Vehicle Detection Using LiDAR

Jacob Heapy

ESPACE - Earth Oriented Space Science and Technology
1.1 Motivation (1)

**Primary Goal:** The automatic detection and segmentation of vehicles in a region using data from an aerial Light Detection and Ranging (LiDAR) system.

**Why Vehicle Detection?**
- Critical for automated traffic monitoring in urban and non-urban areas
- Can also provide city officials:
  - Parking habits (where, when)
  - Road utilisation by vehicle type (car, van, truck)
  - Traffic flow, as described by Flow = Density x Velocity [Toth & Grejner-Brzezinska, 2005]
- All information collected can be used to improve the efficiency of current and future transportation networks
1.1 Motivation (2)

Why Aerial LiDAR?

- **Active Sensing Technique:**
  - Possible to use day and night
  - Able to penetrate vegetation coverage (ex. Tree canopies)

- **Information Content:**
  - 3D geometrical information about a scene
  - Additional information about reflective properties of surfaces

- **Motion Analysis:**
  - Linear scanning means moving objects appear deformed [Yao et al., 2011a]

- **Convenience Through Remoteness:**
  - No work crews disturb traffic flow by installing ground-based sensors
  - Cover large areas easily
1.2 Assumptions

**General Assumptions Include:**

- **Ground Points:**
  - Ground points form a normal distribution of height
  - Vehicles are outliers from the ground points and fit a prior height distribution
  - Ground slope at maximum 26° [Borcs & Benedek, 2015]

- **LiDAR Platform:** We have accurate knowledge of the motion and orientation of the LiDAR platform relative to the scene

- **Weather:** Data is free from interference caused by weather

- **Vehicles:**
  - Aspect ratio of vehicles is known and well defined
  - Extracted vehicles can be described by rectangularity, elongatedness, and from number and intensity of LiDAR returns
    - Rectangularity: measure of how well an object fits in a rectangular bounding box
    - Elongatedness: ratio of the short to long sides of that bounding box
1.3 Problems

**General Problems Include:**

- **Crowding**: Detection difficult in crowded areas like parking lots and intersections

- **Data Gaps**: Missing points in data may result from:
  - Faulty equipment
  - Poor platform control (yaw, pitch, roll)
  - Surface absorption/reflection

- **Ground Truth**: Testing accuracy against “ground truth” typically uses human interpreted “sensed truth”

- **Vehicle Size**: Small vehicles can be difficult to properly detect and lead to inaccurate velocity estimates

- **Velocities**: Slow moving vehicles can have inaccurate velocity estimates
Preliminary Overview of Methodology:

Vehicle Detection w/ LiDAR

Stationary

- Object-Based Point Cloud Analysis
  - Zhang et al., 2014
  - Eum et al., 2017
  - Yu et al. 2019

  - Grid Cell: Watershed
    - Yao et al., 2008

  - Multi-Level Marked Point Process
    - Borcs & Benedek, 2012
    - Borcs & Benedek, 2015
    - Benedek, 2017

  - Support Vector Machine
    - Zhang et al., 2013
    - Wang et al., 2017

Moving

- Motion Artifact Detection
  - Toth & Grejner-Brzezinska, 2005&2006
  - Yao & Stilla, 2011
  - Yao et al., 2011a
Stationary Vehicle Detection

- **Detection and Classification of Vehicles:**
  - Stationary vehicle detection disregards motion artifacts
  - Focus on accurately separating vehicles from complex backgrounds

- **Typical Order of Analysis:**
  1) Separate Object Points and Background
  2) Apply constraints to classify Vehicle Points
  3) Detect and segment vehicles
Preliminary Overview of Methodology:

1. Stationary
   - Object-Based Point Cloud Analysis
     - Zhang et al., 2014

2. Multi-Level Marked Point Process
   - Borcs & Benedek, 2015

3. Support Vector Machine
   - Wang et al., 2017
Stationary Vehicle Detection [Zhang et al., 2014]:

1. **LiDAR Data**
2. **Ground Surface Filtering**
   - Progressive TIN* Densification
3. **Object Classification**
   - Classify as Vehicle or Non-Vehicle.
4. **Connected Component Analysis**
   - Segment based on Fixed Distance Neighbors
5. **Vehicle Extraction**
   - Based on area, rectangularity, and elongation.

