

UC Irvine

UC Irvine Previously Published Works

Title

Truck body type classification using a deep representation learning ensemble on 3D point sets

Permalink

<https://escholarship.org/uc/item/25h8g6vk>

Authors

Li, Yiqiao
Allu, Koti Reddy
Sun, Zhe
et al.

Publication Date

2021-12-01

DOI

"10.1016/j.trc.2021.103461

Peer reviewed

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/355826629>

Truck Body Type Classification using a Deep Representation Learning Ensemble on 3D Point Sets

Article in *Transportation Research Part C Emerging Technologies* · November 2021

DOI: 10.1016/j.trc.2021.103461

CITATIONS

0

READS

122

6 authors, including:



Yiqiao Li

University of California, Irvine

11 PUBLICATIONS 4 CITATIONS

SEE PROFILE



Koti Reddy Allu

University of California, Irvine

3 PUBLICATIONS 0 CITATIONS

SEE PROFILE



Zhe Sun

University of California, Irvine

3 PUBLICATIONS 0 CITATIONS

SEE PROFILE



Yeow Chern Andre Tok

University of California, Irvine

31 PUBLICATIONS 270 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Enhanced Truck Counts for California [View project](#)



Mobility of Older Adults in California: In the Era of Shared Mobility [View project](#)

1 **Truck Body Type Classification using a Deep Representation Learning Ensemble on 3D Point Sets**

2

3 **Yiqiao Li, Corresponding Author**

4 Ph.D. Candidate
5 Department of Civil and Environmental Engineering
6 Institute of Transportation Studies
7 4000 Anteater Instruction and Research Building (AIRB)
8 University of California, Irvine
9 Irvine, CA 92697
10 yiqiall@uci.edu

11

12 **Koti Reddy Allu**

13 Ph.D. Candidate
14 Department of Civil and Environmental Engineering
15 Institute of Transportation Studies
16 4000 Anteater Instruction and Research Building (AIRB)
17 University of California, Irvine
18 Irvine, CA 92697
19 kallu@uci.edu

20

21 **Zhe Sun, Ph.D.**

22 Systems Manager
23 Institute of Transportation Studies
24 4000 Anteater Instruction and Research Building (AIRB)
25 University of California, Irvine
26 Irvine, CA 92697
27 zhes@uci.edu

28

29 **Andre Y.C. Tok, Ph.D.**

30 Testbeds Manager and Assoc. Project Scientist
31 Institute of Transportation Studies
32 4000 Anteater Instruction and Research Building (AIRB)
33 University of California, Irvine
34 Irvine, CA 92697
35 ytok@uci.edu

36

37 **Guoliang Feng**

38 Ph.D. Student
39 Department of Civil and Environmental Engineering
40 Institute of Transportation Studies
41 4000 Anteater Instruction and Research Building (AIRB)
42 University of California, Irvine
43 Irvine, CA 92697
44 guolianf@uci.edu

45

46 **Stephen G. Ritchie, Ph.D.**

47 Professor of Civil Engineering and
48 Director, Institute of Transportation Studies
49 4000 Anteater Instruction and Research Building (AIRB)
50 University of California, Irvine
51 Irvine, CA 92697
52 sritchie@uci.edu

53

54

1 **Abstract**

2 Understanding the spatiotemporal distribution of commercial vehicles is essential for facilitating
3 strategic pavement design, freight planning, and policy making. Hence, transportation agencies
4 have been increasingly interested in collecting truck body configuration data due to its strong
5 association with industries and freight commodities, to better understand their distinct operational
6 characteristics and impacts on infrastructure and the environment. The rapid advancement of Light
7 Detection and Ranging (LiDAR) technology has facilitated the development of non-intrusive
8 detection solutions that are able to accurately classify truck body types in detail. This paper
9 proposes a new truck classification method using a LiDAR sensor oriented to provide a wide field-
10 of-view of roadways. In order to enrich the sparse point cloud obtained from the sensor, point
11 clouds originating from the same truck across consecutive frames were grouped and combined
12 using a two-stage vehicle reconstruction framework to generate a dense three-dimensional (3D)
13 point cloud representation of each truck. Subsequently, PointNet – a deep representation learning
14 algorithm – was adopted to train the classification model from reconstructed point clouds. The
15 model utilizes low-level features extracted from the 3D point clouds and detects key features
16 associated with each truck class. Finally, model ensemble techniques were explored to reduce the
17 generalization error by averaging the results of seven PointNet models and further enhancing the
18 overall model performance. The optimal number of models in the ensemble was determined
19 through a comprehensive sensitivity analysis with the consideration of the average correct
20 classification rate (CCR), the variability of the prediction results, and the computation efficiency.
21 The developed model is capable of distinguishing passenger vehicles and 29 different truck body
22 configurations with an average CCR of 83 percent. The average correct classification rate of the
23 developed method on the test dataset was 90 percent for trucks pulling a large trailer(s).

24 **Keywords:** Light Detection and Ranging (LiDAR), Classification, Deep representation learning,
25 PointNet, Model Ensemble

26

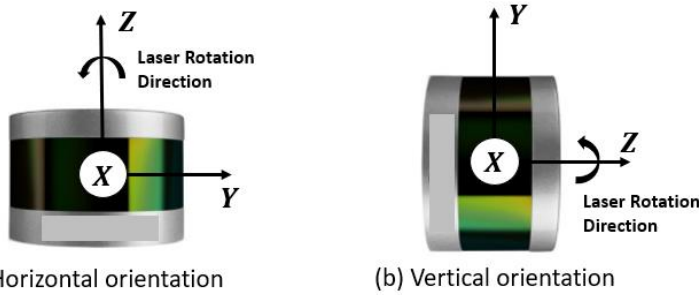
27

28

1 **1. Introduction**

2 Data from the Bureau of Transportation Statistics (BTS) shows that transportation
3 accounted for approximately 8-9 percent of the Gross Domestic Product (GDP) in the U.S. from
4 2010 to 2018 (1), where trucks were the primary contributor that accounted for 80-90 percent
5 across all modes (2). To facilitate sustained economic growth, investments in the highway
6 infrastructure need to adequately accommodate future growth, which requires strategic freight
7 infrastructure planning, effective freight forecasting, and informed policy making (3).
8 Transportation agencies demand high-quality industry-affiliated truck classification data for
9 pavement design (4), truck activity forecasts (5), and to assess the effectiveness of the freight-
10 related policies (6). Existing Weigh-In-Motion (WIM) and Automatic Vehicle Classifier (AVC)
11 systems classify vehicles based on the Federal Highway Administration (FHWA) vehicle
12 classification scheme (7), which categorizes vehicles according to their axle configurations.
13 However, commodity and industry affiliations cannot be effectively inferred from truck axle
14 configuration. Hernandez et al. (8) explored the integration of WIM and inductive loop signature
15 data to classify trucks based on their body configurations, which are associated with the
16 commodity carried and operational characteristics. Although inductive loop sensors are widely
17 deployed in many urban and interstate highway corridors in the US, their coverage remains limited
18 along rural highway corridors that also contribute significantly to the economy (8). However, the
19 implementation of pavement intrusive sensors across the extensive rural highway network is
20 impractical. Therefore, researchers have begun to investigate non-intrusive solutions for collecting
21 vehicle classification data. The rapid advancement of Light Detection and Ranging (LiDAR)
22 technology in recent years provides further opportunities for non-pavement intrusive alternatives
23 to collect detailed vehicle classification data. LiDAR-based vehicle classification was first
24 investigated in the early 2000s (9, 10). Early laser sensors scanned the cross-section of the roadway
25 by taking several range measurements as vehicles traversed the scanning area. Overhead
26 mountings were mostly adopted to capture detailed features from each passing vehicle, (9-11).
27 However, the real-world implementation of such systems is limited by infrastructure availability.
28 Lee and Coifman (13) made an early attempt to develop a side-fire laser scanner for vehicle
29 classification, where a probe vehicle equipped with two laser scanners was parked by the side of a
30 highway. Both laser scanners were mounted side-by-side to scan the road section vertically. The
31 data points obtained from vehicles were grouped into vehicle clusters and eight high-level features
32 were directly extracted to portray the physical characteristics of the vehicle from the vehicle
33 clusters. A decision tree classifier was then used to assign the high-level feature vectors into six
34 distinct classes refined from a length-based classification scheme. However, such laser scanners
35 have an extremely narrow field-of-view, since each laser only scans along a vertical slice across
36 the roadway. When the laser detection zone is occupied by the vehicle traveling at the outermost
37 lane (closest to the shoulder), adjacent vehicles traveling on the corresponding inner lanes will be
38 completely occluded, which could result in significant information loss from other travel lanes. In
39 addition, researcher also explored using low-cost single-beam side-fire LiDAR to get truck body
40 information. Asborno et al. grouped the raw distance measurements from the LiDAR sensor over time to
41 build vehicle signatures and adopted Bayesian combined predictor to classify trucks based on their
42 aggregated body type classes (12). The roadside setup of the LiDAR sensors has been proved to be a cost-
43 efficient and feasible solution for collecting classification data. However, neither the laser scanner nor the
44 single-beam LiDAR sensor is able to provide detailed vehicle profile, which limited the classification
45 accuracy and the total number of vehicles that can be classified.

1 Further advances in vehicle classification research accompanied the advent of LiDAR systems
2 with rotating sensor arrays. These units are typically equipped with 16 or more lasers mounted on
3 a rotating platform that is designed to capture a 3D point cloud representation of the environment.
4 Sahin et al. (14) deployed this advanced LiDAR sensor unit in a vertical orientation to classify
5 specific truck trailer configurations in detail. The vertical orientation of the LiDAR sensor allowed
6 them to directly obtain a denser representation of each vehicle object compared to a horizontal
7 orientation (Figure 1).



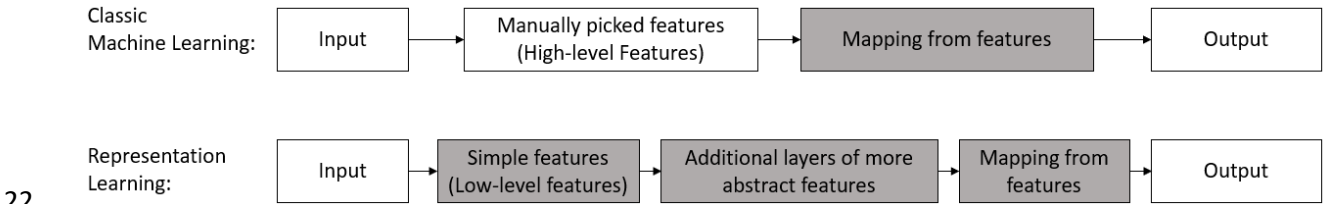
8
9 Figure 1 Illustration of LiDAR Orientation (13)

10 However, the vertical scan limited the horizontal field-of-view of the LiDAR to 45 degrees which
11 is insufficient to capture entire trucks within a single frame. This was addressed by merging
12 multiple frames of a detected vehicle to construct a full 3D profile. The 3D truck profile was
13 subsequently divided into six equal-sized rectangular voxels (volumetric pixels), and high-level
14 features representing physical characteristics of different trailer types were extracted from them.
15 Their model was able to classify 10 different trailers with accuracies ranging from 85 percent to
16 98 percent. The basic assumption of their method was that each truck traverses the LiDAR
17 detection zone (LDZ) with a constant speed. Consequently, the speed changes of the detected
18 vehicle could degrade the performance of their classification. In addition, due to the limited
19 horizontal field-of-view associated with the vertical orientation of the sensor, this approach would
20 experience similar occlusion limitations as that of Lee and Coifman (14). In order to obtain a wider
21 LDZ, Wu et al. (16) adopted a horizontal orientation of the LiDAR sensor for vehicle classification.
22 Unfortunately, the sparse point cloud representation retrieved from the horizontally oriented
23 LiDAR gave insufficient information for detailed truck classification. Their method was only able
24 to classify 8 types of vehicles, where some of the truck-related classes were not well-recognized
25 in their test dataset (15).

26 In summary, the vertical orientation of the LiDAR sensor provided a denser representation of each
27 vehicle object while the field-of-view of the sensor was restricted. Meanwhile, the model
28 performance could vary with traffic conditions due to the constant-speed assumption made in those
29 methods (16, 17). Conversely, the horizontal orientation of the sensor broadens the LDZ, but such
30 placement only captures a sparse point cloud for each vehicle object which gives insufficient
31 information to classify trucks in detail. A dense representation of vehicle objects with a horizontal
32 orientation of the LiDAR unit has not been explored in the literature, which is one of the major
33 motivations of our study. From a classification method perspective, most of the studies extracted
34 high-level features which describe the physical characteristics (e.g. vehicle height, vehicle length,
35 vehicle middle drop, the point density of truck refrigeration unit, etc.) of each predetermined
36 vehicle class from the regularized point cloud, such as 2D projections (9–11) and rectangular
37 voxels (17), or directly from the clustered sparse point cloud (15). Subsequently, classic machine

1 learning algorithms were used to map high-level feature vectors to the predetermined vehicle
2 classes. These methods treat the feature selection and classification process as two separate
3 procedures, where the feature selection process is somehow independent of the classification
4 algorithms. Hence, the optimality of the adopted features is unknown, which may limit the
5 performance of the approach (17).

