Center for Advanced Multimodal Mobility Solutions and Education

Project ID: 2020 Project 04

MACHINE LEARNING-BASED TRAJECTORY OPTIMIZATION OF CONNECTED AND AUTONOMOUS VEHICLES

Final Report

by

Wei Fan (ORCID ID: https://orcid.org/0000-0001-9815-710X)
Pengfei Liu (ORCID ID: https://orcid.org/0000-0001-7217-151X)

Wei Fan, Ph.D., P.E.
Director, USDOT CAMMSE University Transportation Center
Professor, Department of Civil and Environmental Engineering
The University of North Carolina at Charlotte
EPIC Building, Room 3261, 9201 University City Blvd, Charlotte, NC 28223
Phone: 1-704-687-1222; Email: wfan7@uncc.edu

for

Center for Advanced Multimodal Mobility Solutions and Education
(CAMMSE @ UNC Charlotte)
The University of North Carolina at Charlotte
9201 University City Blvd
Charlotte, NC 28223

September 2021

ACKNOWLEDGEMENTS

This project was funded by the Center for Advanced Multimodal Mobility Solutions and Education (CAMMSE @ UNC Charlotte), one of the Tier I University Transportation Centers that were selected in this nationwide competition, by the Office of the Assistant Secretary for Research and Technology (OST-R), U.S. Department of Transportation (US DOT), under the FAST Act. The authors are also very grateful for all of the time and effort spent by DOT and industry professionals to provide project information that was critical for the successful completion of this study.

DISCLAIMER

Table of Contents

List of Figures

List of Tables

EXECUTIVE SUMMARY

Connected and autonomous vehicle (CAV) technologies provide solutions to the existing problems of the transportation systems. As widely known, CAVs can communicate with each other so that they can have coordinated accelerating or decelerating movements. In this manner, CAVs only need a smaller headway which will lead to a higher roadway capacity. For signalized intersections, CAVs can communicate with the signal lights to adjust their speeds when approaching the intersection, so that they can arrive at the intersection during green light. CAVs bring with them many benefits including improving safety, reducing emissions and increasing mobility of the transportation system.

In past decades, numerous research efforts have been made to focus on modeling longitudinal driver behaviors of traditional vehicles. Most microscopic models assume that human drivers react to the stimuli from leading vehicles to keep a safe headway with a desired velocity. In recent years, with the emerging of CAVs, new car following models have been developed to accommodate the longitudinal driving behavior of CAVs. Efforts are needed to calibrate these car following models, and the results are highly related to the data availability, calibration method, and model structure. Despite different mechanisms and software interfaces, when multiple simulation software applications are compared, it seems that errors cannot be eliminated no matter how many parameters are introduced. Machine learning has achieved much success in recent years. It allows the agent to keep learning from observations, actions conducted, and rewards received. When presented with a sequence of states and corresponding actions, extracted from the trajectory data, the algorithm can learn how the vehicles act when faced with varying traffic conditions. The algorithm learns by associating any state observation, such as reaction time, speed, headway, and acceleration rate. The degree to which the agent action matches the vehicle’s action constitutes a reward in the learning sequence. In order to better predict the upcoming states of CAVs under varying traffic conditions, there is a critical need to model the car following trajectory data using machine learning approaches.

This research will compare the prediction accuracy of machine learning method with that of existing car following model using historical trajectory data, and therefore will lead to a better understanding of how CAVs operate in the freeway system.

Chapter 1. Introduction

1.1. Problem Statement

Connected Vehicle (CV) and Autonomous Vehicle (AV) technologies will change the way vehicles are driven in the highway system and have a significant impact on transportation operations, safety, and environment (Campbell and Alexiadis 2016). Driverless Cars (DLC) can keep a shorter headway and maintain consistent acceleration and deceleration rates due to the absence of perception errors and the minimal perception and reaction time (Shi and Prevedouros 2016).

Many studies focused on vehicle trajectory optimization for minimum speed oscillations and minimum conflicts in freeway lane changing and merging. Ntousakis et al. (2016) presented a longitudinal trajectory planning methodology to assist the merging of vehicles on highways. The acceleration and its first and second derivatives are minimized to achieve safe and traffic-efficient merging. Ahn et al. (2013) proposed a rolling-horizon model for an individual CAV control strategy that minimizes fuel consumption and emissions at different grades. Yang and Jin (2014) studied a vehicle speed control strategy to reduce vehicle fuel consumption and emissions. Wang et al. (2014) proposed optimal control models to determine optimal accelerations of a platoon of CAVs to minimize a variety of objective cost functions in a rolling horizon manner. Wang et al. (2016) investigated distributed CAV acceleration control methods to mitigate formation and propagation of moving jams.

Li et al. (2018) formulated a simplified traffic smoothing model to guide movements of CAVs on a general one-lane highway segment. The model confined each vehicle's trajectory as a piecewise quadratic function and let all trajectories in the same platoon share identical acceleration rates. The proposed model was able to optimize traffic performance in terms of fuel efficiency and driving comfort.

Guo et al. (2019) proposed an algorithm for the integrated optimization problem that can simultaneously optimize the trajectories of CAVs and intersection controllers. The results proved the efficiency and sound performance of the proposed optimization framework. The average travel time can be reduced by 35% compared to the adaptive signal control.

Receveur et al. (2017) optimized the trajectory of unmanned terrestrial vehicles so as to reduce consumption, travel time or to improve comfort. Main focuses were on testing different criteria and the possibility of using genetic algorithm to improve the potential field methods. The results showed that potential field methods could be improved by optimizing the path and the correlated motion.

Abbas and Chong (2013) compared the machine learning approach with regression analysis when modeling a car following trajectory data. The results showed that both the machine learning and regression analysis could predict the upcoming acceleration value. However, only the machine learning approach could reproduce the vehicle trajectory, while the regression analysis could ultimately lead to an erroneous model.

Hu and Sun (2019) proposed an online system control algorithm for multilane freeway merging areas in CAV environment based on optimizing vehicles' lane changing and car following trajectories. A simulation platform based on VISSIM was developed for computation and visualization. The results demonstrated that the proposed algorithm outperforms previous co-operative merging algorithms consistently with respect to delays and average travel speeds.

This research will compare the prediction accuracy of machine learning method with that of the existing car following model using historical trajectory data, and therefore will lead to a better understanding of how CAVs operate in the freeway system.

1.2. Objectives

The main objective of this research project is to predict vehicle trajectory in CAV environment using machine learning approach. The objectives of this project are to:

1. To conduct a comprehensive review of the state-of-the-art and state-of-the-practice on CAV trajectory prediction;

2. To develop suitable car following models for CAV driving behavior;

3. To identify potential model parameters for vehicle trajectory prediction using the machine learning method;

4. To compare the prediction results of proposed machine learning method with that of the existing car-following models and provide recommendations on future research directions.

1.3. Expected Contributions

In order to predict vehicle trajectory in CAV environment and develop the guidelines, comparison between the machine learning method and existing car-following model is conducted in this research. The expected contributions from this research are summarized as follows:

1. A review of CAV technologies and CAV trajectory prediction analysis methods;

2. Identification and development of CAV car following models and collect the CAV trajectory data;

3. Guideline on prediction accuracy of vehicle trajectory using the machine learning approach.

1.4. Report Overview

The research will be structured as shown in Figure 1.1. In this chapter, the background and motivation of the study have been discussed, followed by the research objectives and expected contributions.

Chapter 2 presents a comprehensive literature review of the current state-of-the-art and state-of-the-practice of CAV technologies and various methodological approaches to analyze vehicle trajectory with or without CAVs. This chapter gives a clear picture of existing vehicle trajectory prediction methods in consideration of CAVs, possible modeling scenarios, and suitable features to predict the trajectory. To get a better understanding of the capability and feasibility of the machine learning methods, several previous studies using machine learning methods for vehicle trajectory prediction are investigated and presented in as well.

Chapter 3 presents potential freeway segments that have been used to conduct CAV analysis and collect necessary vehicle trajectory data related to the selected freeway segments. A freeway segment is selected in Los Angeles, California. The NGSIM database provides the historical trajectory data of the selected freeway segment. A consolidated historical trajectory data is collected in each lane of the freeway segment. With the information on the trajectory data, researchers can conduct research on the selected freeway segment, make better decisions on trajectory prediction and evaluate the prediction accuracy.

Chapter 4 discusses the previous a state-of-the-practice car-following model, i.e., the Intelligent Driver Model, and the machine learning method. As a newly developed machine learning method, the XGBoost model is proposed to predict vehicle trajectory in this study. In order to precisely predict vehicle trajectory, various features are selected which are essential for any vehicle to decide its acceleration rate. The Root mean square error (EMSE) and Mean absolute error (MAE) are used to compare the results of the proposed XGBoost model and those of the Intellignet Driver Model.

Chapter 5 describes the results of the proposed models in detail. The prediction errors of the XGBoost model and the Intelligent Driver Model are discussed. Also, the feature importance to predict the vehicle trajectory is ranked. So that the most important features that impact vehicle trajectory could be identified.

Chapter 6 will conclude the report with a summary of the prediction results. Directions for future work will also be provided.

