IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 25, 2019, accepted July 6, 2019, date of publication July 10, 2019, date of current version July 30, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2927889

Probabilistic Prediction of Pedestrian Crossing Intention Using Roadside LiDAR Data

JUNXUAN ZHAO[1], YINFENG LI[2], HAO XU[3], AND HONGCHAO LIU[1]

[1]Department of Civil, Environmental, and Construction Engineering, Texas Tech University, Lubbock, TX 79409, USA
[2]College of Civil and Architectural Engineering, North China University of Science and Technology, Tangshan 063210, China
[3]Department of Civil and Environmental Engineering, University of Nevada, Reno, Reno, NV 89557, USA

Corresponding author: Hongchao Liu (hongchao.liu@ttu.edu)

This work was supported by the Nevada Department of Transportation (NDOT) under Grant P224-14-803/TO #15.

ABSTRACT Pedestrians are vulnerable road users that need proactive protection. While both autonomous and connected vehicle technologies aim to deliver greater safety benefits, current designs heavily rely on vehicle-based or on-board sensors and lack strategic real-time interactions with pedestrians who do not have any communication means. As pedestrians are passively protected by the system, they might be put into hazardous situations when vehicle-mounted sensors fail to detect their presence. This paper is part of ongoing research that uses roadside light detection and ranging (LiDAR) sensors to develop a human-in-the-loop system that brings pedestrians into the connected environment. To proactively protect pedestrians, accurate prediction of their intention for crossings at locations, such as unsignalized intersections and street mid-blocks is critical, and this paper presents a modified Naïve Bayes approach for this purpose. It features a probabilistic approach to overcoming the common deficiencies in deterministic methods and provides valuable comparisons between feature-based data processing methods, such as artificial neural network (ANN) and model-based Naïve Bayes approach. A case study was conducted by using a low-cost 16-line LiDAR sensor installed at the roadside. Pedestrians' crossing intention was predicted at a range of 0.5–3 s before actual crossings. The results satisfactorily demonstrated the properties of the modified Naïve Bayes model, as well as its higher flexibility, compared with the ANN approaches in practice.

INDEX TERMS Confidence level, Naïve Bayes, pedestrian crossing intention, roadside LiDAR.

I. INTRODUCTION

Connected-vehicle technology will enable pedestrians, vehicles, roads, and infrastructures to communicate with each other and share vital traffic information through network technologies [1]. It provides a greater range than on-board vehicle equipment, thus allows drivers to receive alert/warning messages much earlier. A critical input to connected-vehicle technology is high-resolution trajectory information of all road users, mainly the real-time presence, location, velocity, and direction data. Despite the recent advancements in vision-based trajectory extraction from image data, video-based sensors still have not overcome their inherent problems. First, illumination still presents significant impacts on video quality and images recorded at night are hard to process; secondly, although the video is full of color information, counting accuracy using image data highly depends on the resolution of cameras and is restricted by the distance; and thirdly, privacy issue remains a big concern. In addition, regardless of the sensing technologies to be used, there is an urgent need to include unconnected vehicles and road users such as pedestrians and cyclists into the loop, considering connected and unconnected vehicles will co-exist for the next couple of decades or even longer [2].

Three-dimensional (3D) Light Detection and Ranging (LiDAR) sensors provide an innovative way to collect trajectory-level data under mixed traffic conditions. These sensors can scan 360° surrounding objects and report accurate location of the objects in 3D point clouds without the influence of illumination conditions. At this moment, LiDAR sensors are primarily used in autonomous vehicles for detecting road users, lane markers, and obstacles [3]–[6]. In this research, it was installed at the roadside to detect vehicles and pedestrians and obtain their real-time trajectories. In general, autonomous vehicles detect objects within a small range around the vehicles using high-cost LiDAR sensors, while cost-efficient roadside LiDAR sensors can detect objects in an extended range. On-board LiDAR sensors can provide a

The associate editor coordinating the review of this manuscript and approving it for publication was Shaohua Wan.

 This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

detailed description of objects by intense point clouds, while data points collected by roadside LiDAR sensors are sparse. On-board LiDAR sensors must work with other sensors such as cameras and radars to support the needs for autonomous driving, but roadside LiDAR sensors can work independently. Despite its many advantages for real-time and trajectory-level data collection, deployment of roadside LiDAR sensors has been fairly limited, due in part to its high cost and in part to its limited applications. However, recent advancement in LiDAR technology, intensive competition, and increasingly enlarged application market will soon make wide deployment possible. Since the methods for on-board LiDAR data processing cannot be directly applied to roadside LiDAR, it is imperative to investigate the essentials of roadside LiDAR ranging from installation strategies to efficient and effective methods for both offline and online data processing.

This study is focused on using roadside LiDAR to capture pedestrians' real-time trajectories and estimating their crossing intention at unsignalized intersections. In previous research, the authors have developed a systematic approach for automatic background filtering, lane identification, pedestrian and vehicle detection and tracking, and integration of multiple roadside LiDAR sensors [7]–[9], which laid a solid foundation for this study. Majority of the studies on estimation of pedestrian behavior are based on on-board video sensors. Hashimoto *et al.* [10] proposed a probabilistic model based on Dynamic Bayesian Network (DBN) for predicting pedestrian behaviors at signalized crosswalks. This model integrated the information of intersection context and pedestrian behaviors. A particle filter was used to estimate pedestrian states, including position, crossing decision, and motion type. The evaluation showed that the proposed model was able to recognize the pedestrian crossing decision in a few seconds from the traffic signal and pedestrian position information, which was assumed to have been obtained from the connected vehicles. However, this assumption is difficult to satisfy since the number of connected vehicles on the roads is limited at present, and thus difficult to provide enough data to test the model. Schneemann and Heinemann [11] proposed a context-based feature descriptor in combination with a support vector machine (SVM) classifier for detecting pedestrian crossing intention in urban environments. The descriptor captured the movement of pedestrians relative to the road and the spatial layout of other scene elements in a generic manner. It showed that context-based data are good indicators for crossing prediction, but the prediction may be delayed due to lack of information about pedestrians' posture and body movement. Contextual information is not easy for a standalone on-board system to obtain, but it is critical for advanced vehicle safety systems. Kwak *et al.* [12] proposed an algorithm to predict a pedestrian's intention using images captured by a far-infrared thermal camera mounted on a moving car at night. Using the dynamic fuzzy automata (DFA) method based on spatial-temporal features (e.g., the distance between the curbs and the pedestrians, pedestrians' speed, and head orientation), the performance of the proposed model was better than the models based on Markovian analysis.

