3-D Data Processing to Extract High-Resolution Micro Traffic Data from Roadside LiDAR Sensors

Yuan Sun

Advisor: Dr. Hao Xu

University of Nevada, Reno
Intelligent Transportation System Lab
November 9th, 2017
Outlines

1. Problem statement.
2. Research objective.
3. Literature review.
4. LiDAR data processing procedures.
5. Tracking validation.
6. Case study.
8. Applications.
10. References.
Problem Statement

- The current connected-vehicle system highly relying on information broadcasted by each vehicle. However, the mixed traffic with connected-vehicles and unconnected-vehicles will exist for the next decades or even longer.

- The traditional traffic sensors such as loop detectors, video detectors, Bluetooth sensors and radar sensors mainly provide macro traffic data such as traffic flow rates, average speeds and occupancy, so the existing sensors cannot provide the micro traffic data needed by connected vehicles.

- The limitation of traditional sensors has also restricted the possible usage of high-resolution micro traffic data. Connected vehicles researches, safety researches and adaptive traffic signal control researches are all limited by the resolution of data input.
Research objective

- This research aims to solve this problem by developing a 3-D data processing procedure to extract high-resolution micro traffic data from roadside LiDAR sensors. It aims at extracting traffic counts, vehicle locations, vehicle speed, vehicle directions, and possible pedestrian information from a VLP-16 LiDAR sensor.
Problem: Mixed traffic would existed, not every vehicle would equipped with communication devices.
Literature Review-Existed Sensors
What is a LiDAR?

LiDAR (also called LIDAR, and LADAR) is a surveying method that measures distance to a target by illuminating that target with a laser light. The name LiDAR, sometimes considered an acronym of Light Detection And Ranging.
Literature Review-Autonomous Vehicles

What are this research’s differences with autonomous vehicles?
1. Fixed LiDAR Location.
2. Have to be able to detect vehicles without good shape.
3. Have to be able to track vehicles without help (features) from optical sensors.

A typical LiDAR 3D view from an autonomous vehicle
Roadside LiDAR data processing—Overview

1. Preprocessing
2. Outlier Removal
3. Ground Segmentation
4. Clustering
5. Key Points Extraction
6. Tracking
Roadside LiDAR data processing: Raw Data
Roadside LiDAR data processing: : Preprocessing

- The csv type data was converted into a special type for 3d cloud processing, and the data within 2 meters radius was deleted directly(recommended by manufacture).
Roadside LiDAR data processing: Set Region Of Interest

ROI setting Before (Left) and After (Right)

The purpose of this step is to identify the region we care about.
Roadside LiDAR data processing: Statistical Outlier Removal

SOR performed Before (Left) and After (Right)

The purpose of this step is to remove the noises, either from data or from sensor.
Roadside LiDAR data processing: Ground Plane Identification and Segmentation

The purpose of this step is to remove the useless ground plane.
Roadside LiDAR data processing: Random sample consensus (RANSAC)

Algorithm:
1. Sample (randomly) the number $s$ of points required to fit the model;
2. Solve for model parameters using samples;
3. For all observations (points) compute the errors $M$ (such as distance of observations compared to what is predicted by the model);
4. If this $M$ within user defined error threshold, go to next step, otherwise resampling;
5. Count the numbers of inliers $I$;
6. If the $I$ is the largest right now, set this model as the best model;
7. Repeat 1-6 until the best model is found.

Algorithm:

\[
\bar{M} = \arg \min_{M} \left\{ \sum_{d \in D} \text{Loss}\left(\text{Err}(d;M)\right) \right\},
\]

\[
\text{Loss}(e) = \begin{cases} 
0 & |e| < c \\
\text{const} & \text{otherwise}
\end{cases}
\]

where $d$ is data, Loss is a loss function, and Err is a error function such as geometric distance.
Roadside LiDAR data processing: Least Median Square (LMedS)

Algorithm:
1. Sample (randomly) the number $s$ of points required to fit the model;
2. Solve for model parameters using samples;
3. For all observations (points) compute the residuals $r_i$ (distance of observations compared to what is predicted by the model);
4. Find the median residual square $r_{\text{median}}^2$;
5. If $r_{\text{median}}^2 < r_{\text{min}}^2$ then set this model the best model for now;
6. Repeat 1-5 until the best model is found.

$\text{minimize } r_{\text{median}}^2$

LMedS does not need any error threshold parameter because it tries to minimize median squared error, this feature leads to better performance in practice.

