An Interaction-Aware Approach for Online Cut-in Behavior Prediction and Risk Assessment for Autonomous Driving

by

Jinwei Zhang

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Mechanical and Mechatronics Engineering

Waterloo, Ontario, Canada, 2020

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

The development of autonomous driving has become one of the biggest trends of the 21st century's technology. However, the promotion and the mass production of autonomous vehicles are still at the beginning stage. The human-driven vehicles will still predominate the traffic. Therefore, understanding the interaction and decision logic between human-driven vehicles, and utilizing it to predict their driving behavior are the keys to the development of autonomous driving techniques. Cut-in behavior is one of the top priorities due to its high risks. Rear-end collisions happen a lot when the lag vehicles cannot predict this abnormal lane change behavior of the front vehicles and response in time. However, related studies on cut-in event prediction and risk assessment have rarely been presented in autonomous driving field. A phase-based design framework is proposed in this work to realize online prediction and risk estimation of the cut-in behavior considering interactions between the involved vehicles. After preprocessing and analyzing of a naturalistic driving dataset, a cut-in behavior predictor and a risk estimator are devised based on Gaussian mixture model and Gaussian mixture regression method. Compared with baseline approaches, both the predictor and estimator designed following the proposed framework achieve enhanced results, which can further improve the driving safety of autonomous vehicles when cut-in behavior occurs.

Acknowledgements

I would like to thank my supervisor Prof. Dongpu Cao for giving me the opportunity and financial support to pursue master's degree and to do this amazing research project in Waterloo Cognitive Autonomous Driving (CogDrive) Lab. Thank you, Prof. Cao, for dedicating yourself and guiding me, advising me, and encouraging me to fulfill all of this.

I also like to thank Prof. Jan P. Huissoon, for being as my co-supervisor. Thank you, Prof. Huissoon, for giving me some valuable advice on each of my milestone reports.

I thank all my fellow CogDrive labmates, the current members and those graduated, without whom this thesis would not have been possible. In particular, thank you, Dr. Huilong Yu and Dr. Guofa Li, for selflessly offering me the insights for my research project and advising me on my academic writing. In addition to being friends of mine, you have been acting as my good examples of excellent researchers that I have been learning a lot from you. I also thank you, Zejian Deng, Wenbo Li, Chen Sun, and Xingxin Chen, for also being my roommates and giving me mental supports in the lockdown time due to the COVID-19 pandemic.

Thank you, all the people whom I worked with in University Choir and Chamber Choir, for taking me a lot of fun in rehearsal after school. A special thank you, University Choir Ensemble Director, Mrs. Liska Jetchick, and Chamber Choir Ensemble Director, Mr. Bob Anderson, for directing us to make many pieces of beautiful songs.

Finally, thank you, my family, for your unconditional love, encouragement, and support to me throughout my life. I can hardly fulfill my master's study and research project without you.

Dedication

To My Parents

With Love

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ACC adaptive cruise control 10

AD autonomous driving 1

AIC Akaike Information Criterion 41

AV autonomous vehicle 1, 19, 55, 60

CA Constant Acceleration 9, 56

CNN Convolutional Neural Network 11

CSAA Constant Steering Angle and Acceleration 9

CSAV Constant Steering Angle and Velocity 9

CTRA Constant Turning Rate and Acceleration 9

CTRV Constant Turning Rate and Velocity 9

CV Constant Velocity 9

DBN Dynamic Bayesian Network 10

DMV Department of Motor Vehicles 30

DNN Deep Neural Network 11

DT Decision Tree 8

EC European Commission 5

EM Expectation-Maximization 38, 53

ENESEMBLE ENabling SafE Multi-Brand pLatooning for Europe 5

EV ego vehicle 5, 19, 47

GMM Gaussian Mixture Model 11, 23, 32, 33, 53

HMM Hidden Markov Model 10

IDM Intelligent Driver Model 12

kNN K-nearest neighbor 12

LCV lane change vehicle 5, 19, 25, 32, 47

LSTM Long Short-term Memory 11

LV lead vehicle 14, 19, 28, 32

MAE mean absolute error 58

MLP Multilayer Perceptron 11

NGSIM Next Generation Simulation 26

NHTSA National Highway Traffic Safety Administration 1, 55

NN Neural Network 11

RF Random Forest 12

RL Reinforcement learning 15

RMSE root mean square error 40

RNN Recurrent Neural Network 11

RRT Rapidly-exploring Random Tree 11

RSA Road Safety Authority 30

RV rear vehicle 28

SAE Society of Automotive Engineers 21, 28

SVM Support Vector Machine 8

THW time headway 7, 49

TTC time-to-collision 7

TTR Time-to-reaction 14

V2V Vehicle-to-vehicle 21

Chapter 1

Introduction

Benefited from the fast advances in AI and high-performance computing, autonomous driving (AD) technique has become one of the hottest topics in recent years because it shows a great potential to further improve the driving safety and commuting efficiency. According to the statistics from US National Highway Traffic Safety Administration (NHTSA), there are 94% of the traffic accidents caused by human mistaken operation [1], and the development of the AD techniques can prevent the human factor related fatalities. It also implies that the techniques could also prevent up to 1,922 traffic fatalities happened in Canada in 2018 [2]. Moreover, the AD techniques are expected to reduce commuting time, traffic congestion and fuel consumption, that it could benefit up to 150 billion US Dollar in the US and 65 billion Canadian Dollar in Canada every year [3] [4]. Therefore, the AD is treated as one of the most influential technologies nowadays, and its development has been paid attention by the universities, leading technology companies, and the policymakers.

The development of AD is unprecedentedly fast, however, the promotion and the mass production of autonomous vehicle (AV) is still at the beginning stage. For a long period of time, human-driven vehicles will still be the majority of the traffic. Human drivers monitor potential risk, continuously make decisions, and adjust their driving behaviors which are often not rational. Since we lack the understanding of human cognition and interaction logic at this stage, it is difficult for autonomous driving to make safe and effective decisions in mixed traffic. Therefore, understanding the interaction and decision logic between human-driven vehicles, and utilizing it to predict their driving maneuvers are the key to the development of AD technique.

Understanding and predicting the aggressive lane change have become one of the top priorities of AD. Lane change maneuvers frequently occur in daily driving. An aggressive

lane change behavior, called cut-in, usually triggers potential threats to the following vehicle. According to statistics, cut-in related rear-end collisions account for 5% of the total traffic accidents [5]. Experienced human drivers are able to handle the cut-in behavior of the front vehicle in most cases by foreseeing its occurrence and evaluating the potential risk level in advance, which can save more time to slows down early and keep a safe distance accordingly. Therefore, the AV should also possess the ability of continuously predicting the cut-in behaviors of nearby vehicles and estimating the risk level by considering their mutual influences. In this case, the AV can save more time to react the upcoming risky situations, and thus to improve the safety.

Since 1930s, the cut-in behavior research has been extensively studied [6]. However, the previous works were based on the data of entire lane change process, which are impossible to be utilized for online prediction of the cut-in behavior. The topic of cut-in behavior prediction has only been investigated in the recent five years, alongside with the raise of AV. In the existing research of cut-in behavior prediction, only a few of them proposed the methods that consider the interaction. Hence, the research question that the thesis proposes is 'how does the AV predict the cut-in behavior and assess its risks online while considering the interaction?'

To answer the question, this thesis proposes a framework of the interaction-aware online cut-in event prediction and risk assessment. The methodology it proposes should 1) consider the interactions between ego vehicle and the cut-in vehicle, and 2) perform online application during the lane change process to improve the prediction and estimation accuracy. In details, the objectives of the research can be listed as followings:

1. To achieve predicting the cut-in events with estimated risks for autonomous vehicles;
2. To implement online cut-in event prediction and risk assessment works, which means that the cut-in event with certain risks can be predicted and estimated in real time;
3. To increase the accuracy of online prediction and online risk estimation by considering the interaction between nearby vehicles.

The remainder of the paper is organized in the following structure. Chapter 2 introduces the related works about cut-in behavior research, cut-in event prediction and risk assessment. Chapter 3 formulates the cut-in scenario and provides the framework of the proposed design methodology. Chapter 4 introduces the data preprocessing procedures for lane change and cut-in event extraction, and Chapter 5 details interaction-aware motion prediction method of lane change vehicle, which is to support the later cut-in event prediction and risk assessment in the next chapter. Section 6 illustrates the design of cut-in

event prediction and risk assessment algorithm with the performance displayed. Concluding remarks and future works are presented in Section 7.

Chapter 2

Literature Review

Overall, this thesis contributes the research fields of cut-in event prediction and risk assessment. Specifically, it attempts to find a way to apply prediction and assessment with considering the inter-vehicle interaction; and, it aims to achieve the online application of the system. In this chapter, we will outline the representative and state-of-the-art works in this field. Starting at introducing the existing definitions of cut-in behavior, this chapter represents the cut-in behavior analysis, which is to investigate the factors that are related to the defined cut-in events. The next section shows a brief review of motion prediction and risk assessment methods, and the topic is narrowed to the cut-in event prediction and risk assessment, according to the results of the cut-in related research. Finally, this chapter ends up with discussing the merits and the limitations of the related works in the final section, which gives ideas to conduct our proposed methods in the later chapters.

2.1 Lane change and cut-in behavior

2.1.1 Concept of lane change behavior

According to Operational Definitions of Driving Performance Measures and Statistics by SAE International [7], the lane change can be defined as 'Lateral movement of a vehicle from (1) a merge lane into a lane of a traveled way, (2) one lane of a traveled way to another lane on the same traveled way with continuing travel in the same direction in the new lane, or (3) a lane on a traveled way to an exit lane departing that traveled way.' The lane change behavior can be classified as discretionary lane change and mandatory lane

change, where the mandatory lane change represents that the driver must leave the lane, such as the lane is merging lane. Additionally, the lane change can be also classified as intentional lane change and unintentional lane change. The unintentional one is also called lane departure, which could be caused by the factor that is not consistent with the driver's will, such as windblown or driving fatigue, etc.

2.1.2 Concept of cut-in behavior

Cut-in behavior is one category of lane change behavior. The general understanding of cut-in behavior is an abnormal lane change action that causes rear-end collisions or triggers significant threats for the vehicle behind. To unify the terminology, the vehicle that applies lane change behavior is called lane change vehicle (LCV), and the subject vehicle that is influenced by the cut-in action is called ego vehicle (EV). There are multiple factors that trigger the risk of EV in the cut-in event, for example, the gap distance between LCV and EV is too close, or LCV applied sudden deceleration, etc. Among these factors, small gap distance is widely recognized as the factor to triggers the risk, so the gap distance is utilized to represent the risk of EV and accordingly to define cut-in behavior. On the other hand, if the drivers of EV feel risky, they will apply emergency brake to keep safe, which is another way to quantitatively define cut-in behavior. In summary, this thesis summarizes the existing cut-in definitions as two categories: distance-based cut-in definitions and response-based cut-in definitions.

Distance-based cut-in definitions

The concept of distance-based cut-in definitions can be illustrated by Project ENabling SafE Multi-Brand pLatooning for Europe (ENESEMBLE) from European Commission (EC) [8], which is 'a lane change manoeuvre performed by vehicles from the adjacent lane to the ego vehicle's lane, at a distance close enough (i.e., shorter than desired inter vehicle distance) relative to the ego vehicle.' The desired inter vehicle distance is called 'gap acceptance'. To define the cut-in events, the following methods proposed the criteria to quantify the gap acceptance. For example, Wang *et al* [9] defined the cut-in accepted gap as 75 meters by manual observations from different testers. From the other articles, the accepted gap is less strict. Zhao *et al* [5], Chen *et al* [10], Aramrattana *et al* [11], Milanes *et al* [12], and Remmen *et al* [13] regarded the cut-in behavior occurred as the accepted gap between the LCV and EV where it is inside of the observation range.

Response-based cut-in definitions

The concept of the response-based cut-in is to regard the deceleration or braking response of EV as the factor of safety threats levels brought by cut-in behavior. This type of defining criteria are more popular than the distance-based ones, and the specific definitions are illustrated below. Ma *et al* [14] was given a criterion of cut-in, that the cut-in occurs as long as the EV applies braking behavior, or the acceleration value is lower than 0 m/s^2. Xie *et al* [15] first extracted the cut-in events from the lane change dataset using the same definition above and utilized K-means method to cluster the extracted cut-in events into three groups according to the average deceleration and maximum deceleration. Each cluster represents the cut-in event with different risk levels.

Kim et al [16] defined the occurrence of cut-in events if the minimum acceleration of EV reaches -0.3 g. This quantitative criterion was conducted from the 31 test drivers' operations on the driving simulator, that the boundary between normal and emergency brake is -0.3 g, where the g stands for the gravitational acceleration of 9.8 m/s^2. Based on the results, Zhou *et al* [17] stated that the cut-in event occurs as the EV possesses braking action for at least 2 seconds, with the minimum acceleration reaching 0.3 g. Feng *et al* [18] extracted the extreme cut-in cases from the naturalistic driving data, with minimum accelerations of EV reach above -0.4 g in all cases.

Other definitions

Besides the mainstream two types of classifications, there is one special method defining the cut-in event that is solely based on the LCV. Wu *et al* [19] utilized the rule-based criteria to differentiate risky cut-in scenarios from lane change events based on the longitudinal acceleration, lateral acceleration, and yaw rate of LCV.