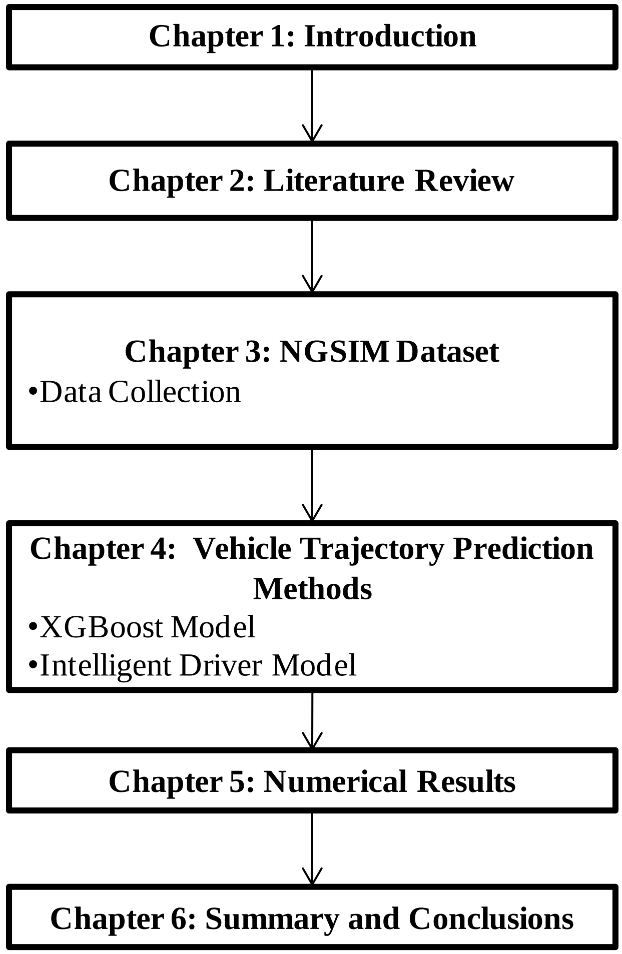

Figure 1.1 Research Structure

Chapter 2. Literature Review

2.1. Introduction

This chapter provides a comprehensive review of the current state-of-the-art and state-of-the-practice on machine learning technologies and various machine learning approaches to analyzing the vehicle trajectory with or without CAVs. This should give a clear picture of existing vehicle trajectory analysis methods in consideration of CAVs, possible modeling scenarios, and suitable parameters to predict vehicle trajectory.

The following sections are organized as follows. Section 2.2 presents definitions of machine learning technologies, followed by current technologies in use and benefits of vehicle trajectory analysis. Section 2.3 details existing vehicle trajectory analysis methods using traditional methods and machine learning approaches. Particular attention will be given to machine learning methods since they are capable of predicting vehicle trajectory based on the current status of the vehicle and its leading vehicle. The vehicle trajectory data used in this study is from the Next Generation SIMulation (NGSIM) program that was launched by FHWA. A description of this dataset is presented in section 2.4. Finally, section 2.5 concludes this chapter with a summary.

2.2. Machine Learning Technology

Nation's economy, safety, and quality of life are influenced by a well-behaved transportation system. Yet, demands in transportation are ever increasing due to trends in population growth, emerging technologies, and the increasing globalization of the economy, which have kept pushing the system to its limits. The increasing rate of the number of vehicles is at a point that has been even more than the overall population increasing rate, which leads to more congested and dangerous roadways. This problem cannot be addressed by just adding more roads or lanes anymore. The construction cost is very high and the time to get the result is too lengthy to catch up with the vehicle increase rate.

One way to improve upon the feet management is by viewing the road as an information highway as opposed to highway for vehicles. The scale of ingested data in the transportation system and even the interaction of various components of the system that generates the data have become a bottleneck for the traditional data analytics solutions. On the other hand, machine learning is a form of Artificial Intelligence (AI) and a data-driven solution that can cope with the new system requirements. Machine learning can quickly and effectively learn the latent patterns of historical data to model the behavior of a system and to respond accordingly in order to automate the analytical model building.

The availability of increased computational power and collection of the massive amount of data have redefined the value of the machine learning-based approaches for addressing the emerging demands and needs in transportation systems.

Machine learning solutions have already begun their promising marks in the transportation industry, where it is proved to even have a higher return on investment compared to the conventional solutions. However, the transportation problems are still rich in applying and leveraging machine learning techniques and need more consideration. The underlying goals for these solutions are to reduce congestion, improve safety and diminish human errors, mitigate unfavorable environmental impacts, optimize energy performance, and improve the productivity and efficiency of surface transportation.

In recent years, machine learning techniques have become an integral part of realizing smart transportation. The development of traffic information acquisition technologies (such as data of GPS trajectories) has provided us with a large amount of traffic data, which in turn paves the road to develop a more accurate vehicle trajectory prediction model based on data mining. Compared with traditional parametric models, data mining algorithms can explore implicit relationships between variables. In Intelligent Transport Networks (ITS) context, accurate prediction of future traffic conditions is essential to mitigate traffic congestion and to respond to the traffic incidents. Statistical machine learning algorithms have also found their way in supporting smart transportation.

Machine learning methods can be characterized based on the type of "learning." There exist several basic types of learning methods, such as: (1) supervised learning where previously labeled data is used to guide the learning process; (2) unsupervised learning, where only unlabeled data is used; (3) reinforcement learning, where the learning process is guided by a series of feedback/reward cycles.

2.2.1. Supervised Learning Technology

Supervised learning method trains a function (or algorithm) to compute output variables based on a given data in which both input and output variables are present. For example, for a given highway, input parameters can be volume (i.e., number of vehicles per hour), current time, and age of the driver, and corresponding output parameter can be the average traffic speed. The learning algorithm utilizes this information for automated training of a function (or algorithm) that computes the speed from a given input. Often, the goal of a learning process is to find a function that minimizes the risk of prediction error that is expressed as a difference between the real and computed output values when tested on a given data set. In such cases, the learning process can be controlled by predetermined acceptable error threshold. The supervised learning process can be thought of as a collection of comments provided by a driving instructor during a lesson in which the instructor explains what should be done (output variables) in different situations (input variables). These comments are adapted by a student driver and turned into a driver behavior. The predetermined thresholds can be thought of as the standards provided by external examiner such as standards published by the Department of Motor Vehicles to pass the driving exam. In this case, the student driver knows the standard way to drive (i.e., actual output) and steps to achieve it (i.e., actual inputs) before he or she starts driving lessons. For the student driver, it becomes an iterative process to achieve acceptable performance. In every iteration, the student driver makes mistakes that are corrected by the driving instructor (i.e., training the new student driver). This iterative process ends when the student successfully gets driving license. Here, two big categories of supervised learning methods, namely, classification and regression, are discussed below.

2.2.1.1 Classification

For a classification problem, the goal of the machine learning algorithm is to categorize or classify given inputs based on the training data set. The training data set in a classification problem includes set of input–output pairs categorized in classes. Many classification problems are binary, i.e., only two classes such as True and False are involved. For example, the individual vehicle's speed data over time can be classified into "speeding" and "not-speeding." Another example of classification is categorical classification, e.g., volume and speed data over time for a highway segment can be classified into levels of service "A," "B," "C," "D," "E," and "F." When a new set of observations is presented to a trained classification algorithm, the algorithm categorizes each observation into a set of predetermined classes. Further details and selected classification methods are provided in subsequent sections.

2.2.1.2 Regression

For a regression problem, the goal of the machine learning algorithm is to develop a relationship between outputs and inputs using a continuous function to help machines understand how outputs are changing for given inputs. The regression problems can also be envisioned as prediction problems. For example, given the historic information about volume and speed for a given highway, the output can be the average speed of the highway for a next time period. The relationship between output variables and input variables can be defined by various mathematical functions (such as linear, nonlinear, and logistic functions).

To summarize, supervised learning depends on the availability of historic data. It is important to note that the data must include input and corresponding known output values in order to train the model. While classification methods are used when the output is of categorical nature, the regression methods are used for the continuous output.

2.2.2. Unsupervised Learning Technology

Unsupervised learning methods depend only on the underlying unlabeled data to identify hidden patterns of data instead of inferring models for known input–output pairs. Consider the same student-driver example, the learning process in this case can be thought of as the student driver with no theoretical instructions for a perfect driving and he/she is driving a vehicle without the driving instructor. Without the presence of correct driving and a driving instructor, the student-driver is forced to drive a vehicle by observing other drivers and deducing the correct pattern of driving. It is important to note that, the perception of "correct driving pattern" may vary for each student driver. Clustering and association are two popular families of methods for unsupervised learning problems.

2.2.2.1 Clustering

Clustering methods focus on grouping data in multiple clusters based on similarity between data points. Usually, clustering methods rely on mathematical models to identify similarities between unlabeled data points. The similarities between data points are identified by various methods such as Euclidean distance. Consider an example of a transportation engineer with a closed circuit television (CCTV) recording of peak hour traffic data for a highway segment without control information such as speed limit of the section. The engineer is trying to identify aggressive drivers, slow drivers, and normal drivers. The engineer's goal is to find clusters such as aggressive drivers, slow drivers, and normal drivers by observing their driving pattern data such as an acceleration and deceleration. In this case, it is important to note that the logic rules of such clusters are defined by the engineer based on his/her own domain expertise.

2.2.2.2 Association

Association method focuses on identifying a particular trend (or trends) in the given data set that represents major data patterns or, the so-called significant association rules that connect data patterns with each other. For example, given crash data of a highway section, finding an association between ages of the drivers involved in the crash, blood-alcohol level of the driver at the time of crash, and time of the day can provide critical information to plan sobriety checkpoint locations and times to reduce crash frequencies as well as fatalities. For the student-driver example, the association method can be thought of as the student-driver associating "normal driver behavior" with certain age group and speed range.

In summary, unsupervised learning tries to identify complex patterns based on the logic provided in the algorithm.

2.2.3. Reinforcement Learning Technology

Reinforcement learning is the training of machine learning models to make a sequence of decisions. The agent learns to achieve a goal in an uncertain, potentially complex environment. In reinforcement learning, an artificial intelligence faces a game-like situation. The computer employs trial and error to come up with a solution to the problem. To get the machine to do what the programmer wants, the artificial intelligence gets either rewards or penalties for the actions it performs. Its goal is to maximize the total reward.

Although the designer sets the reward policy, he/she gives the model no hints or suggestions for how to solve the game. It's up to the model to figure out how to perform the task to maximize the reward, starting from totally random trials and finishing with sophisticated tactics and superhuman skills. By leveraging the power of search and many trials, reinforcement learning is currently the most effective way to hint machine's creativity. In contrast to human beings, artificial intelligence can gather experience from thousands of parallel game plays if a reinforcement learning algorithm is run on a sufficiently powerful computer infrastructure. Table 2.1 provides a summary of different machine learning approaches.