All of the aforementioned methods were developed to serve autonomous vehicles via image processing. As LiDAR data are point clouds that do not have pixel information, the algorithms used in image processing cannot be applied to LiDAR data processing directly. Völz *et al.* [13] introduced a Quantile Regression method to predict pedestrians' time-to-cross when approaching a crosswalk using on-board LiDAR data. The quantile information depicted the complexity and variability of typical pedestrian behaviors and was used to provide a time interval for the possible crossings and an estimation for the associated uncertainty through the size of the time interval. The results showed that Quantile Regression Forest (QRF) produced better results than Linear Quantile Regression (LQR) when the time-to-cross was less than 3 seconds. The largest challenge for this method is how to quickly detect pedestrians' motion change. If pedestrians are blocked by vehicles or roadside objects, it is almost impossible for on-board sensors to detect them. In the authors' previous work [14], we trained a deep autoencoder-artificial neural network (DA-ANN) model using pedestrian trajectories extracted from roadside LiDAR data to predict whether or not a pedestrian walking along the sidewalk will cross the road. However, the model was deterministic in nature thus could only give Yes or No answer (i.e., crossing or non-crossing) without any confidence level information, which limits its application in practice.

In this paper, the authors proposed a probabilistic model based on modified Naïve Bayes method and pedestrian trajectories extracted from roadside LiDAR sensors to predict pedestrian crossing intention before actual arrival at crossing facilities with real-time and quantitative confidence level information. The procedure of prediction includes pedestrian trajectory acquisition, pedestrian feature extraction, prediction model training and evaluation. The results of the case study demonstrated the effectiveness of the proposed model for predicting pedestrian crossing intention at 0.5 to 3 seconds ahead of actual crossings. A comparison analysis was also provided in the case study, which compared the Naïve Bayes results with the results from a deterministic artificial neural network (ANN) approach and verified the proposed probabilistic approach outperforms the deterministic model in real-world situations with adjustable key parameters. For example, the warning threshold of pedestrian crossing signals can be set according to actual requirements.

This paper is structured as follows: Section II introduces the LiDAR sensors and data. Section III presents the methodology for pedestrian crossing intention prediction with roadside LiDAR sensors. Section IV introduces a comprehensive case study to show the model training process and performance evaluation. Section V concludes the findings and discusses future work.

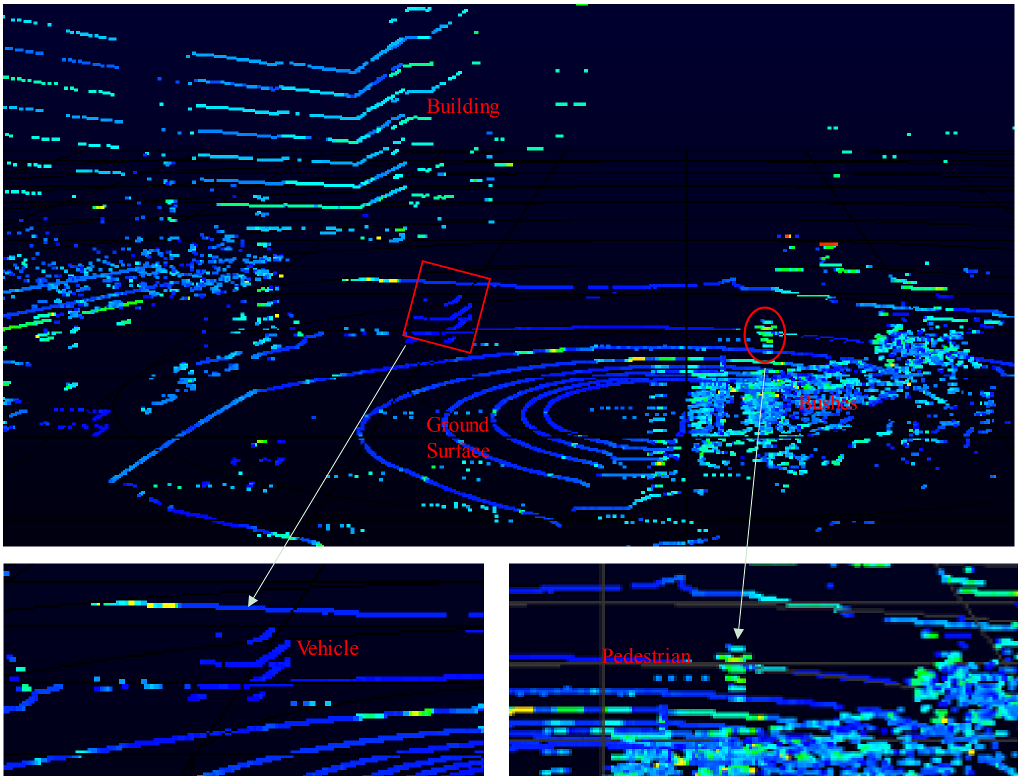


FIGURE 1. Sample LiDAR data from Veloview.