However, it could become worse where the inlier ratio is under 50% since the median will form outliers.
Roadside LiDAR data processing: Clustering

clustering results
The purpose of this step is to identify tracking points.
Roadside LiDAR data processing: Oriented Bounding Box Estimation and Key Pair Extraction

- O’Rourke’s Algorithm (Examine every pair of edge, left), Brute-Force Methods (Random testing), PCA-Based Methods (Right)
Roadside LiDAR data processing: Global Nearest Neighborhood Tracking
Tracking result

First Vehicle Front Speed

First Vehicle Back Speed
Tracking result

- Second Vehicle Front Speed
- Second Vehicle Back Speed
Tracking validation Test Site: Enterprise road
Tracking validation: 1 second OBD

The purpose of this step is to validate the generated speed with Testing vehicles OBD (On Board Device)logger.
Tracking validation Test Site: Blue parking lot
Tracking validation: 0.1 second OBD

The purpose of this step is to validate the stop-go speed with Testing vehicles OBD (On Board Device) logger. Blue points indicate tracking speed, red points indicate logger speed.
Tracking validation: 0.1 second OBD

The purpose of this step is to validate the stop-go speed with Testing vehicles OBD (On Board Device) logger. Blue points indicate tracking speed, red points indicate logger speed.
Tracking validation: 0.1 second OBD

The purpose of this step is to validate the free flow speed with Testing vehicles OBD (On Board Device) logger. Blue points indicate tracking speed, red points indicate logger speed.
Tracking validation: 0.1 second OBD

The purpose of this step is to validate the free flow speed with Testing vehicles OBD (On Board Device) logger. Blue points indicate tracking speed, red points indicate logger speed.
Case Study: Intersection of 15th street and Virginia street
Case Study: Intersection of 15th street and Virginia street
Case Study: Intersection of 15th street and Virginia street

- The datasets collected at this intersection contains hours of traffic data. A portion of this dataset was chosen to study the detailed tracking results. The criteria for choosing such a portion are followed:
  1) This portion has abundant vehicle data, both large buses and small vehicles data existed;
  2) This portion has both stop-go scenario and free flow scenario, so both scenarios could be researched;
  3) This portion has pedestrian crossing the street, thus the pedestrian tracking possibilities could be researched;
  4) This portion has left turn vehicles, so the speed profile of left turn vehicle could be generated.
Case Study: Intersection of 15th street and Virginia street

- 1000 frames were considered meet the selection criteria defined above. A sample frame was presented in the following figure.
Case Study: Vehicle tracking results
Case Study: Pedestrian tracking results
Case Study: summary of tracking results

From the manual observation, a total of 38 objects were existed in this portion of dataset. Among them, there are two pedestrian, 5 left turn vehicles, and three buses.

Among 36 vehicles, 14 vehicles travel southbound at Virginia Street, 17 vehicles travel northbound at Virginia Street, 2 Vehicles travel eastbound at 15th street, and 3 vehicles travel westbound at 15th street.

From the detection result, 39 tracking speed profiles were generated. Two southbound vehicles are counted twice; one northbound vehicle was not detected.
Case Study: multiple counts errors
Case Study: missing counts errors

This vehicle is completed blocked all the way
Case Study: summary

- In sum 37 objects were identified and tracked. Although two same objects were counted twice because blockage of near sensor vehicles, they are still correctly tracked. This give us an accuracy of tracking rate of \( \frac{37}{38} = 97.4\% \) on objects on the roads for this case study.

- The blockages are the major reasons for error detections. If multiple sensors could be deployed at both side of road, and the dataset could be ideally fused, this problem would be solved.
Object recognition from roadside LiDAR sensor - SVM

- A support vector machine constructs a hyper-plane or set of hyper-planes in a high or infinite dimensional space, which can be used for classification, regression or other tasks. Intuitively, a good separation is achieved by the hyper-plane that has the largest distance to the nearest training data points of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier.