* Triangular Irregular Network
3.1 Method 1: Object-Based Point Cloud Analysis (2)

- **Step 1: Ground Surface Filtering**
  - Use a modified Progressive Triangular Irregular Network (TIN)
  - **Seed Points:**
    - Lowest point in a predefined grid
    - Defines the first TIN
    - Smoothing used to expand the number of seed points
    - The TIN is iterated to determine Ground Surface

- **Step 2: Point Classification**
  - Based on height from Ground:
    - Potential Vehicle
    - Non-Vehicle
  - Ground class is removed from dataset

[T] [Zhang et al., 2014]

TIN representation of a house from [Gorte, 2002]
3.1 Method 1: Object-Based Point Cloud Analysis (3)

☐ **Step 3: Connected Component Analysis**
  - Segment the points into different groups based on distance (FDN)
  - **Fixed Distance Neighbors (FDNs):**
    - Select seed points and find other points that belong to this segment
    - Assess neighbors of seed points using fixed distance
    - Add to the seed point segment if they match

☐ **Step 4: Vehicle Extraction**
  - For each segmented area, three parameters are calculated: Area, Rectangularity, and Elongatedness
  - Segmented areas classified as vehicles if they meet the following criteria:

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Threshold</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area</td>
<td>$2.0 &lt; a &lt; 15.0$ m$^2$</td>
</tr>
<tr>
<td>Rectangularity</td>
<td>$0.6 &lt; r &lt; 1.0$</td>
</tr>
<tr>
<td>Elongatedness</td>
<td>$0.25 &lt; e &lt; 0.65$</td>
</tr>
</tbody>
</table>

Modified from [Eum et al., 2017]
3.2 Method 2: Two-Level Point Processes (1)

Stationary Vehicle Detection [Börcs & Benedek, 2015]:

- **LiDAR Data**
  - Energy-Minimizing Classification
    - Terrain, Vegetation, Clutter, Roof, Vehicle
  - Creation of 2D Class Label Map
    - Project 3D data into 2D grid with Class and Intensity
  - Energy Assignment
    - Classify using traffic segments
  - Optimization
    - Minimize Classification Energy
  - Prior Knowledge
Step 1: Classify 3D Points

- Distinguish Terrain, Vegetation, Clutter, Roof, and Vehicle
- Each class assigned an Energy Function
  - Function used to calculate how close a point is to a class in object space
  - i.e. the lower the energy, the better the class fit
- Class fit determined by height, slope, and number of reflections
  - Ex. Terrain class when $\Delta h < 50$ cm in 1 m grid length

Step 2: Projection into 2D

- Create 2D pixel lattice on terrain model
- Project LiDAR points onto pixels
- Class and Intensity based on majority points in pixel

Modified from [Börcs & Benedek, 2015]
3.2 Method 2: Two-Level Point Processes (3)

- **Step 3: Energy Assignment**
  - Use the Two-Level Marked Point Processes (L²MPP) model to assess vehicle candidates depending on energy functions.
  - **Data-Dependant Energy**: Used to evaluate proposed vehicle candidates based on
    - Center of object segment
    - Length and orientation of rectangular bounding box
    - LiDAR point intensity
  - **Prior-Term Energy**: Used to evaluate traffic segments based on
    - Predefined knowledge of vehicle populations
    - Interaction constraints between neighboring objects

  - “Evaluates the hypothesis that [point] \( p \) belongs to \( \xi \) class, marking high quality matches with lower [energy] \( \mu \) values.”
    - 0 = Perfect Match
    - 1 = No Match
3.2 Method 2: Two-Level Point Processes (4)

- **Data-Dependant Energy:** Used to evaluate proposed vehicle candidates:
  - Return intensity, length, width, center of segment