Table 2.1 Summary of Different Machine Learning Approaches

Category	Algorithms
Supervised Learning	Support vector machine
	Decision tree
	Naïve Bayesian classifier
	Neural network
Unsupervised Learning	K-means
	Principal component analysis
	Expectation maximization
	Hierarchical clustering
Reinforcement Learning	Q-learning
	Monte-Carlo based method
	Temporal difference method

2.3. Vehicle Trajectory Analysis Methods

Ensuring safety is a top priority for the autonomous driving and advanced driver assistance systems (ADAS). In order to promise a high degree of safety, the ability to perceive surrounding situations and predict their development in the future is critical. The vehicle in driving encounters various types of dynamic traffic participants such as cars, motor bikes, and pedestrians which could be a potential threat to safe driving. In order to avoid an accident, the system should be able to analyze the pattern in their motion and predict the future trajectories in advance. If the system predicts where the surrounding vehicles are heading in the near future, the vehicle can plan its driving path in response to the situation to come such that the probability of collision is minimized. However, the trajectory of the surrounding vehicles is quite complex to analyze since it is governed by various latent factors determined by complex traffic situations and the state of these latent factors can change dynamically in real time (Park et al. 2018).

Thus far, various vehicle trajectory analysis techniques have been proposed including traditional methods, machine learning methods and deep learning methods.

2.3.1. Traditional Methods

Abbas et al. (2019) presented a multi model-based Extended Kalman Filter (EKF), which is able to predict a set of possible scenarios for vehicle future location. Five different EKF models were proposed in which the current state of a vehicle exists (e.g., a vehicle at an intersection or on a curve path). EKF with Interacting Multiple Model framework was explored and combined for mathematical model creation and probability calculation for that model to be selected for prediction. The results proved that installing flow rules can help the vehicle not to send a new request for path calculation, hence, reduces the network overhead (control messages) up to a certain extent.

Schreier et al. (2014) proposed a Bayesian trajectory prediction and criticality assessment system that allows one to reason about imminent collisions of a vehicle several seconds in advance. The author first inferred a distribution of high-level, abstract driving maneuvers such as lane changes, turns, road followings, etc. of all vehicles within the driving scene by modeling the domain in a Bayesian network with both causal and diagnostic evidences. This was followed by maneuver-based, long-term trajectory predictions, which themselves contain random components due to the immanent uncertainty of how drivers execute specific maneuvers. Taking all uncertain predictions of all maneuvers of every vehicle into account, the probability of the ego vehicle colliding at least once within a time span was evaluated via Monte-Carlo simulations and given as a function of the prediction horizon. This serves as the basis for calculating a novel criticality measure, the Time-To-Critical-Collision-Probability (TTCCP) –a generalization of the common Time-To-Collision (TTC) in arbitrary, uncertain, multi-object driving environments and valid for longer prediction horizons. The authors also conducted analysis in arbitrary road environments with integrated Bayesian approach (Schreier et al. 2016).

Wang et al. (2013) used coordinate transformation method and transform relationship among different variable parameters to establish model of vehicle movement. By using Petri net which had well layering and time sequence, vehicle trajectory, speed, side slip angle, and yaw rate were treated as parameters to describe the movements of vehicle. Domain of discourse and subordinating degree function were confirmed, and fuzzy rules related to controllability and driving comfort were established. Verification tests results showed that the Petri net model can describe the vehicle movement accurately.

Qiao et al. (2014) proposed a hidden Markov model (HMM)-based trajectory prediction algorithm, called Hidden Markov model-based Trajectory Prediction (HMTP). By analyzing the disadvantages of HMTP, a self-adaptive parameter selection algorithm called HMTP∗ is proposed, which captures the parameters necessary for real-world scenarios in terms of objects with dynamically changing speed. In addition, a density-based trajectory partition algorithm was introduced, which helps improve the efficiency of prediction.

Tran and Firl (2014) provided a feature normalization scheme and present a strategy for constructing three-dimensional Gaussian process regression models from two-dimensional trajectory patterns These models can capture spatio-temporal characteristics of traffic situations. Given a new, partially observed and unlabeled trajectory, the maneuver can be recognized online by comparing the likelihoods of the observation data for each individual regression model.

Yao et al. (2013) developed a constant Yaw Rate and Acceleration (CYRA) model based on real human driving data stored in a database. In real-time, the system generated parametric trajectories by interpolating k human lane change trajectory instances from the pre-collected database that are similar to the current driving situation.

Tsang et al. (1999) proposed an algorithm to predict vehicle trajectory based on image processing techniques of the radar return signals. The reliability of the algorithm was evaluated by comparison of its calculated radius of curvature with digital map database, GPS position, and yaw rate data. This approach produced an image which may be processed in a similar way to video data but had the benefit of all weather performance and additional high resolution range and velocity data.

Goli et al. (2018) used Gaussian Process Regression (GPR) to learn motion patterns from historical trajectory data collected with static sensors on the roads. The collected data from vehicles together with GPR models received from infrastructure were then used to predict the future trajectories of vehicles in the scene.

Wang et al. (2019) established a new method for vehicle trajectory prediction (TPVN), which was mainly applied to predict the vehicle trajectory in the short term. Based on the regularity of vehicle movement, the algorithm was helpful to predict the vehicle trajectory so as to estimate the position of the vehicle motion probability. The advantage of the TPVN algorithm was that the calculation result not only predicts the movement behavior of vehicles in different motion patterns but also the probability distribution of all possible trajectories of the vehicle in the future.

Yi et al. (2015) proposed a new vehicle trajectory prediction algorithm for adaptive cruise control (ACC). When vehicle trajectory prediction is not precise enough, it is possible for a neighboring vehicle to be detected as a target. Thus, the authors proposed a new method using both yaw rate and curvature rate to precisely predict vehicle trajectory and to resolve an undesirable case in ACC system.

Liu et al. (2014) developed a driving behavior estimation and classification model based on Hidden Markov Models (HMMs). The lane change behavior was estimated by observing the vehicle state emissions in the beginning stage of a lane change procedure, and then classified by the classifier before the vehicle crosses the lane mark. Ye et al. (2016) also used Hidden Markov Model for vehicle trajectory prediction.

In summary, traditional methods were capable of evaluating the vehicle trajectory. A variety of vehicle trajectory analysis studies using traditional methods have been done to achieve this goal. Table 2.2 exhibits a summary of the vehicle trajectory analysis studies based on traditional methods reviewed in this section.

Table 2.2 Vehicle Trajectory Analysis Studies Based on Traditional Methods

No.	Author, Year	Model
1	Abbas et al., 2019	Extended Kalman Filter
2	Schreier et al., 2014	Bayesian
3	Wang et al., 2013	Fuzzy colored petri net
4	Qiao et al., 2014	Hidden Markov models
5	Tran and Firl, 2014	Gaussian process regression models
6	Schreier et al., 2016	Bayesian network
7	Tsang et al., 1999	Automotive radar image processing
8	Goli et al., 2018	Gaussian process regression
9	Wang et al., 2019	TPVN
10	Yi et al., 2015	MATLAB/Simulink
11	Liu et al., 2014	Hidden Markov models
12	Ye et al., 2016	Hidden Markov models
13	Yao et al., 2013	Constant Yaw Rate and Acceleration (CYRA) model

2.3.2. Machine Learning Methods

Ju et al. (2019) presented a multi-layer architecture Interaction-aware Kalman Neural Networks (IaKNN) which involves an interaction layer for resolving high-dimensional traffic environmental observations as interaction-aware accelerations, a motion layer for transform the accelerations to interaction-aware trajectories, and a filter layer for estimating future trajectories with a Kalman filter. Experiments on the NGSIM dataset demonstrated that IaKNN outperforms the state-of-the-art methods in terms of effectiveness for trajectory prediction.

Xing et al. (2019) proposed a joint time-series modeling approach for leading vehicle trajectory prediction considering different driving styles. The proposed method enabled a precise and personalized trajectory prediction for the leading vehicle based on limited inter-vehicle communication signals, such as the vehicle speed and acceleration of the front vehicles. The feature importance of driving style recognition was also evaluated based on the Maximal Information Coefficient (MIC) algorithm. Then, a personalized joint time series modeling (JTSM) method based on the Long Short-Term Memory (LSTM) Recurrent Neural Network model (RNN) was proposed to predict the front vehicle trajectories. Results indicated that the proposed personalized JTSM approach shows a significant advantage over the baseline algorithms.

Woo et al. (2018) developed a method to predict trajectories of surrounding vehicles using a long short-term memory (LSTM) network. Trajectory prediction of surrounding vehicles is attracting a lot of attention now, and it is expected to apply to advanced driver-assistance systems (ADAS). Although many prediction methods using a deep learning framework have been proposed, most of them only focused on the subject vehicle even though surrounding vehicles largely affect the driving pattern of the subject. To solve this problem, the proposed method took into account the relationship between the subject and surrounding vehicles using the LSTM network. It was demonstrated that the proposed method successfully achieves the goal of the trajectory prediction.

Massaoud et al. (2019a) presented an approach to predict the motion of vehicles surrounding a target vehicle in a highway environment. The approach was based on an LSTM encoder-decoder that uses a social pooling mechanism to model the interactions between all the neighboring vehicles. Messaoud et al. (2019b) brought relational recurrent neural networks (RRNNs) to tackle the vehicle motion prediction problem. The authors proposed a RRNNs based encoder-decoder architecture where the encoder analyzes the patterns underlying in the past trajectories and the decoder generates the future trajectory sequence. The proposed method outperformed LSTM encoder decoder in terms of RMSE values of the predicted trajectories.

Dai et al. (2019) developed a spatio-temporal LSTM-based trajectory prediction model (STLSTM) which includes two modifications. The authors embed spatial interactions into LSTM models to implicitly measure the interactions between neighboring vehicles. The authors also introduced shortcut connections between the inputs and the outputs of two consecutive LSTM layers to handle gradient vanishment.