6 This study investigated a new truck classification method using a LiDAR sensor array in a
7 horizontal orientation, utilizing a reconstruction procedure that combines frames of sparse point
8 clouds to generate a dense point cloud representation of vehicle objects to facilitate accurate truck
9 classification, while preserving the panoramic LDZ. First, vehicle point clouds were extracted by
10 removing the background and clustering the residual points into objects. Then, a two-stage vehicle
11 point cloud reconstruction framework (18) was adopted to enrich the sparse point cloud obtained
12 from the horizontally oriented sensor. Objects associated with the same vehicle from consecutive
13 frames were grouped and combined to generate a dense 3D point cloud representation of each
14 vehicle. In contrast to previous studies, which used a classic machine learning framework, this
15 research adopted the PointNet deep representation learning algorithm to train the classification
16 model from the preprocessed point cloud data. The difference between representation learning and
17 classic machine learning is illustrated in Figure 2. The algorithm extracted low-level features from
18 the point clouds and identified critical features during the training process. Finally, two model
19 ensemble techniques, simple model averaging and Bayesian model averaging, were explored to
20 reduce the variability of model predictions for the test dataset and further enhance the model
21 performance.



22
23 Figure 2 Difference between Classic Machine Learning and Representation Learning (19)

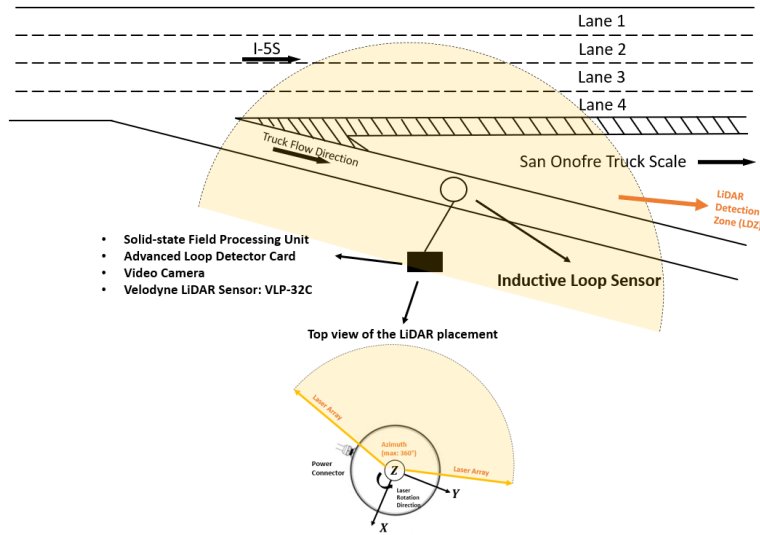
24 The remaining sections of this paper are organized as follows. The second section describes the
25 data collection effort and the dataset used in this study. The third section presents the truck
26 classification framework from the preprocessing step to the model training step. The fourth section
27 discusses the classification results and compares the model with the latest LiDAR-based truck
28 classification models from literature (17).

29

30 **2. Data Description**

31 **2.1 Study Site Layout**

32 The data used in this study were collected from the entrance ramp to the San Onofre truck scale
33 from the Southbound I-5 Freeway in Southern California (as shown in Figure 3). The LiDAR
34 sensor is horizontally placed on the top of the control cabinet. The z-axis of the LiDAR sensor is
35 aligned perpendicular to the truck flow direction. The yellow sector presented in Figure 3
36 illustrates the approximated LiDAR Detection Zone (LDZ). Data under free flow and congested
37 conditions were observed at the study site and included in the model development.

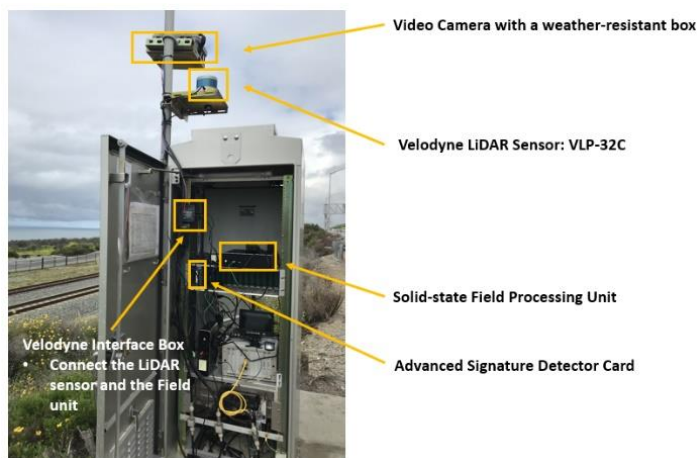


1
2 Figure 3 Layout of the Detection Site

3 2.2 Data Collection Setup

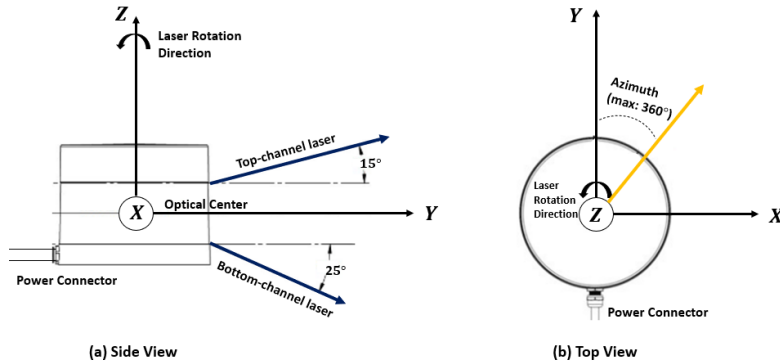
4 A video camera, an advanced loop detector card, and a Velodyne VLP-32c LiDAR unit were
 5 installed at the study site as shown in Figure 4. The lateral distance between the centroid of the
 6 LiDAR sensor and the adjacent traffic lane is approximately 6 meters. All three sensors were
 7 connected to a solid-state field processing unit. The video camera and loop detector were used to
 8 establish data groundtruth. The Velodyne VLP-32c sensor has 32 infra-red lasers paired with infra-
 9 red detectors mounted on a motorized rotating platform to provide distance measurement between
 10 the sensor and objects (13).

11 The LiDAR was configured to scan the surroundings at a frequency of 10 rotations per second and
 12 180-degree LDZ, with each rotation generating a single 3D point cloud frame. The LiDAR sensor
 13 was horizontally mounted on a platform attached to the roadside pole of an existing traffic control
 14 cabinet.



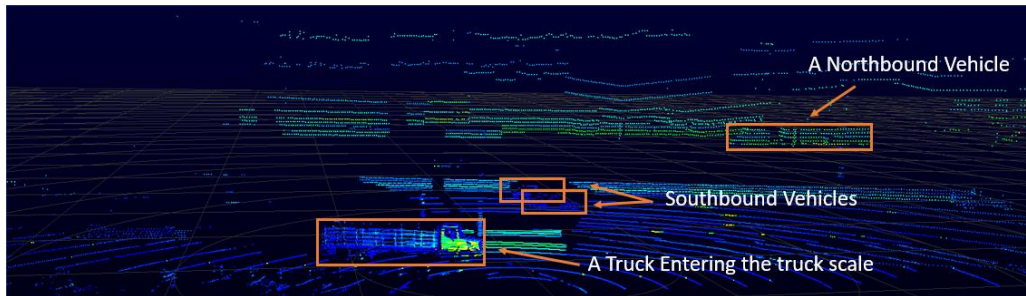
15
16 Figure 4 System Setup

1 The LiDAR sensor was mounted 2.05 meters above the ground plane and the top laser channel
2 elevation angle was 15 degrees (Figure 5), which allowed the sensor to capture both the top and
3 side view of passing vehicles. As shown in Figure 5, the z -axis is perpendicular to the direction of
4 the traffic and the laser array rotates about the z -axis.



5 (a) Side View
6 (b) Top View
7 Figure 5 Illustration of the LiDAR Sensor

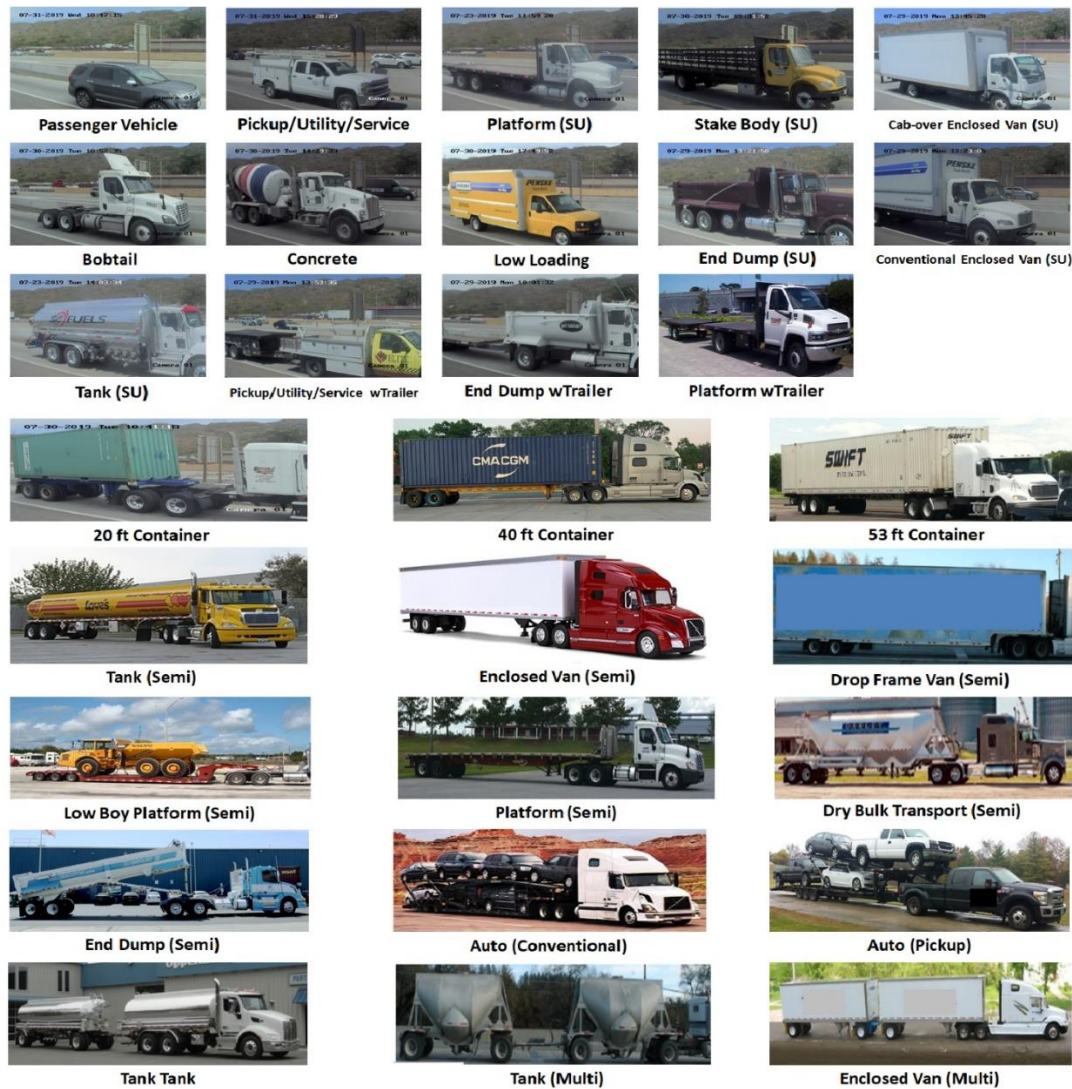
8 A sample of raw point cloud data frame showing a vehicle entering the truck scale within the LDZ
9 is presented in Figure 6.



10 Figure 6 The Raw Point Cloud of the Detection region

11 12 2.3 Data Description

13 The video data from the camera, inductive loop signature data from the loop detector, and point
14 cloud data from the LiDAR sensor, were collected simultaneously. The data were collected from
15 multiple time periods across 7 days between July 18th, 2019, and July 31st, 2019. Overall, point
16 clouds associated with 7,655 vehicles were labeled, representing 29 different types of trucks
17 (including a class labeled as “Other,” which represented trucks not belonging to any of the classes
18 defined in Figure 7) as well as passenger vehicles (Figure 7). 70 percent of the data were used for
19 training while the remaining 30 percent were reserved for testing. In addition, a temporally
20 independent test dataset was collected on 8/1/2019. This dataset was continuously collected from
21 8:20 in the morning to 20:00 in the evening. During this data collection period, 1,472 vehicle
22 records were captured by loop detectors and have been labeled via the corresponding video images
23 while 1,489 vehicle objects were detected by the LiDAR sensor using our algorithm. This temporal
24 continuous and independent test set was used to validate the truck classification method from the
25 preprocessing step to the classification algorithm.