- **Prior-Term Energy:** Used to evaluate traffic segments.
  - Relative orientation, overlap with other objects, interaction with the road network

Modified from [Börcs & Benedek, 2015]
Step 4: Optimization

Multiple Births and Deaths (MBD): Used to optimize both vehicle and traffic segment classification in four steps:

- **Birth Step: (Visit all pixels)**
  - With a certain probability generate a new object (birth)
  - Object gets random length, width, and orientation
  - Add new object to existing traffic segment if close, otherwise generate new traffic segment

- **Death Step:**
  - Consider energy function of each new object
  - Calculated Data-Dependant and Prior-Term Energy
  - Objects above a certain energy value removed (death)
3.2 Method 2: Two-Level Point Processes (6)

- **Step 4: Optimization continued...**

  - **Re-Arrangement:**
    - Add white gaussian noise to objects close together
    - Calculate the energy required to exchange the objects into a new group
    - Re-arrange if energy is low enough

  - **Convergence:**
    - If previous three steps do not converge, change birth and death thresholds.
    - Repeat until minimum energy achieved.

[Börcs & Benedek, 2015]
Stationary Vehicle Detection [Wang et al., 2017]:

- **LiDAR Data**
- **Terrain Classification**
  - Progressive Morphological Filter
- **Point Segmentation**
  - Objects from Non-Terrain and Blobs from Terrain
- **OCSVM Classification**
  - Classify Objects and Blobs using SVM
- **Vehicle Extraction**
  - Based on integration of Object and Blob classification

* Minimum Covariance Determinant
3.3 Method 3: One-Class Support Vector Machine-MCD* (2)

- **Step 1: Terrain Classification**
  - Low outliers (below terrain) removed by limiting lower $\delta h$ to all neighbours
  - Use a Progressive Morphological Filter
    - Removes non-terrain objects based on elevation difference to neighbors

- **Step 2: Point Segmentation**
  - Objects segmented from the Non-Terrain data
    - Neighborhood Distance threshold
    - Minimum Object Points threshold
  - Blobs segmented from terrain data
    - Extract areas of missing terrain data
    - Filter out small missing areas using opening operation

* Minimum Covariance Determinant
3.3 Method 3: One-Class Support Vector Machine-MCD* (3)

[Wang et al., 2017]
3.3 Method 3: One-Class Support Vector Machine-MCD* (4)

- **Step 3: OCSVM Classification**
  - Used when only one class is of interest and a decision must be made between true and false classification
  - Build a hyperplane in Feature Space that has maximum separation between Vehicle Class and Non-Vehicle class
    - Plane is “learned” using training data of vehicle data
  - **Minimum Covariance Determinant (MCD)**
    - Used to optimize the “learned” hyperplane
    - State-of-art covariance estimator for multidimensional Gaussian data
    - Assumes data is an elliptical distribution
    - Allows for a larger class distribution than just Gaussian data

[Wang et al., 2017]
3.3 Method 3: One-Class Support Vector Machine-MCD* (5)

- **Step 3: OCSVM Classification con...:**
  - Object Classification (Non-Terrain) features
    - Area, Elongatedness, Planarity, Vertical Position, and Vertical Range
  - Blob Classification (Terrain Missing Data) features
    - Area, Max-Length, Min-Length, and Rectangularity

- **Step 4: Vehicle Extraction**
  - Combine results of Object and Blob Classifications
  - When both results agree on object, extraction complete
  - If methods disagree, preference given to method based on number of points available for classification
    - If few points available, result of Blob classification used. If many points available, result of Object classification used

* Minimum Covariance Determinant

[Wang et al., 2017]
4.1 Dataset from Zhang et al. [2014]

- **Provider**: John Chance Land Surveys (private source)
- **Sensor Type**: FLI-MAP 400 (Pulsed LiDAR)
- **Point Density**: Average 40 pts/m²
- **Area of dataset**: 40 000 m² (200m x 200m)
- **Number of Vehicles (Total)**: 136