II. LiDAR SENSOR AND DATA

In this research, a 360° 16-line VLP-16 LiDAR sensor manufactured by Velodyne was used. 16 lasers rotate horizontally by an internal motor with a speed of 5 to 20 rotations per second, which can generate about 300,000 3D points per second. It can scan over a range of 2 meters (6.5 feet) to 100 meters (328 feet), with a 30° vertical field of view (+/−15° with 2° interval). The VLP-16 LiDAR sensor reports coordinates in spherical coordinates (r, ω, α), which can be converted to Cartesian coordinates (x, y, z). The output LiDAR data include the location information of measured points relative to the LiDAR sensor in XYZ coordinates, intensity, laser ID, azimuth, the distance between a data point and the sensor, and timestamp [15]. Fig. 1 shows a sample LiDAR data frame visualized by the Veloview software that comes with the LiDAR product. There were pedestrians, vehicles, bushes, buildings and ground surfaces in the raw LiDAR data, but only pedestrians and vehicles were the objects of interest.

III. PROPOSED PROBABILISTIC PREDICTION MODEL

The proposed prediction model predicts crossing probabilities of pedestrians walking on a sidewalk or at an intersection corner in a time interval. First, it was trained offline using pedestrians' historical trajectory-level movement features, which were extracted from roadside LiDAR data. Then the real-time probabilistic crossing intention prediction was achieved by applying a pedestrian' current movement features to the trained model and obtaining a predicted result. A final prediction of this pedestrian at the current time was made based on the predicted results for the current and two previous time intervals. At last, a deterministic ANN model was also tested and evaluated with the same datasets, thus allowing a fair comparison of the two models.

A. PEDESTRIAN TRAJECTORY ACQUISITION

From raw roadside LiDAR data to road users' real-time trajectories, the authors have developed a complete roadside LiDAR data processing procedure in the previous works [9]. The main steps were in the order of background filtering, lane identification, object clustering, pedestrian/vehicle classification, and tracking. The extracted trajectories include XYZ position, the total number of data points, distance to LiDAR, tracking ID, frame number, velocity, direction, timestamp, and pedestrian/vehicle label of each road user. Pedestrian trajectories were the initial data for pedestrian crossing intention prediction. While going through the details of these methods is unnecessary and beyond the objective of this paper, for illustrative purpose, Fig. 2(a-f) show the results after each step of data processing. Fig. 2(a) presents the raw LiDAR data in a 3D Cartesian coordinate system; Fig. 2(b) shows LiDAR point clouds after background filtering. The remaining noise points will be excluded in the following clustering step; Fig. 2(c) demonstrates the identified road boundaries; Fig. 2(d) shows the clustering and classification

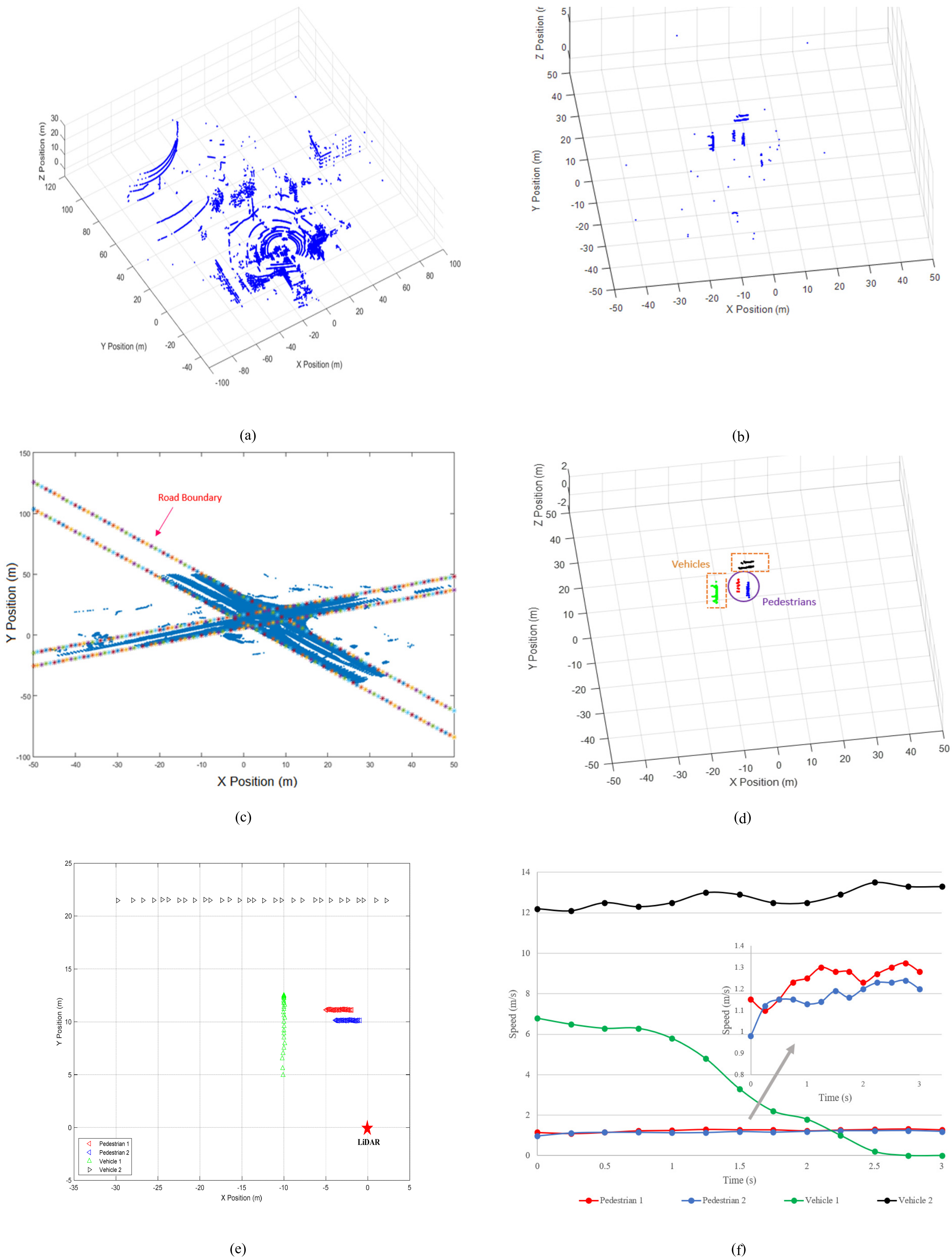


FIGURE 2. **(a) Raw LiDAR data. (b) LiDAR data after background filtering. (c) Road lane identification. (d) Clustering and classification. (e) Extracted trajectories of pedestrians and vehicles. (f) Speed information of pedestrians and vehicles.**

results of two pedestrians and two vehicles in one sample data frame; Fig. 2(e) and Fig. 2(f) show the trajectories and speed information of the same pedestrians and vehicles during a three-second interval as an example.

