Writeup: Track 3D-Objects Over Time

Please use this starter template to answer the following questions:

1. Write a short recap of the four tracking steps and what you implemented there (filter, track management, association, camera fusion). Which results did you achieve? Which part of the project was most difficult for you to complete, and why?

Step-1: Extended Kalman Filter

The first step is to implement an extended Kalman filter.

Kalman Filter Equations Summary

Prediction step:

$$\mathbf{x}^{-} = \mathbf{F}\mathbf{x}^{+}$$

$$\mathbf{P}^{-} = \mathbf{F}\mathbf{P}^{+}\mathbf{F}^{T} + \mathbf{Q}$$

Update step:

$$\gamma = \mathbf{z} - \mathbf{H}\mathbf{x}$$

$$\mathbf{S} = \mathbf{H}\mathbf{P}^{-}\mathbf{H}^{T} + \mathbf{R}$$

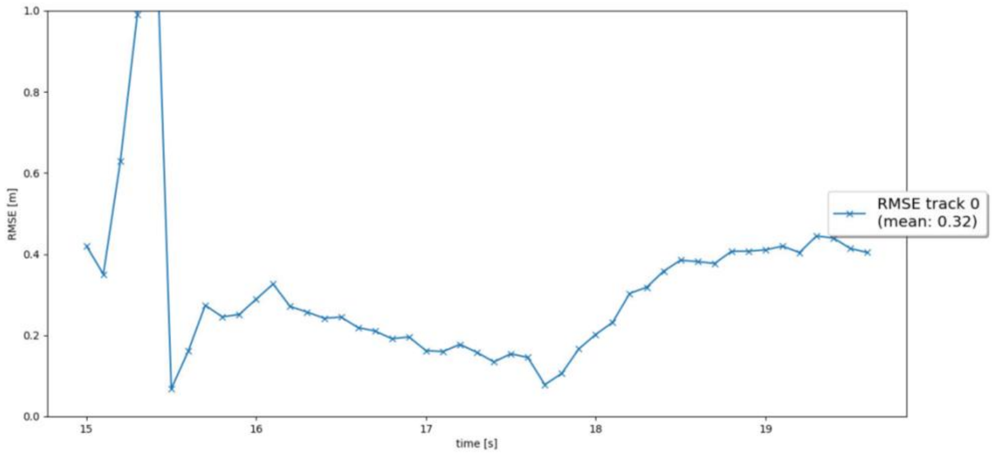
$$\mathbf{K} = \mathbf{P}^{-}\mathbf{H}^{T}\mathbf{S}^{-1}$$

$$\mathbf{x}^{+} = \mathbf{x}^{-} + \mathbf{K}\gamma$$

$$\mathbf{P}^{+} = (\mathbf{I} - \mathbf{K}\mathbf{H})\,\mathbf{P}^{-}$$

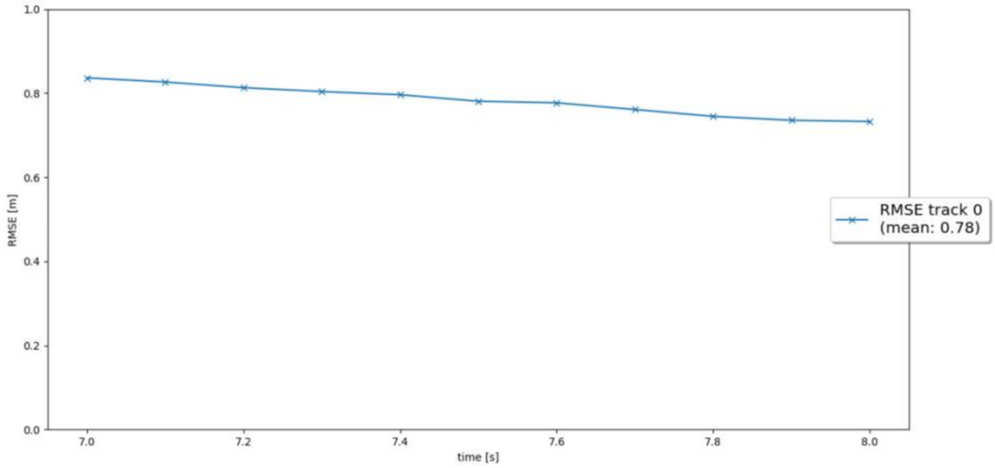
There are basically two steps: predict and measure. The prediction step is to predict x and P based on the motion model. The measurement step is to update x and P based on the measurement error and the covariances.