**Ground Truth**: Human interpretation

Modified from [Zhang et al., 2014]
4.2 Dataset from Börcs & Benedek [2015]

- **Provider**: Astrium GEO-Inf. Services Hungary (private source)
- **Sensor Type**: Aerial Discrete Return LiDAR
- **Point Density**: Average 8 pts/m²
- **Area total**: 319 000 m²
- **Number of Datasets**: 7
- **Number of Vehicles (Total)**: 1009

<table>
<thead>
<tr>
<th>Feature / Data set</th>
<th>#1</th>
<th>#2</th>
<th>#3</th>
<th>#4</th>
<th>#5</th>
<th>#6</th>
<th>#7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main road traffic</td>
<td>×</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>×</td>
</tr>
<tr>
<td>Roadside parking</td>
<td>×</td>
<td>×</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parking square</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td></td>
<td></td>
<td></td>
<td>×</td>
</tr>
<tr>
<td>Curved Road</td>
<td>×</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cluttered traffic</td>
<td>×</td>
<td>×</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>×</td>
</tr>
<tr>
<td>Area in 10^{-3}km²</td>
<td>46</td>
<td>65</td>
<td>39</td>
<td>47</td>
<td>37</td>
<td>39</td>
<td>46</td>
</tr>
<tr>
<td>Point num ·10⁴</td>
<td>45</td>
<td>33</td>
<td>35</td>
<td>38</td>
<td>27</td>
<td>36</td>
<td>36</td>
</tr>
</tbody>
</table>

Modified from [Börcs & Benedek, 2015]
4.3 Dataset from Wang et al. [2017]

- **Provider**: ISPRS for urban object classification *Vaihingen* (open source)
- **Sensor Type**: Leica ALS50 (Pulsed LiDAR)
- **Point Density**: Average 4 pts/m²
- **Area per Dataset**: Not specified
- **Number of Vehicles (Total)**: 287

Modified from [Wang et al., 2017]
5 Results: General Evaluation Formulas

- All three methods evaluated based on Precision, Recall, and F-Score
  - Based on TP (True Positives), FP (False Positives), FN (False Negatives)

- Precision (Correctness)
  - Error of commission (user’s accuracy)
  - How much of the classified data should we trust
  - Precision = TP/(TP+FP)

- Recall (Completeness)
  - Error of omission (producer’s accuracy)
  - How much of the actual data was classified correct
  - Recall = TP/(TP+FN)

- F-Score: “harmonic mean of precision and recall”
  - Maximum penalization for low precision or recall
  - Combined metric used in optimization
  - F-Score = 2*(Precision*Recall)/(Precision+Recall)
5.1 Results Method 1: Object-Based Point Cloud Analysis

- **Methods**
  - **Our OBPCA method**

- Tested at 3 different point densities
  - Correctness (Precision)
  - Completeness (Recall)

- **F-Score**: “harmonic mean of precision and recall”
  - F-Score = \(2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}\)

<table>
<thead>
<tr>
<th>Density ((\text{pt/m}^2))</th>
<th>F-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>82.3%</td>
</tr>
<tr>
<td>20</td>
<td>78.7%</td>
</tr>
<tr>
<td>10</td>
<td>69.4%</td>
</tr>
</tbody>
</table>

Modified from [Zhang et al., 2014]
5.2 Results Method 2: Two-Level Point Processes

<table>
<thead>
<tr>
<th>Set</th>
<th>NumV*</th>
<th>Object level F-rate %</th>
<th>Pixel level F-rate%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>D-PCA</td>
<td>h-max</td>
</tr>
<tr>
<td>All</td>
<td>1009</td>
<td>77</td>
<td>82</td>
</tr>
</tbody>
</table>

*NumV = Number of real Vehicles in the test set

Modified from [Börcs & Benedek, 2015]

- F-rate (F-score) is the “harmonic mean of precision and recall”
  - F-rate = 2*(Precision*Recall)/(Precision+Recall)

