### Meets Specifications

Nice work,  
Congratulations on completing this project with and outstanding submission.  
You nailed it.  
The code and reasoning showcased in your work shows a good understanding of the concepts in the lessons.

Keep up the good work as you proceed. All the best!

## Check out the following for further learning.

[Cameras in Processing (2D and 3D)](https://medium.com/@behreajj/cameras-in-processing-2d-and-3d-dc45fd03662c)  
[What is the best feature to track an object in video sequences?](https://www.researchgate.net/post/What_is_the_best_feature_to_track_an_object_in_video_sequences)  
[How to Detect and Track Object With OpenCV](https://www.intorobotics.com/how-to-detect-and-track-object-with-opencv/)  
[Camera Tracking for Augmented Reality Media](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.130.9199&rep=rep1&type=pdf)

### FP.0 Final Report

The writeup / README should include a statement and supporting figures / images that explain how each rubric item was addressed, and specifically where in the code each step was handled.

Impressive job with the writeup and graphs, examining how each rubric was handled. 

### FP.1 Match 3D Objects

Code is functional and returns the specified output, where each bounding box is assigned the match candidate with the highest number of occurrences.

The bounding box matching code is functional and it is returning correct results. 

### FP.2 Compute Lidar-based TTC

Code is functional and returns the specified output. Also, the code is able to deal with outlier Lidar points in a statistically robust way to avoid severe estimation errors.

Plane fitting seems like a good idea and your results look good.Another way is to use the [IQR range values](https://www.khanacademy.org/math/statistics-probability/summarizing-quantitative-data/box-whisker-plots/a/identifying-outliers-iqr-rule) to remove outliers.

### FP.3 Associate Keypoint Correspondences with Bounding Boxes

Code performs as described and adds the keypoint correspondences to the "kptMatches" property of the respective bounding boxes. Also, outlier matches have been removed based on the euclidean distance between them in relation to all the matches in the bounding box.

Everything is good here. Excellent work! 

### FP.4 Compute Camera-based TTC

Code is functional and returns the specified output. Also, the code is able to deal with outlier correspondences in a statistically robust way to avoid severe estimation errors.

Again excellent job removing the outliers and computing the TTC using key-point matches.

### FP.5 Performance Evaluation 1

Several examples (2-3) have been identified and described in detail. The assertion that the TTC is off has been based on manually estimating the distance to the rear of the preceding vehicle from a top view perspective of the Lidar points.

You correctly showed image examples where estimates are higher. Furthermore, you have given a well sound reason.

### FP.6 Performance Evaluation 2

All detector / descriptor combinations implemented in previous chapters have been compared with regard to the TTC estimate on a frame-by-frame basis. To facilitate comparison, a spreadsheet and graph should be used to represent the different TTCs.

Really outstanding comparison with very interesting conclusions to extract.