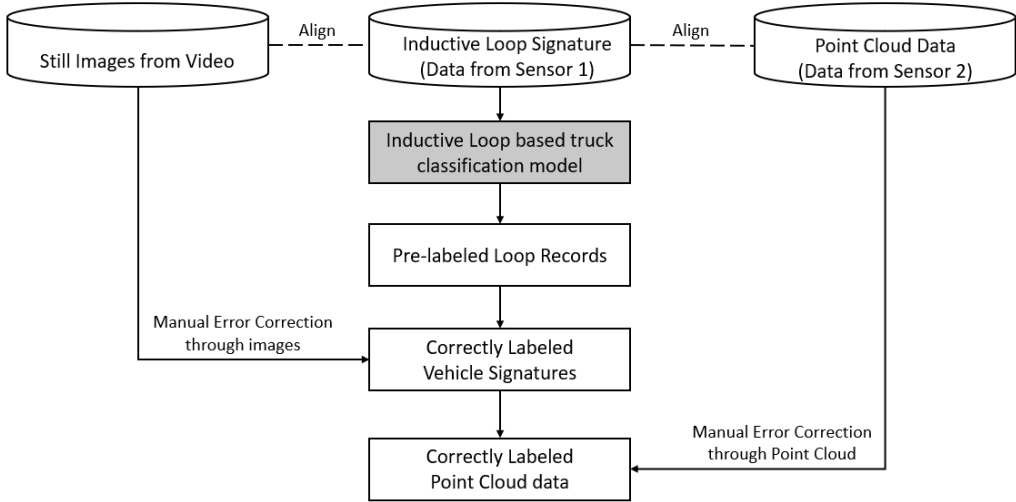


1
 2 Figure 7 Illustration of Vehicle Body Configuration used in the study (Note: SU: Single-Unit Trucks, Semi:
 3 Tractors pulling Semi-Trailer, Multi: Tractors pulling Multiple Trailers. Tank Tank: Tank tractor pulling a
 4 tank trailer)

5
 6 **2.4 Semi-automatic Data Labeling Method**

7 Data labeling has been considered a critical but labor-intensive process in many existing studies
 8 on vehicle classification. Conventionally, the data labeling process requires a large amount of time
 9 to visually verify the body configuration of each detected vehicle through images from the video
 10 camera, and then to manually record the corresponding vehicle characteristics. In this study, a
 11 semi-automatic data labeling strategy was developed to improve the efficiency of the data labeling
 12 process and further enrich the training dataset. The semi-automatic data labeling method using an
 13 inductive loop sensor was intended as an efficient data labeling mechanism to train the LiDAR-
 14 based classification model. Therefore, inductive loop sensors are not required for the LiDAR-
 15 based model implementation and calibration.

1 The overall semi-automatic data labeling process is illustrated in Figure 8. First, the vehicle records
 2 from the three data sources were synchronized and aligned. When a vehicle triggered the inductive
 3 loop sensor, its corresponding images from the video camera were cropped based on the
 4 synchronized timestamp. A time window with a range of ± 0.1 seconds around the time instance
 5 that the vehicle hit the loop sensor was created. The vehicle point cloud object which contained
 6 frames belonging to the time window was subsequently aligned with the vehicle image. Then, this
 7 study adopted a signature-based truck classification model to pre-label vehicle images (20).
 8 Afterwards, misclassified data were manually corrected from visual validation of the pre-labeled
 9 dataset. Finally, the labels were applied to corresponding point clouds as the data sources were
 10 aligned. The signature-based truck classification model had an overall 72.4 percent accuracy (20).
 11 Therefore, the labels of only around 30 percent of vehicles needed to be corrected, which
 12 significantly reduced the workload of data processing. This semi-automatic data labeling strategy
 13 greatly accelerated the labeling process for emerging sensors (Sensor 2 in Figure 8) with the
 14 support of the existing technology (Sensor 1 in Figure 8).

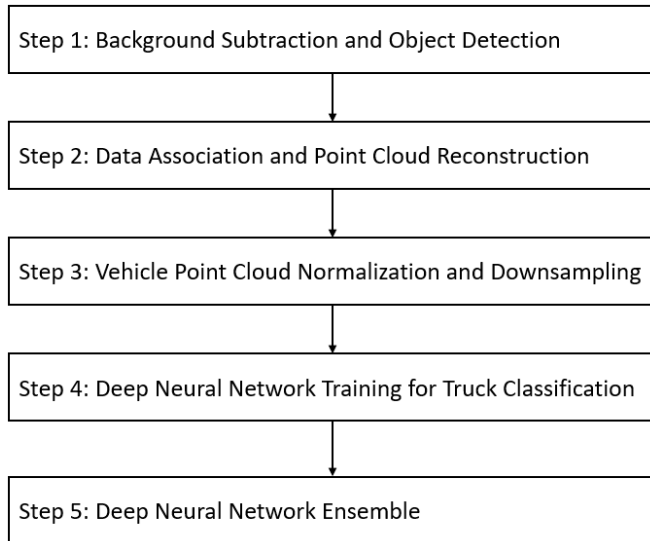


15
 16 Figure 8 Semi-automatic Data Labeling Framework

17
 18
 19
 20
 21
 22
 23
 24
 25

1 **3. Methodology**

2 This section details the data processing procedure and the classification framework developed in
3 this study. The overall workflow of our truck classification method is shown in Figure 9.



4

5 Figure 9 The Workflow of the New LiDAR-based Truck Classification Method

6

7

8 **3.1 Data preprocessing**

9 **3.1.1 Background Subtraction and Object Detection**

10 A large set of points in the raw point cloud was associated with static objects, e.g. ground, buildings.
11 These background points were required to be subtracted from the raw point cloud to reduce the
12 data processing time and improve the accuracy of the proposed classification method.

13 A point cloud background subtraction method was developed based on spatial occupancy. The
14 algorithm started with aggregating the point cloud data over an initiation period t . The points
15 originating from each of the 32 laser channels of the LiDAR were distributed on a conical surface,
16 with the LiDAR unit being the apex. The elevation angles of the channels are listed below:

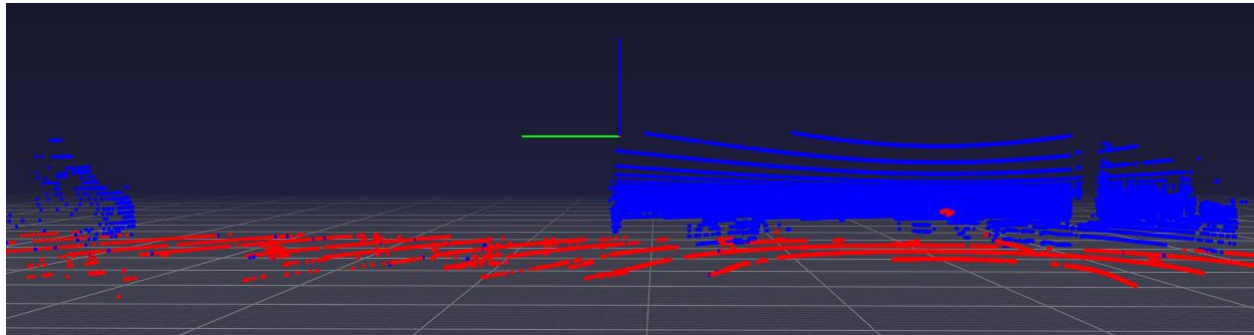
1 Table 1 Laser Channel Angles

ID	Elevation Angles (deg)	ID	Elevation Angles (deg)
0	15	16	-1.667
1	10.3333	17	-2
2	7	18	-2.333
3	4.667	19	-2.667
4	3.333	20	-3
5	2.333	21	-3.333
6	1.667	22	-3.667
7	1.333	23	-4
8	1	24	-4.667
9	0.667	25	-5.333
10	0.333	26	-6.148
11	0	27	-7.254
12	-0.333	28	-8.843
13	-0.667	29	-11.31
14	-1	30	-15.639
15	-1.333	31	-25

2

3 The cumulative number of points within each cell on the conical surface was subsequently counted,
 4 defined by a horizontal angular resolution θ , the channel number I , and the radical distance
 5 resolution r . The cells that were occupied by the background objects were deemed to have more
 6 points, which was identified using a threshold δ . These background cells were used as a mask to
 7 filter out background points in other frames. The initiation period t should be long enough to
 8 accumulate enough points in the environment, but not too long to waste computational resources.
 9 Smaller point/range resolution values led to smaller cells, which depicted the background with
 10 better resolution. However, too-small cells led to a much longer initialization time and higher
 11 computational complexity. The horizontal angular resolution of the LiDAR sensor, which is the
 12 angular difference between two adjacent points on the same channel, is also an important factor to
 13 consider when choosing θ . The Lidar used in this study has an angular resolution of 0.2° . θ should
 14 be larger than 0.2° so that at least one point falls into each cell at each frame.

15 The threshold δ should be tuned based on all other parameters. The following parameters were
 16 used in this study: $t = 60sec$, $\theta = 1^\circ$, $r = 0.1m$, $\delta = 10$. The background subtraction result of
 17 frame 126,010 on July 19, 2019, is illustrated in Figure 10, where red represents background and
 18 blue represents foreground.



19

20 Figure 10 Background Subtraction Result (Red: background; Blue: foreground)

1 To effectively identify individual vehicles, the foreground points were partitioned based on their
2 similarity. Density-based spatial clustering of applications with noise (DBSCAN) is a popular
3 clustering algorithm that generally works well in existing point cloud studies (21). The algorithm
4 groups points based on their proximity. It also marks isolated points as outliers, which makes the
5 algorithm less sensitive to background points that were not removed from the previous step. The
6 algorithm takes two parameters:

- 7 • *minPoints*: the minimum number of points to form a cluster. Its value should be close to
8 the point cloud size of a typical vehicle in the dataset.
- 9 • *eps*: the searching radius to form a cluster. If *eps* is too small, a large proportion of the
10 points will not be clustered; whereas for a large *eps*, most of the points will be allocated to
11 the same cluster.

12 Tuning DBSCAN parameters is a sophisticated process. In addition to the heuristic (21) offered
13 by the authors of the DBSCAN, researchers have proposed many different approaches (e.g.)to
14 determine the optimal parameters. To find optimal parameters and to conduct sensitivity analysis
15 for DBSCAN are beyond the scope of our research. Instead, we estimate these parameters
16 following engineering heuristics. We acknowledge that the resulted parameters are sub-optimal
17 but are adequate for the vehicle classification task.

18 For the test dataset, *eps* should be slightly smaller than jam density spacing such that two adjacent
19 stopped vehicles are not clustered as the same vehicle. However, if *eps* is too large, the truck's
20 cabin could be identified as a separate object from the trailer. We chose *eps* to be 1.5 meters. The
21 physical meaning of *eps* is the maximum distance between two points to be considered as one
22 cluster. It is chosen to be slightly smaller than the jam density such that even in completely stopped
23 traffic, two adjacent vehicles are not grouped into one cluster. For this reason, it does not need to
24 be tuned under different traffic conditions. The other parameter *minPoints* is estimated based on
25 the size of a typical truck. The value is chosen such that a typical semi-truck can be detected when
26 it is on the off-ramp (measured from the tip of the offramp divider), which is around 60 meters
27 from the sensor. Below is the process to estimate the number of points captured by LiDAR when
28 the distance between the truck's front face and LiDAR is d . The scenario is illustrated in Figure
29 11 with a side view and top-down view,

30 where

31 w - width of the truck

32 h - height of the truck

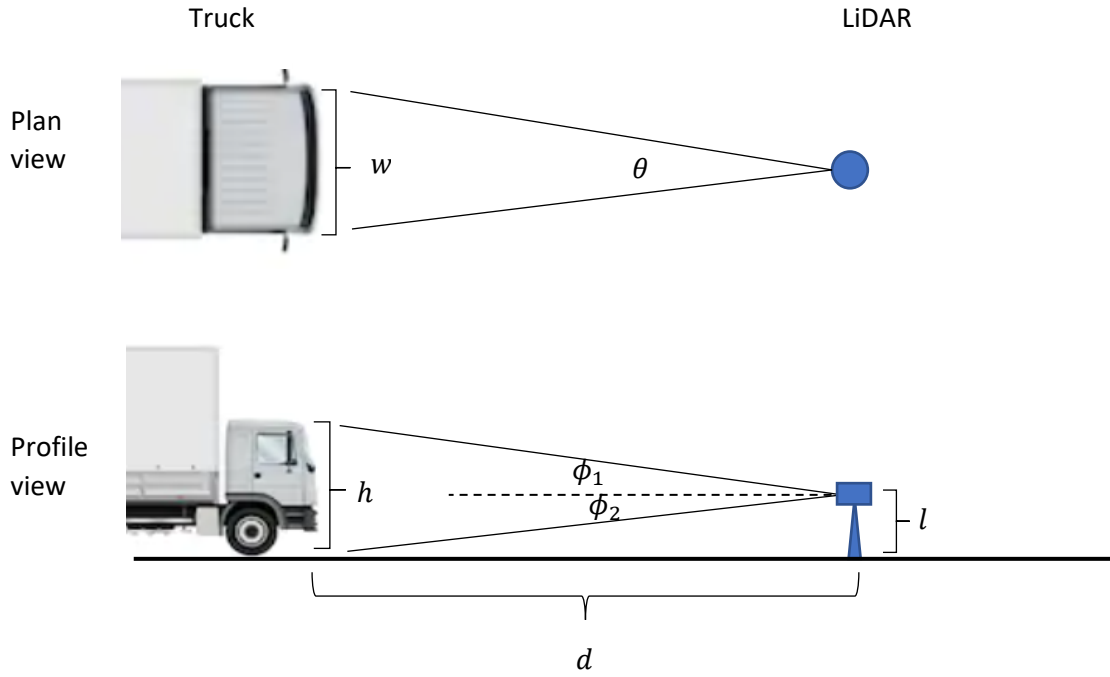
33 ϕ_1 - angle of elevation from the LiDAR to the truck

34 ϕ_2 - angle of depression from the LiDAR to the truck

35 θ - beam width of the truck viewed by the LiDAR

36 l - mounting height of the LiDAR

37



1
 2 Figure 11 Detection Illustration
 3 From the top-down view,

$$4 \quad a = \rho\theta \approx \frac{\rho w}{d} \quad (1)$$

5 Where
 6 ρ - azimuth resolution of the sensor, represented as the number of points within a unit angle.
 7 α - number of points per laser channel.