The results of the literature review show that there is no unified definition of cut-in behavior. From the two categories of definitions, each one has its pros and cons. Based on the distance-based definitions, the cut-in behaviors are easy to be measured and can be instantly detected for online application. However, the criteria of the definitions ignore the other factors that may trigger the risk level of EV. Conversely, the advantage of response-based cut-in definitions is that they can comprehensively concern multiple factors that impact the risk level of EV. But their drawback is that response latency exists between the actuation of cut-in behavior and driver's response of EV, meaning that the instant cut-in detection is not achievable. Hence, this type of definition is usually used for offline analysis.

2.2 Cut-in behavior research

Cut-in behavior research is one category of driver behavior research, which essentially focuses on investigating the related factors that cause certain driving maneuvers. The cut-in behavior research can be classified as two parts: The first part is to investigate the relationship between cut-in intention and gap acceptance [20] [9], and the second part is to investigate the factors that cause the cut-in event with different risk levels of EV. Since the second part is the field that the thesis concerns about, the following reviewed papers are all about investigating the factors that cause the cut-in event with certain risk levels.

Zhou *et al* [17] found out that when the EV applied the emergency brake and the acceleration was smaller than -0.3 g, the cut-in event occurs with the corresponding time headway (THW) and gap distance being less than 2 s and 60 m respectively. They also found out both THW and time-to-collision (TTC) can characterize the risk level of cut-in, but THW is more sensitive than TTC for human drivers.

Zhao *et al* [21] mainly investigated how do gap distance and velocity difference influence the driver's subjective feelings of the risk levels. In detail, several testers were invited to view different cut-in scenes and were asked to mark the five levels of risk scores for each one. Then the dynamic factors, namely velocity difference and gap distance of EV and LCV, were utilized to characterize the collected risk scores. The result of the experiment showed that the gap distance was better to characterize the risk levels than the velocity differences. It means that the gap distance is more sensitive to represent the driver's risk levels. In summary, this paper investigated two key cut-in related factors with comparison, but it only proved that they could characterize the cut-in event instead of the lane change scenario with no risk.

The two approaches above directly found the single factors to characterize the risk level of EV. However, the corresponding results were not ideal because the risk level of EV may be jointly related to multiple cut-in related factors. To prove the assumption, Ma *et al* [14] proposed an approach that first classified the road and traffic condition into four cases: 1) open road and smooth traffic, 2) open road and congested traffic, 3) closed road and smooth traffic, and 4) closed road and congested traffic. Then in each case, it proposed a set of rules based on THW and speed difference to characterize the braking behavior of EV. Compared with the previous approaches, it conducted better results.

Another approach is to utilize the machine learning concept to characterize the braking behavior with multiple cut-in features. Xie et al [15] first extracted six cut-in indicators: speed of EV, longitudinal gap distance, lateral gap distance, longitudinal speed difference, lateral speed difference, longitudinal acceleration difference. Then it proposed an ensemble

learning model consisting of a Decision Tree (DT) and a Support Vector Machine (SVM) to characterize three types of braking responses with different risk levels. In general, the machine learning methods could conduct good results of characterizing cut-in behavior, but this experiment did not check the characterization ability of each feature. In other words, it is unknown for the feature that is good or bad for characterizing the model.

Concerning that the cut-in event is a process and lasts several seconds, Feng *et al* [18] proposed an analytical approach to characterize the cut-in event by extracting the features at different time segments of the lane change. At the time of starting braking behavior, its risk level characterized by THW was significantly correlated with velocity difference, lane change direction of LCV, and vehicle type of LCV. After the braking was applied, its risk level was significantly correlated with longitudinal velocity of LCV. In conclusion, The risk level of EV was triggered by the different related factors in different time processes if the cut-in event occurs.

The approaches above have extensively analyzed the cut-in related factors with pros and cons commented. Based on their results, the velocity difference, longitudinal gap distance, and THW are proved to be the representative factors that relate to the cut-in behaviors and braking behavior of EV. Additionally, they are proved that the braking behavior of EV is correlated with multiple factors of cut-in behavior. However, most analyses ignored the fact, that lane change behavior was a continuous process, and the cut-in events could have occurred at any time stamp of the lane change scenario. Knowing the time frame of the cut-in event occurred is crucial for online prediction and risk assessment, we believe the research gap is urgently needed to be filled, that the lane change process is supposed to be segmented, and does further analysis on every one of them. Feng *et al* [18] and Kim *et al* [16] proposed some preliminary ideas of splitting the lane change process into two segments, but both splitting methods are too coarse to get the accurate time of cut-in occurred, and the finer splitting method is expected to be explored.

2.3 Prediction and risk assessment

Safety is crucial to AD. The concept of risk is the safety threat that the vehicle may encounter in the near future. Hence, in order to assess the risk associated with a specific situation, it is necessary to utilize specific methods to predict how this situation will evolve in the future.

The cut-in prediction is one category of motion prediction. Based on the existing methodologies and analytical results in the previous section, the cut-in prediction and risk

assessment can be applied by extracting the features that indicate the occurrence of the upcoming cut-in events with levels of risks. The features are uniformly called cut-in indicators, and the risk score is defined as the intensity of the braking behavior. In the later paragraphs, the general motion prediction and risk assessment methods are first reviewed, and previous applications about cut-in prediction and risk assessment are introduced subsequently.

2.3.1 Methods of motion prediction

In specific, the motion prediction is to conduct the future motion of vehicles based on the current or history situation. Lefevre *et al* [22] did a comprehensive review of the motion prediction and risk assessment methods published in 2014. Since then, there are lots of new contributions to motion prediction methods published. Hence, the following review content focuses on the newly published research contributions on the motion prediction field from 2014 to 2020. The motion prediction methods can be classified into three categories: 1) Physics-based motion prediction, 2) Maneuver-based motion prediction and 3) Interaction-aware motion prediction methods.

Physical-based motion prediction

The physical-based motion prediction is a fundamental prediction type that predicts the future trajectory according to the current state and its possible kinetic models of the target vehicle. There are several types of prediction methods applied to the kinematic model. For the point-mass kinematic model, Constant Velocity (CV), Constant Acceleration (CA), Constant Turning Rate and Velocity (CTRV), and Constant Turning Rate and Acceleration (CTRA) are mainly implemented. For the more complex kinematic models, such as three degrees of freedom (3-DOF) bicycle model, Constant Steering Angle and Velocity (CSAV) and Constant Steering Angle and Acceleration (CSAA) are derived.

Another alternative method so far is to use interpolation methods to predict the trajectory, which shows valid simulation with the coordination of the lane change driving maneuver. Kim *et al* [23] proposed to use CV methods fifth-order polynomial to interpolate the lane change trajectory after the lane change intention is already computed on the highway scenario. Additionally, Yoon *et al* [24] also proposed a step response of the third-order linear system as lateral trajectory simulation on the highway scenario.

The following methods were proposed to better describe the uncertainties of predicted trajectory. [25] mentioned to use Kalman Filter to simulate the prediction results with cor-

responding distributions. Alternatively, [26] proposed Monte Carlo Simulation to generate multiple future trajectories, and the belief of each trajectory was conducted in further.

Maneuver-based motion prediction

According to Lefevre *et al* [22], the definition of maneuver is 'a set of trajectories that shares the same features'. It can be defined by people. For instance, 'turn left', 'turn right', 'left-lane change', 'slow down' are some of the typical human-defined driving maneuvers. Conversely, the maneuvers can also be grouped and summarized by the clustering algorithm. A vivid comparison about the differences is Deo *et al* [27] and Kasper *et al* [28]. The maneuvers of the former article were defined by authors, and those six driving maneuvers on the highway scenario were: constant speed with lane-keeping/left lane change/right lane change, and deceleration with lane-keeping/left lane change/right lane change. The latter article proposed 27 maneuvers on the highway scenario, produced by the clustering algorithm.

The idea of maneuver-based motion prediction comes from the intuition of human drivers. In some driving situations, human drivers foresee the future trajectory based on the early recognition of the maneuver that drivers intend to perform. Similarly, the motion prediction problem can be solved by having the history trajectory to deduce the future trajectory of one driving maneuver. The existing models that apply the maneuver-based motion prediction can be classified as 'Bayesian family', 'Neural network family', 'Gaussian mixture family', 'Typical machine learning family', and 'Fuzzy logic family'.

For Bayesian family, researchers have implemented various types of Bayesian methods. Starting from the simplest one, Wyder *et al* [29] implemented the Naive Bayesian Filter to estimate left-turn, right-turn, and stop maneuver as vehicles approaching intersection with better performances, comparing with CTRV model. The most widely used method of Bayesian Family is Hidden Markov Model (HMM). By assuming the trajectory evolution process with Markov properties, Lefevre *et al* [30] implemented HMM to estimate driver's lane change intention on the highway and encoded the prediction results to assist designing personalized adaptive cruise control (ACC) module. Ernst *et al* [31] implemented an ensemble learning method, which was to incorporates the results of both fuzzy logic and HMM to estimate the driving behaviors on the intersection. On the highway scene, Deo *et al* [27] also utilized HMM to predict maneuver intentions of surrounding vehicles independently. Comparing with HMM, Dynamic Bayesian Network (DBN) is a more complex form that can define the evolution process of trajectory more thoroughly. Lefevre *et al* [32] built a DBN to estimate the driver's intention at the intersection to quantify the potential degree of risk. Li *et al* [33] proposed a framework to classify the lane change and

lane-keeping intentions on the highway. In summary, the methods coming from Bayesian family are widely implemented, but in most cases, they are preferred to be used on the intersection scene instead of the highway scene.

Neural Network (NN) has gained a lot of attention in recent decades. Due to its excellence in non-linear data representation, it becomes a trendy type of models to realize the prediction problem. A typical NN-based intention estimation framework is developed by Yoon *et al* [24]. In the proposal, researches trained a Multilayer Perceptron (MLP) to differentiate the left/right lane change and lane-keeping intention by feeding the current states of ego and surrounding vehicles. Besides that, the Deep Neural Network (DNN) methods have been explored rapidly recently. The Long Short-term Memory (LSTM) of Recurrent Neural Network (RNN) is one of the most popular prediction methods nowadays. Xin *et al* [34] utilized LSTM to handle lane change/lane-keeping estimation; and Khosroshahi *et al* [35] proposed a cooperative learning model consisting of an LSTM and a histogram to detect the turning intention on the intersection scenes. LSTM is also applied to intention estimation on the roundabout. Zyder *et al* [36] proposed an LSTM framework to estimate which exit (north, east, south) the vehicles would leave. Besides LSTM, some other DNN methods are also utilized to do similar jobs. Lee *et al* [37] proposed a method to check the lane change/lane-keeping intention using Convolutional Neural Network (CNN): it firstly transformed the surrounding sensor data (road geometry, ego position, surrounding vehicle positions) with different historical time stamps to bird-eye view feature maps. Then it feeds the image-based feature maps into CNN to get the intentions with probabilities of the surrounding vehicles. In summary, the NN-based methods were widely used on both intersection and highway scenes.

The Gaussian Mixture Model (GMM) and its variation models are one of the most popular methods for trajectory prediction. Wiest *et al* [38] utilized the GMM and Variational GMM to predict the trajectories with uncertainties at the intersection. The performance showed that it succeeded in achieving valid prediction under 2-second prediction horizon. Additionally, it proved that the Variational GMM model showed better prediction results due to its stronger representation of the nonlinear system. Besides that, there are several other methods that consist of the GMM model to fulfill the trajectory prediction. Schlechtriemen *et al* [39] achieved similar tasks by utilizing the GMM for longitudinal prediction. For lateral prediction, it uses Rapidly-exploring Random Tree (RRT) as the lateral prediction to generate possible future paths and smooth the trajectory using vehicle models. Wiest *et al* [40] proposed a Bernoulli GMM as the prediction method and achieve two seconds prediction before the vehicle approaches intersections. Li *et al* [41] proposed a prediction method consisting of GMM with particle filters to predict the trajectory of surrounding vehicles with lane change or lane-keeping maneuvers.

For typical machine learning models, Yi *et al* [25] and Ward *et al* [42] utilized K-nearest neighbor (kNN) to classify lane change/lane-keeping intentions on the highway and stop/go/yield intentions on the merging scene. SVM was also used by Wissing *et al* [43] and Izquierdo *et al* [44] to estimate the lane change/lane-keeping intentions. Xie *et al* [45] used SVM to estimate if a lane change intention was aggressive, moderate, or courteous behavior. Two additional typical approaches to be utilized were RF. Barbier *et al* [46] and Kruber *et al* [47] applied the Random Forest (RF) to estimate maneuver intention by classifying turning behavior on the intersection.

Fuzzy logic is widely used in the control field, and it can also be regarded as machine learning prediction models. Mohajerin *et al* [48] and Ernst *et al* [31] proposed a fuzzy logic-based model to estimate the driving intention by classifying turning intention and lane change intention; Yi *et al* [25] took advantages of Fuzzy c-mean to cluster the trajectories as different maneuver intentions.