- Object Level = How well method segments objects
- Pixel Level = How well method classifies pixels **AND** segments them to objects
5.3 Results Method 3: One-Class Support Vector Machine-MCD*

OCSVM-MCD

Modified from [Wang et al., 2017]

- 2 LiDAR Datasets: (ISPRS Vaihingen at 4 pts/m²)

- F-rate (F-score) is the “harmonic mean of precision and recall”
  - F-rate = 2*(Precision*Recall)/(Precision + Recall)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>F-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>95.1%</td>
</tr>
<tr>
<td>II</td>
<td>91.6%</td>
</tr>
</tbody>
</table>

* Minimum Covariance Determinant
6.1 Discussion Method 1: Object-Based Point Cloud Analysis (1)

**Shortcomings of Method:**

- **Sensitive to Density:** Method is sensitive to changes in point-cloud density.
  - F-Score reduced 13% by change from 40 to 10 pt/m²
  - For LiDAR, 40 pt/m² can be impractical for large areas

- **Difficulty with Vegetation:** Method is unable to detect vehicles under vegetation

**Violated Assumptions:**

- **Shape of Return:** As point density decreases, small defining features disappear.
  - Criteria for “vehicle” is too broad, resulting in incorrect segmentation and false positives
  - Due to reliance on strict definitions of Rectangularity and Elongatedness

**What this Method Delivers:**

- Vehicle detection from **high density** LiDAR data, in **non-vegetated** regions, with an **F-Rate** of ~82%

*[Zhang et al., 2014]*
Proposed Improvements to Method [Eum et al., 2017]:

- Decision algorithm for detection
  - Same ground filtering and connected component analysis (segmentation) as Zhang et al. [2014]
  - Fixed violations due to reliance on strict elongatedness and rectangularity:
    - All thresholds removed
    - Detection using complex decision tree algorithm
  - Decision tree created from training data
    - Considers vehicle’s area, elongatedness, and rectangularity in horizontal and vertical

- Shows that Object-Based Point Cloud Analysis can be improved and should not be dismissed as a topic

- Eum at al. [2017] still relied on 20 pt/m² and managed an F-score of 85%, not addressing all issues with method

6.1 Discussion Method 1: Object-Based Point Cloud Analysis (2)
6.2 Discussion Method 2: Two-Level Point Processes (1)

**Shortcomings of Method:**
- Dependence on number of returns and intensity of returns
  - If data is not available, the accuracy is affected
  - Vehicles of different compositions may have different return intensities
    - Ex. Vehicles with glass roofs (Teslas, Mercedes, etc.)

**Violated Assumptions:**
- None

**What this Method Delivers:**
- Vehicle detection from LiDAR data that considers **intensity and number of returns**, in all urban regions, with an object-level **F-Score** of ~97%
- **Traffic segments** based on vehicle group orientation
- Expandable ability to detect **rooftop parking** and **vehicle motion**

[Börcs & Benedek, 2015]
Further Evaluation: [Börcs & Benedek, 2015]

☐ To test the robustness of method and assumptions, authors evaluated:

❖ Relevance of Energy Terms
  ▪ Method evaluation performed five times, each time removing a different energy term (example: return intensity)
  ▪ Removing an energy term reduces the number of equations to be optimized
  ▪ F-Score calculated from each different evaluation, showing how detection accuracy depends on the energy term
  ▪ Method was least sensitive to the intensity return energy term (4% reduced), most sensitive to the bounding box energy term (31% reduced)

❖ Dependence on Point Cloud Density
  ▪ Evaluation performed on downscaled point cloud of 4 pts/m²
  ▪ F-Score reduced 4% at object level and 3-7% at pixel level
6.3 Discussion Method 3: One-Class Support Vector Machine-MCD* (1)

Shortcomings of Method: [Wang et al., 2017]
- Difficulty with Vegetation: Method is unable to consistently detect vehicles under vegetation
- Close Vehicles: Method tends to group close vehicles into one larger vehicle

Violated Assumptions:
- None

What this Method Delivers:
- Vehicle detection from low density LiDAR data, in non-vegetated regions, with an F-Score of ~93%
  - Allows for data with gaps/missing regions
  - Does not require information on intensity or number of returns

* Minimum Covariance Determinant
Further Evaluation:

- To test the robustness of OCSVM-MCD as a classifier, authors evaluated it using seven benchmark UCI datasets used for testing classification algorithms.
- In each benchmark, method was compared to OCSVM without the MCD and the FOCSVM as classic classification methods.
- OCSVM-MCD outperformed both other methods in each benchmark test.