RMSE of lidar tracking



Step-2: Track Management

The second step is to implement the tracking lifecycle management



Step-3: Data Association

The thrid step is to implement the association of measurements to tracks and to handle unassociated tracks and measurements. We use a single nearest neighbor data association measured by Mahalanobis distance and use gating to ease associations.

- As a distance measure for this decision, the $\mathbf{Mahalanobis\ distance}$ is used (note that it actually contains a squared distance):

$$d(\mathbf{x}, \mathbf{z}) = \gamma^T \mathbf{S}^{-1} \gamma = (\mathbf{z} - h(\mathbf{x}))^T \mathbf{S}^{-1} (\mathbf{z} - h(\mathbf{x}))$$

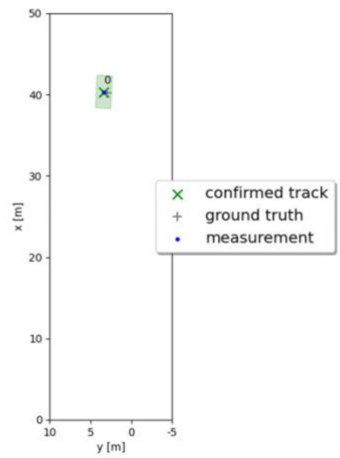
- Say we have N tracks and M measurements.
- The $\mathbf{association\ matrix\ A}$ is an $N \times M$ matrix that contains the Mahalanobis distances between each track and each measurement:

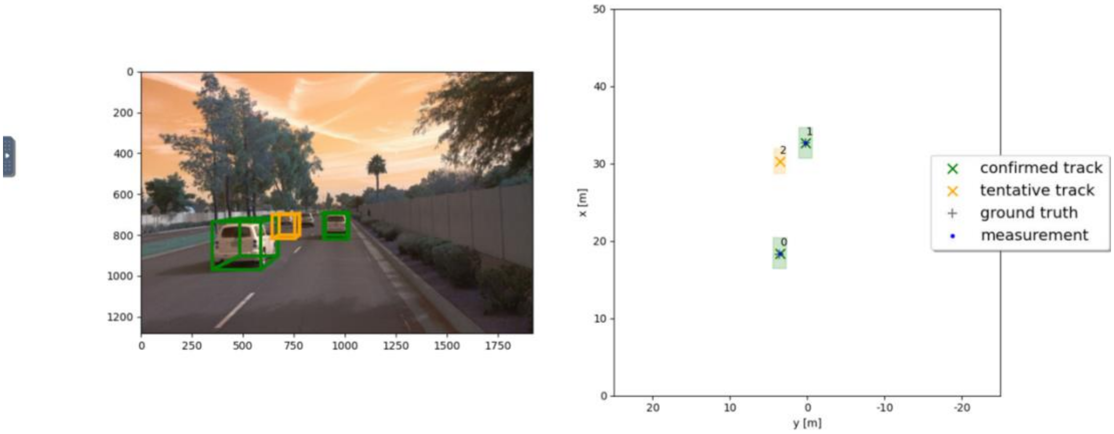
$$\mathbf{A} = \begin{pmatrix} d(\mathbf{x}_1, \mathbf{z}_1) & d(\mathbf{x}_1, \mathbf{z}_2) & \dots & d(\mathbf{x}_1, \mathbf{z}_M) \\ \vdots & \vdots & & \vdots \\ d(\mathbf{x}_N, \mathbf{z}_1) & d(\mathbf{x}_N, \mathbf{z}_2) & \dots & d(\mathbf{x}_N, \mathbf{z}_M) \end{pmatrix}$$

$$d\left(\mathbf{x}, \mathbf{z}\right) \leq F_{\chi^2}^{-1}\left(0.995 | \dim_{\mathbf{z}}\right)$$

- If a measurement lies outside a track's gate, we can set the distance to infinity, for example:

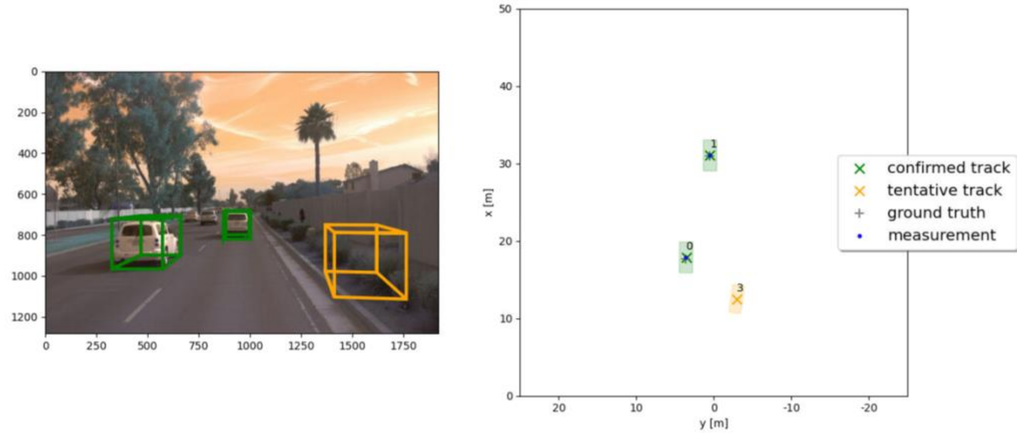
$$\mathbf{A} = \begin{pmatrix} d(1,1) & \infty & \infty \\ d(2,1) & d(2,2) & d(2,3) \\ d(3,1) & d(3,2) & d(3,3) \\ \infty & d(4,2) & d(4,3) \end{pmatrix}$$

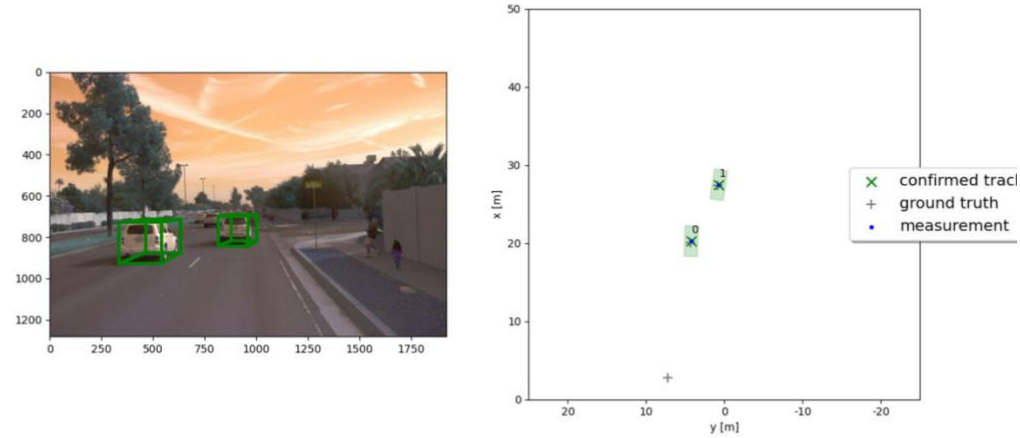


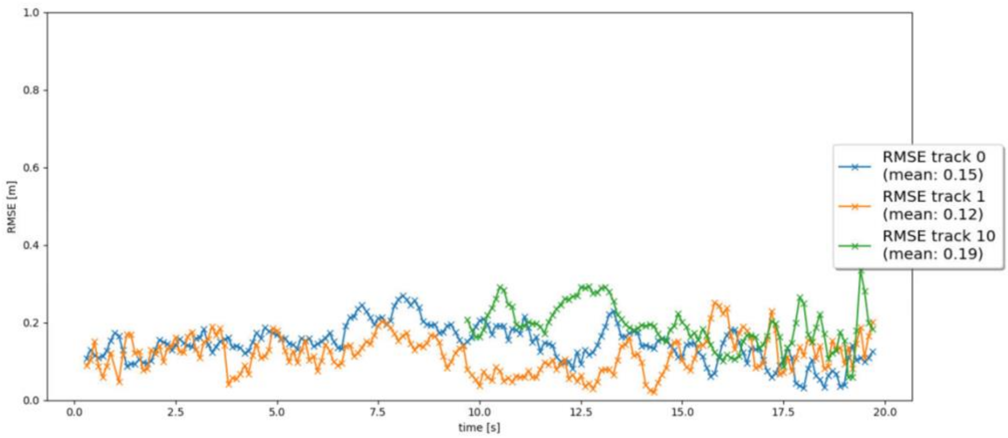


Step-4: Camera Sensor fusion

The fourth step is to implement camera fusion.