Interaction-aware motion prediction

Based on the Physical-based motion prediction and Maneuver-based motion prediction methods, this type of methods applies the prediction by considering the inter-dependencies of the traffic participants. Hence, it can theoretically handle the prediction in complex traffic scenarios and reach better prediction results. The interaction-aware prediction models are rarely investigated, and it can be classified as 'Planning-based', 'Neural network-based', 'Bayesian network-based', and 'Game theory-based' models.

Planning-based methods transform the prediction works into the multi-agent planning problem. One way to solve the prediction problem is to use the idea of optimization, that every vehicle is assumed to have the same intelligence and cooperatively apply the best planning strategy according to the cost function. To reduce the computational complexity, the problem usually transforms the road geometry to grid maps. Bahram *et al* [49] proposed a method that overlay occupancy grids on the highway geometry and the longitudinal trajectory prediction of the vehicles are interactively predicted based on the occupancy percentage in the future time frames. Ding *et al* [50] also designed a cost function that assisted the nearby vehicles to plan future trajectories in both urban and highway driving scenes. Deo *et al* [51] implemented similar work by designing the cost function as the energy function, and the trajectory prediction work was based on minimizing the global energy that every target vehicle reaches. Another way to solve the planning-based prediction is to use the available driving models to predict future trajectories. For instance, Schulz *et al* [52] proposed a method that applied Intelligent Driver Model (IDM) for longitudinal

prediction, and also Diehl *et al* [53] and Li *et al* [54] also mentioned to use the IDM as a baseline method in their studies.

Similar to the 'Neural Network Family' of Maneuver-based motion prediction, Neural network-based methods also encode the history information to the deep neural network for prediction. The only difference is that their encoding strategy concerns the interaction between vehicles. One type of approach is to encode the driving information into images and take advantage of the excellent representation of CNN to images. For example, Lee *et al* [37] proposed a method on the highway scenario, that abstracted the bird-eye view scene to vectorized images first. Then it utilized CNN to predict the vectorized image in the next a few seconds. Similarly, Djuric *et al* [55] used similar methods to implement the scene prediction in the intersection, but its vectorized image contains more details in road geometry. Besides image vectorization, Hu *et al* [56] also implemented DNN to understand the interaction between nearby vehicle and outputs the coefficients of future trajectory. Dai *et al* [57] created a unified dual LSTM structure, that the upper network predicted the trajectory of each vehicle only based on its history information, and the lower network fixed the predicted trajectories by checking if the predicted future trajectories were valid for consisting the interaction mechanism. Altche *et al* [58], Kim *et al*, [59] and Park *et al* [60] also proposed LSTM methods that encoded the information about not only the target vehicles but also its surrounding ones, and the model predicted the trajectory with lane-keeping and lane change driving intentions and following trajectories.

Bayesian network is able to describe the cause and effect of the complex system by considering the probabilities. Hence, it is recognized by some researches as an appropriate model to construct the interaction mechanisms between agents for prediction. The prediction work can be fulfilled by multiple payers consisting of HMM. Li *et al* [41] provided two layers of HMM for interaction-aware prediction on the highway, where the upper layer was to provide the potential situations for each target vehicle, and the lower layer was to generate a set of future trajectories based on each situation. Geng *et al* [61] proposed a model that consisted of HMM and ontology models to cooperatively describe the interaction and achieves the turning, lane change/lane-keeping driving maneuvers. Moreover, the prediction work can also be achieved by utilizing DBN. The prediction models that Schulz *et al* [52] and Li *et al* [33] proposed consisted of a DBN for predicting vehicle intentions. Due to the model complexity, those two approaches discretized the reachable road into a grid map to simplify the prediction scenes.

Similar to the Planning-based prediction methods, Game theory-based model can describe the inter-vehicles interaction and the outcome is going to find an optimal solution. But the difference is that the agent can be either cooperative or non-cooperative and thus achieving different prediction results [62] [63].

In summary, the existing methods of motion prediction are classified as physical-based, maneuver-based, and interaction-aware motion prediction methods, according to the model complexity. As the simplest type of prediction models, physical-based motion prediction methods are outdated for the current research. However, in some cases, they are cooperatively applied with some advanced methods to optimize the prediction performance. Maneuver-based prediction methods become the most popular type of methods among the three because they balance the model complexity and prediction accuracy. Interaction-aware prediction methods are less popular among the other two types, due to its advanced but complicated model structures. Theoretically, their prediction accuracy can be significantly enhanced with a longer prediction horizon. However, since the human interaction mechanisms have not been fully discovered, the existing game theory and dynamic Bayesian network approaches can hardly expose the advantages of considering the interactions. However, the neural network approach can potentially deal with the interactions by utilizing the self-learning strategy, but the machine-generated interaction is untraceable due to its 'black box' property.

2.3.2 Methods of risk assessment

The risk assessment methods are able to evaluate the risk levels of the current situations or upcoming situations according to the prediction results. The methods of risk assessment can be divided as five categories, which are: time-based, kinematic-based methods, statistic-based and potential field-based.

Time-based risk assessment

This type of methods assesses the level of risks simply from the time-related factors. For example, the THW, TTC, and Time-to-reaction (TTR). THW is calculated by taking the time stamp that passes between the LV and EV reaching the same location [64], which can be calculated as

$$THW = \frac{(X_{\mathrm{LV}} - X_{\mathrm{EV}})}{V_{\mathrm{EV}}},$$

where the X_{LV} and X_{EV} represent the displacement of the lead vehicle (LV) and EV respectively, and V_{EV} is the current speed of EV. THW is usually used for longitudinal behavior metrics, such as lane-following scenes. TTC is defined as the time required for the collision of two vehicles by assuming that they move at their current velocities [65]. Based on the

TTC concept, some methods evolved to calculate TTC by assuming constant accelerations [66]. TTC is a simple but effective metric for evaluating the collision risks, and it is widely used for collision avoidance or collision alarming products. However, the TTC is not as sensitive as the speed difference between objects is similar.TTR is to compute the remaining time that is necessary to take the action to avoid collisions. There are some works investigating the driver reaction time [67] [68]. This type of methods are more close to reality as it considers the driver response when facing the actions and the actual braking behaviors which are related to the vehicle dynamics [67] [69]. However, the assessment would fail if the driver possessed the unexpected driving behavior, and the calculation is more complex than the THW and TTC, meaning that the TTR is more computationally expensive.

Kinematic-based methods

The kinematic-based methods achieve the risk assessment by utilizing the kinematic-related factors, such as distance or accelerations. The most straightforward applications are CV model and CA model. One typical risk assessment method based on distance factors are Mazda braking distance metrics [70]. It utilized the current accelerations, velocities of both vehicles, and the driver response to compute the minimum braking distance for the vehicle. Additionally, the required minimum deceleration to avoid collisions is used as another metric of kinematic-based methods [71] [72].

Statistic-based methods

The statistic-based methods are mainly data-driven, that the assessment models are built based on either the collision probability or the naturalistic driving dataset. The collision probability is to calculate how likely the reachable sets of each vehicle intersect with each other based on the probabilistic motion prediction results. The motion prediction part is mainly approached by Monte Carlo simulation [73] [26] [74] and Markov model [75]. The naturalistic driving dataset can be utilized for assessing the driving risks by studying general driving behavior. For instance, the related features of the braking maneuver that represents the risk level can be extracted using K-means clustering methods [76]. The risk related features can also represent the risk levels based on the NN [77]. Moreover, the Q-learning, one category of Reinforcement learning (RL), was proposed to estimate the collision risk, that the reward function was designed to augment the quick risk avoidance [78].

Potential field-based methods

The potential field-based methods assess the risk levels by examining the effect of repulsive and attractive artificial potential fields. Originally the potential field is proposed for mobile robots to avoid the obstacles. The attractive potential fields are for guiding the robot to the target position, and the repulsive potential fields are placed on the obstacles for keeping the robot away from them. The potential field force can be represented as the negative gradient of the attractive and repulsive potential field. The potential field-based methods are applied for risk assessment of the cut-in behavior [79].

2.3.3 Cut-in event prediction and risk assessment

Based on the existing methodologies and analytical results in Section 2.2, the cut-in prediction can be applied by extracting the features that indicate the occurrence of the upcoming cut-in events when the EV driver applies braking behavior. The features are uniformly called cut-in indicators, and the risk score is defined as the intensity of the braking behavior. To the best of our knowledge, there are not so many papers that doing the related field so far. Hence, we present the previous works that are representative of the cut-in prediction and risk assessment. The works can be classified into two categories, which are indicator-based methods and motion prediction-based methods.

Indicator-based methods

As the name suggests, the indicator-based method is to extract the cut-in indicators from the current or history motion states of the 'lane change scenes, and then applies the analytical or machine learning methods to map to the future braking behavior of the EV.

Kim *et al* [16] proposed an analytical method, namely Range-range Plot, to estimate the braking behavior of EV. In detail, it extracted the longitudinal and lateral gap distances between EV and LCV as cut-in indicators. After the prediction is applied, the authors found that the EV is going to apply an emergency brake ($a \geq 0.3\ g$) if the longitudinal gap distance is small ($\leq$ 12m) and LCV is at the original lane. After LCV crossed the lane and was at the target lane, the EV would attempt to release the emergency brake and switch to a normal deceleration. The prediction results successfully reflected the change in risk levels of the cut-in process, but the experiment has two drawbacks. First, it failed to state clearly the prediction horizon. And second, all the experiment was done on the driving simulator, which has the gap to the real situation.

Wu *et al* [19] and Remmen *et al* [13] proposed machine learning methods to estimate the braking behavior of EV. In specific, Wu *et al* extracted THW and longitudinal speed differences as cut-in indicators. Then it utilized the linear regression model to map the indicators to maximum braking jerk, average deceleration, and maximum deceleration of EV respectively. Fair prediction results were conducted, but the experiment exposed the same problem as mentioned above, which failed to state the prediction horizon.

Similarly, Remmen *et al* extracted the longitudinal gap distance, lateral gap distance, longitudinal speed difference, lateral speed difference from the current and previous time frames as cut-in indicators. Then it utilized ensemble learning consisting of four machine learning methods to apply the cut-in prediction and risk assessment. The result showed that the model reaches 83% accuracy for detecting the ongoing cut-in events and 61% accuracy for predicting cut-in events with the prediction horizon of 1 second. Even if it proves that the proposed model was able to predict the upcoming cut-in event, the experiment was not flawless. As a suggestion, it is supposed to select the cut-in indicators with a high correlation to the upcoming cut-in event.

Motion prediction-based methods

Different from indicator-based methods that solely extract the cut-in indicators from history data, this type of methods first predicts the future motion of LCV and then checks if the cut-in event will occur with estimated risks. The core of the method is to predict the future motion of LCV, and a few existing works have been implemented. Chen *et al* [10] proposed a constant velocity model to predict the lateral trajectory of the cut-in vehicles. Meanwhile, Liu *et al* [80] applied the motion prediction on LCV using the proposed HMM. Based on the predicted trajectory segment, it further classified if the future motion was a cut-in event or a normal lane change. The application was based on motion prediction methods.

Summary

The two methods above have proven to predict the cut-in event with estimated risks, however, each method has its drawbacks. For indicator-based methods, some existing methods do not state the prediction horizon, and some methods failed to evaluate the performance of each cut-in indicator. For motion prediction-based methods, this research field is even less rarely investigated. Those methods should be both utilized to conduct theoretical better results.

2.4 Summary and comments

In summary, this chapter introduces the existing cut-in definitions to quantitatively recognize cut-in behavior, the research of cut-in behavior, and the applications about cut-in prediction and risk assessment, with their pros and cons commented. For the cut-in definitions, a more comprehensive definition of cut-in event should be proposed with the criteria concerning about 1) the gap distance between EV and LCV and 2) the braking response of EV. For cut-in related research, most existing approaches implemented the offline analysis, and none of them analyzed the specific time of cut-in occurrence. Therefore, an online-based analysis is needed for the later online cut-in event prediction and risk assessment. For the application part, a new method is needed to propose with considering the interaction between EV and LCV. Additionally, the prediction work needs to take advantage of the results of the motion prediction of LCV, which can be given more useful indicators and increase the prediction and estimation accuracy.

Chapter 3

Problem Statement and Proposed Method Overview

In order to enable the AV to predict the cut-in behavior of LCV and assess its risk, this chapter focuses on expanding the research question into a practical problem. In particular, it first describes the specific lane change scene on the highway, then it formulates the problem by raising some constraints, including the constraints of the physical settings and the cognitive mutual influence between vehicles. Finally, our method of cut-in event prediction and risk assessment is proposed, and the corresponding framework is shown.

3.1 Lane change scene

Based on the former experience in the real application, we perceive a lane change scene on the highway that contains the EV, the LCV, and the LV. The diagram of the lane change scene is shown in Fig. 3.1. According to the figure, the highway is simplified as a two-lane road, and the car with lane change behavior is named as LCV marked in yellow. When LCV starts lane change, the lane it occupies is called the original lane; and the lane that LCV occupies at the end of lane change is called the target lane. The EV is the vehicle that is behind LCV at the target lane, and the LV is the vehicle that is in front of the LCV at the target lane. For EV, only two related vehicles are considered since the motion of LCV and LV can be fully detected by EV.