B. MODIFIED NAÏVE BAYES PREDICTION MODEL

For classifying sequence data such as trajectories, feature-based classifications (e.g., SVM and ANN) and distance-based classifications (e.g., K nearest neighbor) can only provide classification labels for the sequence without the information of quantitative confidence levels, while model-based classifications (e.g., Naïve Bayes, Markov Model, and Hidden Markov Model) assume that sequences in a class are generated by an underlying model and described by probability distributions. The Naïve Bayes sequence classifier is a basic generative model [16] and can be trained efficiently in a supervised learning setting by learning conditional probabilities of features in different classes, even with a small dataset [17], this makes it suitable for predicting pedestrian crossing intention in real time. It requires a strong (naïve) assumption that all features in the sequences must be independent of each other. The algorithm of Naïve Bayes classification is summarized as follows:

Given:

A sequence $X = (x_1, x_2, \ldots, x_n)$ has n independent features. x_i $(i = 1, 2, \ldots, n)$ represents each of the n features.

Class: $Y = (y_1, y_2, \ldots, y_j)$ represents j classes. y_m $(m = 1, 2, \ldots, j)$ represents each of the j classes.

Classify the sequence X into one of the above classes.

According to the Bayes' theorem:

$$P\left(y_j|x_1, x_2, \ldots, x_n\right) = \frac{P\left(y_j\right) \mathrm{P}\left(x_1, x_2, \ldots, x_n|y_j\right)}{\mathrm{P}\left(x_1, x_2, \ldots, x_n\right)} = \frac{P\left(y_j\right) \prod_{i=1}^{n} \mathrm{P}\left(x_i|y_j\right)}{\mathrm{P}\left(x_1, x_2, \ldots, x_n\right)}$$

$$P\left(y_j|x_1, x_2, \ldots, x_n\right) \propto P\left(y_j\right) \prod_{i=1}^{n} \mathrm{P}\left(x_i|y_j\right)$$

$$\widehat{y_j} = \arg\max_{y} P\left(y_j\right) \prod_{i=1}^{n} \mathrm{P}\left(x_i|y_j\right) \tag{1}$$

where $P(y_j|x_1,x_2,\ldots,x_n)$ is the conditional probability of assigning a sequence $(x_1,x_2,\ldots,x_n)$ to each of j possible classes y_j. $\prod_{i=1}^{n} \mathrm{P}\left(x_i|y_j\right)$ represents the multiplication of each conditional probability $P(x_i|y_j)(i = 1, 2, \ldots, n)$. $\arg\max_{y} p(y_j) \prod_{i=1}^{n} \mathrm{P}(x_i|y_j)$ gives a value y at which $p(y_j) \prod_{i=1}^{n} \mathrm{P}(x_i|y_j)$ is maximized. $\propto$ means proportional to and $\hat{y}_j$ means an estimation of y_j value.

According to (1), the sequence X will be classified into the class that has the highest conditional probability value $\hat{y}_j$.

Based on the obtained trajectory information, the values of each feature are discrete. For example, Fig. 3 shows the data distribution of extracted pedestrian trajectories with crossing/non-crossing labels in terms of velocity (mph) and direction (rad) features using 20 crossing trajectories and 20 non-crossing trajectories (for better illustration) as examples. It is easy to find that the direction values of blue crossing points were relatively clustered within a range of 2.3 to 2.5rad, while the velocity values of orange non-crossing points were relatively aggregated within the 2.0 to 3.5mph range. In order to apply the Naïve Bayes algorithm, the values of each feature should be segmented into different ranges for probability calculation and the optimal combination of features was used as the input of the Naïve Bayes model. Therefore, how to determine the best segmentation of the data in each feature needs to be solved first.

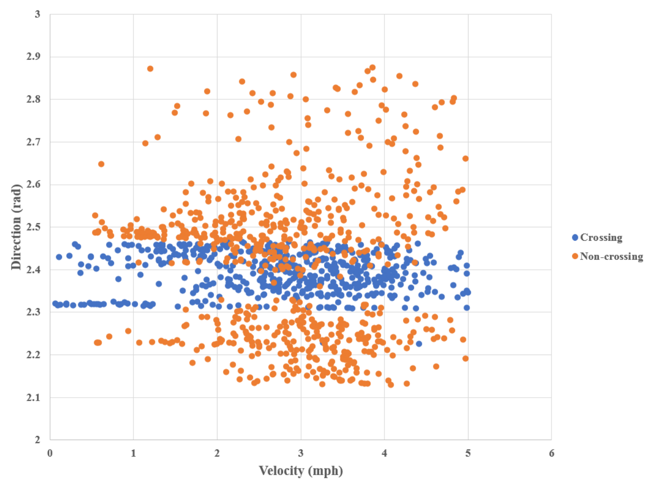


FIGURE 3. Sample trajectory data in velocity and direction features.