The SVM should maximize the distance between the two decision boundaries. Mathematically, this means we want to maximize the distance between the hyperplane defined by $\mathbf{w}^T \mathbf{x} + b = -1$ and the hyperplane defined by $\mathbf{w}^T \mathbf{x} + b = 1$. This distance is equal to $\frac{2}{\|\mathbf{w}\|}$. This means we want to solve $\max_{\mathbf{w}} \frac{2}{\|\mathbf{w}\|}$. Equivalently we want $\min_{\mathbf{w}} \frac{\|\mathbf{w}\|}{2}$.

\[
\min_{\mathbf{w}, b} \frac{\|\mathbf{w}\|}{2},
\]
\[
s.t. \quad y^{(i)} (\mathbf{w}^T \mathbf{x}^{(i)} + b) \geq 1 \quad \forall i \in \{1, \ldots, N\}
\]
Object recognition from roadside LiDAR sensor – feature selection

- Selecting great features is the key for correctly identify objects. From literature review, we could learn that the features falls into three types. Point features use a histogram to describe its features. Local features defined a small area to describe the features. Global features using the whole data cloud geometric information.

- In this research, for the reason that the shape of cloud points are kept changing, hence it is very unstable to use local features or global features to describe the clouds. In this research, the three largest eigenvectors, as well as the cloud point’s number, are used as features to achieve this task.
Object recognition from roadside LiDAR sensor – feature selection training table

<table>
<thead>
<tr>
<th>Identification</th>
<th>Eigenvector1</th>
<th>Eigenvector2</th>
<th>Eigenvector3</th>
<th>Point numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.6086</td>
<td>1.37415</td>
<td>0.508575</td>
<td>80</td>
</tr>
<tr>
<td>0</td>
<td>1.6487</td>
<td>1.37721</td>
<td>0.377087</td>
<td>67</td>
</tr>
<tr>
<td>0</td>
<td>1.62817</td>
<td>1.38084</td>
<td>0.404202</td>
<td>60</td>
</tr>
<tr>
<td>0</td>
<td>1.61167</td>
<td>1.207</td>
<td>0.401193</td>
<td>62</td>
</tr>
<tr>
<td>0</td>
<td>1.61415</td>
<td>1.23093</td>
<td>0.457184</td>
<td>59</td>
</tr>
<tr>
<td>0</td>
<td>1.49094</td>
<td>1.23088</td>
<td>0.576513</td>
<td>61</td>
</tr>
<tr>
<td>0</td>
<td>1.39088</td>
<td>0.74139</td>
<td>0.373193</td>
<td>15</td>
</tr>
<tr>
<td>0</td>
<td>1.36448</td>
<td>0.583993</td>
<td>0.342423</td>
<td>14</td>
</tr>
<tr>
<td>0</td>
<td>1.39329</td>
<td>0.68799</td>
<td>0.651142</td>
<td>19</td>
</tr>
<tr>
<td>0</td>
<td>1.39027</td>
<td>0.767213</td>
<td>0.582005</td>
<td>19</td>
</tr>
<tr>
<td>0</td>
<td>1.37912</td>
<td>0.814165</td>
<td>0.649501</td>
<td>21</td>
</tr>
<tr>
<td>0</td>
<td>1.36489</td>
<td>0.687317</td>
<td>0.552423</td>
<td>18</td>
</tr>
<tr>
<td>0</td>
<td>1.34833</td>
<td>0.539579</td>
<td>0.476426</td>
<td>15</td>
</tr>
<tr>
<td>0</td>
<td>1.32924</td>
<td>0.695193</td>
<td>0.417093</td>
<td>18</td>
</tr>
<tr>
<td>1</td>
<td>3.70775</td>
<td>1.83262</td>
<td>1.24179</td>
<td>68</td>
</tr>
<tr>
<td>1</td>
<td>3.88347</td>
<td>1.84832</td>
<td>1.77437</td>
<td>102</td>
</tr>
<tr>
<td>1</td>
<td>3.94735</td>
<td>1.86505</td>
<td>1.71187</td>
<td>106</td>
</tr>
<tr>
<td>1</td>
<td>3.93755</td>
<td>1.91298</td>
<td>1.66293</td>
<td>109</td>
</tr>
<tr>
<td>1</td>
<td>3.91298</td>
<td>1.76556</td>
<td>1.59001</td>
<td>109</td>
</tr>
<tr>
<td>1</td>
<td>3.96283</td>
<td>1.77208</td>
<td>1.10843</td>
<td>109</td>
</tr>
<tr>
<td>1</td>
<td>4.0024</td>
<td>1.80707</td>
<td>1.3942</td>
<td>120</td>
</tr>
<tr>
<td>1</td>
<td>3.87637</td>
<td>1.88138</td>
<td>1.49816</td>
<td>148</td>
</tr>
<tr>
<td>1</td>
<td>11.0421</td>
<td>2.98472</td>
<td>2.28897</td>
<td>118</td>
</tr>
<tr>
<td>1</td>
<td>11.1075</td>
<td>2.99588</td>
<td>2.16623</td>
<td>114</td>
</tr>
<tr>
<td>1</td>
<td>11.1406</td>
<td>2.90252</td>
<td>2.22607</td>
<td>108</td>
</tr>
<tr>
<td>1</td>
<td>10.8162</td>
<td>3.04487</td>
<td>2.04528</td>
<td>192</td>
</tr>
<tr>
<td>1</td>
<td>10.9052</td>
<td>3.03557</td>
<td>2.12502</td>
<td>186</td>
</tr>
<tr>
<td>1</td>
<td>10.1597</td>
<td>4.61755</td>
<td>3.13038</td>
<td>323</td>
</tr>
<tr>
<td>1</td>
<td>10.3946</td>
<td>4.7539</td>
<td>3.11835</td>
<td>335</td>
</tr>
</tbody>
</table>
Object recognition from roadside LiDAR sensor – correctly identified cars.
Object recognition from roadside LiDAR sensor – correctly pedestrians/buses.
Object recognition from roadside LiDAR sensor – wrongly identified cars.
Object recognition from roadside LiDAR sensor – wrongly identified objects
Object recognition from roadside LiDAR sensor – summary.