Nikhil and Morris (2018) proposed a convolutional neural network (CNN) based human trajectory prediction approach. Unlike more recent LSTM-based moles which attend sequentially to each frame, their model supports increased parallelism and effective temporal representation.

Xiao et al. (2018) discussed a new paradigm of GPS and OBD Integration (GOI) based on GPS receiver and on-board diagnostics (OBD) reader, which offers a feasible way for large-scale trajectory collection especially suitable for private cars. The authors proposed a vehicle positioning approach by employing supporting vector machine for regression (SVR) to achieve accurate and reliable vehicle position and trajectory prediction based on GOI.

Han et al. (2007) focused on detecting and predicting the changing lane intention and action of the preceding vehicle. The algorithm employed SVM for driving pattern recognition by integrating two different cues: motion cue and appearance cue, which was trained using two class feature sets extracted from examples of lane changing and lane keeping video sequences.

Izquierdo et al. (2017) evaluated two kinds of artificial neural networks over two different datasets to predict its trajectories. A Support Vector Machine classifier was used to classify the action that will be carried out. Two different architectures of neural networks were used in order to perform the lateral position prediction. The first one was a Nonlinear Autoregressive Neural Network (NARNN) which is especially indicated to predict time series in dynamical models. The second one was a feed-forward neural network (FFNN) which is indicated when a mapping between inputs and outputs is desired.

Boubezoul et al. (2009) proposed a vehicles trajectories analysis in bend within a suitable Support Vector Machine (SVM) algorithm framework. The goal of this study was to predict the failure trajectories by using real data measurements.

Choi et al. (2019) developed an arterial trajectory prediction model that predicts the next intersections that a vehicle will visit based on its previously visited intersections. The proposed model was based on Artificial Neural Networks and was trained and tested on one-year Bluetooth data from Brisbane, Australia.

Walker (2019) presented an attention-based recurrent neural network that is capable of accurately predicting the Cartesian trajectories of multiple human driven vehicles over a 3s prediction-horizon. The network was trained on a dataset that contains various potentially dangerous collision scenarios, where broken down vehicles block lanes on a motorway.

Pecher et al. (2016) used three approaches to vehicle trajectory prediction, along with extensions, and assessed their accuracy in an urban road network. These included an approach based on the intuition that drivers attempt to reduce their travel time, an approach based on artificial neural networks (ANN), and an approach based on Markov models.

Kim et al. (2017) proposed an efficient vehicle trajectory prediction framework based on recurrent neural network. The proposed trajectory prediction method employed the recurrent neural network called long short-term memory (LSTM) to analyze the temporal behavior and predict the future coordinate of the surrounding vehicles. The proposed scheme fed the sequence of vehicles' coordinates obtained from sensor measurements to the LSTM and produced the probabilistic information on the future location of the vehicles over occupancy grid map.

In summary, machine learning methods are capable of predicting vehicle trajectory. A variety of vehicle trajectory analysis studies using machine learning methods have been conducted to achieve this goal. Table 2.3 exhibits a summary of the vehicle trajectory analysis studies based on machine learning methods reviewed in this section.

Table 2.3 Vehicle Trajectory Analysis Studies Based on Machine Learning Methods

No.	Author, Year	Model
1	Ju et al., 2019	Kalman Neural Networks
2	Xing et al., 2019	Joint time series model
3	Woo et al., 2018	LSTM
4	Messaoud et al., 2019a	Relational recurrent neural networks
5	Messaoud et al., 2019b	Social pooling mechanism
6	Zhao et al., 2019	Multi-agent tensor fusion
7	Dai et al., 2019	Spatio-temporal LSTM
8	Nikhil and Morris, 2018	CNN
9	Han et al., 2007	SVM
10	Izquierdo et al., 2017	NARNN and FFNN
11	Boubezoul et al., 2009	SVM
12	Choi et al., 2019	ANN
13	Walker, 2019	RNN
14	Pecher et al., 2016	ANN
15	Kim et al., 2017	LSTM
16	Park et al., 2018	LSTM

2.3.3. Deep Learning Methods

Cheng and Sester (2018) proposed using a Long Short–Term Memory (LSTM) recurrent neural network based deep learning approach to model user behaviors. It encoded user position coordinates, sight of view, and interactions between different types of neighboring users as spatio–temporal features to predict future trajectories with collision avoidance. The real–world data–driven method can be trained with pre-defined neural networks to circumvent complex manual design and calibration. The results showed that ViewType-LSTM, which mimics how a human sees and reacts to different transport modes and can well predict mixed traffic trajectories in a shared space at least in the next 3 s, and was also robust in complicated situations.

Yoon and Kum (2016) developed a probabilistic lateral motion prediction algorithm based on multilayer perceptron (MLP) approach. The MLP model consisted of two parts; target lane and trajectory models. In order to develop an intuitive and accurate prediction algorithm, a lane-based trajectory prediction model was introduced based on the fact that vehicles drive within a lane except for during lane changes. The proposed MLP model outputs probabilities of how likely a vehicle followed each trajectory and each lane for a given input of vehicle position history including current position.

Choi et al. (2018) proposed a deep learning approach to learning and predicting network-wide vehicle movement patterns in urban networks. Inspired by recent success in predicting sequence data using recurrent neural networks (RNN), specifically in language modeling that predicts the next words in a sentence given previous words, this research aimed to apply RNN to predict the next locations in a vehicle's trajectory, given previous locations, by viewing a vehicle trajectory as a sentence and a set of locations in a network as vocabulary in human language.

Jeong et al. (2017) presented simulation results demonstrating the effectiveness of employing a deep neural network (DNN) for vehicle trajectory prediction. The DNN was trained to output the trajectory of the vehicle for the following few seconds.

Jiang et al. (2019) designed three kinds of deep neural networks: Long Short Term Memory (LSTM), Gated Recurrent Units (GRU), and Stacked Auto encoders (SAEs) to predict the position and the velocity of the forward vehicles. The performance of these three network models were verified on the NGSIM I-80 dataset which consists of real trajectories of vehicles on multi-lanes. The results demonstrated that in the three deep neural networks that were designed, the LSTM model performed better than GRU model and SAEs model in the area of trajectory prediction.

In summary, deep learning based method is capable of predicting vehicle trajectory. A variety of deep learning-based vehicle trajectory analysis studies have been done to achieve this goal. Table 2.4 exhibits a summary of the deep learning based vehicle trajectory analysis studies reviewed in this section.

Table 2.4 Vehicle Trajectory Analysis Studies Based on Deep Learning Methods

No.	Author, Year	Model
1	Cheng and Sester, 2018	LSTM
2	Yoon and Kum, 2016	Multilayer perceptron approach (MLP)
3	Choi et al., 2018	RNN
4	Jeong et al., 2017	Deep neural network
5	Jiang et al., 2019	LSTM, GRU, and SAEs

2.4. NGSIM Vehicle Trajectory Data

Traffic microsimulation models are becoming widely used and valuable tools in modeling existing and planned future transportation networks and conditions. These models can help transportation professionals make important decisions on such topics as new roadway alignments and configurations, new interchange configurations and locations, the addition of freeway auxiliary lanes, work zone management strategies and plans, operational and intelligent transportation system (ITS) strategies and plans, coordination and timing of traffic signals, and the addition of high-occupancy toll lanes. Although many of the microsimulation models used today are robust and provide a wide range of analysis options, some gaps and limitations still exist that can affect the accuracy of their results.

Since the 1970s, the Federal Highway Administration (FHWA) has been a leader in the development of traffic simulation models. Before FHWA took a leadership role, no commercial traffic simulation packages existed in the marketplace. To help achieve wider acceptance of the use of microsimulation systems and ensure the tools provide accurate results, FHWA's Traffic Analysis Tools Program launched the Next Generation SIMulation (NGSIM) program. NGSIM is a unique public-private partnership between FHWA and commercial microsimulation software developers, the academic research community, and the traffic microsimulation community. In undertaking this partnership, FHWA acts as a market facilitator and uses focused public resources to influence and stimulate the commercial simulation market by fostering a cooperative environment of public-private coordination.

Through the NGSIM program, FHWA develops several driver behavioral algorithms, along with supporting documentation and validation datasets, which represent the fundamental logic within microscopic traffic simulation models. These algorithms describe the interactions of multimodal travelers, vehicles, and highway systems and the influence posed by traffic control devices, delineation, congestion, and other features of the environment. NGSIM products include:

- Real-world datasets and their corresponding data descriptions. These datasets consist of detailed vehicle trajectory, wide-area detector, and supporting data for researching driver behavior. The vehicle trajectory data, which were collected using digital video cameras, are particularly valuable due to the unprecedented level of detail and accuracy: the precise location of each vehicle on a 0.5- to 1.0-kilometer section of roadway is recorded every one-tenth of a second.
- Core simulation algorithms, which are mathematical models that replicate fundamental driver behavior logic, such as how drivers follow each other or how and when drivers choose to change lanes, which are the foundational logic within traffic simulation models. The algorithms developed and validated under the NGSIM program are based on

collected real-world datasets and are intended to fill the gaps and limitations of current simulation models.

- Documentation of the core algorithms, including the theory and logic behind the algorithms, and documentation of the validation datasets.

From the outset, the NGSIM team sought input and advice from the traffic simulation community through the formation of three stakeholder groups that together represent the different perspectives of the community. These stakeholder groups included a traffic modelers group that represents researchers and others that develop driver behavior models, a software developers group that represents private vendors responsible for developing and maintaining commercial traffic simulation software, and a model users group that represents the practitioners who use traffic simulation models for decision making.