The main idea for segmentation is to divide each type of feature data into different ranges and then require the total number of data points in each range to satisfy the minimal amount requirement (*PointThres*). The number of ranges was predefined between the minimal number of divisions (*Nsegmin*) and the maximal number of divisions (*Nsegmax*). The training data (*TData*) with labels were first divided into *Nsegmax* ranges, and checked to ensure the data in each range satisfy the *PointThres* requirement. If not, the number of ranges was decreased by one and then checked the requirement again. This way, the maximal allowable number of segmentations (*MaxAllowseg*) can be obtained. Note that for cases in which the number of segmentations was less than *MaxAllowseg*, the data in each segmentation always met the *PointThres* requirement. After obtaining the *MaxAllowseg* information, probabilities can be calculated for the cases where the number of segmentations ranged from one to *MaxAllowseg*. The output of the training process was a trained prediction model that included the probability information for all the allowable segmentations. The training algorithm is described in Algorithm 1.

For the evaluation process, the inputs were unclassified data (*Data*), trained prediction model, and probability threshold for crossing warning (*WarnThres*). If there is no prediction model, the probability of correctly predicting crossing or non-crossing intention is 50%. This *WarnThres* parameter can be adjusted based on actual needs. A lower *WarnThres* value means there is a higher possibility of crossing intention prediction, which improves pedestrian safety. Given a data record, the data value of each feature can be fitted into one appropriate range/interval based on the trained model. Since the features were independent of each other, the corresponding probabilities can be multiplied directly.

Algorithm 1 Training Process

Input: TData, Nsegmin, Nsegmax, PointThres.
Output: MaxAllowseg, Count, Interval, Probability (Trained prediction model).

Begin
1. DataRange ← (max(TData)-min(TData))
2. Total ← length (TData)
3. // Find the maximal allowable number of segmentations.
4. For i = Nsegmax:-1:Nsegmin
5. Unit ← DataRange/i
6. For j = 1: i
7. Interval(j) ← min(Data) + j∗Unit
8. Count(j) ← n∈interval(j-1, j)
9. Endfor
10. If min(Count) ≥ PointThres
11. MaxAllowseg ← i
12. Break
13. Endif
14. Endfor
15. // Calculate probabilities for all the allowable number of segmentations.
16. For t = 1: MaxAllowseg
17. Probability(t) = Count (t)/Total
18. Endfor
19. Return: MaxAllowseg, Count, Interval, Probability.

Algorithm 2 Evaluation Process

Input: Data, WarnThres, Prediction Model (Probability).
Output: Classification label with quantitative confidence level information.
Note: Crossing (label = 1) and Non-crossing (label = 0).

Begin
1. // Find the max probability for label = 1 case and label = 0 case.
2. P(label=1) ← Data, Probability
3. P(label=0) ← Data, Probability
4. P_{max}(label=1) ← max(P(label=1))
5. P_{max}(label=0) ← max(P(label=0))
6. // Normalization.
7. P'_{max}(label=1) ← P_{max}(label=1)/(P_{max}(label=1) + P_{max}(label=0))
8. P'_{max}(label=0) ← P_{max}(label=0)/(P_{max}(label=1) + P_{max}(label=0))
9. // Determine the classification label.
10. If P'_{max}(label=1) ≥ WarnThres
11. Label ← 1 (Crossing)
12. else
13. Label ← 0 (Non-crossing)
14. Endif
15. Return: Label, P'_{max}.

Considering the different segmentations, two maximal probabilities for crossing and non-crossing labels were chosen, and the corresponding segmentations were considered as the optimal segmentations. It cannot be guaranteed that the sum of these two probabilities will be equal to one, since the chosen segmentations may be different for crossing and non-crossing cases. The next step was to normalize two selected maximal probabilities. If the normalized maximal crossing probability is greater than the predefined *WarnThres* value, the data record will be assigned a crossing label. Otherwise, the data record will be classified into the non-crossing case. The evaluation algorithm is described in Algorithm 2.

In order to improve prediction accuracy against incorrect predictions occurring in a small number of frames along the entire trajectory, three continuous frames were used for aggregate prediction. The final crossing/non-crossing label was determined by combining the prediction results for the current frame and the two previous frames by the majority rule. The final probabilities were used to provide quantitative confidence level information for crossing and non-crossing predictions.

IV. CASE STUDY

A. MODEL TRAINING

In this study, a VLP-16 LiDAR sensor with 10Hz rotation frequency was horizontally installed (about 6ft above the ground [18]) at the intersection of North Sierra Street and

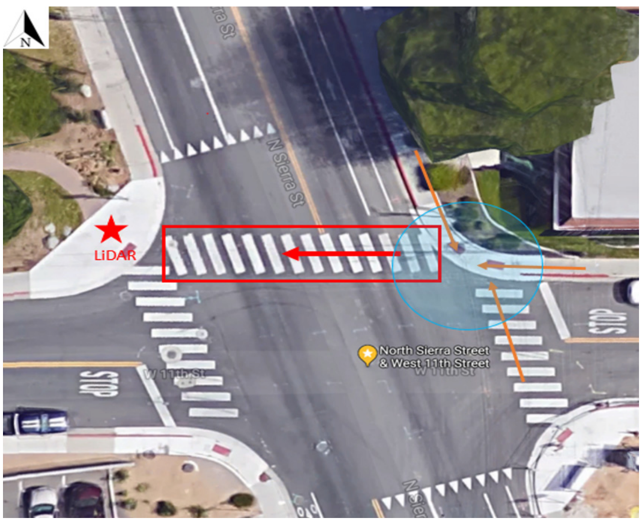


FIGURE 4. Google Maps of the selected intersection.