- In sum, in this research, with the global features and RBF kernel SVM classifier, the accuracy of identification is $\frac{35}{38} = 92.1\%$. Besides, using decision tree classifier would yield same accuracy of identification.

<table>
<thead>
<tr>
<th>SVM Type</th>
<th>Bus</th>
<th>Car</th>
<th>Pedestrians</th>
<th>Bicyclist</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear Kernel</td>
<td>2 for 2</td>
<td>29 for 34</td>
<td>2 for 2</td>
<td>5 for 0</td>
<td>0.86842</td>
</tr>
<tr>
<td>RBF Kernel</td>
<td>2 for 2</td>
<td>31 for 34</td>
<td>2 for 2</td>
<td>3 for 0</td>
<td>0.92105</td>
</tr>
<tr>
<td>Poly Kernel</td>
<td>2 for 2</td>
<td>28 for 34</td>
<td>2 for 2</td>
<td>6 for 0</td>
<td>0.84211</td>
</tr>
</tbody>
</table>
Applications

1. Connected vehicles.
2. Safety.
3. Mobility.

FIGURE 3 LOS B driving cycle of the selected highway scenario example
Summary

This research developed a data processing procedure for detection and tracking of multi-lane multi-vehicle speed trajectories with a roadside Light Detection and Ranging (LiDAR) sensor. Different from existing perception methods for autonomous vehicle system, this procedure was developed specifically to extract trajectories from a roadside LiDAR sensor.

The developed procedure has been tested against intersection of Evans Street and Enterprise Road, a two way stops sign intersection; and Kietzke lane, an arterial road with 40 mph speed limit. Then, the data extraction procedure has been validated by comparing tracking results and speeds logged from a testing vehicle through the on-board diagnostics interface (OBD-I), at a parking lot of University of Nevada, Reno. The validation results suggest that the tracking speed matches real driving speed accurately.
Summary

- A case study was conducted to examine the accuracy of tracking multiple objects on the roads. 1000 data frames from intersection of 15th Street and Virginia Street was used as source data frames. The proposed data processing framework successfully tracked 37 objects out of 38 objects on the road, which gives an accuracy of 97.4%. Then a support vector machine based algorithm was developed to differentiate pedestrian/bicyclist and car/bus. This algorithm correctly classifies 35 objects among 38 objects, which gives an accuracy of 92.1%. The result of case study indicates that the proposed data processing framework has a satisfactory tracking and clustering accuracy and could be used for traffic mobility estimation.

- This data processing procedure not only could be applied to extract high-resolution trajectories for connected-vehicle applications, it could also be valuable to practices in traffic safety, traffic mobility and fuel efficiency estimation. The ordinary Rectangular Rapid Flash Beacon (RRFB) could be upgraded in order to automatically detect pedestrians, this is especially important during night time. An adaptive traffic signal plan which could adapts to special events or economic changes also become feasible from this research. Driving cycle developments, which used to largely rely on sampling vehicles, could become much more accurate because this research enables the possibilities to extract every vehicle’s speed profile. In sum, this research result provides a reliable way to accurately extract high-resolution traffic data from a roadside LiDAR sensor, and would benefit researches in connected vehicles, traffic safety, traffic mobility and fuel consumption estimation.
References


References (Continued)


Thanks!