[Wang et al., 2017]

<table>
<thead>
<tr>
<th>Dataset</th>
<th># Features</th>
<th># Objects</th>
<th># Train Targets</th>
<th># Test Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Biomed</td>
<td>5</td>
<td>194</td>
<td>102</td>
<td>25/67</td>
</tr>
<tr>
<td>Breast</td>
<td>9</td>
<td>699</td>
<td>367</td>
<td>91/241</td>
</tr>
<tr>
<td>Heart</td>
<td>13</td>
<td>303</td>
<td>123</td>
<td>41/139</td>
</tr>
<tr>
<td>Import</td>
<td>25</td>
<td>159</td>
<td>71</td>
<td>17/71</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>34</td>
<td>351</td>
<td>180</td>
<td>45/126</td>
</tr>
<tr>
<td>Sonar</td>
<td>60</td>
<td>208</td>
<td>89</td>
<td>22/97</td>
</tr>
<tr>
<td>Arrhythmia</td>
<td>278</td>
<td>420</td>
<td>190</td>
<td>47/183</td>
</tr>
</tbody>
</table>

* Minimum Covariance Determinant
Recall the Primary Goal: “The automatic detection and segmentation of vehicles in a region using data from an aerial LiDAR system.”

- Do the methods achieve this goal? Yes, ...to a certain degree.

- **Method 1**: Not comparable, due to reliance on high density LiDAR
  - Detection and Segmentation: At 10 pts/m², F-Score is 69%

- **Method 3**: Does well in achieving goal.
  - LiDAR Data: Accounts for degraded LiDAR data, uses only point cloud
  - Detection and Segmentation: At 4 pts/m², F-Score is 93%
    - Region: Not capable of detection in regions with dense vegetation

- **Method 2**: Achieves goal.
  - LiDAR Data: Uses all aspects of LiDAR data (intensity, return number)
  - Detection and Segmentation: At 8 pts/m², F-Score is 97%
  - Region: Capable of detection in regions with dense vegetation
7 Opinion: Vehicle Detection with LiDAR (1)

☐ **Is it possible?**
  ☐ Yes, Methods 2 & 3 show that vehicle detection is possible with F-scores in mid- to high-90% range under “reasonable” assumptions:
    ▪ Approximate size, shape, and LiDAR signal response of vehicles

☐ **What is it useful for?**
  ☐ Single pass vehicle detection or short-term (<1 day) traffic monitoring.
  ☐ NOT very useful for long-term traffic monitoring when airborne.

☐ **New Research.**
  ☐ 2019 - Yu et al.: Object Extraction using 3D Point Clouds
    ▪ Extract light poles, trees, and vehicles from mobile LiDAR (MLS)
    ▪ Method used for vehicle detection, but also applied to other classification data
  ☐ 2017 - Benedek: Marked Point Process
    ▪ Analyzed buildings from aerial images, automated PCB quality control using images, and vehicle detection from ALS and MLS supplemented with RGB
Future of this Field?

- Object detection and classification using LiDAR
  - Many objects simultaneously (not just vehicles)
  - Both mobile and aerial LiDAR (MLS and ALS)

- Vehicle detection using LiDAR will be a biproduct of mapping for the purpose of extracting whole scenes of objects

- LiDAR will be used in conjunction with other methods (like optical imagery) to improve classification

From [Yu et al., 2019]

From [Benedek, 2017]
References (1)


References (3)