The NGSIM team and stakeholder groups first conducted a market assessment of traffic microsimulation models, including identifying their limitations and prioritizing the NGSIM algorithm research needs based on the objective of improving the core behavioral algorithms in microsimulation software. The NGSIM team then formulated high-level plans for collecting data and developing and validating the algorithms. These plans ensured that the research would be conducted through a consistent, rigorous process. The teams also provided an infrastructure for free and open sharing of data by developing data formats and a Web site for online dissemination of NGSIM products. The high-level system plans and projects infrastructure that enabled the team to proceed with the primary tasks of collecting real-world datasets, developing core simulation algorithms, and validating the algorithms using commercial simulation software.

As a result of the core simulation algorithms developed through NGSIM ultimately being incorporated into commercial simulation models, transportation practitioners are able to use microsimulation software more confidently knowing that traffic simulation experts developed the models' algorithms using high-quality, real world datasets. Improving the core algorithms ultimately lead to more reliable and valid transportation decisions, which is critical in the current environment of both shrinking transportation budgets and growing demand for accountable and efficient transportation investments. Enabling reliable and valid transportation decisions through improved traffic simulation modeling is the ultimate goal of the NGSIM program.

Li et al. (2019) proposed a coordination and trajectory prediction system (CTPS), which had a hierarchical structure including a macro-level coordination recognition module and a micro-level subtle pattern prediction module which solves a probabilistic generation task. The NGSIM US-101 highway dataset was used to extract training and testing data. The proposed system was tested on multiple driving datasets in various traffic scenarios, which achieves better performance than baseline approaches in terms of a set of evaluation metrics. The results also showed that using categorized coordination can better capture multi-modality and generate more diversified samples than the real-valued coordination, while the latter can generate prediction hypotheses with smaller errors with a sacrifice of sample diversity.

Wissing et al. (2018) presented a novel trajectory prediction approach utilizing a combination of maneuver classification and probabilistic estimation of temporal properties with a model based trajectory representation. The three parts of the prediction framework were evaluated on the NGSIM data set. It showed, that based on a good performance of the maneuver prediction as well as the time-to-lane-change estimation, lane change trajectories with high accuracy can be predicted.

Deo and Trivedi (2018) presented an LSTM model for interaction aware motion prediction of surrounding vehicles on freeways. The model assigned confidence values to maneuvers being performed by vehicles and outputted a multi-modal distribution over future motion based on them. The approach was compared with the prior art for vehicle motion prediction on the publicly available NGSIM US-101 and I-80 datasets. The results showed an improvement in terms of RMS values of prediction error.

Ding et al. (2013) discussed in detail the effectiveness of Back-Propagation (BP) neural network for prediction of lane-changing trajectory based on the past vehicle data and compared the results between BP neural network model and Elman Network model in terms of the training time and accuracy. Driving simulator data and NGSIM data were processed by a smooth method and then used to validate the availability of the model. The test results indicated that BP neural network might be an accurate prediction of driver's lane-changing behavior in urban traffic flow.

Tomar et al. (2010) employed a multilayer perceptron (MLP) to train itself from existing NGSIM field data and predict the future path of a lane changing vehicle. The impact and effectiveness of the proposed technique was demonstrated. Prediction results showed that an MLP is able to give the future path accurately only for discrete patches of the trajectory and not over the complete trajectory.

Altché and Fortelle (2017) presented a first step towards consistent trajectory prediction by introducing a long short-term memory (LSTM) neural network, which is capable of accurately predicting future longitudinal and lateral trajectories for vehicles on highway. Unlike previous work focusing on a low number of trajectories collected from a few drivers, this network was trained and validated on the NGSIM US-101 dataset, which contains a total of 800 hours of recorded trajectories in various traffic densities, representing more than 6000 individual drivers.

Past research has sought a better understanding of how to utilize the NGSIM vehicle trajectory data. Based on the literature review as presented above, Table 2.5 exhibits a summary of the existing studies using NGSIM data.

Table 2.5 Summary of Studies Using NGSIM Data

No.	Author, Year	Method
1	Li et al., 2019	Bayesian deep learning
2	Wissing et al., 2018	maneuver classification algorithm
3	Deo and Trivedi, 2018	LSTM
4	Ding et al., 2013	Back-propagation neural network
5	Tomar et al., 2010	Multilayer perceptron (MLP)
6	Altché and Fortelle, 2017	LSTM

2.5. Summary

A comprehensive review and synthesis of the current state-of-the-art and state-of-the-practice on vehicle trajectory prediction methods using machine learning technologies have been discussed and presented in the preceding sections. This is intended to provide a solid reference for and assistance in formulating vehicle trajectory analysis methods and developing effective research strategies for future tasks.

Chapter 3. NGSIM Dataset

3.1. Introduction

As discussed in the literature review conducted in Chapter 2, this chapter will identify potential freeway segment and collect necessary trajectory data related to the selected freeway segment. The case study is conducted in Los Angeles, California. To support the development of algorithms for driver behavior at microscopic levels, the Next Generation SIMulation (NGSIM) computer program has been collecting detailed, high-quality traffic datasets. The NGSIM datasets represent the most detailed and accurate field data collected to date for traffic microsimulation research and development. The US Highway 101 (US 101) dataset is one of several datasets collected under the NGSIM program.

The following sections are organized as follows. Section 3.2 presents information on the selected freeway segment. Section 3.3 presents the trajectory data related to the selected freeway segment. Finally, section 3.4 concludes this chapter with a summary.

3.2. The Potential Freeway Segment

3.2.1. Layout of the US Highway 101

Researchers for the NGSIM program collected detailed vehicle trajectory data on southbound US 101, also known as the Hollywood Freeway, in Los Angeles, CA, on June 15th, 2005. The study area is approximately 640 meters (2,100 feet) in length and consists of five mainline lanes throughout the section. An auxiliary lane is present through a portion of the corridor between the on-ramp at Ventura Boulevard and the off-ramp at Cahuenga Boulevard. Eight synchronized digital video cameras, mounted from the top of a 36-story building adjacent to the freeway, recorded vehicles passing through the study area. NG-VIDEO, a customized software application developed for the NGSIM program, transcribed the vehicle trajectory data from the video. This vehicle trajectory data provided the precise location of each vehicle within the study area every one-tenth of a second, resulting in detailed lane positions and locations relative to other vehicles. The map of the selected signalized intersection is shown in Figure 3.1.

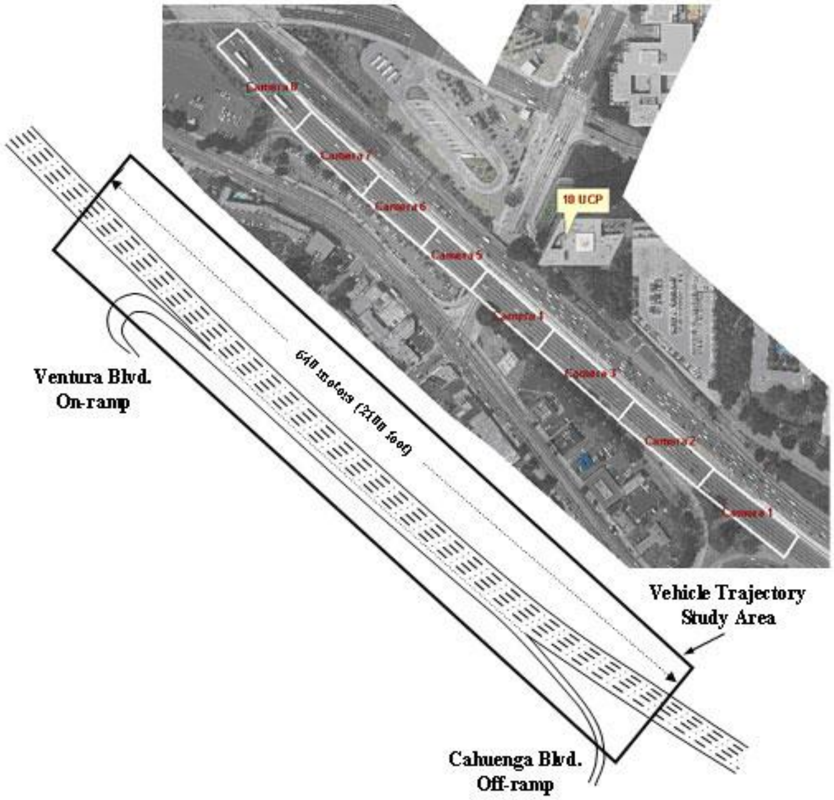

Figure 3.1 The Map of the Selected Freeway Segment

3.2.2. US 101 Dataset

A total of 45 minutes of data are available in the full dataset, segmented into three 15 minute periods: 7:50 a.m. to 8:05 a.m.; 8:05 a.m. to 8:20 a.m.; and 8:20 a.m. to 8:35 a.m. These periods represent the buildup of congestion, or the transition between uncongested and congested conditions, and full congestion during the peak period. In addition to the vehicle trajectory data, the US 101 dataset also contains computer-aided design and geographic information system files, aerial ortho-rectified photos, loop detector data, raw and processed video, weather data, and aggregate data analysis reports.

In this study, we consider a 15 minute segment of vehicle trajectories on the US101 highway. Since different vehicle type has different car following behavior, only passenger cars are involved in the analysis. The time period is between 7:50am and 8:05am, June 15th, 2005. In total, the selected dataset includes trajectories for 1,993 individual vehicles, recorded at 10 Hz.

3.3. Feature Extraction

The NGSIM dataset provides vehicle speed, position, acceleration rate, and headway of each individual vehicle. In this study, the objective is to predict the acceleration rate for the object vehicle, which is the determining factor of vehicle trajectory. Under the CAV environment, the object vehicle can receive information from its leading vehicle. The acceleration rate of the object vehicle is then predicted according to the status of both the object vehicle and its leading vehicle. The following features are defined for predicting the acceleration rate for the object vehicle:

- Lateral position of the object vehicle x which is the lateral position of the vehicle based on the leftmost edge of the road
- Longitudinal position of the object vehicle y
- Speed of the object vehicle v
- Space headway between object vehicle and its leading vehicle sp
- Lateral position of the leading vehicle x_l
- Longitudinal position of the leading vehicle y_l
- Speed of the leading vehicle v_l
- Acceleration rate of the leading vehicle a_l

3.4. Summary

To better investigate vehicle trajectory with the machine learning method, NGSIM dataset is used to provide historical vehicle trajectory data. A freeway segment of US 101 is selected in Los Angeles, California. The study area is approximately 640 meters (2,100 feet) in length and consists of five mainline lanes throughout the section. An auxiliary lane is present through a portion of the corridor between the on-ramp and the off-ramp. The basic information on the selected freeway segment is discussed. Traffic volume of the study period and extracted features are shown. This is a basic preparation for predicting vehicle trajectory with CAV technologies in the future tasks.