8 From the side view,

$$9 \quad \phi_1 = \arctan\left(\frac{h-l}{d}\right) \quad (2)$$

$$10 \quad \phi_2 = \arctan\left(\frac{l}{d}\right) \quad (3)$$

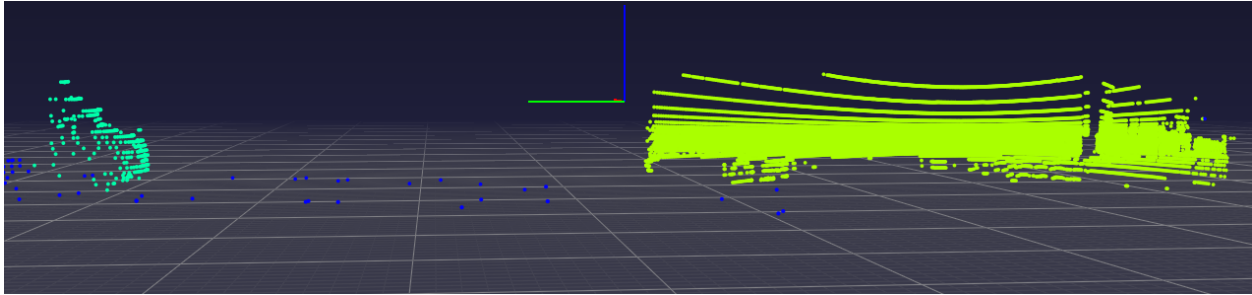
11 For the VLP-32c LiDAR with a 10Hz sampling rate, $\rho = 10$ points per degree. The mounting
 12 height of the LiDAR $l = 2.05m$. The size limit of trucks defined by the California Department of
 13 Transportation (Caltrans)¹ is used for the calculation, where $w = 2.59m$ (8.5 feet), $h =$
 14 $4.26m$ (14 feet). Then, $\phi_1 = 2.11^\circ$, $\phi_2 = 1.96^\circ$. Based on ϕ_1, ϕ_2 , and the elevation angles of
 15 the laser channels in Table 1, the number of laser channels that cover the truck, $\beta = 10$. The total
 16 number of points captured by the Lidar from the truck's front face at distance $d = 60m$ is thus,

¹ <https://dot.ca.gov/programs/traffic-operations/legal-truck-access/restrict-process>

1
$$N = \alpha \cdot \beta \approx \frac{\beta \rho w}{d} = 247 \quad (4)$$

2 To accommodate trucks that are slightly smaller than the typical size. We choose *minPoints* to
 3 be 200, which is slightly smaller than *N*.

4 The object detection result of frame 126,010 on July 19, 2019, is illustrated in Figure 10. The
 5 outliers are represented by blue points, while objects are represented in other distinct colors.



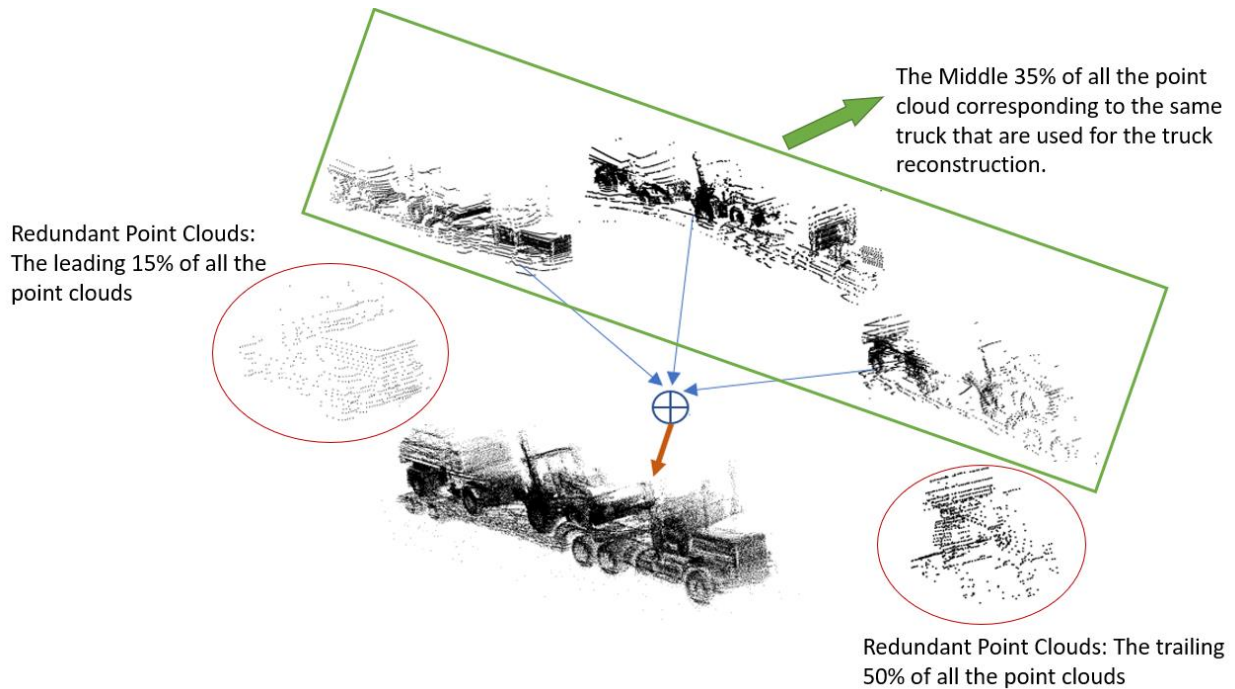
6
 7 Figure 12 Object Detection Result (Blue: outliers; Green: vehicle 1; Yellow: vehicle 2)

8

9 **3.1.2 Data Association and Truck Point Cloud Reconstruction**

10 More detailed truck point clouds for truck classification purposes were reconstructed using the
 11 framework proposed by Allu et al. (18). In this approach, the truck body point clouds were
 12 reconstructed in two steps. Each scan of the side-fire Lidar sensor captured all the truck bodies
 13 presented in the LDZ either partially or completely (side views). For example, Figure 12 presents
 14 one scan of the LiDAR sensor after object detection. This scan captured the front portion of vehicle
 15 1 and the full side view of vehicle 2. The scan obtained rich information of the front portion of the
 16 truck when it entered the LDZ, the side of the truck when it was in the midsection part of the LDZ,
 17 and the rear portion of the vehicle when it was leaving the LDZ. A more detailed truck body can
 18 be reconstructed by aggregating all the point clouds captured while traversing the LDZ. The
 19 objective of the first step was to identify and group the point clouds of the same truck from all the
 20 LiDAR scans during its traversal through the LDZ. In the proposed framework, the point clouds
 21 of the same trucks in two successive LiDAR scans were mapped using a combined strategy of
 22 Kalman Filter-based tracking and Hungarian Algorithm-based data association (23). The
 23 centroid's location of the minimum oriented bounding box of the ground projected truck point
 24 cloud data was used as the state variable for tracking purposes.

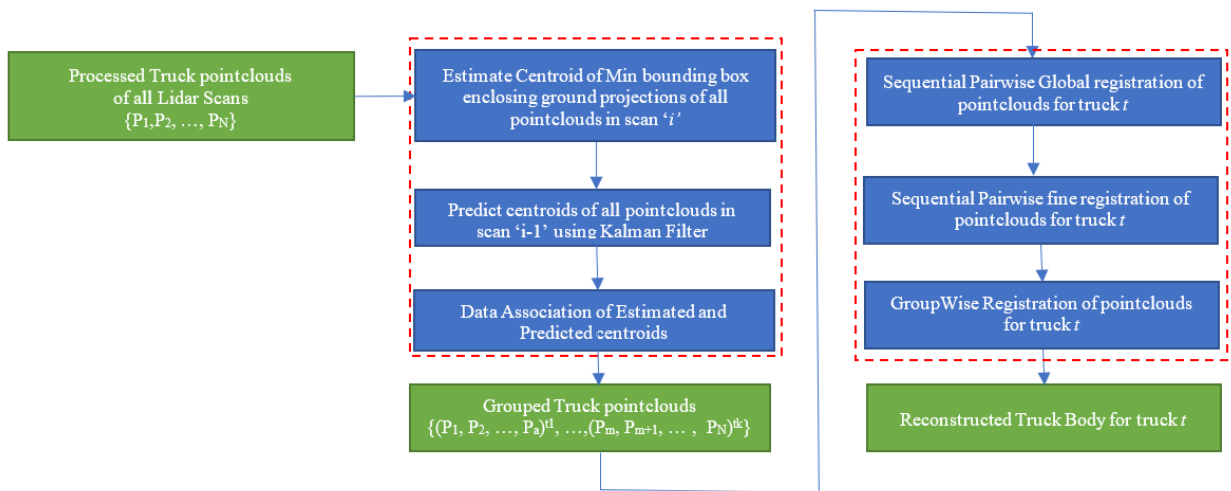
25 Once the point clouds corresponding to the same truck were grouped, the first 15 percent and the
 26 last 50 percent of all the point clouds were treated as redundant frames and excluded from the
 27 vehicle reconstruction process to improve the computation efficiency. This threshold was
 28 determined through a heuristic process with an objective function formulated as minimizing the
 29 number of statistical outlines of the reconstructed point cloud. As Figure 13 shows, the information
 30 within the vehicle point clouds represented in the green box overlaps the information captured in
 31 the leading and trailing point clouds represented in the red circles. For example, the red circle on
 32 the left-hand side presents the tractor unit of this Low Boy Platform which has been effectively
 33 captured by the point clouds within the green box. Thus, both the leading and trailing redundant
 34 point clouds were excluded without loss of reconstruction detail.



1

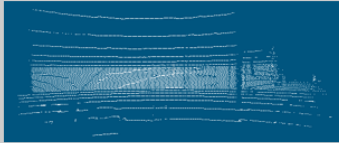
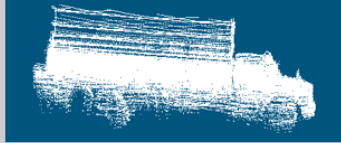
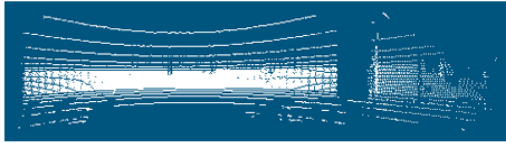
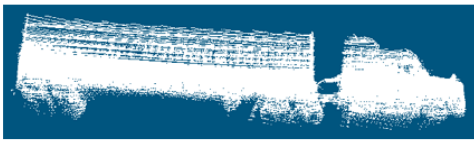

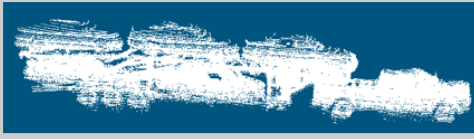


2 Figure 13 Illustration of Point Cloud Reconstruction

3 Subsequently, more detailed truck body point clouds were reconstructed in the second step by
 4 globally aligning the selected point clouds to one of the truck's midsection point clouds as
 5 reference. This global alignment was accomplished by initially computing sequential pairwise
 6 transformation matrices using a Random Sample Consensus (RANSAC)-based coarse registration
 7 strategy. These coarse transformation matrices were further fine-tuned using the Iterative Closest
 8 Point (ICP) algorithm. The fine-tuned pairwise transformation matrices along with the
 9 corresponding point clouds were globally aligned using a pose-graph-based Simultaneous
 10 Localization and Mapping (SLAM) optimization algorithm as proposed in Choi et al (24). The
 11 two-staged proposed framework is presented in Figure 14 below. The results of the object
 12 detection and vehicle reconstruction by sample truck body types are presented in Figure 15.