11th Street in Reno, Nevada to collect field data for 30 minutes (18,000 frames). The case study analyzed the pedestrian trajectories on the east side of North Sierra Street, and the selected crosswalk was marked with a red rectangle in Fig. 4. Three main pedestrian approach directions were southbound, northbound, and westbound. In total, 598 crossing trajectories (17,511 trajectory data points) and 622 non-crossing trajectories (17,786 trajectory data points) within the selected study area were extracted from the collected roadside LiDAR

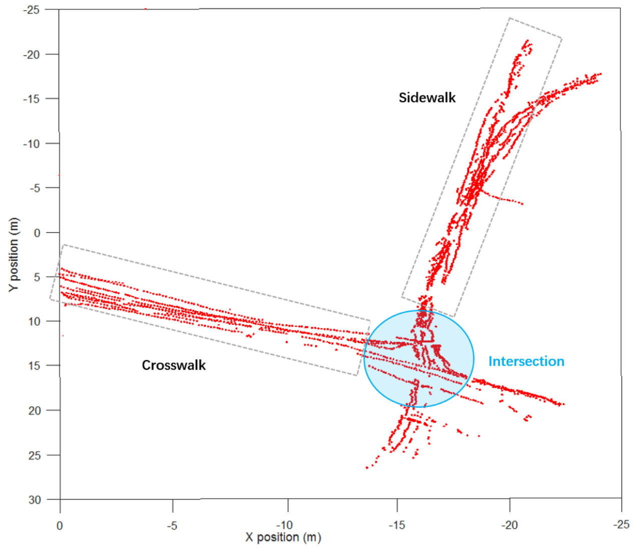
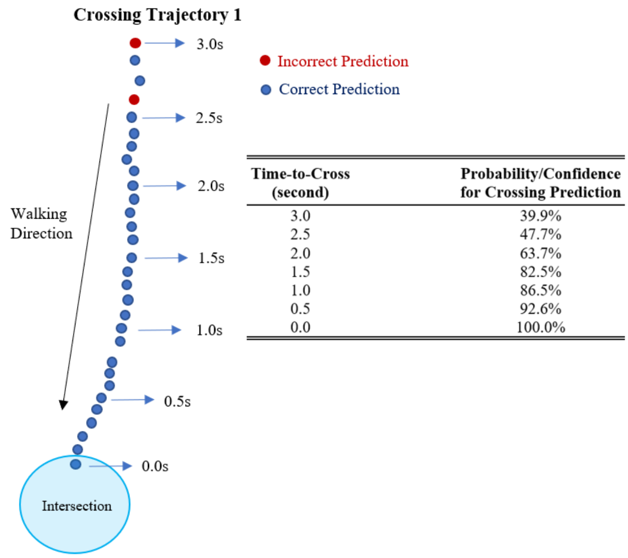


FIGURE 5. Sample pedestrian trajectory data.



Time-to-Cross (second)	Probability/Confidence for Crossing Prediction
3.0	39.9%
2.5	47.7%
2.0	63.7%
1.5	82.5%
1.0	86.5%
0.5	92.6%
0.0	100.0%

FIGURE 6. Crossing prediction for a sample crossing trajectory.

data. These trajectories recorded each pedestrian's walking behavior 3 to 0 seconds before actual crossing/non-crossing at the crossing facility. Fig. 5 demonstrates sample pedestrian trajectories extracted from roadside LiDAR data. The LiDAR sensor was located at the origin (0,0) and not shown in the plot. These trajectories were used for model training and evaluation.

Using the obtained pedestrian trajectory data, four features – X position (m), Y position (m), Velocity (mph), and Direction (rad), were selected for model training. Without losing the generality, the authors assumed these four features were independent of each other in practice [19], [20].

Based on the proposed modified Naïve Bayes algorithm, the minimal number of segmentations (*Nsegmin*) and maximal number of segmentations (*Nsegmax*) were set to one and ten, respectively. The required minimal number of data points (*PointThres*) in each segmentation was five. Using the modified Naïve Bayes method, a pedestrian crossing intention prediction model was obtained.

B. MODEL EVALUATION

To validate and evaluate the performance of the trained pedestrian crossing intention prediction model, a testing dataset collected from the same intersection that had not been seen by the trained model was applied to the obtained model. Here, ten crossing and ten non-crossing trajectories from the testing dataset were used to demonstrate the prediction process and results as examples. The probability threshold for crossing warning (*WarnThres*) was equal to 40.0%. In Fig. 6, the crossing trajectory 1 shows the pedestrian's walking path from 3 to 0 seconds before actual crossing. The blue and red dots represent the correct and incorrect crossing predictions. If the total predicted time length is T and the current time is t, then the Time-to-Cross is equal to $T - t$ (e.g., the time before actual crossing). It is easy to see that the probability of crossing predictions became increasingly higher as the pedestrian approached to the crossing facilities. At three seconds before crossing, the predicted probability for that pedestrian to cross the road upon arrival at the crossing facility was only 39.9% (less than *WarnThres* 40.0%, therefore gave an incorrect prediction – non-crossing), while the probability increased to 86.5% at one second before crossing. At the moment of actual crossing, the predicted probability for crossing was 100.0%.

For all ten crossing (label = 1) and ten non-crossing (label = 0) trajectories, prediction accuracies based on Time-to-Cross are listed in Table 1(a) and Table 1(b). Note that since the crossing warning threshold was set to 40% (less than 50%, which is randomly guessing), there was a higher probability of correct predictions in crossing than non-crossing. The total average prediction accuracies of crossing and non-crossing pedestrians with 40% warning threshold were roughly 97% and 84%, respectively. The performance/accuracy of the prediction model is restricted by: 1) pedestrians have greater motion variability than vehicles, which makes prediction more difficult, especially for irrational behaviors; 2) the collected LiDAR data for pedestrians at farther distances are limited, which affects the detailed description of pedestrians' behaviors; 3) the accuracy of extracted trajectory information.

To examine the model's property in real-time calculation, we implemented the algorithm in MATLAB and recorded the computation time on a regular Dell desktop (Intel Core i7-4790 CPU (3.60G Hz) and 16GB of RAM), it took only about 0.20 milliseconds to process one single trajectory data point and make a prediction, which was ideal for real-time applications.

TABLE 1. (a) Prediction accuracy for ten crossing trajectories. (b) Prediction accuracy for ten non-crossing trajectories.