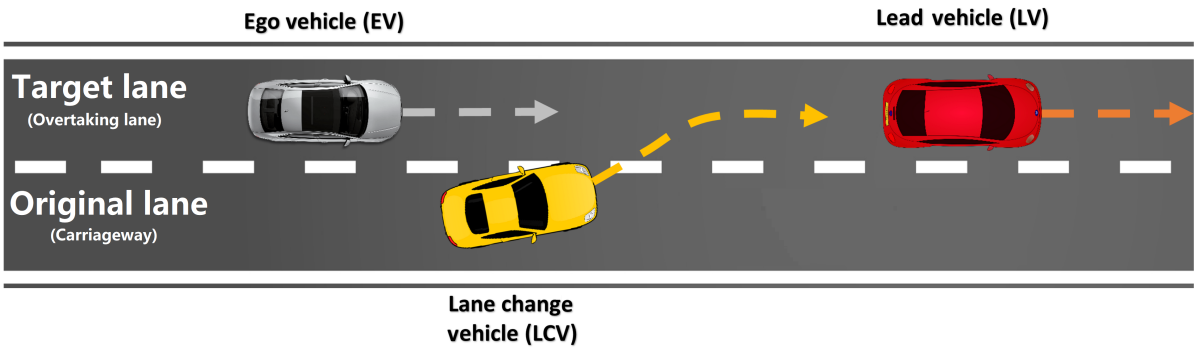


Figure 3.1: Diagram of lane change scene

3.2 Constraints

In the real application, the lane change scenes are not as simple as what we expect in the diagram above. Instead, there are lots of variations, which makes the problem complex. To clarify the problem, we specify the necessary constraints, which can be categorized as physical constraints and cognitive constraints, and the details contents of each constraint are shown below.

3.2.1 Physical constraints

Physical constraints aim at regulating the variations in physical settings. For example, the lane number of the road segment needs to be determined. In addition, we also need to emphasize the number of traffic participants with their types, and what are their driving maneuvers according to the physical law. Specifically, the physical constraints are categorized as road geometry, traffic entity type, and driving behaviors of each vehicle.

- **Road geometry:** On the highway, two adjacent lanes with the same direction are only considered. It consists of one carriageway (right) and one overtaking lane (left). The boundary between lanes is marked as the dashed line, meaning that lane change behavior is lawfully allowed. No extra ramp or merging lane is shown on the road segment.
- **Traffic entity types:** All traffic participants are vehicles in the road segment, including EV, LCV, and LV. No other traffic participants (i.e. pedestrian, static obstacles, etc.) are shown in the road segment.

- **Vehicle driving behaviors:** LCV applies left lane change, starting on the carriageway and ending up on the overtaking lane. During the lane change process of LCV, the LV, and EV possess lane-keeping behavior.

One thing worth emphasizing is that this thesis only investigates the lane change behavior. In other words, it means that the LCV has presumably made the decision of applying lane change. Anything related to lane change decision making is not going to be considered.

3.2.2 Cognitive constraints

To further formulate the problem, we also consider limiting the constraints that are caused by cognitive variations. The actions of the traffic participants not only depend on physical settings but also depend on the states of its nearby participants. For example, the driver's lane change action could be applied because there were enough gaps in the target lane. Also, while the EV noticed that the vehicle is changing the lane, it might keep the speed or slow down to maintain the sufficient gap distance. Hence, for each traffic agent, it is necessary to determine its surrounding participants that may affect its action.

In this problem, we check which surrounding vehicles the driver pays attention to under different driving maneuvers. Based on the research results of Society of Automotive Engineers (SAE) [7], we find out that under lane-keeping behavior, the average ratio of driver's gazing time in the mirror is around 5% over the total driving time. So we can speculate that the drivers generally pay attention to the front view when the vehicle applies lane-keeping. The article also shows that under the lane change behavior, the average ratios of driver's gazing time in the mirror, on the target lane, and on the original lane are around 10%, 60%, and 30% respectively. Also during the lane change, the driver's gazing ratio in the mirror is continuously decreasing while on the target lane is increasing. Based on the result, we can speculate that while the vehicle is changing lane, the driver pays the most attention to the LV; meanwhile, it pays much less attention to the EV after the lane change decision has been made.

This lane change is assumed to occur in mixed traffic, so the LCV and LV are assumed as human drivers, with their movements captured by the perception system of EV. Additionally, the Vehicle-to-vehicle (V2V) communication is assumed to be unavailable in the mixed traffic. Hence, only EV and LCV can be fully captured during the lane change and the other nearby vehicles who may be occluded or do not affect the cut-in event are not considered.

By having the supporting conditions above, we make a few cognitive constraints for the necessary vehicles shown in the problem.

- **LV:** The motion of the LV has influence on the motion of the LCV and the EV;
- **LCV:** The motion of the LCV has influence on the motion of EV, and has slight influence on the motion of LV;
- **EV:** The motion of EV has slight influence on the motion of the EV and the LCV.

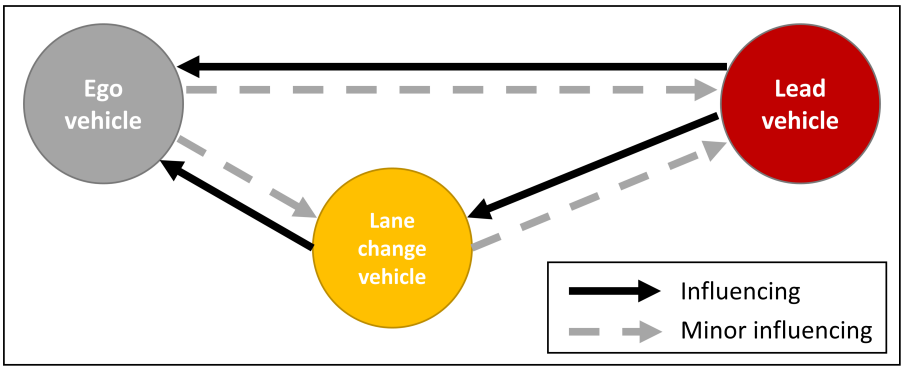


Figure 3.2: Diagram of cognitive constraints between vehicles

We also visualize the cognitive constraints by plotting a diagram, as shown in Fig. 3.2. Each node with its name represents the corresponding vehicle. The black arrow from A to B represents that the motion of vehicle A has strong influence on B, while the grey dashed arrow from A to B represents that the motion of vehicle A has slight influence on B.

By checking those arrows that link two nodes, we find that some pairs of nodes influence each other, meaning that the interaction between those node pairs exists. To simplify the diagram, we get rid of the links representing slight influences. The simplified diagram is shown in Fig. 3.3. Therefore, the simplified cognitive constraints become the following points below.

- **LV:** The motion of the LV has influence on the motion of the LCV and the EV;
- **LCV:** The motion of the LCV has influence on the motion of EV;

Having those design constraints from both physical and cognitive settings finalized, it is clear enough to propose a series of methods to predict the cut-in event and assess the risks and solve the formulated problem.

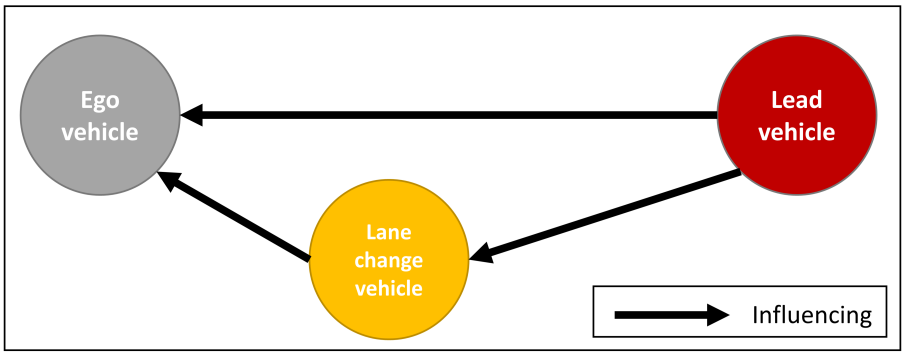


Figure 3.3: Simplified diagram of cognitive constraints between vehicles

3.3 Proposed methods

In this section, we propose the methods of solving the formulated problem above and explain the idea according to the diagram of the framework. In detail, we explain how to decompose the big problem into several small steps and clarify the proposes, corresponding input, and expected outcomes of each step. Finally, we indexed the steps to the corresponding later chapters of the article.

The diagram of the corresponding framework is shown in Fig. 3.4. In the data extraction step, the valid lane change data is extracted from the naturalistic driving dataset, which is then used to label the 'cut-in event' and 'normal lane change event' in the event labeling step. In the feature extraction and selection step, the extracted lane change data is first utilized to predict the future trajectory of LCV based on history trajectories of LCV and LV. Then the data is split into four phases, and the corresponding interaction-related features are extracted from history trajectories of EV, LCV, and predicted trajectory of LCV. After that, the drivers' cut-in responded acceleration data is extracted and normalized as the risk scoring labels in the risk score and labeling step. Having the phase-based features, the event labels, and the risk score labels prepared, the online cut-in event prediction and the corresponding risk assessment are implemented by utilizing the GMM in the final step.

The remaining part of the thesis is organized as the following structures: Chapter 4 introduces the data extraction, and Chapter 5 presents the trajectory prediction of LCV in the feature extraction and selection step and the performance of trajectory prediction. Chapter 6 details 1) phase-based splitting concept and the feature selection approach and 2) illustrates the principle of the GMM based predictor and estimator with their performance results and discussions. Concluding remarks are presented in Section 7.

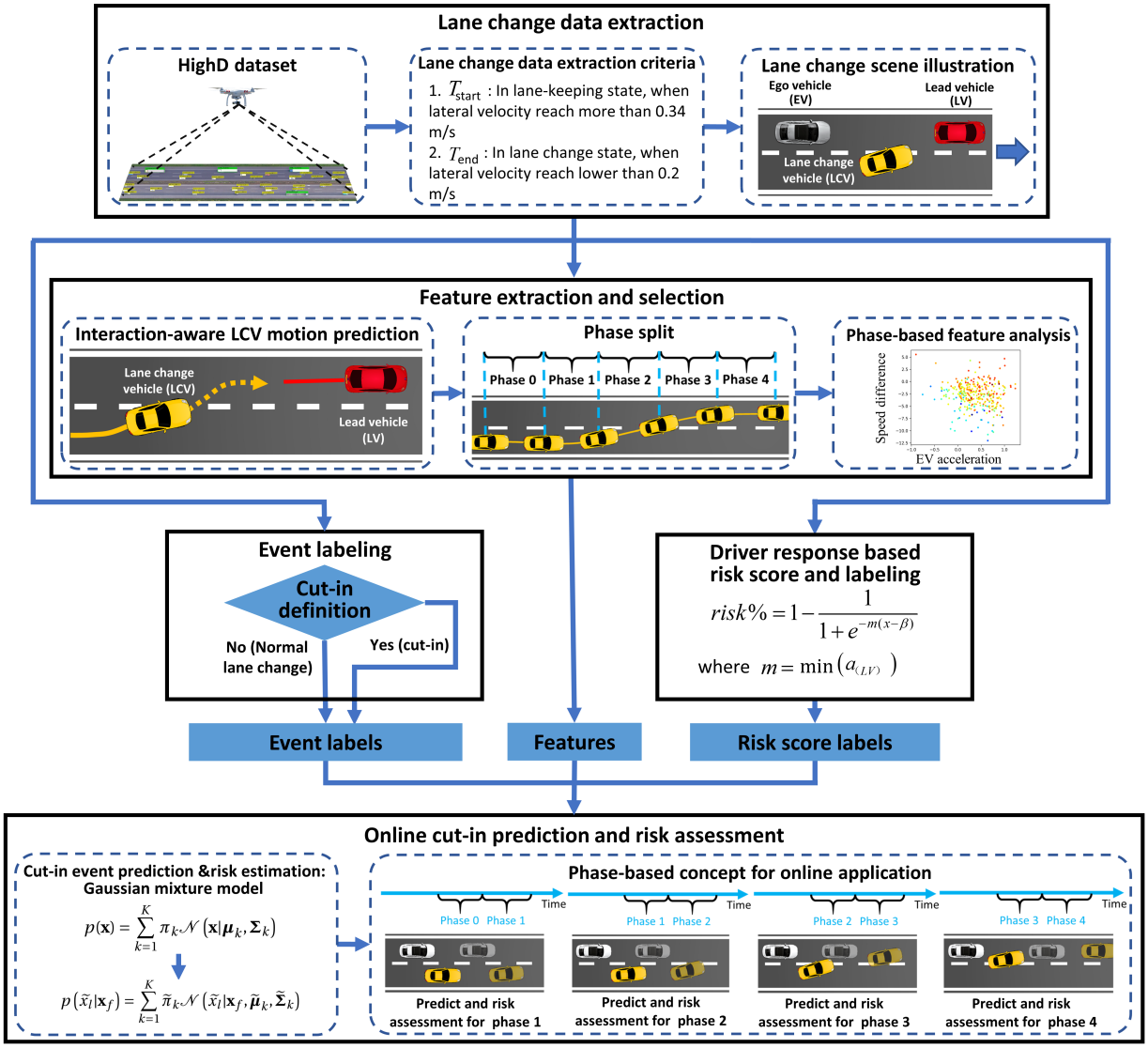


Figure 3.4: Diagram of the framework

Chapter 4

Lane Change and Cut-in Event Extraction for Experiment Data

The data is the foundation of the later experiment, and its quality can directly affect the later performances of data-driven methods. In this chapter, our goal is to prepare the two types of data for 1) LCV motion prediction and 2) cut-in event prediction and risk assessment respectively. In detail, we select HighD dataset[81] as our data source, and we show how to extract two types of valid lane change data by proposing the extraction criteria. Finally for cut-in event prediction and risk assessment, since the cut-in event is one type of the lane change event, we present the cut-in event extraction criteria to extract the cut-in scenes among those lane change scenes.