Chapter 4. Vehicle Trajectory Prediction Methods

4.1. Introduction

CAVs can adjust their maneuvers based on the surrounding vehicles. According to the current condition of its leading vehicle, CAV should be able to predict its future trajectory in advance. To quickly and accurately decide its next step acceleration rate is essential for CAVs to avoid an accident. As a newly developed technology, machine learning methods are proved to have advanced calculation ability in vehicle trajectory prediction area. This chapter will quantify the prediction accuracy of the proposed machine learning method and compare the results to those of a traditional state-of-the-art method.

This chapter is organized as follows. Section 4.2 presents the XGBoost model for vehicle trajectory prediction. Section 4.3 describes the Intelligent Driver Model. Section 4.4 presents the comparison methods for the proposed models. Finally, in section 4.5, a summary concludes this chapter.

4.2. XGBoost Model

XGBoost is a prevalent boosting tree algorithm employed in industry because of its accuracy and high efficiency in prediction. In fact, XGBoost is developed from gradient boosting decision tree (GBDT) algorithm and employed in classification and regression problems with multiple decision trees (Xu et al., 2019). XGBoost can prevent over-fitting by normalizing the objective function. The details of the model are illustrated as follows.

A dataset is assumed as $D = \{(x_i, y_i)\}(i = 1,2, \dots, n)$, and the model has k trees. The model result $\widehat{y_i}$ is expressed as:

$$\widehat{y_i} = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{4.1}$$

where F is the hypothesis space, and $f(x)$ denotes a decision tree:

$$F = \{f(x) = \omega_{q(x)}\} \tag{4.2}$$

where $\omega_{q(x)}$ represents the score of each leaf node; $q(x)$ is the number of leafs.

When a new tree is developed to fit the residual errors of last tree, the predicted score for the t-th tree can be calculated as:

$$\widehat{y_i}^t = \widehat{y_i}^{t-1} + f_t(x) \tag{4.3}$$

The objective function is as follows:

$$J^{(t)} = \sum_{i=1}^{n} L(y_i, \widehat{y_i}) + \Omega(f_t) \tag{4.4}$$

where L is the loss function, Ω is a penalizing term, and:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} {\omega_j}^2 \tag{4.5}$$

where γ is a parameter represents the complexity of the leaf; Tdenotes the number of the leaves; λ is a parameter scaling the penalty; and $\omega$is the vector of scores on each leaf.

Unlike the general gradient boosting methods, the XGBoost employs the second-order Taylor expansion to the loss function. Formula (4.4) is then simplified as follows:

$$J^{(t)} = \sum_{i=1}^{n} \left[L(y_i, {\widehat{y_i}}^{t-1}) + g_i f_t(x_i) + \frac{1}{2} h_i {f_t}^2(x_i) \right] + \Omega(f_t) \tag{4.6}$$

$$g_i = \frac{\partial L(y_i, {\widehat{y_i}}^{t-1})}{\partial {\widehat{y_i}}^{t-1}} \tag{4.7}$$

$$h_i = \frac{\partial^2 L(y_i, {\widehat{y_i}}^{t-1})}{\partial {\widehat{y_i}}^{t-1}} \tag{4.8}$$

Then, the final objective function can be generated as follows:

$$\begin{aligned} J^{(t)} &= \sum_{i=1}^{n} \left[g_i \omega_{q(x_i)} + \frac{1}{2} h_i \omega_{q(x_i)}^2 \right] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} {\omega_j}^2 \\ &= \sum_{j=1}^{T} \left[\left(\sum_{i \in I_j} g_i \right) \omega_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) \omega_j^2 \right] + \gamma T \end{aligned} \tag{4.9}$$

where $I_j = \{i | q(x_i) = j\}$ is the set of data point indices belonged to the j-th leaf. Since the same score is assigned to all the data points on the same leaf, the index of the summation in the second line can be revised. The terms $g_i$and $h_i$denote the first and second derivativesof the loss function. Let $G_j = \sum_{i \in I_j} g_i$ and $H_j = \sum_{i \in I_j} h_i$, then the final objective function is changed to a quadratic function as follows:

$$J^{(t)} = \sum_{j=1}^{T} \left[G_j \omega_j + \frac{1}{2} \left(H_j + \lambda \right) \omega_j^2 \right] + \gamma T \tag{4.10}$$

Finally, the optimal solution of the optimized objective function can be generated:

$$\omega_j^* = -\frac{G_j}{H_j + \lambda} \tag{4.11}$$

$$J^* = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T \tag{4.12}$$

4.3. Intelligent Driver Model

The Intelligent Driver Model (IDM) produces better realism than most of the deterministic car following models (Treiber et al. 2000). The fundamental of the IDM is to calculate the acceleration rate of the object vehicle by considering both the ratio of desired velocity versus actual velocity and the ratio of desired headway versus actual headway. The calculation of acceleration rate is expressed as follows:

$$a = a_m [1 - (\frac{v}{v_0})^{\delta} - (\frac{s^*(v, \Delta v)}{s})^2] \tag{4.13}$$

$$s^*(v, \Delta v) = s_0 + s_1 \sqrt{\frac{v}{v_0}} + vT + \frac{v \times \Delta v}{2\sqrt{a_m b}} \tag{4.14}$$

where

a = acceleration rate of the object vehicle;

a_m = maximum acceleration;

v = current velocity of the object vehicle;

v_0 = desired velocity;

δ = acceleration exponent;

$s^*(v, \Delta v)$ = desired minimum headway;

Δv = speed difference between the object vehicle and the leading vehicle;

s = current headway between the object vehicle and the leading vehicle;

s_0 = linear jam distance;

s_1 = non-linear jam distance;

T = desired headway;

b = comfortable deceleration.

Table 4.1 presents the values of all the parameters in the proposed IDM in this study.

Table 4.1 Values of Parameters in the IDM

Parameters	**Values**	**Parameters**	**Values**
a_m	0.73 m/s^2	s_1	3 m
v_0	29 m/s	T	0.6 s
δ	4	b	1.67 m/s^2
s_0	2 m		

4.4. Model Comparison

The Root mean square error (RMSE) and Mean absolute error (MAE) are employed to examine the performance of the proposed models.

RMSE calculates the average of square errors between predicted values and actual values:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{N}(y_i^* - y_i)^2} \tag{4.15}$$

Mean absolute error (MAE) is calculated by averaging the absolute errors between predicted values and actual values:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i^* - y_i| \tag{4.16}$$

where N is the number of data points; y_i^* and y_i represent the predicted and actual values.

4.5. Summary

This chapter presents the machine learning method for vehicle trajectory prediction under CAV environment. The object vehicle is able to get the real time condition of itself and its leading vehicle, such as speed, position, and acceleration rate. According to this information, CAV could predict its acceleration rate for the next time step. Features that are necessary for the prediction are summarized. The prediction errors are quantified and compared with that of an existing car-following model, the Intelligent Driver Model, by using the RMSE and MAE.

Chapter 5. Numerical Results

5.1. Introduction

This chapter presents the numerical results of the vehicle trajectory prediction. The collected trajectory data is divided into two subsets, the training set and the testing set. The training set is used to train the XGBoost model. And the testing set is used to calculate the prediction error of the proposed model. The testing set is also used to calculate the acceleration rate by the IDM. The prediction errors are compared between the XGBoost model and the IDM. The chapter is organized as follows. Section 5.2 describes the prediction errors of the proposed models. Section 5.3 presents the feature importance ranking for vehicle trajectory prediction. Finally, in section 5.4, a summary concludes this chapter.

5.2. Performance of the Models

In this study, the RMSE and MAE are employed to evaluate the prediction accuracy of the XGBoost model and the IDM. Table 5.1 shows the RMSE and MAE values for the proposed models. As one can see from the table, the RMSE and MAE of the XGBoost model are 3.9953 and 2.6950, respectively, which are smaller than the errors of the IDM (i.e., 6.2748 and 4.7164). This illustrates the superiority of the XGBoost model in the prediction of vehicle trajectory.

Table 5.1 Comparison of the Two Models in Acceleration Rate Prediction

Algorithm	**RMSE**	**MAE**
XGBoost	3.9953	2.6950
IDM	6.2748	4.7164

Figure 5.1 shows the predicted and observed values in a predict horizon of 30 seconds. As can be seen in the figure, the XGBoost model can effectively predict the acceleration rate of the object vehicle. The prediction results of the IDM are inferior to those of the XGBoost model. By comparing the prediction results, one can conclude that the XGBoost model is more reliable for vehicle trajectory prediction than the IDM.