(a)

Time-to-Cross (second)	Trajectory										Prediction Accuracy
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	
3.0	0	1	1	1	1	1	1	1	1	1	90%
12.5	1	1	1	1	1	1	1	1	1	1	100%
2.0	1	1	1	1	1	1	1	1	1	1	100%
1.5	1	1	1	1	1	1	1	1	0	1	90%
1.0	1	1	1	1	1	1	1	1	1	1	100%
0.5	1	1	1	1	1	1	1	1	1	1	100%
0.0	1	1	1	1	1	1	1	1	1	1	100%
Total Average Accuracy	Prediction results: 1(crossing) and 0 (non-crossing) Warning threshold = 40%										97%

(b)

Time-to-Cross (second)	Trajectory										Prediction Accuracy
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	
3.0	0	0	0	1	0	0	0	0	0	1	80%
12.5	0	0	0	0	0	0	1	0	1	1	70%
2.0	0	1	0	0	0	0	1	0	0	0	80%
1.5	1	1	0	0	0	0	0	1	0	0	70%
1.0	0	0	0	0	1	0	0	0	0	0	90%
0.5	0	0	0	0	0	0	0	0	0	0	100%
0.0	0	0	0	0	0	0	0	0	0	0	100%
Total Average Accuracy	Prediction results: 1(crossing) and 0 (non-crossing) Warning threshold = 40%										84%

TABLE 2. Sensitivity analysis for crossing warning threshold.

Crossing Warning Threshold (WarnThres)	Crossing Accuracy	Non-crossing Accuracy
35%	100.0%	33.3%
36%	99.5%	37.1%
37%	98.9%	55.3%
38%	98.8%	69.6%
39%	98.3%	84.0%
40%	98.0%	84.9%
41%	97.7%	86.0%
42%	97.6%	87.2%
43%	97.5%	87.8%
44%	96.9%	88.5%
45%	96.1%	89.1%
46%	94.9%	90.7%
47%	94.4%	92.0%
48%	93.5%	93.4%
49%	93.1%	94.0%

Next, the authors also conducted a sensitivity analysis of prediction accuracy and crossing warning threshold (*WarnThres*). As shown in Table 2, with a decrease of the crossing warning threshold, the crossing prediction accuracy increased, and non-crossing prediction accuracy decreased. It indicates that if the goal is to ensure the accuracy of crossing prediction is as good as possible, the crossing warning threshold should be decreased, while if the prediction accuracies for both crossing and non-crossing are sought, the crossing warning threshold should be increased to a relatively high value. Therefore, the selection of the warning threshold depends on the objective of applications in practice.

To further evaluate the performance of the trained modified Naïve Bayes crossing prediction model, the authors further conducted a comparison analysis by using the same training and testing datasets to train and evaluate a model based on the ANN. The backpropagation (BP)-neural network is a multilayer feed-forward artificial neural network [21]. It is composed of an input layer, hidden layer(s), an output layer, and neurons in each layer. The input data is fed into the input layer. Then, the activity of each hidden layer is determined by the inputs and the weights that connect the input layer and hidden layer. A similar process occurs between the hidden layer and output layer. The transmission from one neuron in one layer to another neuron in the next layer is independent. The output layer produces the estimated outcomes. The comparison information (error) between target outputs and estimated outputs is given back to the input layer as a guide to adjust the weights in the next training round. Through this iteration process, the neural network gradually learns the inner relationship between input and output by adjusting the weights for each neuron in each layer to reach the best accuracy. When the minimal error is reached, or the number of iterations is beyond the predefined value, the training process is terminated with fixed weights [22]. The determined ANN model structure and functions are summarized as follows:

1) *Feature Selection*: X position, Y position, Velocity, Direction.
2) *The number of hidden layers*: 1
3) *The number of hidden neurons*: 5
4) *Training function*: Scaled conjugate gradient backpropagation.
5) *Transfer function*: Soft max transfer function.
6) *Performance function*: Mean squared error performance function.

The average prediction accuracies for all collected crossing and non-crossing trajectories were around 91% and 93%, respectively. The results clearly indicated the proposed modified Naïve Bayes model had advantages over the ANN model in the following two main aspects: 1) it can provide quantitative confidence level information for prediction results; and 2) the crossing warning threshold parameter can be adjusted according to the actual needs in real-world situations.

V. CONCLUSION

This paper presents a new probabilistic approach to predicting pedestrians' crossing intention before their actual arrival at the crosswalk using roadside LiDAR sensors. The proposed model is based on a modified Naïve Bayes method and trajectory-level movement data and featured by providing real-time and quantitative confidence level information with the predictions. The major steps include pedestrian trajectory acquisition, pedestrian feature extraction, prediction model training and evaluation. A comprehensive case study demonstrated the effectiveness of the proposed method using real data, and the trained modified Naïve Bayes prediction model had more flexible applications in pedestrian crossing predictions than the popular ANN model. Applications of such developments not only support proactive protection of pedestrians in future connected-vehicle environments, moreover, probabilistic predictions with adjustable parameters can also help with other real-world applications as a standalone safety enhancement technology, such as real-time warning systems at unsignalized intersections and mid-block crosswalks, and in smart signals to automatically adjust the start and termination time of pedestrian phases. Ongoing and future studies along this path are being focused on model evaluation and adjustments with an increased level of data variety and complexity. A new experimental site is being developed at the City of Las Vegas with multiple 32-laser LiDAR sensors installed at critical locations for better detection. With much heavier pedestrian volume and more versatile pedestrian behaviors, advanced pedestrian behavior prediction models would be expected.

REFERENCES

[1] *Connected Vehicle Basics, Intelligent Transportation Systems Joint Program Office at United States Department of Transportation*. Accessed: Jul. 8, 2019. [Online]. Available: https://www.its.dot.gov/index.htm

[2] *National Connected Vehicle Field Infrastructure Footprint Analysis*, Accessed: Jul. 8, 2019. [Online]. Available: http://sp.stsmo.transportation.org/Documents/Exec%20Summary%20Final.pdf

[3] B. Li, T. Zhang, and T. Xia, "Vehicle detection from 3D Lidar using fully convolutional network," 2016, *arXiv:1608.07916*. [Online]. Available: https://arxiv.org/abs/1608.07916

[4] H. Wang, B. Wang, B. Liu, X. Meng, and G. Yang, "Pedestrian recognition and tracking using 3D LiDAR for autonomous vehicle," *Robot. Auton. Syst.*, vol. 88, pp. 71–78, Feb. 2017.