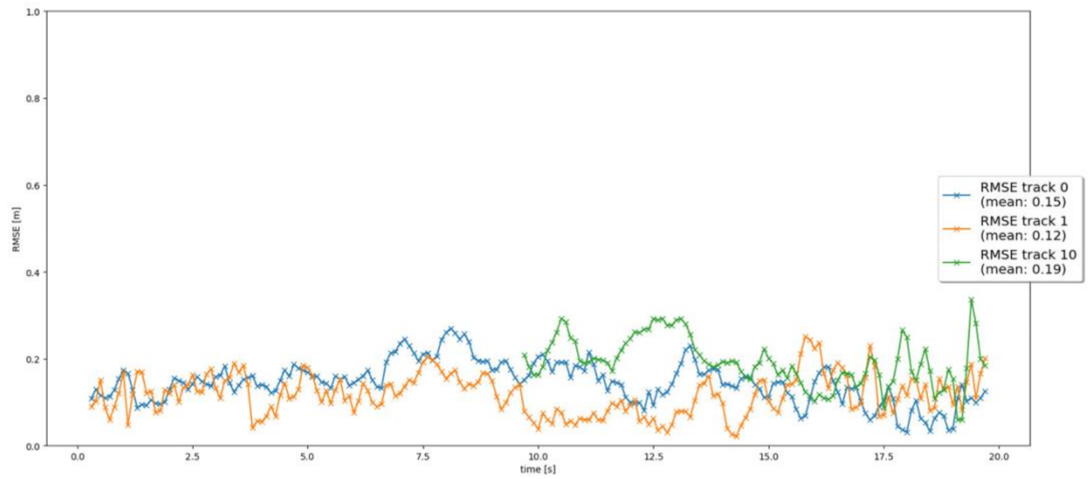
Most Difficult Part

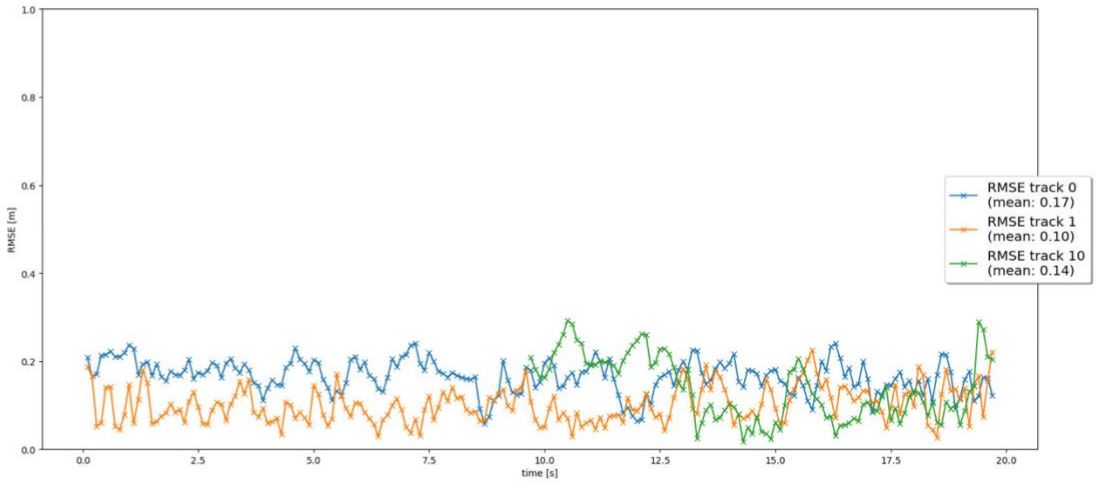
Measurement part is most difficult especially the part fuse the camera and lidar coordinate

2. Do you see any benefits in camera-lidar fusion compared to lidar-only tracking (in theory and in your concrete results)?

The most important aspect of Lidar is the spatial projection which is better than a camera.

Although in the industry there are still discussions related with the camera only or sensor fusion approach.





In project, Lidar camera fused tracking provide slightly better results

3. Which challenges will a sensor fusion system face in real-life scenarios? Did you see any of these challenges in the project?

Speed of sensor fusion since lidar data is usually much larger than camera file and the speed to combine tracking from different sensors in a real time (10ms) environment is hard.

To built a world coordinate for all sensor and construct deep learning based on such general system shall be more beneficial, especially in huge amount of data that can be fed in transformer kind of model which usually perform better with big data.

Improvement opportunity:

- Robust association algorithm like GNN