13

14 Figure 14 Point Cloud Reconstruction Framework (18)

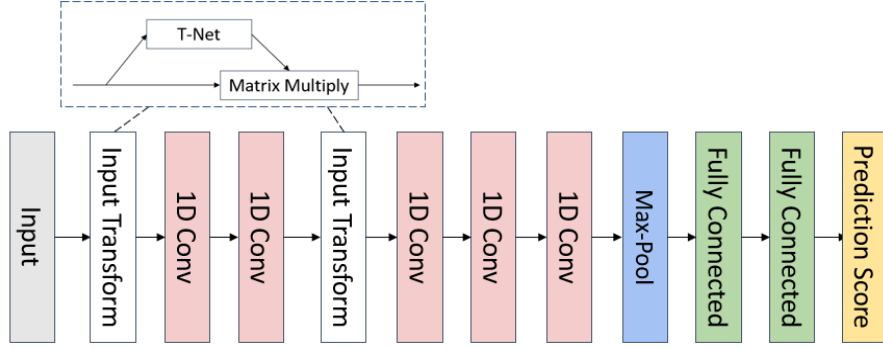
Body Type	Single Frame of Object Detection Results	Vehicle Registration Results
Conventional Single-Unit Enclosed Van		
Semi-trailer Enclosed Van		
Auto Transport with a Pickup Tractor		
Tank with a Tank Trailer		

1
2 Figure 15 Object Detection and Vehicle Reconstruction Results

3
4 3.2 PointNet-based Truck Classification Model

5 3.2.1 The Deep Representation Learning Algorithm: PointNet

6 The reconstructed 3D point cloud is an irregular type of geometric data structure, where each point
7 is represented by its cartesian coordinates (x, y, z) . A conventional convolution neural network
8 requires a regular data format such as image pixels and 3D voxels as inputs. Therefore, the point
9 cloud cannot be directly fed into a typical convolutional architecture. Point clouds are generally
10 transformed to other data types for classification purposes. In the literature, transportation
11 researchers have extracted high-level physical characteristics from either raw points (15) or
12 transformed point clouds, e.g., 2D images (9–11, 16) or 3D voxels (17) to solve the truck
13 classification problem. Finally, these high-level features are used as inputs to classic machine
14 learning algorithms. However, such data transformations and aggregations can introduce
15 quantization error and further conceal the natural invariances of the point cloud data (25), which
16 affects the accuracy and the variety of types of a truck that can be classified. In order to
17 accommodate the intrinsic characteristics of the point cloud data structure for improving truck
18 classification accuracy, a novel deep neural network architecture – PointNet (25) was adopted in
19 this research. This neural network architecture can directly incorporate point clouds as inputs and
20 detect critical features for classification from the raw inputs. However, due to the limitation of the
21 “black box” method, the specific key features were not identified in this paper. The architecture of
22 PointNet is shown in Figure 16.



1
2 Figure 16 PointNet Architecture (25)

3 PointNet benefits primarily from two components of its architecture: the shared multi-layer
4 perceptron (MLP) and the max-pooling function. The shared MLP is constructed using 1D
5 convolution with a kernel size of 1, which provides a dense connection across points with the
6 shared parameters (weight and bias terms). This means that the spatial encoding of each point can
7 be learned by the shared MLP. A max-pooling layer is applied as a symmetric function to gather
8 information from all the points, in order to resolve the invariance to permutation issue of the point
9 cloud data structure. A function $f(x_1, x_2, \dots, x_N)$ of N variables are invariant under random
10 permutation if the function value does not change over the permutation of its variables. The generic
11 representation of symmetric functions can be written as:

$$12 \quad f(x_1, x_2, \dots, x_N) = f(x_N, x_2, \dots, x_1) = f(x_2, x_1, \dots, x_N) = \dots \quad (5)$$

13 The max-pooling function extracts the global critical feature of each truck point cloud and the
14 overall model structure is able to learn the skeleton of each object. PointNet ideally fits the task of
15 truck body classification since the truck body types are generally invariant and distinct in shape.

16

17 3.2.2 PointNet for Truck Classification Model

18 In this research, the PointNet architecture was adopted to classify truck body types in detail. Prior
19 to the training process, the reconstructed point cloud needed to be regularized. First, the variable
20 number of data points in the reconstructed point clouds was uniformly downsampled to a common
21 number of points, as input into the PointNet. The downsampling process contained three steps.
22 First, a regular voxel grid with a resolution of 5 percent was generated for each reconstructed truck
23 point cloud, where those points were bucketed into voxels. Second, each occupied voxel was
24 represented by a single point, which was calculated by taking the average of all points within each
25 voxel grid. Finally, 1024 points suggested by Qi et al (25) were randomly sampled from the
26 uniformly downsampled point cloud. A truck point cloud k can be written as a 3D point set, $p^k =$
27 $\{(x_j^k, y_j^k, z_j^k) | j = 1, \dots, n\}$, where $n = 1024$ in this study. After the downsampling process, the
28 centroid of the truck point cloud k was moved to the (0,0,0) point in the coordinate and was
29 represented as $p^{kc} = \{(x_j^{kc}, y_j^{kc}, z_j^{kc}) | j = 1, \dots, n\}$. The operation along the x axis is presented in
30 Equation 6, where y and z follow the same calculation.

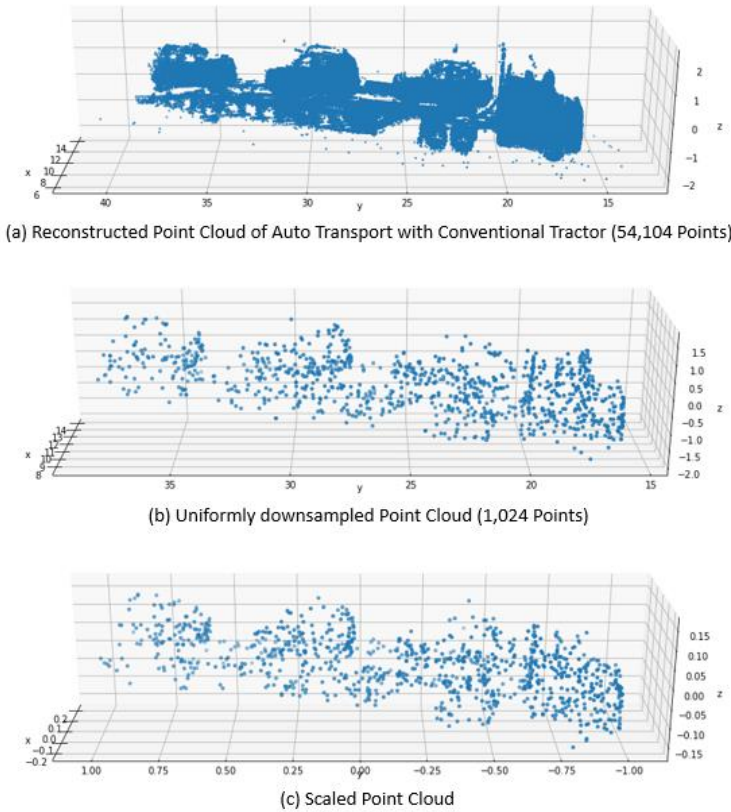
$$31 \quad x_j^{kc} = x_j^k - \frac{\max\{x_j^k\} - \min\{x_j^k\}}{2} \quad (6)$$

1 Then, the truck point cloud k was normalized to a unit sphere and denoted by $p^{kcn} =$
 2 $\{(x_j^{kcs}, y_j^{kcs}, z_j^{kcs}) | j = 1, \dots, n\}$. The operation along the x -axis is presented in Equation 7, where y
 3 and z follow the same calculation.

$$4 \quad x_j^{kcs} = \frac{x_j^{kc}}{\max\{x_j^{kc}\}} \quad (7)$$

5 The point cloud preparation step is shown in Figure 17. Here, a reconstructed point cloud of an
 6 auto transport with a conventional tractor is taken as an example.

7

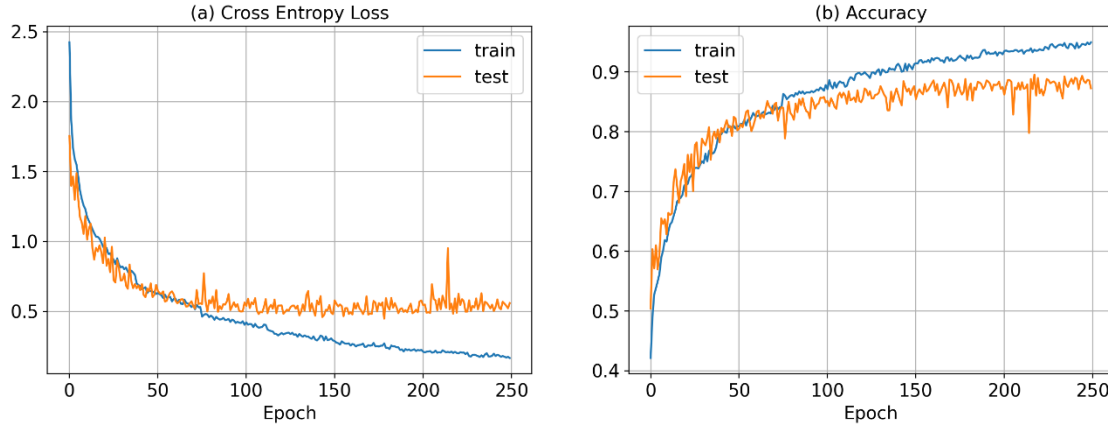


8

9 Figure 17 Point Cloud Preprocessing

10 Two data augmentation methods were applied during the model training process (25). First, each
 11 training instance was randomly rotated along the z -axis. Second, each point of the truck point cloud
 12 was jittered with a Gaussian noise which followed a $N(0, 0.02)$ distribution to increase the diversity
 13 of the training instances. The data augmentation process was used to ensure that the model learned
 14 the shape of the vehicle point cloud which was invariant to its position and orientation in the
 15 coordinate system. For instance, the point generated from the lasers may distribute differently for
 16 a vehicle traveling different lanes. The sample jittering process imitates the vehicle point clouds
 17 located across various lanes on the roadway. The rotation process simulates truck lane changing
 18 behaviors and turning movements.

1 The truck classification model was trained on 5,364 reconstructed truck point clouds with 32
 2 synthetically modified data of every training instance via the aforementioned data augmentation
 3 method. This model was trained with an RTX 2080 super GPU and took approximately 3 hours to
 4 converge. The learning process of the truck classification model is presented in the learning curve
 5 (Figure 18). The model accuracy on both training and test datasets improved in a similar trend
 6 until 100 epochs. After this point, the model performance gradually plateaued on the test dataset
 7 but continued improving on the training dataset. The model converged after 250 epochs.



8
 9 Figure 18 Learning Curves

10
 11 3.3 Model Averaging

12 A multiple layer structure with nonlinear activation functions on each layer provides deep neural
 13 networks with the theoretical ability to approximate any complex mapping function (19). However,
 14 deep neural network models generally suffer from high variance issues, where model performance
 15 varies significantly by dataset (26). Hence, model averaging strategies were explored to reduce
 16 the model variance and further enhance the model performance. The simplest way to apply model
 17 averaging on deep neural networks is to train multiple deep neural networks with different initial
 18 values and have all the models cast their votes. In this study, two model averaging methods were
 19 explored, and are explained in the next subsection.

20 3.3.1 Simple Model Averaging (SMA)

21 Let $m_a = \{m_1, m_2, \dots, m_n\}$ denote n PointNet models trained with various initial values. c_b
 22 denotes the class labels. $p(c_b|m_a)$ represents the probability that model m_a made a prediction on
 23 class c_b . The equation of SMA is shown below.

24
$$\hat{c} = \underset{c_b \in C}{argmax} \sum_{a=1}^n p(c_b|m_a) \quad (8)$$

25 SMA assumes that m_a produced an equal contribution to the final decision and gave the prediction
 26 results by averaging all the votes of the candidate models.

27

1 3.3.2 Bayesian Model Averaging (BMA)

2 Unlike simple model averaging, which treats candidate models m_a equally, Bayesian model
3 averaging assign a prior probability, presenting the subjective credibility of the model predicting
4 a certain class. The posterior probability derived from the candidate models was used as the final
5 prediction score (27, 28). In the case of a class c to be predicted based on training dataset D_{train}
6 using n PointNets with initial value drawn from a normal distribution, the Bayesian model
7 averaging provides final predictions based on the law of total probability:

$$8 \quad p(c) = \sum_{a=1}^n p(c_b|m_a)p(m_a|D_{train}) \quad (9)$$

$$9 \quad \hat{c} = \underset{c_b \in C}{argmax} \sum_{a=1}^n p(c_b|m_a)p(m_a|D_{train}) \quad (10)$$

10 As Equation 10 presents, the averaged model assigns higher weights to the candidate model which
11 performs better for the specific class (Note that Equation 8-11 were applied to each individual
12 vehicle record). The final prediction relies on the weighted average of the prediction scores.
13

14 4. Results

15 4.1 Evaluation Metrics

16 There are several evaluation metrics shown in Equations 11-14 that can be used to assess the
17 performance of classification models.

$$18 \quad Precision = \frac{TP}{TP + FP} \quad (11)$$

$$19 \quad Recall = \frac{TP}{TP + FN} \quad (12)$$

$$20 \quad Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (13)$$

$$21 \quad F1score = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (14)$$

22 where TP represents the true positive and TN represents the true negative.