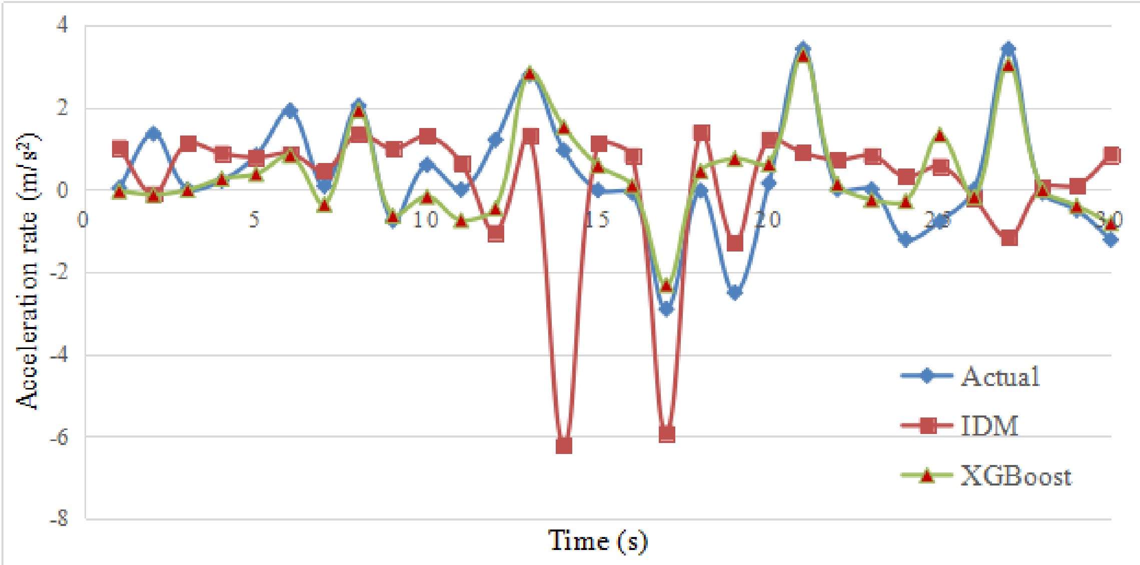

Figure 5.1 Comparison of the Predicted Results and the Actual Data

5.3. Feature Importance

To further explore the impact of each feature on the vehicle trajectory prediction, the relative importance of the eight input features in the XGBoost model is calculated. The feature importance is ranked based on the F score, which is a measurement of the frequency that a variable is selected for splitting. The feature will get higher score if it is used to make decisions in the decision trees more frequently. The importance ranking of the input features are displayed in Figure 5.2. It can be seen from the figure, the longitudinal position, lateral position, and the velocity of the object vehicle are the most important features to predict the vehicle trajectory.

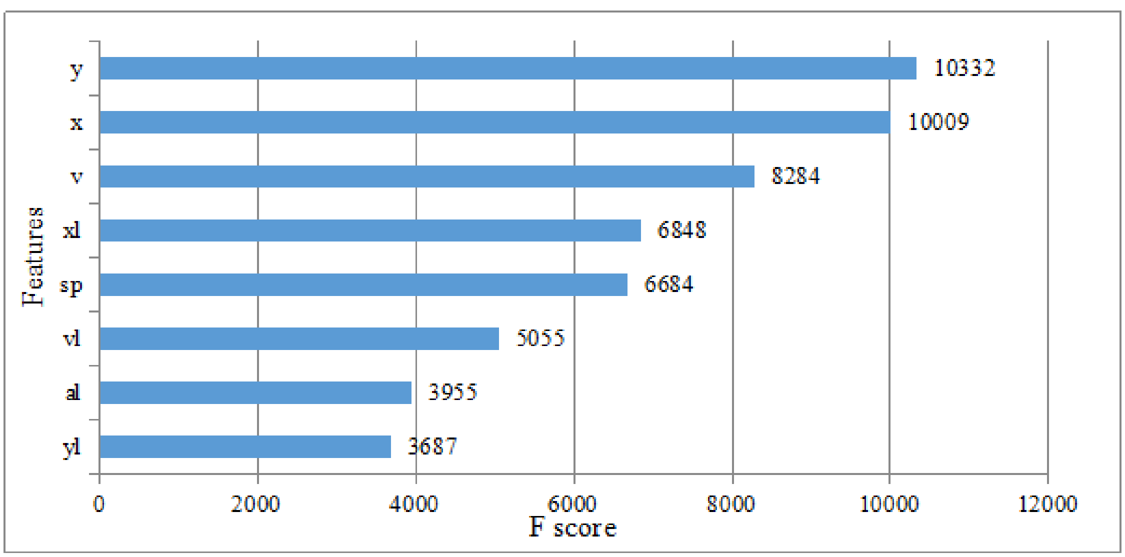

Figure 5.2 Feature Importance Ranking

5.4. Summary

This chapter focuses on describing the vehicle trajectory prediction results using XGBoost model. The predicted results are compared with the IDM, which is a traditional car following model. The NGSIM dataset is utilized to train and test the proposed XGBoost model. The predicted results show that the XGBoost model gets higher prediction accuracy than the IDM model. The longitudinal position of the object vehicle is the most important feature to predict the vehicle trajectory.

Chapter 6. Summary and Conclusions

6.1. Introduction

This chapter will summarize the results and illustrate the limitation of this study. The following sections are organized as follows. In section 6.2, the background and main results of this study are reviewed and a summary of conclusions for the vehicle trajectory prediction using machine learning method is discussed. Section 6.3 presents a brief discussion of the limitations of the current approaches and possible directions for further research are also given.

6.2. Summary and Conclusions

Connected and autonomous vehicle (CAV) technologies provide solutions to the existing problems of the transportation systems. As widely known, CAVs can communicate with each other so that they can have coordinated accelerating or decelerating movements. In this way, CAVs only need a smaller headway which will lead to a higher roadway capacity. For signalized intersections, CAVs can communicate with the signal lights to adjust their speeds when approaching the intersection, so that they can arrive at the intersection during green light. CAVs bring with them many benefits including improving safety, reducing emissions and increasing mobility of the transportation system.

In past decades, numerous research efforts have focused on modeling longitudinal driver behaviors of traditional vehicles. Most microscopic models assumed that human drivers react to the stimuli from leading vehicles to keep a safe headway with a desired velocity. In recent years, with the emerging of CAVs, new car following models have been developed to accommodate the longitudinal driving behavior of CAVs. Efforts are needed to calibrate these car following models, and the results are highly related to the data availability, calibration method, and model structure. Despite different mechanisms and software interfaces, when multiple simulation software applications are compared, it seems that error cannot be eliminated no matter how many parameters are introduced. Machine learning has achieved much success in recent years. It allows the agent to keep learning from observations, actions conducted, and rewards received. When presented with a sequence of states and corresponding actions, extracted from the trajectory data, the algorithm can learn how the vehicles act when faced with varying traffic conditions. The algorithm learns by associating any state observation, such as reaction time, speed, headway, and acceleration rate. The degree to which the agent action matches the vehicle’s action constitutes a reward in the learning sequence. In order to better predict the upcoming states of CAVs under varying traffic conditions, there is a critical need to model the car following trajectory data using the machine learning approach.

This research compared the prediction accuracy of machine learning method with that of the existing car following model using historical trajectory data, and therefore led to a better understanding of how CAVs operate in the freeway system.

To better predict the vehicle trajectories, the XGBoost model was developed to predict vehicle trajectories in CAV environment. The predicted results were compared with the IDM, which is a traditional car following model. The NGSIM dataset was used to train and test the XGBoost model. The predicted results proved that the XGBoost model gets higher prediction accuracy than the IDM model. The longitudinal position of the object vehicle was the most important feature to predict the vehicle trajectory. The results of this study could help guide the machine learning approaches in the area of vehicle trajectory prediction.

6.3. Directions for Future Research

In this section, some of the limitations of the study are presented and directions for further research are also discussed.

The case study in this study only focused on simple freeway segment. Future studies will focus on more complicated scenarios, such as freeways with multiple ramps and weaving sections, and arterial road with multiple intersections. Mixed traffic environment will be considered including both trucks and passenger cars. Developing an advanced car following model under lane change situations is another research direction to go. Future research efforts will also investigate other machine learning models to predict vehicle trajectory considering lane changing in different roadway scenarios.

References

1. Abbas, M. M., and Chong, L. (2013). *Car-Following Trajectory Modeling with Machine Learning: Showcase for Merits of Artificial Intelligence* (No. 13-4712).
2. Abbas, M. T., Jibran, M. A., Afaq, M., & Song, W. C. (2019). An adaptive approach to vehicle trajectory prediction using multimodel Kalman filter. *Transactions on Emerging Telecommunications Technologies*, e3734.
3. Ahn, K., Rakha, H. A., and Park, S. (2013). Ecodrive Application: Algorithmic Development and Preliminary Testing. *Transportation Research Record*, 2341(1), 1–11.
4. Altché, F., & de La Fortelle, A. (2017, October). An LSTM network for highway trajectory prediction. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 353-359). IEEE.
5. Boubezoul, A., Koita, A., & Daucher, D. (2009, July). Vehicle trajectories classification using support vectors machines for failure trajectory prediction. In *2009 International Conference on Advances in Computational Tools for Engineering Applications* (pp. 486-491). IEEE.
6. Campbell, R. and Alexiadis, V. (2016). *Connected Vehicle Impacts on Transportation Planning: Analysis of the Need for New and Enhanced Analysis Tools, Techniques, and Data - Highway Capacity Manual Briefing* (No. FHWA-JPO-16-365).
7. Cheng, H., & Sester, M. (2018, June). Mixed traffic trajectory prediction using LSTM–based models in shared space. In *The Annual International Conference on Geographic Information Science* (pp. 309-325). Springer, Cham.
8. Choi, S., Kim, J., Yu, H., & Yeo, H. (2019, October). Real-time Prediction of Arterial Vehicle Trajectories: An Application to Predictive Route Guidance for an Emergency Vehicle. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)* (pp. 4030-4036). IEEE.
9. Choi, S., Yeo, H., & Kim, J. (2018). Network-wide vehicle trajectory prediction in urban traffic networks using deep learning. *Transportation Research Record*, *2672*(45), 173-184.
10. Dai, S., Li, L., & Li, Z. (2019). Modeling vehicle interactions via modified LSTM models for trajectory prediction. *IEEE Access*, *7*, 38287-38296.
11. Deo, N., & Trivedi, M. M. (2018, June). Multi-modal trajectory prediction of surrounding vehicles with maneuver based lstms. In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1179-1184). IEEE.
12. Ding, C., Wang, W., Wang, X., & Baumann, M. (2013). A neural network model for driver's lane-changing trajectory prediction in urban traffic flow. *Mathematical Problems in Engineering*, *2013*.
13. Goli, S. A., Far, B. H., & Fapojuwo, A. O. (2018, June). Vehicle Trajectory Prediction with Gaussian Process Regression in Connected Vehicle Environment $\star$. In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 550-555). IEEE.
14. Guo, Y., Ma, J., Xiong, C., Li, X., Zhou, F., and Hao, W. (2019). Joint optimization of vehicle trajectories and intersection controllers with connected automated vehicles: Combined dynamic programming and shooting heuristic approach. *Transportation research part C: emerging technologies*, *98*, 54-72.
15. Han, F., Tan, Y., & Eledath, J. (2007). Preceding Vehicle Trajectory Prediction by Multi-Cue Integration. In *MVA* (pp. 575-578).