4.1 Dataset

The available public driving and traffic datasets have a wide range of variety, with each having different research purposes. Therefore, selecting an appropriate data source as further development is necessary. In this section, we first show the list of criteria that the dataset should possess, and then we will briefly introduce the dataset, HighD dataset we decide to use, and how it meets our criteria.

4.1.1 Dataset selection criteria

Before listing the criteria for the late dataset selection, we recalled the real application of the research. To realize the trajectory prediction and the cut-in prediction, we need to detect the movement of the surrounding vehicles. Although the personal driving style is an important factor to determine a driver's future actions, it cannot be directly detected. Hence, we expect the dataset to cover a variety of different lane change settings, and all of those are driven by different drivers. In this case, although we still don't know the factor of the driving style of each driver, this drawback can be recovered by increasing the amount of learning data, therefore increasing the prediction performance. On the other hand, the dataset should possess transcendent precision, so the later prediction algorithms could possess the property of robustness. In summary, we listed the criteria for data selection.

- **Precision:** The dataset should possess the up-to-date precision from data sampling and data preprocessing;
- **Quantity of lane change scenes:** The dataset should contain a large amount of naturalistic lane change scenes. Each of them should be sampled randomly on different types of road conditions, and they should be generated by random drivers.

4.1.2 The HighD Dataset

Based on our criteria above, we evaluated the existing popular public traffic dataset. The naturalistic traffic dataset, Next Generation Simulation (NGSIM) [82], and the HighD Dataset [81] are commonly used for the related research. Based on Table 4.1, we can see that the HighD dataset has general better data quality and data variety. HighD Dataset contains more precise trajectories with existing labels to be used, such as the number of lane change and neighbor vehicle ID in each trajectory, which makes further development much easier than NGSIM. About the data variety, the number of lane change trajectory from HighD dataset is around twice as many as it from NGSIM, meaning that it covers more types of driving maneuvers. But still, NGSIM has one advantage that cannot be replaced by HighD Dataset. Since NGSIM was recorded on the road segment with more lanes and heavier traffic, it shows stronger levels of interaction than HighD dataset. The rate of lane change is a good indicator to reflect the idea because more vehicles would choose to change lane in a heavy or congested traffic. Based on the discussion results above, we chose HighD Dataset as our data source for further research.

Table 4.1: A comparison between HighD Dataset and NGSIM Dataset

		HighD	NGSIM
Data quality	Erroneous trajectories	**Less**	More
	Captured road range	400-420 m	**500-640 m**
	Number of lanes	2-3	**5-6**
	Preprocessing procedure	**Yes**	No
	Pre-labeled information	**Yes**	No
Data variety	Traffic type	**Light and heavy traffic**	Heavy traffic
	Vehicle/trajectory number	**>110,000**	9,206
	Number of lane change	**> 11,000**	5,600
	Rate of lane change	0.10	**0.45**
	Merging scene	Yes	Yes

4.2 Lane change scene extraction

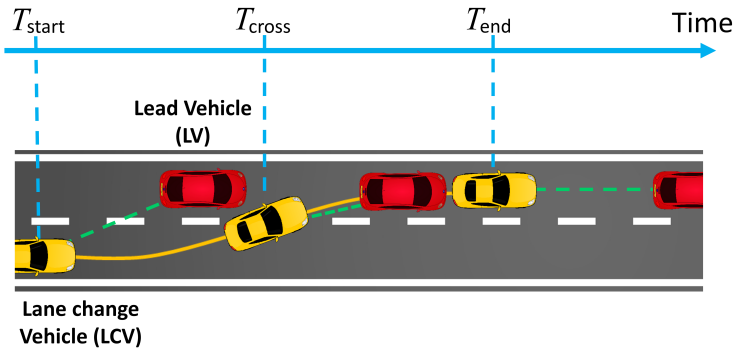
To solve the problem described in Chapter 3.3, two units need to be realized: 1) motion prediction of LCV and 2) cut-in event prediction and risk assessment. The two units require two different lane change data.

Because HighD dataset has a limited recording range (420 meters), some lane change trajectories cannot be captured completely. An extraction algorithm is proposed to extract and the complete lane change scenes for 1) LCV motion prediction and 2) cut-in event prediction and risk assessment respectively based on the following criteria:

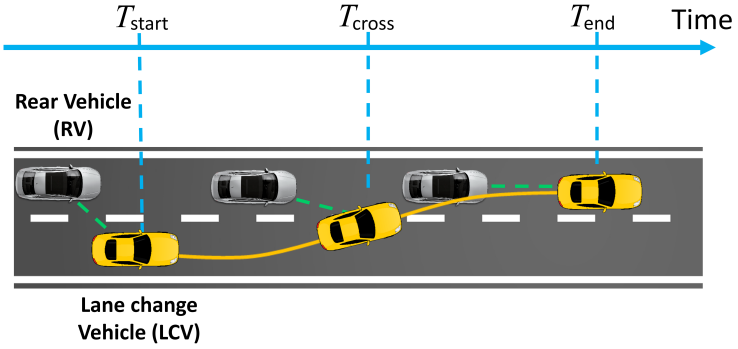
1. The LCV occupies more than one lane during the process;
2. Each lane change scene must include the time of starting lane change (T_{start}), crossing the lane marking (T_{cross}), and finishing lane change (T_{end}). It satisfies $T_{cross} \geq T_{\text{start}}$ and $T_{\text{end}} \geq T_{\text{cross}}$. The explanation of T_{start}, T_{cross} and T_{end} are shown in the Table 4.2;
3. For the lane change data of LCV motion prediction, the corresponding LV must be within the recording range and possess lane-keeping behavior during the time interval $[T_{\text{start}}, T_{\text{end}}]$;
4. For the lane change data of cut-in event prediction and risk assessment, the corresponding rear vehicle (RV) must be within the recording range and possess lane-keeping behavior during the time interval $[T_{\text{start}}, T_{\text{end}}]$,

The corresponding diagrams of lane change data are shown in Fig. 4.1. After applying the algorithm, there are 478 valid left lane change scenes extracted for LCV motion prediction, and 864 valid left lane change scenes extracted for cut-in event prediction and risk assessment. Overall, the minimum and the maximum value of lane change duration are 3.60 and 10.04 seconds respectively. The average duration value is 5.41 seconds and the standard deviation value is 0.90 seconds.

The definition of T_{start} originates from the J2944 by SAE [7]. It is proved as a conservative strategy to specify the start of lane change maneuver from lane-keeping maneuver. Conversely, the definition of T_{end} is observed manually. When the lateral velocity decreases and approaches 0.2 m/s, the LCV lateral deviation to the target lane converges to a certain value, which is 0.3 m.



(a) Lane change data for LCV motion prediction



(b) Lane change data for cut-in event prediction and risk assessment

Figure 4.1: Lane change data for LCV motion prediction

4.3 Interaction-aware cut-in event extraction

To distinguish the cut-in events from the extracted lane change scenes, the differences between the cut-in and normal lane change events need to be clarified. This section introduces a new method of extracting cut-in events by considering the potential interactions between LCV and RV.

The conditions of cut-in events are summarized based on the description of the cut-in event [16]: while the LCV is changing lanes and the driving space of RV is greatly limited, the driver of RV has to apply emergency brake to avoid crashing. In conclusion, the two conditions below are satisfied if the cut-in event occurs:

1. The longitudinal gap distance between LCV and RV is small enough and the interaction exists;

Table 4.2: Explanation of the critical timestamps of lane change

Time	Explanation
T_{start}	While the lateral velocity of LCV is increasing, the time stamp when lateral velocity of LCV approaches to 0.34 m/s.
T_{cross}	After the T_{start} occurs, the time stamp when the geometrical center of LCV covers one of the lane markings.
T_{end}	After the T_{cross} occurs and while the lateral velocity of LCV is decreasing, the time stamp when lateral velocity of LCV approaches to 0.2 m/s.

2. RV applies an emergency brake.

To quantify the first condition, the 'two-second rule', which is recommended by the New York State Department of Motor Vehicles (DMV) [83] and Road Road Safety Authority (RSA) in Ireland [84], is used to regulate the gap distance of two vehicles. The 'two-second rule' states that the driver is supposed to keep time headway for no less than 2 seconds for a safe car following. On the contrary, if the car-following distance is too close and the interaction occurs, the rule is violated, which could potentially cause accidents. Additionally, during the lane change process, the time headway value would reach the minimum at T_{cross}. In summary, one way to determine whether the gap distance is close enough between vehicles is to apply the 'two-second rule' at the time step T_{cross}.

To quantify the second condition, the minimum accelerations of the RV are extracted and plotted as a histogram, shown in Fig. 4.2. Considering the occurrence rate of the emergency event as a random variable, this paper proposes a definition, that the rate is consistent with the 'one-sigma' rule: 31.4%. However, the area 31.4% occupies the two equally extreme parts in the histogram, with the top 15.7% and the tail 15.7%. Eventually, the histogram with the minimum 15.7% acceleration cases is chosen. It means that from all lane change scene data, the scenes with 15.7% smallest acceleration values are regarded to apply emergency brakes. The corresponding threshold that represents the boundary between the normal brake and emergency brake is 0.92 m/s^2.

Based on the two conditions above, the quantitative requirements of the cut-in event are shown in Equation 4.1, with $THW_{(\text{RV})}$ representing time headway of the RV and $a_{(\text{RV})}$

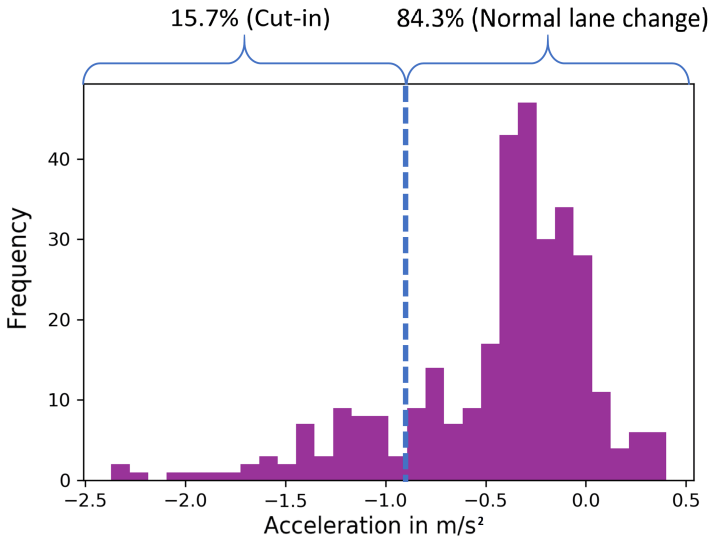


Figure 4.2: Minimum acceleration of the RV in left lane change scenes

representing acceleration of the RV. By applying the cut-in event extraction algorithm, 75 cut-in events are extracted from 478 lane change events.

$$\begin{cases} THW_{(\mathrm{RV})}[T_{\mathrm{cross}}] < 2 \text{ s} \\ \\ \min(a_{(\mathrm{RV})}[T_{\mathrm{start}} : T_{\mathrm{end}}]) < -0.92 \text{ m/s}^2 \end{cases} \tag{4.1}$$

In summary, the cut-in extraction criteria have several merits. First, it is deduced from the commonly-agreed conception of the 'cut-in event'. Moreover, it describes the levels of interaction between the LCV and the RV based on the 'two-second rule' from transportation authorities. The deceleration threshold is reasonable as it is summarized based on a relatively large volume of naturalistic lane change data.

Chapter 5

Interaction-aware Motion Prediction of Lane Change Vehicle

In this chapter, we present the design of an interaction-aware motion prediction algorithm for LCV. The algorithm is able to present real-time prediction, meaning that it outputs the predicted future trajectory segment based on any parts of the lane change segment. Additionally, the predicted future movement of LCV considers the potential mutual influence between itself and the lead vehicle. In detail, the design is presented at the following steps: First, the structure of the algorithm with the configurations, including input, output, consumed duration, and predicted horizon, is specified. Then, the training set is processed from lane change data, which includes trimming, data augmentation, and dimension reduction. Later the training phase and testing phase using GMM is presented. Finally, the results of motion prediction are discussed.

5.1 Prediction structure formulation

As was discussed in Section 3.3, the future motion of LCV is influenced by LCV itself and LV. Therefore, the history segments of their trajectories should be both concreted. For LCV, the lateral distance regarding the crossed lane boundary is important, since it reflects the process of the lane change maneuver. For both trajectories, the longitudinal displacement of the trajectory is redundant, because it can be done by integrating the profile of the longitudinal velocity. However, the longitudinal position difference between the two vehicles is necessary, as it potentially dominates the level of interaction between the two vehicles. In summary, the history factors for motion prediction are:

- Longitudinal and lateral history velocity profiles of LCV and LV;
- The lateral deviation percentage of LCV regarding the crossed lane boundary;
- Lateral behavior of LCV - left lane change or right lane change;
- Longitudinal gap distance between LV and LCV.