23 FP and FN indicate false positive and false negative, respectively.

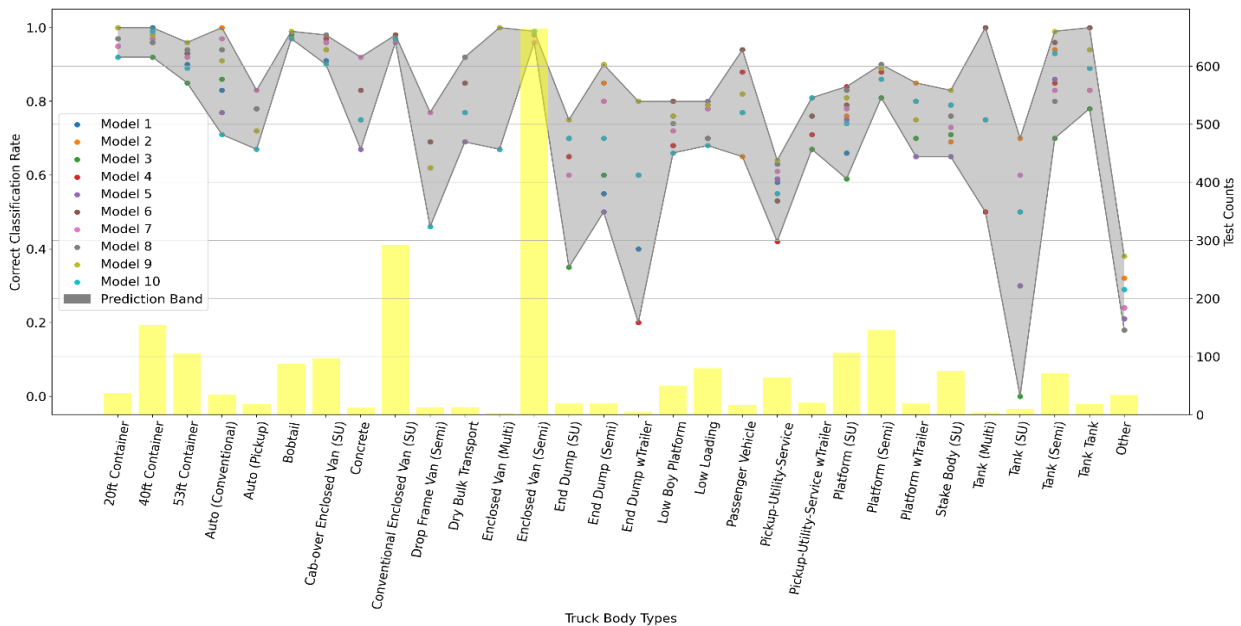
24 As shown in Equation 12, the recall value for each individual class depicts the percentage of
25 vehicles in a certain class that is correctly classified. This value is also commonly referred to as
26 the correct classification rate (CCR) in vehicle classification research (15, 17, 29). The CCR was
27 adopted to assess the model performance of each class. From an overall model perspective, the
28 accuracy measurement summarizes the total number of vehicles that are correctly classified.
29
30
31

1 However, such measurement is biased towards the major classes, since it takes both *TP* and *TN*
 2 into consideration. For example, if most of the training instances in a majority class are correctly
 3 classified and the model barely recognized any instances in a minority class, the model will still
 4 be identified as an accurate model in terms of the accuracy measurement. Therefore, this study
 5 also considered metrics such as average CCR across all classes and F1, which does not involve
 6 class distribution in its calculation, to evaluate the overall model performance.

7

8 4.2 Ensemble PointNet Results

9 Ten PointNet models were trained with different initial values and the same model structure.
 10 Figure 19 shows the CCR of each class from Ten different models on the test dataset, and the
 11 number of test samples is presented in the bar plot.



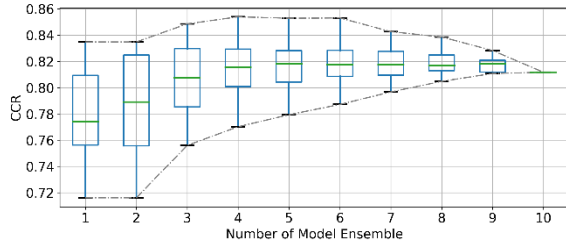
12

13 Figure 19 Model Variance on Test Dataset

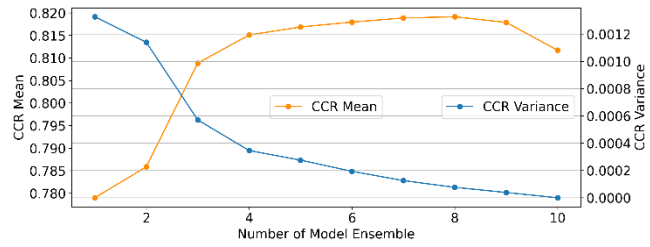
14 The same model structure yielded high variance in predicting certain classes. For instance, model
 15 5 obtained a CCR of 50 percent on predicting tractors with an end dump trailer, whereas model 9
 16 was able to achieve a CCR of 90 percent on predicting the same class. Conversely, model 1
 17 obtained a CCR of 78 percent for Auto transport with a pickup tractor. But the CCR for model 9
 18 on predicting the same class was only 72 percent. Figure 19 thus reveals the need for an ensemble
 19 model. Auto (Conventional) and Auto (pickup) distinguish the tractor units of auto transport trucks.
 20 “wTrailer” is used to designate a straight truck unit pulling a small trailer. “Other” represents all
 21 the truck types that do not fit the definition of the previous 29 classes. “Tank Tank” represents a
 22 tank tractor with a tank trailer, which specifically belongs to Class 14 in the California-modified
 23 FHWA scheme (30).

24 In order to select the optimal number of ensembles under both SMA and BMA strategies, a
 25 sensitivity analysis was conducted using various combinations of ten PointNet models. Figure 20
 26 (a) presents the CCR distribution of the different number of model ensembles. For instance, the

1 first boxplot in Figure 20 (a) presents the results of 10 single PointNet models. The second boxplot
 2 considers all possible combinations for choosing 2 models to the ensemble at a time from 10
 3 PointNet without considering the order of each individual model. Figure 20 (a) shows that the
 4 variability of prediction results reduces as the ensemble size increases. In Figure 20 (b), the mean
 5 CCR reached the highest values at the ensemble size of 7 and the CCR variance gradually
 6 converged. Therefore, the optimal number of models within the ensemble is determined as 7 for
 7 the SMA ensemble strategy.



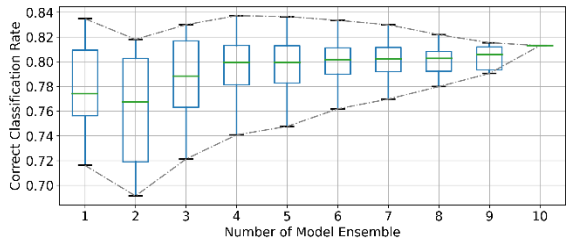
8 (a)



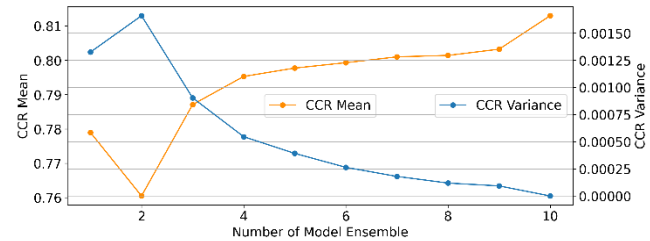
9 (b)

10 Figure 20 Simple Averaging Model Selection

11 Figure 21 presents the sensitivity analysis results on the BMA approach. According to the boxplots
 12 shown on the left-hand side, the variability of the model prediction results followed a reducing
 13 trend as the ensemble size increased. Figure 21 (b) shows that the elbow point was located at an
 14 ensemble size equal to 9 for both the CCR mean and variance values. Thus, the optimal number
 15 of ensembles for BMS is 10.



16 (a)



17 (b)

18 Figure 21 Bayesian Ensemble Model Selection

19 Results from both Figure 20 and Figure 21 indicate that both ensemble strategies are able to reduce
 20 the variability of the model results as well as improve the average CCR values over adopting only
 21 a single model.

22 The results from SMA and BMA are presented in Table 2.

23 Table 2 Model Comparison

	Ensemble Numbers	Computation Time	Average CCR	Average F1 Score	Number of Class CCRs larger than SMA/BMA
SMA	7	Around 21 hours	0.83	0.85	8
BMA	10	Around 30 hours	0.81	0.83	3

24

1 The SMA model slightly outperforms the BMA model in terms of both CCR and F1 score values.
2 In addition, the individual class-level CCR of SMA is superior to the BMA models on 8 different
3 classes. The BMA is able to achieve a similar level of CCR and F1 values with 10 model ensembles
4 which significantly compromised the computation efficiency, where the marginal costs of the
5 ensemble model training are approximately 3 hours. The Bayesian-based approach relies on a
6 reliable prior distribution to improve the model performance. With the limited data points that can
7 be used for the prior distribution, the prior information may not be sufficiently useful to improve
8 the overall model performance which could be a disadvantage of the BMS model. Therefore, with
9 the consideration of average CCR and the computation time, the SMA strategy was suggested for
10 the use of the truck classification model.

11 Moreover, average F1 scores that are larger than the average CCR score indicate that the
12 performance of the ensembled PointNet was not significantly biased towards the majority class
13 since the minority class presented low variations in their body type design. Hence, the ensemble
14 PointNet presents a promising result in solving real-world truck body type classification problems.

15 A comparison between the CCR values in Table 3 and Table 4 shows that the model was less
16 competent in predicting single-unit trucks, where the average CCR for single-unit vehicles was 77
17 percent, while performance for trucks with trailers was 85 percent. The “Other” truck type in Table
18 3 presents the other truck types with the body configuration of the truck pulling trailer(s) whereas
19 the “Other” truck type in Table 4 represents the other single-unit trucks.

20 Table 3 Confusion Matrix for trucks with trailer(s) (SMA Approach)

	20ft Container	40ft Container	53ft Container	Auto (Conv)	Auto (Pickup)	Drop Frame (Semi)	Dry Bulk Transport	Enclosed Van (Multi)	Enclosed Van (Semi)	End Dump (Semi)	End Dump wTrailer	P/U/S wTrailer	Low Boy Platform	Platform (Semi)	Platform wTrailer	Tank (Multi)	Tank (Semi)	Tank Tank	Other	Test Count*	CCR	Classic ML Approach (17)
20ft Container	36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	37	0.97	0.96
40ft Container	0	154	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	154	1.00	0.98
53ft Container	0	0	99	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	1	106	0.93	-
Auto (Conv)	0	0	0	35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35	1.00	0.91
Auto (Pickup)	0	0	0	1	14	0	0	0	0	0	0	3	0	0	0	0	0	0	0	18	0.78	-
Drop Frame (Semi)	0	0	0	0	0	8	0	0	5	0	0	0	0	0	0	0	0	0	0	13	0.62	-
Dry Bulk Transport	0	0	0	0	0	0	10	0	1	0	0	0	0	0	0	0	2	0	0	13	0.77	-
Enclosed Van (Multi)	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	3	1.00	-
Enclosed Van (Semi)	0	0	3	0	0	0	0	0	659	0	0	0	0	0	0	0	0	0	3	665	0.99	0.94
End Dump (Semi)	0	0	0	0	0	0	0	0	0	18	0	0	0	0	0	0	2	0	0	20	0.90	0.85
End Dump wTrailer	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	1	0	1	5	0.60	-
P/U/S wTrailer	0	0	0	0	0	0	0	0	1	0	0	18	0	0	1	0	0	0	1	21	0.86	-
Low Boy Platform	0	0	0	1	0	0	0	0	0	0	0	1	39	8	0	0	0	0	0	49	0.80	-
Platform (Semi)	0	0	1	0	0	0	0	0	1	1	0	1	5	135	0	0	2	0	0	146	0.92	0.94
Platform wTrailer	0	0	0	0	0	0	0	0	0	0	2	2	0	16	0	0	0	0	0	20	0.80	-
Tank (Multi)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	4	1.00	-
Tank (Semi)	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	69	0	1	71	0.97	0.97	
Tank Tank	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	16	1	1	18	0.89	-
Other	0	0	2	0	0	1	0	1	5	1	0	0	1	2	0	0	1	0	9	23	0.39	-

21 Note: Cells labeled with red colors represent CCR lower than 0.80. The yellow cells highlight the correctly classified numbers. The grey cells point to the main causes of the misclassification. “Cov” is
 22 short for “Conventional”. “P/U/S” is short for “Pickup/Utility/Service Trucks”. ML stands for “Machine Learning”. *The test set used in (17) was different from the test set used in this study.