16. Hu, X., and Sun, J. (2019). Trajectory optimization of connected and autonomous vehicles at a multilane freeway merging area. *Transportation Research Part C: Emerging Technologies, 101*, 111-125.
17. Izquierdo, R., Parra, I., Muñoz-Bulnes, J., Fernández-Llorca, D., & Sotelo, M. A. (2017, October). Vehicle trajectory and lane change prediction using ann and svm classifiers. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1-6). IEEE.
18. Jeong, D., Baek, M., & Lee, S. S. (2017, October). Long-term prediction of vehicle trajectory based on a deep neural network. In *2017 International Conference on Information and Communication Technology Convergence (ICTC)* (pp. 725-727). IEEE.
19. Jiang, H., Chang, L., Li, Q., & Chen, D. (2019, September). Trajectory Prediction of Vehicles Based on Deep Learning. In *2019 4th International Conference on Intelligent Transportation Engineering (ICITE)* (pp. 190-195). IEEE.
20. Ju, C., Wang, Z., Long, C., Zhang, X., Cong, G., & Chang, D. E. (2019). Interaction-aware kalman neural networks for trajectory prediction. *arXiv preprint arXiv:1902.10928*.
21. Kim, B., Kang, C. M., Kim, J., Lee, S. H., Chung, C. C., & Choi, J. W. (2017, October). Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 399-404). IEEE.
22. Li, J., Ma, H., Zhan, W., & Tomizuka, M. (2019, June). Coordination and trajectory prediction for vehicle interactions via Bayesian generative modeling. In *2019 IEEE Intelligent Vehicles Symposium (IV)* (pp. 2496-2503). IEEE.
23. Li, X., Ghiasi, A., Xu, Z., and Qu, X. (2018). A piecewise trajectory optimization model for connected automated vehicles: Exact optimization algorithm and queue propagation analysis. *Transportation Research Part B: Methodological*, *118*, 429-456.
24. Liu, P., & Kurt, A. (2014, October). Trajectory prediction of a lane changing vehicle based on driver behavior estimation and classification. In *17th international IEEE conference on intelligent transportation systems (ITSC)* (pp. 942-947). IEEE.
25. Messaoud, K., Yahiaoui, I., Verroust-Blondet, A., & Nashashibi, F. (2019b, October). Relational recurrent neural networks for vehicle trajectory prediction. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)* (pp. 1813-1818). IEEE.
26. Messaoud, K., Yahiaoui, I., Verroust-Blondet, A., & Nashashibi, F. (2019a, June). Non-local Social Pooling for Vehicle Trajectory Prediction. In *2019 IEEE Intelligent Vehicles Symposium (IV)* (pp. 975-980). IEEE.
27. Nikhil, N., & Tran Morris, B. (2018). Convolutional neural network for trajectory prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 0-0).
28. Ntousakis, I. A., Nikolos, I. K., and Papageorgiou, M. (2016). Optimal vehicle trajectory planning in the context of cooperative merging on highways. *Transportation research part C: emerging technologies*, *71*, 464-488.
29. Park, S. H., Kim, B., Kang, C. M., Chung, C. C., & Choi, J. W. (2018, June). Sequence-to-sequence prediction of vehicle trajectory via LSTM encoder-decoder architecture. In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1672-1678). IEEE.
30. Pecher, P., Hunter, M., & Fujimoto, R. (2016, May). Data-driven vehicle trajectory prediction. In *Proceedings of the 2016 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation* (pp. 13-22).

31. Qiao, S., Shen, D., Wang, X., Han, N., & Zhu, W. (2014). A self-adaptive parameter selection trajectory prediction approach via hidden Markov models. *IEEE Transactions on Intelligent Transportation Systems*, *16*(1), 284-296.
32. Receveur, J. B., Victor, S., and Melchior, P. (2017). Multi-criteria trajectory optimization for autonomous vehicles. *IFAC-PapersOnLine*, *50*(1), 12520-12525.
33. Schreier, M., Willert, V., & Adamy, J. (2014, October). Bayesian, maneuver-based, long-term trajectory prediction and criticality assessment for driver assistance systems. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (pp. 334-341). IEEE.
34. Schreier, M., Willert, V., & Adamy, J. (2016). An integrated approach to maneuver-based trajectory prediction and criticality assessment in arbitrary road environments. *IEEE Transactions on Intelligent Transportation Systems*, *17*(10), 2751-2766.
35. Shi, L. and Prevedouro, P.D. (2016). Autonomous and Connected Cars: HCM Estimates for Freeways with Various Market Penetration Rates, *Transportation Research Procedia*, 15, 2016, pp. 389-402.
36. Tomar, R. S., Verma, S., & Tomar, G. S. (2010, November). Prediction of lane change trajectories through neural network. In *2010 International Conference on Computational Intelligence and Communication Networks* (pp. 249-253). IEEE.
37. Tran, Q., & Firl, J. (2014, June). Online maneuver recognition and multimodal trajectory prediction for intersection assistance using non-parametric regression. In *2014 IEEE Intelligent Vehicles Symposium Proceedings* (pp. 918-923). IEEE.
38. Treiber, M., Hennecke, A., & Helbing, D. (2000). Congested traffic states in empirical observations and microscopic simulations. *Physical review E*, *62*(2), 1805.
39. Tsang, S. H., Hoare, E. G., Hall, P. S., & Clarke, N. J. (1999, October). Automotive radar image processing to predict vehicle trajectory. In *Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348)* (Vol. 3, pp. 867-870). IEEE.
40. Walker, R. (2019, October). Predicting Time-Series Trajectories of Human Driven Vehicles, using an Unsupervised Attention-Based Recurrent Neural Network. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)* (pp. 2278-2285). IEEE.
41. Wang, C., Qin, J., Guo, M., & Xu, Y. (2013). Prediction of vehicle trajectory based on Fuzzy colored petri net. *TELKOMNIKA Indonesian Journal of Electrical Engineering*, *11*(10), 5824-5832.
42. Wang, L. L., Chen, Z. G., & Wu, J. (2019). Vehicle trajectory prediction algorithm in vehicular network. *Wireless Networks*, *25*(4), 2143-2156.
43. Wang, M., Daamen, W., Hoogendoorn, S. P., and van Arem, B. (2014). Rolling horizon control framework for driver assistance systems. Part I: Mathematical formulation and non-cooperative systems. *Transportation research part C: emerging technologies*, *40*, 271-289.
44. Wang, M., Daamen, W., Hoogendoorn, S. P., and van Arem, B. (2016). Connected variable speed limits control and car-following control with vehicle-infrastructure communication to resolve stop-and-go waves. *Journal of Intelligent Transportation Systems*, *20*(6), 559-572.
45. Wissing, C., Nattermann, T., Glander, K. H., & Bertram, T. (2018, November). Trajectory prediction for safety critical maneuvers in automated highway driving. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)* (pp. 131-136). IEEE.
46. Woo, H., Sugimoto, M., Wu, J., Tamura, Y., Yamashita, A., & Asama, H. (2018). Trajectory prediction of surrounding vehicles using LSTM network. In *2013 IEEE Intelligent Vehicles Symposium (IV)*.

47. Xiao, Z., Li, P., Havyarimana, V., Hassana, G. M., Wang, D., & Li, K. (2018). GOI: A novel design for vehicle positioning and trajectory prediction under urban environments. *IEEE Sensors Journal*, *18*(13), 5586-5594.
48. Xing, Y., Lv, C., & Cao, D. (2019). Personalized Vehicle Trajectory Prediction Based on Joint Time Series Modeling for Connected Vehicles. *IEEE Transactions on Vehicular Technology*.
49. Xu, Y., Zhao, X., Chen, Y., & Yang, Z. (2019). Research on a Mixed Gas Classification Algorithm Based on Extreme Random Tree. *Applied Sciences*, *9*(9), 1728.
50. Yang, H., & Jin, W. L. (2014). A control theoretic formulation of green driving strategies based on inter-vehicle communications. *Transportation Research Part C: Emerging Technologies*, 41, 48-60.
51. Yao, W., Zhao, H., Bonnifait, P., & Zha, H. (2013, June). Lane change trajectory prediction by using recorded human driving data. In *2013 IEEE Intelligent Vehicles Symposium (IV)* (pp. 430-436). IEEE.
52. Ye, N., Zhang, Y., Wang, R., & Malekian, R. (2016). Vehicle trajectory prediction based on Hidden Markov Model.
53. Yi, S. G., Kang, C. M., Lee, S. H., & Chung, C. C. (2015, June). Vehicle trajectory prediction for adaptive cruise control. In *2015 IEEE Intelligent Vehicles Symposium (IV)* (pp. 59-64). IEEE.
54. Yoon, S., & Kum, D. (2016, June). The multilayer perceptron approach to lateral motion prediction of surrounding vehicles for autonomous vehicles. In *2016 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1307-1312). IEEE.
55. Zhao, T., Xu, Y., Monfort, M., Choi, W., Baker, C., Zhao, Y., ... & Wu, Y. N. (2019). Multi-agent tensor fusion for contextual trajectory prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 12126-12134).