Note that h_h and h_f represent the horizon of history and future trajectory respectively. Then the predicted trajectory of LCV can be formulated using future factors, which are longitudinal and horizontal velocity profiles.

Since the prediction algorithm will be built using GMM, which is one type of machine learning problem. The feature vector can be constructed by concatenating the history factors, and the label vector can be constructed by concatenating the future factors. However, the size of both the feature vector and the label vector would become very large and redundant if the data sampling frequency, history, and future profile duration increases. For example, the HighD Dataset has a sampling frequency of 25 Hz. If the h_h and h_f are 1 second and 4 seconds respectively, the feature and label vector size would be 103 and 200, respectively. The redundant sizes will drag the computation speed for both model training and trajectory prediction. Another problem is, one of the features, the later velocity of LCV would be relatively small number varies much small, making the signal-to-noise ratio large and worse prediction precision. Therefore, dimension reduction and feature augmentation are crucial tasks to do before the prediction.

5.2 Data preparation

In this section, we introduce the method of data preparation for later training, including trimming the lane change data, feature augmentation, and feature dimension reduction to accord with the formulated structure.

5.2.1 Trajectory trimming

Having a bunch of lane change data with various time duration, we trimmed the data for the feature vector (history velocity profile) and label vector (future velocity profile) based on specified history horizon h_h and predicted horizon h_f respectively. For each lane change data, we first specify the current time stamp T_c for several times. The T_c starts at

time stamp of the start lane change . Then the next T_c is specified at 0.2 seconds after the last T_c. The T_c is pushed forward recursively until the T_c approaches to the time stamp of the end lane change. For more details about the time stamps of start lane change and end lane change, see Section 4.2.

Then at each current time stamp T_c, we trimmed the longitudinal velocity profile of LCV $v_{x(LCV)}$, lateral velocity profile of LCV $v_{y(LCV)}$, longitudinal velocity profile of LV $v_{x(LV)}$, and lateral velocity profile of LV $v_{y(LV)}$ at the time interval of $[T-h_h, T_c]$ as history velocity profiles, shown as

$$\begin{aligned} &v_{x(LCV)}\left[T_c - h_h, T_c\right] \\ &v_{y(LCV)}\left[T_c - h_h, T_c\right] \\ &v_{x(LV)}\left[T_c - h_h, T_c\right] \\ &v_{y(LV)}\left[T_c - h_h, T_c\right]. \end{aligned} \tag{5.1}$$

We also trimmed the longitudinal velocity profile of LCV $v_{x(LCV)}$, lateral velocity profile of LCV $v_{y(LCV)}$ at the time interval of $[T_c, T_c+h_f]$ as future velocity profiles, shown as

$$\begin{aligned} &v_{x(LCV)}\left[T_c, T_c + h_f\right] \\ &v_{y(LCV)}\left[T_c, T_c + h_f\right]. \end{aligned} \tag{5.2}$$

and Equation 5.1 and 5.2 are the history velocity profiles and the future velocity profiles respectively for the data point.

5.2.2 Feature augmentation

As mentioned in Section 5.1, the profile data is supposed to apply feature augmentation and dimension reduction. For feature augmentation, we simply transformed the longitudinal and lateral velocity profile into polar coordinates, which is $v\psi$-coordinate. The v represents the speed scalar, and ψ represents heading angle of the vehicle. In each time step T, the scalar velocity $v[T]$ and heading angle $\psi[T]$ of the vehicle are given as:

$$v[T] = \sqrt{(v_x[T])^2 + (v_y[T])^2} \quad , \tag{5.3}$$

and

$$\psi[T] = \arctan \frac{v_y[T]}{v_x[T]} \quad , \tag{5.4}$$

where $v_x[T]$ and $v_y[T]$ are longitudinal and lateral velocity value in time step T.

In the real implementation, we transformed the $\psi[T]$ from radius into degree for more obvious observation of lateral movement.

5.2.3 Feature dimension reduction

For dimension reduction, we applied Chebyshev polynomial to fit each profile. The coefficients of the fitted polynomial are utilized as a part of the feature vector for the later GMM model. First, we defined the Chebyshev polynomial Y_n of degree n as

$$Y_n(x) = \cos(n \arccos(x)) \quad , \tag{5.5}$$

that contains trigonometric property but is able to represent as polynomial. When $n = 0$ and $n = 1$, the two polynomials are shown as

$$Y_0(x) = 1 \tag{5.6}$$
$$Y_1(x) = x \tag{5.7}$$

and the higher order polynomials can be calculated with iteration, as given

$$Y_{n+1}(x) = 2xY_n(x) - Y_{n-1}(x) \quad , \text{where} \quad n \geq 1. \tag{5.8}$$

To fit a function $f(x)$ in the interval $[-1, 1]$, the Chebyshev coefficients can be defined as

$$c_n = \frac{2}{N} \sum_{k=0}^{N-1} f\left(x_k\right) Y_n\left(x_k\right) \tag{5.9}$$

where x_k are the N zeros of $Y_n\left(x_k\right)$. The discrete velocity profile can be approximated using the least square approximation to calculate the coefficients, that representing the velocity

profile. Based on the function complexity of the velocity profiles, the highest degree is determined as n. The time intervals of the velocity profiles are mapped as the interval $[-1, 1]$ for easier function reconstruction. It reduces the feature dimension by transforming the velocity profile with size $\mathbb{R}^{h \times f}$ to the array of coefficients $\mathbf{c} \in \mathbb{R}^{n+1}$, where h, f represent segment horizon and sampling frequency respectively.

The highest degree of the function contains, the better approximation ability it has with higher numbers of coefficients. Hence, an optimal degree of n can balance the good fitting performance and small dimension size. To find the optimal highest degree number n, we compared the fitting performance under different numbers of highest degrees. In detail, we extracted a longitudinal speed profile from an arbitrary trajectory of HighD dataset. The trajectory lasts eight seconds with 200 samples in total. Then we applied least square regression to fit it as Chebyshev polynomial with the highest degree from 1 to 6. Finally, we recovered the polynomial coefficients to be the trajectories. The performance results are shown in Fig. 5.1.

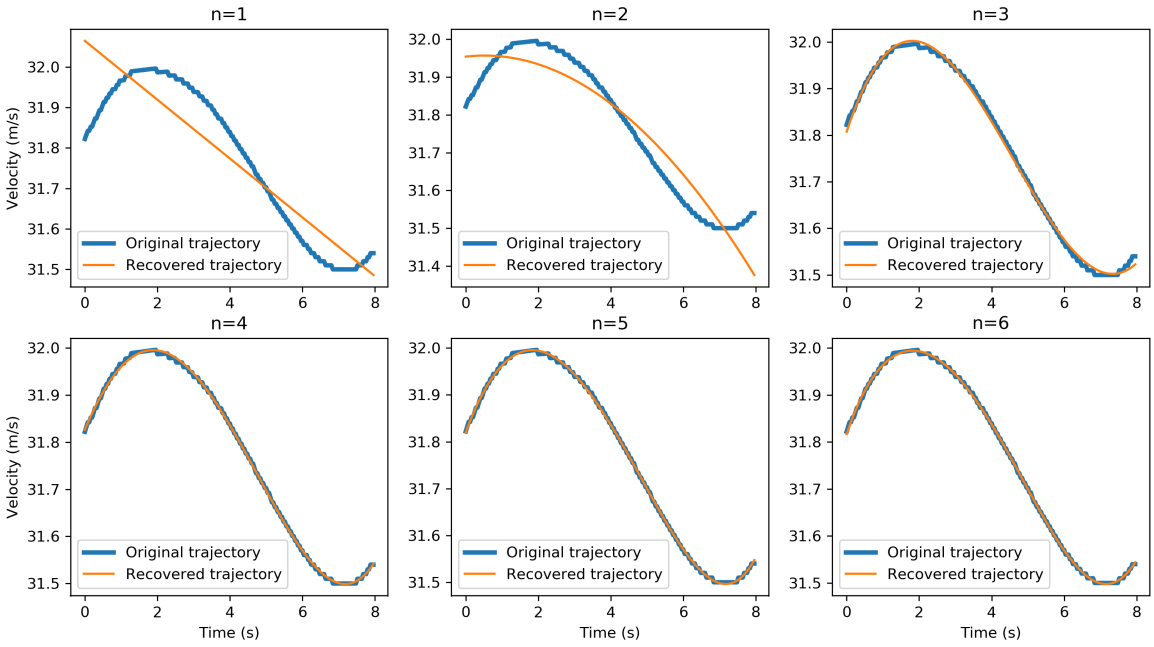


Figure 5.1: Performance of the Chebyshev polynomial fit

Based on the results, we conclude that the $4th$-degree Chebyshev polynomial fit reaches excellent performances of data approximation while also reaching the purpose of feature dimension reduction. The reduced features greatly reduce the computational load of the

learning model, which will assist the online application performance of the latter prediction.

5.2.4 Processed data format

After the three steps of data preparation, including trajectory trimming, feature augmentation, and dimension reduction, the processed training data are concatenated as the input feature vector, shown as

$$\boldsymbol{x}_f = \begin{bmatrix} \boldsymbol{C}_{hist-v(LCV)} & \boldsymbol{C}_{hist-\psi(LCV)} & \boldsymbol{C}_{hist-v(LV)} & \boldsymbol{C}_{hist-\psi(LV)} & d_{y(LCV)} & b_{y(LCV)} & \Delta d_x \end{bmatrix}, \tag{5.10}$$

where the first four factors are *n-th* degree Chebyshev coefficient arrays of LCV history velocity profile, LCV history heading angle profile, LV history velocity profile, and LV history heading angle profile respectively. The fifth factor is lateral deviation percentage of LCV regarding the crossed lane boundary at the current time frame. The sixth factor is the categorical value of lateral behavior of LCV, which '0' stands for left lane change and '1' stands for right lane change. The seventh factor is the longitudinal gap distance of two vehicles at the current time frame.

The corresponding processed label data are concatenated as the input label vector, shown as

$$\boldsymbol{x}_l = \begin{bmatrix} \boldsymbol{C}_{fut-v(LCV)} & \boldsymbol{C}_{fut-\psi(LCV)} \end{bmatrix}, \tag{5.11}$$

where the two factors represent *n-th* degree Chebyshev coefficient arrays of LCV future velocity profile, LCV history heading future profile. It is also the ground truth in the prediction phase.

5.3 Gaussian mixture model-based motion prediction

In this section, we present the motion prediction algorithm based on Gaussian mixture model (GMM), due to its excellent representation property. In details, we represent the training step and the prediction step of the model, and finally show the reconstruction step, that rover the predicted result to the predicted trajectory.

5.3.1 Training step

The GMM is a function that describes probability density parametrically. Its structure is formed by an arbitrary number of weighted Gaussian probability density functions. The generic formula of GMM is shown as:

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) \ , \tag{5.12}$$

where $\boldsymbol{x}$ is a d-dimensional random variable, and $\mathcal{N}\left(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)$ is a multivariate normal distribution with mean $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{\Sigma}_k$. K represents the number of mixed Gaussian kernels of GMM, and π_k is the weighting coefficient that satisfies:

$$\begin{cases} 0 \leq \pi_k \leq 1 \\ \\ \sum_{k=1}^{K} \pi_k = 1. \end{cases} \tag{5.13}$$

In the training phase, the Expectation-Maximization (EM) algorithm iteratively consumes all pieces of training data and produces the model parameters [85]. The form of the input data point is the concatenation of input feature vector $\boldsymbol{x}_f$ and input label vector $\boldsymbol{x}_l$, shown as

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_f & \boldsymbol{x}_l \end{bmatrix}, \tag{5.14}$$

where the format of $\boldsymbol{x}_f$ and $\boldsymbol{x}_l$ is shown in Section 5.2.4

Before the training starts, there are three parameters need to be specified. The Gaussian kernel number K is to determine the complexity of GMM, while the convergence threshold h_{conv} and the maximum iteration J_{max} are to jointly determine the model accuracy and training speed.