23 Table 4 Confusion Matrix for Single-unit vehicles (SMA Approach)

	Bobtail	Cab-over Enclosed Van (SU)	Concrete	Conv Enclosed Van (SU)	End Dump (SU)	Low Loading	Passenger Vehicle	P/U/S	Platform (SU)	Stake Body (SU)	Tank (SU)	Other	Test Counts*	CCR
Bobtail	87	0	0	0	0	0	0	0	1	0	0	0	88	0.99
Cab-over Enclosed Van (SU)	0	95	0	1	0	1	0	0	0	0	0	0	97	0.98
Concrete	0	0	9	0	0	0	0	0	1	1	0	1	12	0.75
Conv Enclosed Van (SU)	0	1	0	289	0	1	0	1	0	0	0	0	292	0.99
End Dump (SU)	0	2	0	1	14	1	0	1	1	0	0	0	20	0.70
Low Loading	0	1	0	14	0	63	0	1	0	0	0	1	80	0.79
Passenger Vehicle	0	0	0	0	0	1	14	2	0	0	0	0	17	0.82
P/U/S	0	2	0	2	1	3	5	39	2	8	0	2	64	0.61
Platform (SU)	1	1	0	2	1	0	0	4	90	4	0	3	106	0.85
Stake Body (SU)	0	1	0	0	0	0	0	11	4	59	0	0	75	0.79
Tank (SU)	0	0	0	0	1	0	0	0	3	5	0	0	10	0.50
Other	0	0	0	1	0	1	0	3	3	2	1	9	20	0.45

24 Note: Cells labeled with red colors represent CCR lower than 0.80. The yellow cells highlight the correctly classified numbers. The grey cells point to the main causes of the misclassification. “Cov” is
 25 short for “Conventional”. “P/U/S” is short for “Pickup/Utility/Service Trucks”. *The test set used in (17) was different from the test set used in this study.

1 This was likely caused by the similarity across body types. For example, with different shapes of
 2 commodities or devices carried, single-unit platform trucks shared similar body configurations
 3 with pickup/utility/service trucks, single-unit stake body trucks, single-unit tank trucks, and single-
 4 unit dump trucks. In addition, the “passenger vehicle” class included 4-tire small pickups which
 5 shared a similar profile with 6-tire utility pickups that were categorized in the
 6 “pickup/utility/service” class.

7 Table 3 presents the confusion matrix of 18 truck body types, including tractors pulling semi-
 8 trailers, tractors pulling a large single trailer, and tractors pulling multiple trailers. Body type
 9 confusion occurred primarily among auto transports, low boy platforms, and semi-trailer platform
 10 trucks. Similar to the issues shown in single-unit vehicles, the loading on the trailers was likely the
 11 cause of misclassifications across these three types. Classifying trucks in more detail naturally
 12 increases the chances of misclassification among similar body types. Therefore, balancing the total
 13 number of trucks that can be classified and high CCR values across all classes is critical.

14 The latest LiDAR-based truck classification method adopted and compared several classic
 15 machine learning algorithms (17). In their approaches, the truck point clouds were first segmented
 16 into six large voxels. Then, relevant features were extracted by using the feature engineering
 17 approach. Those features were predefined before feeding them into the classifiers. Therefore, the
 18 features were yet to be optimized, which limited their models to classify trucks in detail. For
 19 example, essential features may be missed during the feature engineering process prior to the
 20 classifier training. On the contrary, the deep representation learning approach incorporates the
 21 feature selection process into the overall optimization scheme. The raw points were directly
 22 learned by the PointNet and essential features were identified during the learning process (19). The
 23 ensemble PointNet optimally selects the key features and learns the skeleton of each type of vehicle
 24 to classify various truck types accurately in high levels of detail. Furthermore, the ensemble
 25 strategies enhanced the generality of the PointNet and further improve the precision of predictions.

26 The ensemble PointNet has also been tested under different traffic speed conditions. Since trucks
 27 pulling large trailers may generally travel at a slower speed than the single-unit vehicles or single-
 28 unit vehicles pulling a small trailer, these two types of vehicles have been evaluated separately.

29 Table 5 presents the classification accuracy across three different speed ranges. A Pearson’s Chi-
 30 Square statistical test has been performed to verify the independence of classification accuracy
 31 across different speed ranges. As Table 7 presents, the p-values of 0.313 and 0.111 for the
 32 corresponding truck types showed the null hypothesis cannot be rejected at the significance level
 33 of 5 percent. Therefore, there is no significant difference in the classification results across speed
 34 values.

35 Table 5 Classification Results under Different Speed Ranges

Average Speed mph	Truck pulling Large Trailers	Single Unit Vehicle w/o a small trailer
<20	0.96 (683)	0.84 (261)
20 - 40	0.96 (637)	0.89 (632)
40>	-	0.83 (12)

36 Note: the number in the bracket presents the sample size within each corresponding speed range

37 The coefficient of variance of speeds was also calculated to describe the fluctuation of each
 38 vehicle's speed. Similarly, the classification accuracy across different speed fluctuation levels is
 39 presented in Table 6. The classification model preserves a high accuracy on predicting trucks even

1 if their speed changes significantly due to acceleration or deceleration as they traverse the LiDAR's
 2 field of view. Thus, our model is superior to the latest classification model which performs
 3 reconstruction based on the assumption of constant vehicle speeds (17). The results of Pearson's
 4 Chi-Square test shows that the model performance is independent of speed fluctuations (Table 7).

5 Table 6 Classification Results under Different Speed Variance from each Individual Vehicle

Coefficient of Variance of Speeds	Truck with Large Trailers	Single Unit Vehicle w/o a Trailer
>0.8 (High Fluctuation)	1.00 (16)	1.00 (9)
0.2-0.8 (Midian Fluctuation)	0.96 (1,095)	0.84 (122)
< 0.2 (Low Fluctuation)	0.95 (1,260)	0.88 (774)

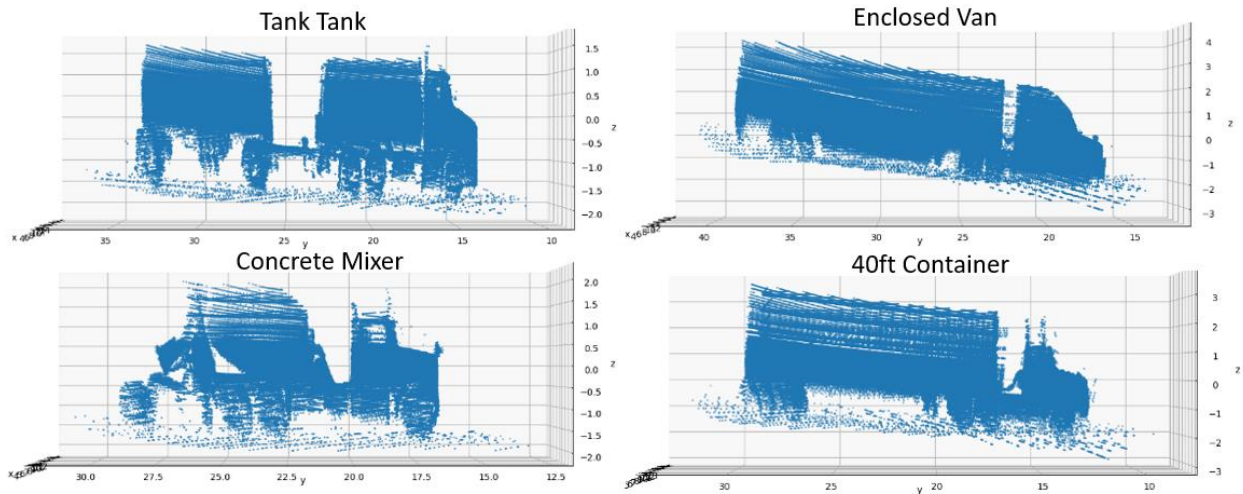
6 Note: the number in the bracket presents the sample size within each corresponding range of coefficient of variance of speed.

7 Table 7 p-values of Pearson's Chi-Square Test

	Average Speed	Speed Fluctuations
Truck pulling Large Trailers	0.313	0.355
Single Unit Vehicle w/o a small trailer	0.111	0.207

8

9 Some of the minor failure cases of background subtraction have been observed in the test dataset
 10 and presented in Figure 22. For the truck point clouds shown in Figure 22, some points on the
 11 ground plane around the truck point clouds were not successfully eliminated in the background
 12 subtraction step. However, the ensemble PointNet demonstrated sufficient robustness to the noise
 13 data and was nonetheless able to correctly classify them.



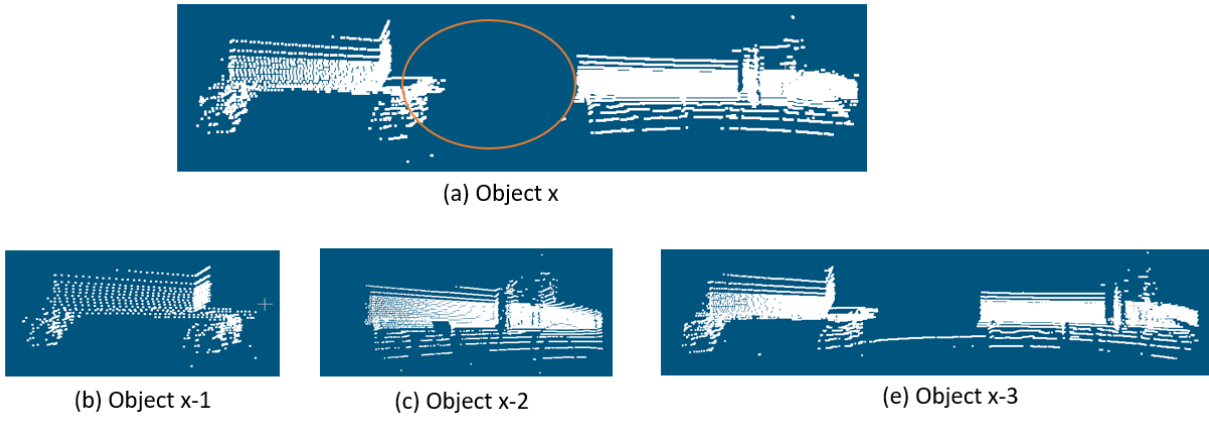
14

15 Figure 22 Minor Failure Cases of the Background Subtraction Algorithm

16 **4.3 Model Results on Temporal Independent Dataset**

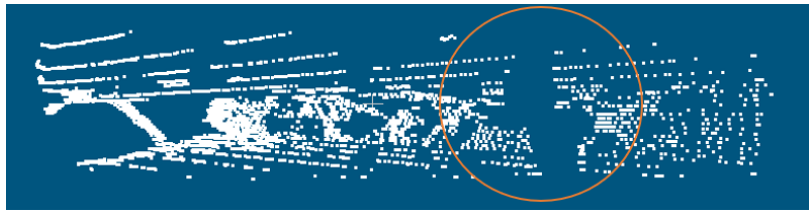
17 In order to test the model's ability on collecting truck classification counts, the developed classification
 18 model has been evaluated on a temporally independent dataset where the LiDAR, video, and loop data were
 19 collected simultaneously and continuously over approximately 12 hours. The video and loop data were used
 20 as ground truth to evaluate the performance of the LiDAR-based truck classification model. During the data
 21 collection period, there are 1,472 ground truth vehicles were recorded by inductive loops while 1,489
 22 vehicle objects were detected through the object detection and data association process. 4 (0.3 percent)
 23 vehicles were miss-detected, and 18 (1.2 percent) vehicles were over-counted by the LiDAR sensors. The

1 miss- and over-counted cases were primarily caused by the failure of the object detection algorithm due to
 2 the sparseness of the point cloud. Figure 23 presents the object detection results of a tank truck pulling a
 3 tank trailer. As Figure 23(a) shows, the points on the connector between the tractor and trailer were too
 4 sparse to be detected by the object detection algorithm in one of the point clouds associated with object x.
 5 In this case, the tractor and the trailer unit were separated into two different objects: Object x-1 and Object
 6 x-2. However, when the vehicle traveled to a certain position in the LDZ, more points located at the tractor-
 7 trailer connector can be captured by the LiDAR sensor (Figure 23(c)), where the Object x was identified as
 8 the Object x-3. Therefore, ultimately, Object x was over-counted as three different objects (Object x-1, x-
 9 2, and x-3) after the data preprocessing step. In the temporal independent test dataset, only 1 out of 8 dump
 10 trucks pulling a trailer presented the issue as such.



