viewed in the attached comparison_results.xlsx file.

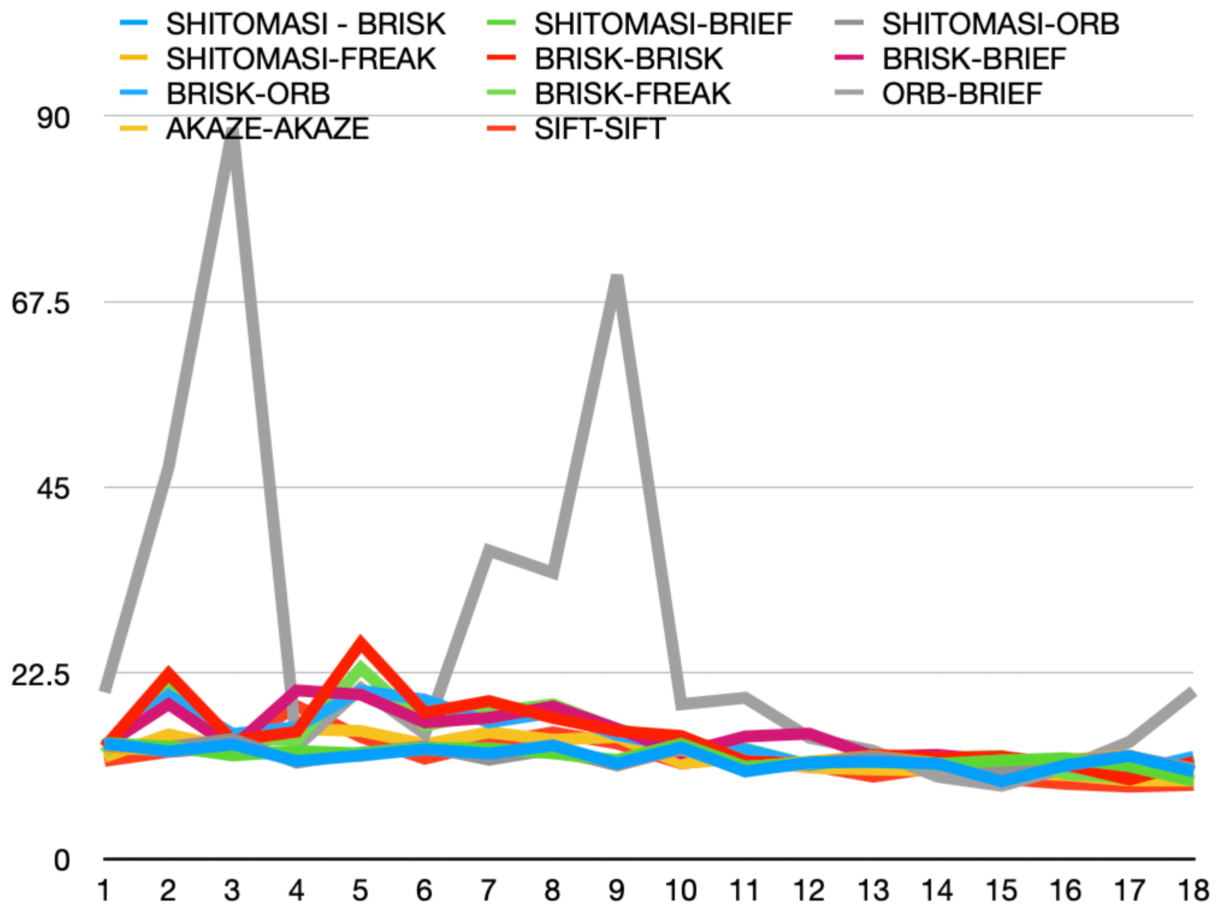


Chart showing CAMERA TTC values for few of the combinations

We can see form the chart, SHITOMASI and BRISK detectors give consistent results while ORB detectors has deviations.