5.3.2 Prediction step

As the training is done, the trained model with parameters π_k, $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$, describes the joint mixture distribution of $\boldsymbol{x}_f$ and $\boldsymbol{x}_l$. Motion prediction of LCV is to compute the conditional distribution of estimated label $\widetilde{\boldsymbol{x}}_l$ given a test feature vector $\boldsymbol{x}_f$ and trained GMM. The conditional distribution can also be written as:

$$
\begin{aligned}
p\left(\widetilde{\boldsymbol{x}}_l|\boldsymbol{x}_f\right) &= \frac{p\left(\boldsymbol{x}_f, \widetilde{\boldsymbol{x}}_l\right)}{\int p\left(\boldsymbol{x}_f, \widetilde{\boldsymbol{x}}_l\right) d\widetilde{\boldsymbol{x}}_l} \\
&= \sum_{k=1}^{K} \widetilde{\pi}_k \mathcal{N}\left(\widetilde{\boldsymbol{x}}_l|\boldsymbol{x}_f, \widetilde{\boldsymbol{\mu}}_k, \widetilde{\boldsymbol{\Sigma}}_k\right),
\end{aligned} \tag{5.15}
$$

which is still the Gaussian mixture form. The parameters of the conditional mixture are given as:

$$
\widetilde{\pi}_k = \frac{\pi_k \mathcal{N}\left(\boldsymbol{x}_f|\boldsymbol{\mu}_{k,x_f}, \boldsymbol{\Sigma}_{k,x_f x_f}\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\boldsymbol{x}_f|\boldsymbol{\mu}_{j,x_f}, \boldsymbol{\Sigma}_{j,x_f x_f}\right)} \tag{5.16}
$$

$$
\widetilde{\boldsymbol{\mu}}_k = \mu_{k,x_l} + \boldsymbol{\Sigma}_{k,x_l,x_f} \boldsymbol{\Sigma}_{k,x_f x_f}^{-1} \left(\boldsymbol{x}_f - \boldsymbol{\mu}_{k,x_f}\right) \tag{5.17}
$$

$$
\widetilde{\boldsymbol{\Sigma}}_k = \boldsymbol{\Sigma}_{k,x_l x_l} - \boldsymbol{\Sigma}_{k,x_l x_f} \boldsymbol{\Sigma}_{k,x_f x_f}^{-1} \boldsymbol{\Sigma}_{k,x_f x_l}, \tag{5.18}
$$

where

$$
\boldsymbol{\mu}_k = \begin{bmatrix} \boldsymbol{\mu}_{k,x_f} \\ \boldsymbol{\mu}_{k,x_l} \end{bmatrix} \tag{5.19}
$$

$$
\boldsymbol{\Sigma}_k = \begin{bmatrix} \boldsymbol{\Sigma}_{k,x_f x_f} & \boldsymbol{\Sigma}_{k,x_f x_l} \\ \boldsymbol{\Sigma}_{k,x_l x_f} & \boldsymbol{\Sigma}_{k,x_l x_l} \end{bmatrix} \tag{5.20}
$$

is the partition of means and covariance of the GMM.

The mean and the covariance of the estimated label are specified as the full conditional distribution, which are given as:

$$
\widetilde{\boldsymbol{\mu}} = \sum_{k=1}^{K} \widetilde{\pi}_k \widetilde{\boldsymbol{\mu}}_k \tag{5.21}
$$

$$
\widetilde{\boldsymbol{\Sigma}} = \sum_{k=1}^{K} \widetilde{\pi}_k^2 \widetilde{\boldsymbol{\Sigma}}_k \ . \tag{5.22}
$$

5.3.3 Reconstruction step

In the reconstruction step, the predicted results are able to be recovered to the predicted trajectory. The estimated mean value $\widetilde{\boldsymbol{\mu}}$ is selected as the predicted results. It is also consisted with the concatenation of two Chebyshev coefficient arrays, with one represents predicted future velocity profile and the other represents predicted future heading angle profile, shown as

$$\widetilde{\boldsymbol{x}}_l = \widetilde{\boldsymbol{\mu}} = \begin{bmatrix} \widetilde{\boldsymbol{C}}_{fut-v(LCV)} & \widetilde{\boldsymbol{C}}_{fut-\psi(LCV)} \end{bmatrix}. \tag{5.23}$$

By reconstructing the $\widetilde{\boldsymbol{C}}_{fut-v(LCV)}$ and $\widetilde{\boldsymbol{C}}_{fut-\psi(LCV)}$ through Chebyshev polynomial, we get the predicted velocity profile $\widetilde{v}_{x(LCV)}$ and heading angle profile $\widetilde{v}_{x(LCV)}$ in the time interval $[T_c, T_c + h_f]$, where T_c represents the current time step and h_f represents prediction horizon. After transforming them into XY-coordinate, we get the longitudinal and lateral speed profile of predicted LCV. Finally, the trajectory of LCV can be calculated through simple integrations.

5.4 Results and discussion

In this section, we do a comprehensive evaluation of the motion prediction algorithm. In order to get the optimal configuration of the model design, we try to answer the different questions:

1. What is the optimal number of Gaussian kernel of the model?
2. Does the model with interaction-aware features show better performance on prediction than the same model without interaction-aware features?

To answer the questions above, we design the experiments on the following evaluation procedures. To answer the first question, we train several models using different Gaussian kernel number K and examine the accuracy of each model by calculating in longitudinal, lateral, and Euclidean root mean square error (RMSE) under different prediction horizons. The example format of accuracy metrics is shown in Table 5.1. To answer the second question, we prepare another training model with the same configurations and training

data, except that the input training vector only contains the Chebyshev polynomial efficient arrays of LCV. The next section shows the detailed procedures of deducing the optimal configuration of the model, and the later section compares the performances between our purposed interaction-aware motion prediction algorithm and the same model without considering interaction-aware features.

Table 5.1: Example format of performance results

	Prediction horizon			
Accuracy	1	2	3	$\cdots$
Longitudinal	$\cdots$	$\cdots$	$\cdots$	$\cdots$
Lateral	$\cdots$	$\cdots$	$\cdots$	$\cdots$
Euclidean	$\cdots$	$\cdots$	$\cdots$	$\cdots$

5.4.1 Influences of Gaussian kernel number to performances

To evaluate how does the Gaussian kernel number influence the prediction performance, we first collect $20,000$ numbers of data points in total. Training for the GMM is done by using the aforementioned EM algorithm with numbers of iteration $J_{max} = 1000$ and convergence threshold value $h_{conv} = 0.001$.

Since the amount of data influences the complexity of the model, therefore we find the optimal Gaussian kernel number K for each data. The optimal K is specified based on Akaike Information Criterion (AIC), which is a standard to evaluate how good a model can fit the data. A good machine learning model possesses the model complexity that neither underfit nor overfit the data, and the optimal complexity can be located using the plot of AICs under different model complexity. To find the optimal K, we plot the AICs under different K as shown in Fig. 5.2, where x-axis represents Gaussian kernel number K of GMM and y-axis represents the AIC values. The optimal K number is the 'elbow point' of the profile, which is marked in red From the figure, we can see that the optimal $K = 1200$.

To prove that if the optimal condition reaches if $K = 1200$, we use the same data to train four models with different K and test their accuracy. Those four models are consisted of 100, 600, 1200, and 1700 Gaussian kernels respectively. The training and testing follow the '10 times 10-fold' cross-validation principle. Therefore, 90% of the data set are used for training, and the remaining 10% are used for testing. The results of the models are shown in Table 5.2, and the corresponding visualization of the performance is shown in Fig. 5.4.

Table 5.2: Performances of the prediction models

(a) GMM with $K = 100$

	Prediction horizon (in seconds)			
RMSE	1	2	3	4
Longitudinal	0.0058	0.0362	0.1094	0.2352
Lateral	0.0062	0.0292	0.0641	0.0990
Euclidean	0.0094	0.0516	0.1387	0.2728

(b) GMM with $K = 600$

	Prediction horizon (in seconds)			
RMSE	1	2	3	4
Longitudinal	0.0051	0.0275	0.0769	0.1564
Lateral	0.0056	0.0218	0.0431	0.0621
Euclidean	0.0084	0.0387	0.0956	0.1781

(c) GMM with $K = 1200$

	Prediction horizon (in seconds)			
RMSE	1	2	3	4
Longitudinal	0.0047	0.0226	0.0605	0.1215
Lateral	0.0051	0.0184	0.0354	0.0505
Euclidean	0.0077	0.0322	0.0763	0.1401

(d) GMM with $K = 1700$

	Prediction horizon (in seconds)			
RMSE	1	2	3	4
Longitudinal	0.0046	0.0210	0.0553	0.1097
Lateral	0.0046	0.0172	0.0329	0.0470
Euclidean	0.0075	0.0300	0.0700	0.1271

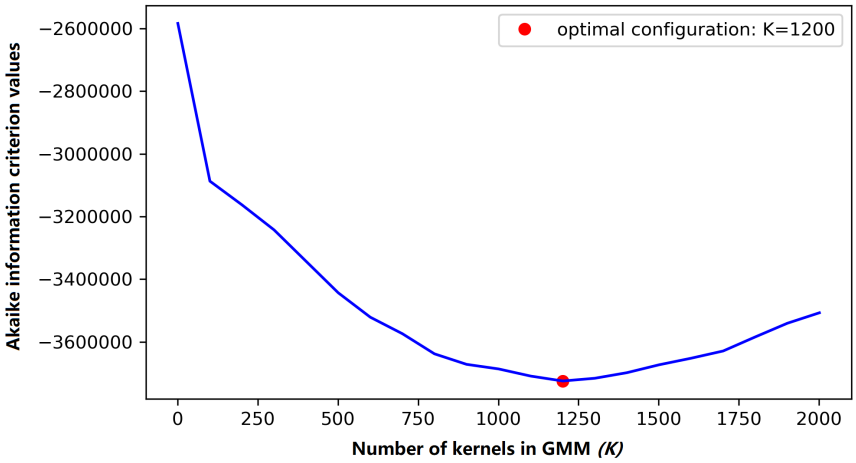


Figure 5.2: AIC values under different Gaussian kernel numbers of GMM

There are some general findings based on the plot. Under similar cases, the performances of the lateral prediction are better than the longitudinal since the scale of movement on longitudinal is much larger than the lateral one. And the RMSE does not show the linear relationship with the prediction horizon - the RMSE possesses the exponential growth as the prediction horizon increases.

To prove if the model complexity reaches optimal as $K = 1200$, we test the performance of the model under different K. From the results, we can see that the accuracy is going to converge as the K increases from 100 to 1700. In detail, when K increases from 100 to 600, the accuracy is greatly improved. As the K continue increasing from 600 to 1200, the accuracy is still improving with a smaller scale. As the K increase from 1200 t0 1700, the performance shows only a minor increase. Based on the results, we can see that the model is underfitting and approach to overfitting as K increases. Hence, we can conclude that the $K = 1200$ cannot be guaranteed to be the optimal condition, but it is close to the optimal condition. The result of a typical prediction with K=1200 is visualized and shown in Fig. 5.3.

5.4.2 Effect of interaction-aware features on prediction

To test the effect of the interaction-aware feature on model performance, we compare the proposed method with the equivalent model without interaction-aware features, and it

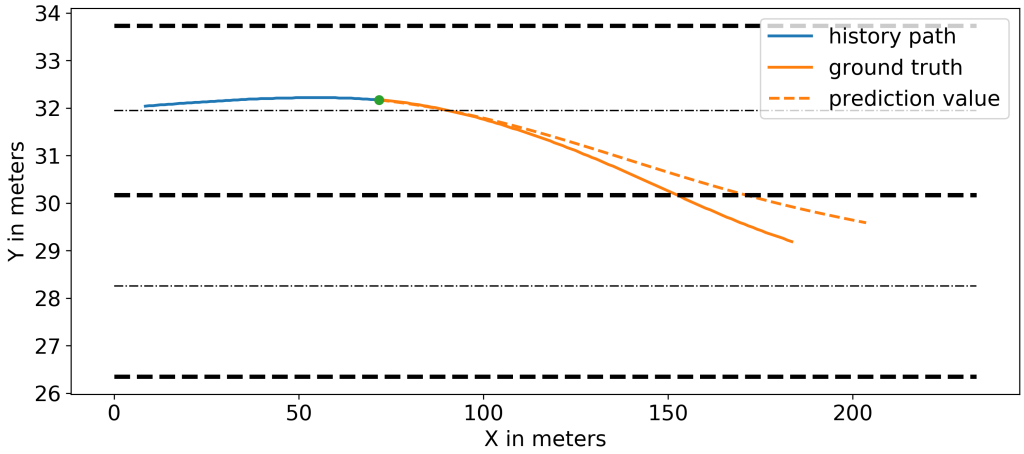


Figure 5.3: An example of the LCV motion prediction result

is regarded as a baseline method.

The performances of the comparison are shown in Table 5.3. Note that the two models are trained under the same configuration: the number of Gaussian kernel components is $K = 200$, and the training is also done by using the aforementioned EM algorithm with numbers of iteration $J_{max} = 1000$ and convergence threshold value $h_{conv} = 0.001$. Additionally, the training and testing data are identical, with the amount of $14,000$. Both models are tested by '10 times 10-fold' cross-validation. The horizons of history trajectory are both set as $h_h = 1$ second and the horizons of predicted trajectory are both set as $h_f = 6$ seconds. To make the later comparison more straightforward, we visualize the results and plot as shown in Fig. 5.5.

Based on the figure, we can see that both the baseline model and the proposed model share some similarities. Their performances of the lateral prediction are better than the longitudinal one. The RMSE shows the exponential growth as the prediction horizon is linearly increasing. However, compared with their performances, we found out that the proposed model shows obvious better performances on the longitudinal accuracy. As for the lateral prediction, however, shows minor better performances than the baseline model. The reason is that the interaction-aware features of the proposed model are the history trajectory of LV with the lane-keeping maneuver. Therefore, the longitudinal factor shows more useful information than the lateral ones. Based on the discussion above, we conclude that the interaction-aware features are necessary for improving prediction accuracy.