11
 12 Figure 23 Object Detection Error Case 1 (Overcounting)

13 Figure 24 presents the object detection result of an auto transport truck. The gap between the trailer body
 14 caused the error of the object detection algorithm. The algorithm detected two objects in this frame and
 15 grouped them separately. However, each identified object that contained less than 15 frames was treated as
 16 noises in the preprocessing steps. The frame numbers of the noise objects were determined based on the
 17 vehicle speed within the LDZ. In the study, the diameter of the LDZ is around 50 meters. The LiDAR
 18 sensor rotated with a speed of 10 frames per second. Assuming the vehicle travels at a speed of 80 mph, at
 19 least 15 frames will be captured by the LiDAR sensor. Hence, the vehicle objects with less than 15 frames
 20 were discarded. This vehicle record was missed by the LiDAR detection algorithm due to the sparseness of
 21 its single frame. Such cases only account for 0.3 percent of the independent test dataset, and 3 percent
 22 within the auto transport trucks.



23
 24 Figure 24 Object Detection Error Case 2 (Miscounting)

25 The average CCR and F1 scores for the temporal independent site were 0.85 and 0.83 respectively.

26

5. Conclusion, Discussion, and Future Works

This research presented a novel LiDAR-based truck classification method through the implementation of a truck point cloud reconstruction framework that was able to retain a wide LDZ without compromising classification accuracy. The data used in the development of the model was collected from a horizontally-oriented multi-array 3D LiDAR sensor, which had the ability to capture a wide field of view of the roadway. In this case, even though vehicles traveling in the outermost lanes presented in front of the LiDAR sensor for a short period of time and occluded vehicles traveling in the corresponding inner lanes, the point cloud originating from those fully occluded vehicles could be retrieved from consecutive frames. The sparse point clouds from individual frames resulting from a low vertical resolution were enriched by aggregating multiple frames associated with the same truck. Subsequently, PointNet – a novel deep representation learning algorithm – was adopted to classify vehicles based on their detailed body types. PointNet successfully learned the basic characteristics of each truck class by selecting the critical features from each preprocessed point cloud. Finally, two model ensemble strategies, SMA and BMA, were explored to improve the generality of the model and to further enhance the model performance. A detailed sensitivity analysis was conducted to select the optimal number of model ensembles with the consideration of average CCR, prediction variability, and computation efficiency of the ensemble model. Our LiDAR-based truck classification model was tested on vehicles traveling in a variety of speed ranges including conditions with high-speed fluctuations. The performance of our model was not significantly affected by traffic conditions according to the statistical test results. A closer assessment of the reconstructed point cloud found that some points on the ground plane were not eliminated in the background subtraction step. However, the ensemble PointNet demonstrated sufficient resilience to the noise data and was still able to classify the affected vehicles correctly. Further, the model has also been evaluated on a temporally independent dataset. The vehicle counts obtained by the LiDAR sensor were compared to the vehicle counts obtained from the loop detector. Overall, 0.3 percent of vehicles were miss-detected, and the LiDAR sensor overcounted vehicles by 1.2 percent. These issues are primarily caused by the occasional occurrence of sparseness within an individual frame of the vehicle which results in the error of the object detection algorithm and subsequently leads to misassignment in the data association step. However, such an issue only accounted for less than 2 percent of the overall dataset and will be addressed by further improving the object detection algorithm in the future.

The novel framework developed was able to classify heavy-duty trucks in much more detail, with a close relationship to their industry affiliations. For instance, the new model could accurately distinguish low boy platforms from general flatbed trucks, where these two types of platform trucks are designed to carry different types of payloads. This model was able to classify 30 different vehicle types (advantageously mainly trucks) and achieve an average class CCR of 90 percent for a truck with large trailer (s) and 83 percent for single-unit vehicles with or without a small trailer. Remarkably, the proposed method was able to distinguish 53ft containers and semi-trailer enclosed vans with over 95 percent CCR even though they share very similar physical characteristics.

The multi-array rotating LiDAR sensor is one of the most critical components for autonomous vehicles. The cost of LiDAR sensors is likely to reduce significantly with the maturity of the technology following the mass production of the autonomous vehicle in the future. This expected trend will increase the potential of using LiDAR sensors as a cost-efficient alternative for roadside truck data collection.

1 In the future, object detection will be further enhanced to better accommodate the occasional
2 occurrence of sparseness point cloud issues. Furthermore, multi-lane truck classification
3 applications can be explored as the horizontal orientation of the LiDAR permits capturing a full
4 360-degree field of view.

5

6

7 **Acknowledgment**

8 This study was made possible through funding received by the California Department of
9 Transportation (Caltrans) and the Pacific Southwest Region University Transportation Center
10 (PSR-UTC). The authors would like to thank Caltrans and PSR-UTC for their support of
11 university-based research, and especially for the funding received for this project. The authors also
12 gratefully appreciate the support provided by Tom Shepard of Caltrans District 11 for his effort in
13 providing access to the study site and setting up field sensors. The contents of this paper reflect
14 the views of the authors who are responsible for the facts and the accuracy of the data presented
15 herein. The contents do not necessarily reflect the official views or policies of the State of
16 California. This paper does not constitute a standard, specification, or regulation.

17

18

1 Reference:

- 2 1. Bureau of Transportation Statistics. U.S. Gross Domestic Product (GDP) Attributed to
3 Transportation Functions. [https://www.bts.gov/content/us-gross-domestic-product-gdp-](https://www.bts.gov/content/us-gross-domestic-product-gdp-attributed-transportation-functions-billions-current-dollars)
4 [attributed-transportation-functions-billions-current-dollars](https://www.bts.gov/content/us-gross-domestic-product-gdp-attributed-transportation-functions-billions-current-dollars). Accessed Jul. 18, 2020.
- 5 2. Statistics Bureau of Transportation. Transportation's Contribution To The Economy.
6 [https://www.bts.gov/browse-statistical-products-and-data/transportation-economic-](https://www.bts.gov/browse-statistical-products-and-data/transportation-economic-trends/tet-2017-chapter-2)
7 [trends/tet-2017-chapter-2](https://www.bts.gov/browse-statistical-products-and-data/transportation-economic-trends/tet-2017-chapter-2). Accessed Jul. 18, 2020.
- 8 3. Beagan, D., D. Tempesta, and K. Proussaloglou. *Quick Response Freight Methods*
9 *(QRFM)*. 2019.
- 10 4. AASHTO. *Freight Data Guide for Improved Transportation Planning*. 2018.
- 11 5. Schaefer, Ron; Worth, Monica; Heilman, Jonathan; Kehoe, N. *Freight Demand Modeling*
12 *and Data Improvement*. 2017.
- 13 6. American Planning Association. *Surface Transportation Policy Guide*. 2019.
- 14 7. Federal Highway Administration. *Traffic Monitoring Guide FHWA*. 2013.
- 15 8. Aschauer, D. A. Highway Capacity and Economic Growth. *Economic Perspectives*, 1990,
16 pp. 14–24.
- 17 9. Abdelbaki, H. M., K. Hussain, and E. Gelenbe. A Laser Intensity Image Based Automatic
18 Vehicle Classification System. *IEEE Conference on Intelligent Transportation Systems,*
19 *Proceedings, ITSC*, 2001, pp. 460–465. <https://doi.org/10.1109/itsc.2001.948701>.
- 20 10. Hussain, K. F., and G. S. Moussa. Laser Intensity Vehicle Classification System Based on
21 Random Neural Network. *Proceedings of the Annual Southeast Conference*, Vol. 1, 2005,
22 pp. 131–135. <https://doi.org/10.1145/1167350.1167372>.
- 23 11. Sandhawalia, H., J. A. Rodriguez-Serrano, H. Poirier, and G. Csurka. Vehicle Type
24 Classification from Laser Scanner Profiles: A Benchmark of Feature Descriptors. *IEEE*
25 *Conference on Intelligent Transportation Systems, Proceedings, ITSC*, No. Itsc, 2013, pp.
26 517–522. <https://doi.org/10.1109/ITSC.2013.6728283>.
- 27 12. Asborn, M. I., C. G. Burris, and S. Hernandez. Truck Body-Type Classification Using
28 Single-Beam Lidar Sensors. *Transportation Research Record*, Vol. 2673, No. 1, 2019, pp.
29 26–40. <https://doi.org/10.1177/0361198118821847>.
- 30 13. Velodyne Acoustics Inc. *VLP-32C User Manual*. 2018.
- 31 14. Lee, H., and B. Coifman. Side-Fire Lidar-Based Vehicle Classification. *Transportation*
32 *Research Record*, No. 2308, 2012, pp. 173–183. <https://doi.org/10.3141/2308-19>.
- 33 15. Wu, J., H. Xu, Y. Zheng, Y. Zhang, B. Lv, and Z. Tian. Automatic Vehicle Classification
34 Using Roadside LiDAR Data. *Transportation Research Record*, Vol. 2673, No. 6, 2019,
35 pp. 153–164. <https://doi.org/10.1177/0361198119843857>.
- 36 16. Vatani Nezafat, R., O. Sahin, and M. Cetin. Transfer Learning Using Deep Neural
37 Networks for Classification of Truck Body Types Based on Side-Fire Lidar Data. *Journal*
38 *of Big Data Analytics in Transportation*, Vol. 1, No. 1, 2019, pp. 71–82.

- 1 <https://doi.org/10.1007/s42421-019-00005-9>.
- 2 17. Sahin, O., R. V. Nezafat, and M. Cetin. Methods for Classification of Truck Trailers
3 Using Side-Fire Light Detection and Ranging (LiDAR) Data. *Journal of Intelligent*
4 *Transportation Systems: Technology, Planning, and Operations*, Vol. 0, No. 0, 2020, pp.
5 1–13. <https://doi.org/10.1080/15472450.2020.1733999>.
- 6 18. Allu, K., Z. Sun, and A. Tok. LiDAR-Based Reconstruction for Truck Surveillance. *UC*
7 *Irvine: Institute of Transportation Studies*, 2020.
- 8 19. Goodfellow, I., Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.
- 9 20. Hernandez, S. V. *Integration of Weigh-In-Motion and Inductive Signature Data for Truck*
10 *Body Classification*. 2014.
- 11 21. Ester, M., H.-P. Kriegel, J. Sander, and X. Xu. A Density-Based Algorithm for
12 Discovering Clusters in Large Spatial Databases with Noise. *Kdd*, Vol. 96, 1996, pp. 226–
13 231. <https://doi.org/10.1016/B978-044452701-1.00067-3>.
- 14 22. Kriegel, H.-P., P. Kröger, K. Kröger, J. Sander, and A. Zimek. Density-Based
15 Clustering. <https://doi.org/10.1002/widm.30>.
- 16 23. Bewley, A., Z. Ge, L. Ott, F. Ramos, and B. Upcroft. Simple Online and Realtime
17 Tracking. *Proceedings - International Conference on Image Processing, ICIP*, Vol. 2016–
18 August, 2016, pp. 3464–3468. <https://doi.org/10.1109/ICIP.2016.7533003>.
- 19 24. Choi, S., Q.-Y. Zhou, and V. Koltun. Robust Reconstruction of Indoor Scenes. *IEEE*
20 *Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5556–5565.
- 21 25. Qi, C. R., H. Su, K. Mo, and L. J. Guibas. PointNet: Deep Learning on Point Sets for 3D
22 Classification and Segmentation. *Proceedings - 30th IEEE Conference on Computer*
23 *Vision and Pattern Recognition, CVPR 2017*, Vol. 2017-Janua, 2017, pp. 77–85.
24 <https://doi.org/10.1109/CVPR.2017.16>.
- 25 26. Zhou, Z. H., J. Wu, and W. Tang. Ensembling Neural Networks: Many Could Be Better
26 than All. *Artificial Intelligence*, Vol. 137, No. 1–2, 2002, pp. 239–263.
27 [https://doi.org/10.1016/S0004-3702\(02\)00190-X](https://doi.org/10.1016/S0004-3702(02)00190-X).
- 28 27. Hoeting, J. A., D. Madigan, A. E. Raftery, C. T. Volinsky, A. E. Raftery, and C. T.
29 Volinsky. Bayesian Model Averaging: A Tutorial (with Discussion). *Statistical Science*,
30 Vol. 14, No. 4, 1999, pp. 382–417.
- 31 28. Raftery, A. E., T. Gneiting, F. Balabdaoui, and M. Polakowski. Using Bayesian Model
32 Averaging to Calibrate Forecast Ensembles. *Monthly Weather Review*, Vol. 133, No. 5,
33 2005, pp. 1155–1174. <https://doi.org/10.1175/MWR2906.1>.
- 34 29. Hernandez, S. V., A. Tok, and S. G. Ritchie. Integration of Weigh-in-Motion (WIM) and
35 Inductive Signature Data for Truck Body Classification. *Transportation Research Part C:*
36 *Emerging Technologies*, Vol. 68, 2016, pp. 1–21.
37 <https://doi.org/10.1016/j.trc.2016.03.003>.
- 38 30. Quinley, R. WIM Data Analyst’s Manual: FHWA Report IF-10-018. 2010, p. 183.