[5] H. Wang, X. Lou, Y. Cai, and L. Chen, "A 64-line Lidar-based road obstacle sensing algorithm for intelligent vehicles," *Sci. Program.*, vol. 2018, Nov. 2018, Art. no. 6385104. doi: 10.1155/2018/6385104.

[6] J. Jung, E. Che, M. J. Olsen, and C. Parrish, "Efficient and robust lane marking extraction from mobile Lidar point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 147, pp. 1–18, Jan. 2019.

[7] J. Wu, H. Xu, and J. Zhao, "Automatic background filtering and lane identification with roadside LiDAR data," *IEEE Intell. Transp. Syst. Mag.*, to be published. doi: 10.1109/MIS.2019.2918115.

[8] J. Zhao, H. Xu, D. Wu, and H. Liu, "An artificial neural network to identify pedestrians and vehicles from roadside 360-degree LiDAR data," in *Proc. 97th Transp. Res. Board Annu. Meeting*, Washington DC, USA, Jan. 2018, pp. 18-03129-1–18-03129-5.

[9] J. Zhao, H. Xu, H. Liu, J. Wu, Y. Zheng, and D. Wu, "Detection and tracking of pedestrians and vehicles using roadside LiDAR sensors," *Transp. Res. C, Emerg. Technol.*, vol. 100, pp. 68–87, Mar. 2019.

[10] Y. Hashimoto, Y. Gu, L. T. Hsu, M. Iryo-Asano, and S. Kamijo, "A probabilistic model of pedestrian crossing behavior at signalized intersections for connected vehicles," *Transp. Res. C, Emerg. Technol.*, vol. 71, pp. 164–181, Oct. 2016.

[11] F. Schneemann and P. Heinemann, "Context-based detection of pedestrian crossing intention for autonomous driving in urban environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Daejeon, South Korea, Oct. 2016, pp. 2243–2248.

[12] J.-Y. Kwak, B. C. Ko, and J.-Y. Nam, "Pedestrian intention prediction based on dynamic fuzzy automata for vehicle driving at nighttime," *Infr. Phys. Technol.*, vol. 81, pp. 41–51, Mar. 2017.

[13] B. Völz, H. Mielenz, R. Siegwart, and J. Nieto, "Predicting pedestrian crossing using quantile regression forests," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Gothenburg, Sweden, Jun. 2016, pp. 426–432.

[14] J. Zhao, H. Xu, J. Wu, Y. Zheng, and H. Liu, "Trajectory tracking and prediction of pedestrian's crossing intention using roadside LiDAR," *IET Intell. Transp. Syst.*, vol. 13, no. 5, pp. 789–795, May 2019.

[15] *Velodyne VLP-16 User's Manual and Programming Guide*. Accessed: Jul. 8, 2019. [Online]. Available: https://usermanual.wiki/Pdf/VLP16Manual.1719942037.pdf

[16] D. D. Lewis, "Naive (Bayes) at forty: The independence assumption in information retrieval," in *Proc. Eur. Conf. Mach. Learn.*, Berlin, Germany, Apr. 1998, pp. 4–15.

[17] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proc. 23rd Int. Conf. Mach. Learn.*, Pittsburgh, PA, USA, Jun. 2016, pp. 161–168.

[18] J. Zhao, H. Xu, J. Wu, C. Lin, and H. Liu, "Detection range analysis of roadside LiDAR sensors," in *Proc. 98th Transp. Res. Board Annu. Meeting*, Washington, DC, USA, Jan. 2019, pp. 05458-1–05458-19.

[19] H. Zhang, "The optimality of naive Bayes," in *Proc. 17th Int. FLAIRS Conf.*, Marco Island, FL, USA: AAAI Press, May 2004.

[20] C. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*, Cambridge, U.K.: Cambridge Univ. Press, 2008.

[21] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 4th ed. Cambridge, MA, USA, Academic, 2009.

[22] S. C. Wang, *Interdisciplinary Computing in Java Programming*, vol. 743. Berlin, Germany: Springer, 2012.

JUNXUAN ZHAO received the B.E. degree in electrical engineering and automation from Anhui Agricultural University, China, in 2013, the M.S. degree in electrical engineering from Colorado State University, in 2015, and the M.S. degree in civil engineering from Texas Tech University, in 2018, where she is currently pursuing the Ph.D. degree with the Civil, Environmental, and Construction Engineering Department. Her research interests include intelligent transportation systems, GIS in Transportation, and GNSS.

YINFENG LI received the B.S. and Ph.D. degrees from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2010 and 2015, respectively. He is currently an Associate Professor with the College of Civil and Architectural Engineering, North China University of Science and Technology. His research interests include air traffic flow management, airspace management, and traffic data analysis.

HAO XU received the Ph.D. degree in civil engineering from Texas Tech University, in 2011. He is currently an Associate Professor with the Department of Civil and Environmental Engineering, University of Nevada, Reno. His main areas of research include intelligent transportation systems, traffic operation and control, and development of applications for transportation engineering.

HONGCHAO LIU received the Ph.D. degree in civil engineering from the University of Tokyo, Tokyo, Japan. He was a Postdoctoral Researchers with the PATH Program, Institute of Transportation Studies, University of California at Berkeley, Berkeley, CA, USA. He is currently a Professor with the Department of Civil, Environmental and Construction Engineering, Texas Tech University, Lubbock, TX, USA. His research interests include traffic operation and control, intelligent transportation systems, data analytics, and transportation system modeling. He is also a Past Member of the Artificial Intelligence and Advanced Computing Applications Committee, Transportation Research Board, National Research Council, USA.