As can be seen in comparison_results.xlsx file, Harris detector gives -Inf and Nan values and is hence very unreliable.

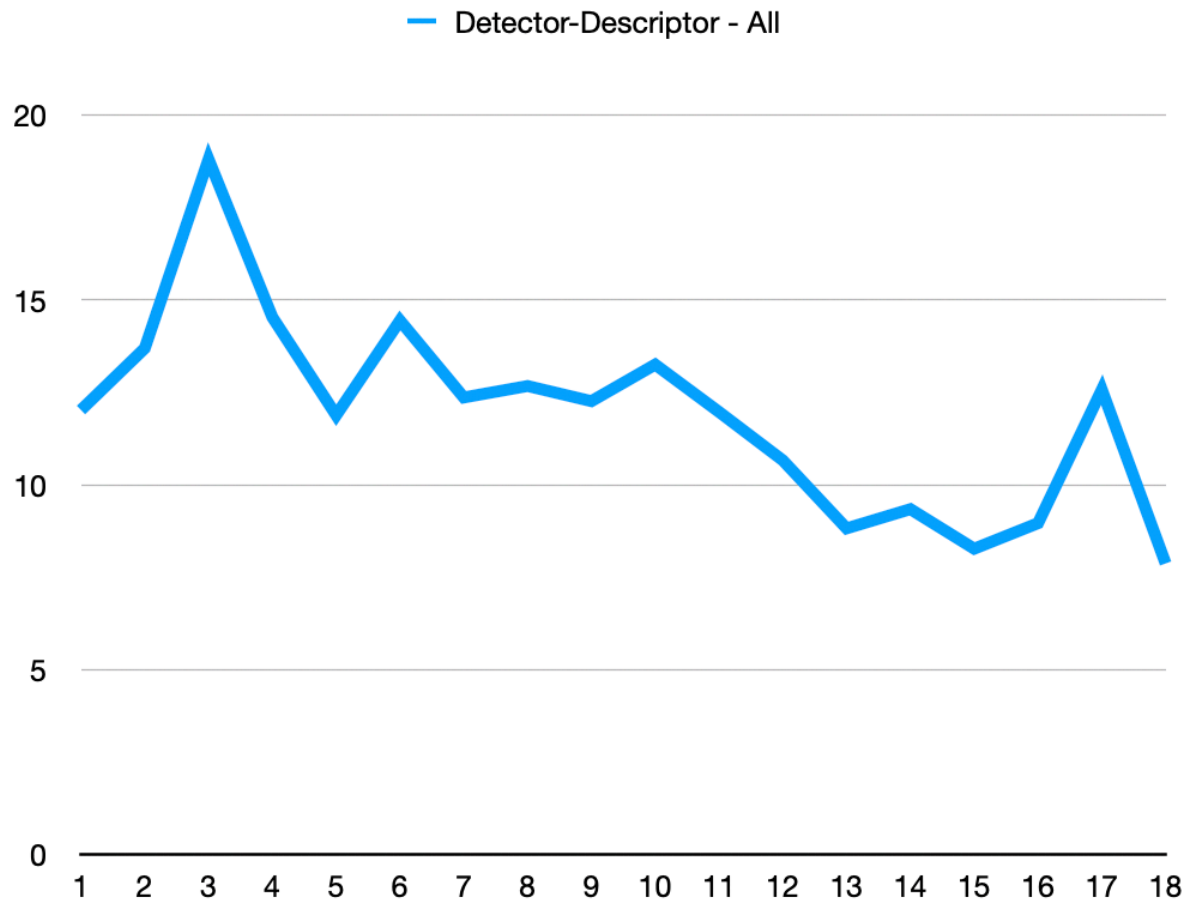


Chart showing LIDAR TTC Values.

As can be seen LIDAR based TTC estimation is reliable and consistent compared to Camera based TTC estimation. For Camera based estimation, results depend on the detector/descriptor type used.