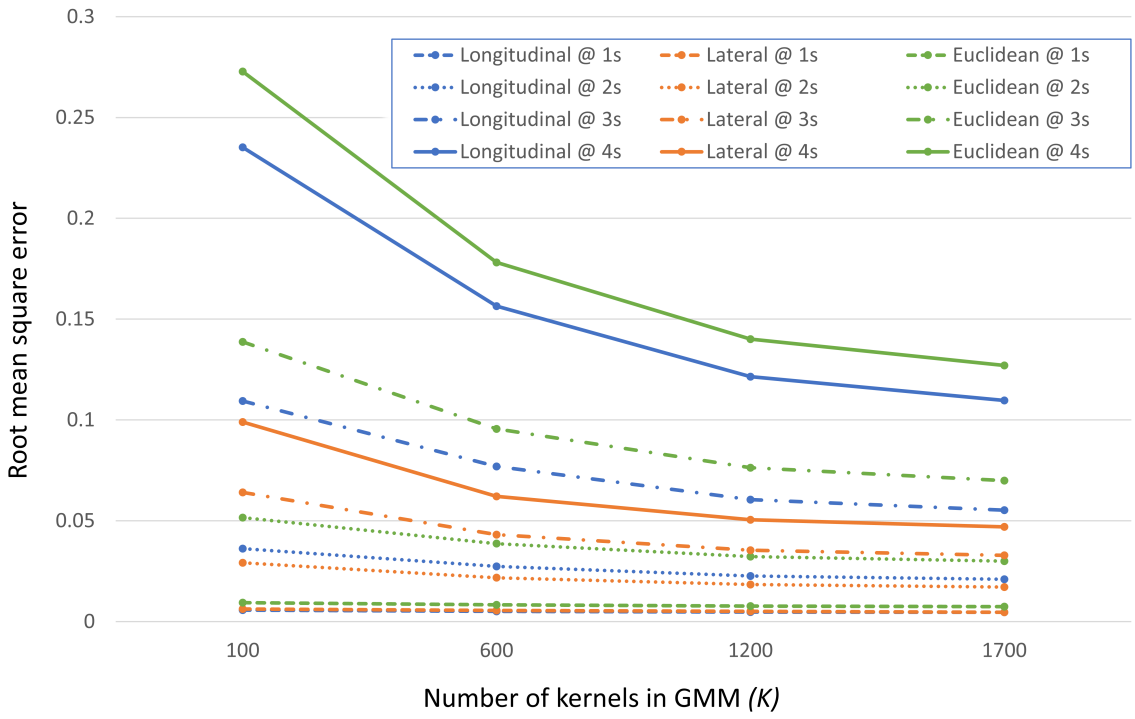


Figure 5.4: RMSE under different amount of Gaussian kernels

Table 5.3: Performances of the proposed and the baseline method

(a) Proposed interaction-aware prediction

	Prediction horizon (in seconds)					
RMSE	1	2	3	4	5	6
Longitudinal	0.0058	0.0309	0.0866	0.1766	0.3011	0.4594
Lateral	0.0067	0.0267	0.0516	0.0752	0.0932	0.1063
Euclidean	0.0097	0.0452	0.1100	0.2048	0.3295	0.4854

(b) Baseline prediction

	Prediction horizon (in seconds)					
RMSE	1	2	3	4	5	6
Longitudinal	0.0065	0.0376	0.1124	0.2391	0.4206	0.6556
Lateral	0.0071	0.0302	0.0614	0.0923	0.1177	0.1372
Euclidean	0.0104	0.0534	0.1391	0.2721	0.4546	0.6874

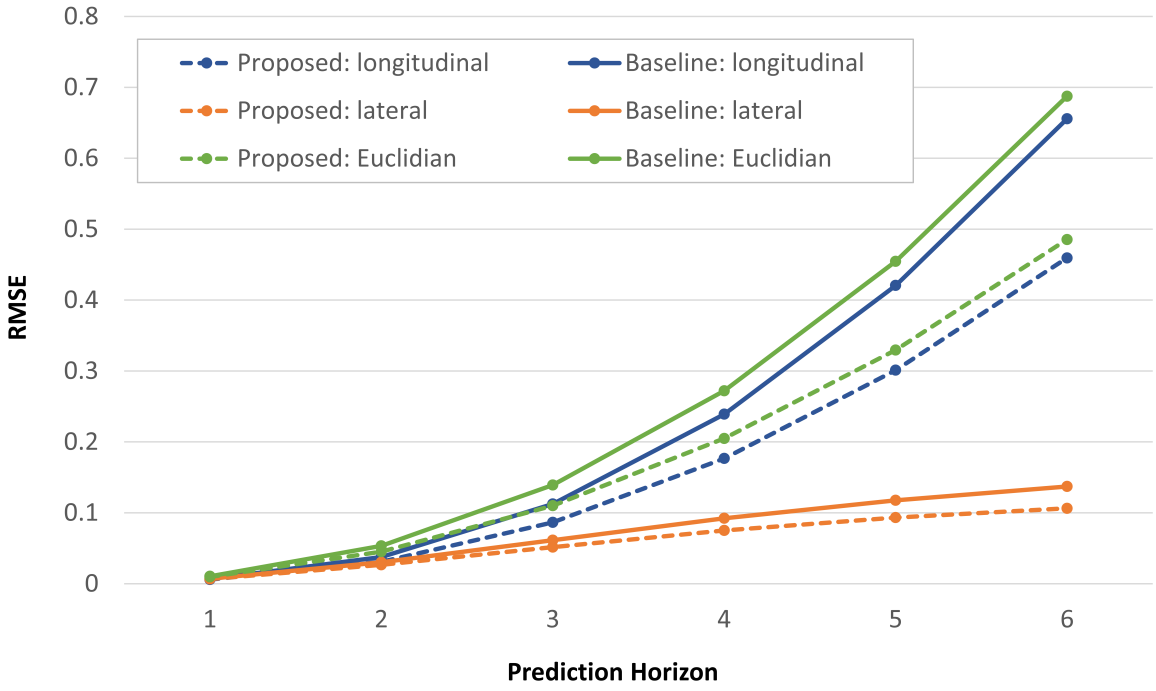


Figure 5.5: Performances of the proposed and the baseline prediction methods

Chapter 6

Online Cut-in Event Prediction and Risk Assessment

In this chapter, we present the design of online cut-in event prediction and risk assessment algorithms. As mentioned in Chapter 4.2, the typical lane change duration lasts from 3.6 to 10.4 seconds. During a such long process, the driving actions will keep changing and the cut-in event may be thereby triggered. To achieve the online cut-in event prediction and risk assessment, we developed a phase-based splitting method to segment the extracted lane change process into several different phases, and the two algorithms are applied to predict the cut-in event and estimate the risk of the next phase based on the current phase.

6.1 Structure formulation

As was discussed in Chapter 3.3, the occurrence of the cut-in event is related to two factors: one is the abnormal lane change of the LCV, and the other one is determined by the state of the ego vehicle EV. Therefore, the history information of both LCV and EV is supposed to be concerned. Additionally, by knowing the predicted trajectory of LCV, the cut-in prediction can be better handled. To get the predicted trajectory of LCV, Chapter 5 presents the comprehensive implementation of LCV motion prediction design. Hence, the history information of EV, LCV, and predicted future information of LCV are going to be concerned about the cut-in prediction and risk estimation.

To realize the online prediction and risk estimation, we proposed a phase-based splitting method. In the state-of-the-art, the cut-in related research was extensively investi-

gated, but only a few publications focus on the cut-in event prediction and estimation. In the existing methods, the prediction and estimation methods are only applied once in each lane change event, which cannot reach the online prediction. Therefore, our methods filled the research gap by dividing the lane change process into four phases, and our proposed phase-based prediction and estimation algorithms can be applied four times in each phase to reach the online application. The diagram of the online application in a lane change process is shown in Fig. 6.1,

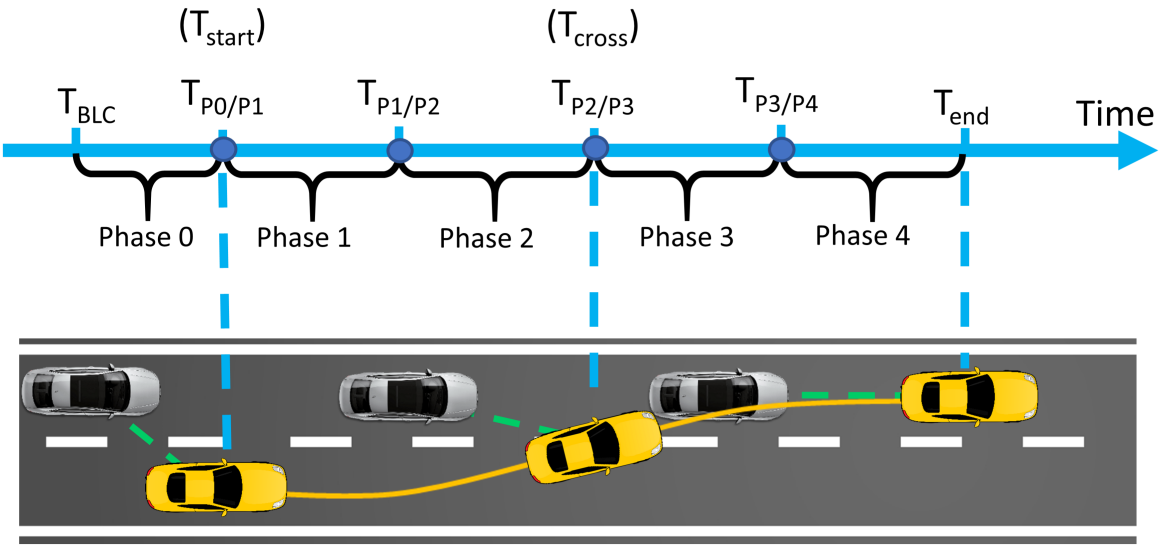


Figure 6.1: Diagram of online cut-in event prediction and risk estimation

where the blue dots marked on the timeline are the time stamps of each application. $T_{\mathrm{P0/P1}}$, $T_{\mathrm{P1/P2}}$, $T_{\mathrm{P2/P3}}$, $T_{\mathrm{P3/P4}}$ mean the time stamps of the edges of the two neighbor phases. 'P0', 'P1', 'P2', 'P3', and 'P4' represent 'Phase 0', 'Phase 1', 'Phase 2', 'Phase 3' and 'Phase 4' respectively. When the current time T_c reaches $T_c = T_{P[n]/P[n+1]}$, the detailed procedures are:

1. Predict the future motion of LCV in the time interval $[T_c, T_c + h_f]$, where h_f stands for the prediction horizon;
2. Based on the predicted motion, estimate the ending point of the next phase $\widetilde{T}_{P[n+1]/P[n+2]}$, and trim the predicted trajectory to the time interval $[T_c, \widetilde{T}_{P[n+1]/P[n+2]}]$. Note that $T_c + h_f > \widetilde{T}_{P[n+1]/P[n+2]}$;

3. Extract the features from the history trajectories of EV and LCV in Phase n, and from the trimmed predicted trajectory of LCV in the time interval $[T_c, \widetilde{T}_{P[n+1]/P[n+2]}]$;

4. Apply phase-based cut-in event prediction and risk estimation based on the feature, and be given the predicted event and estimated risk.

The following sections are presented in the following way. First, we introduce the phase-based splitting method. Then, the feature extraction including the necessary analysis is presented. Later, the design of event prediction and risk estimation is shown, which is followed by the results and discussion.

6.2 Phase-based splitting method

lane change is a complex action and cut-in is a kind of lane change. Splitting the lane change into several phases is able to decompose the maneuver into simple actions. The previous research of lane change [86] states that the lane change process can be divided into four segments based on the analysis of the steering wheel profile. An algorithm is proposed to split the cut-in into four phases, and the detailed criteria are shown in Table 6.1. Before lane change (P1) starts, 'P0' is defined for representing the time interval before the lane change. The P0 starts at 2.5 seconds before P0/P1, which represents that the driver starts the lane change event at the average time of 2.5 seconds before the lane change event is observed by other vehicles [87].

When P0 ends and P1 starts, the variable representing the deviation to the lane boundary is defined as Y_{dev}, with the current value recorded as d_{start}. In addition, the criteria of $\frac{2}{3}d_{\text{start}}$ utilized to split P1/P2, and P3/P4 can make four duration distributions relatively even. The duration distributions of four phases are shown in Fig. 6.2. This phase-based splitting method will be applied in the phase split of cut-in events.

6.3 Phase-based feature selection

After each trajectory is split into four phases, the dynamic features are extracted in each phase, shown in Table 6.2, with X, V_x, V_y, a_x, and a_y representing longitudinal distance, longitudinal velocity, lateral velocity, longitudinal acceleration, and lateral acceleration respectively. The THW and ψ represent THW and heading angle respectively.

Table 6.1: Phase split criteria

Time frame	**Criteria**
P0 starts	$T_{start} - 2.5$s
P0 ends/P1 starts	T_{start}
P1 ends/P2 starts	when $Y_{dev} = \frac{2}{3} d_{start}$ on the current lane
P2 ends/P3 starts	T_{cross}
P3 ends/P4 starts	when $Y_{dev} = \frac{2}{3} d_{start}$ on the target lane
P4 ends	T_{end}

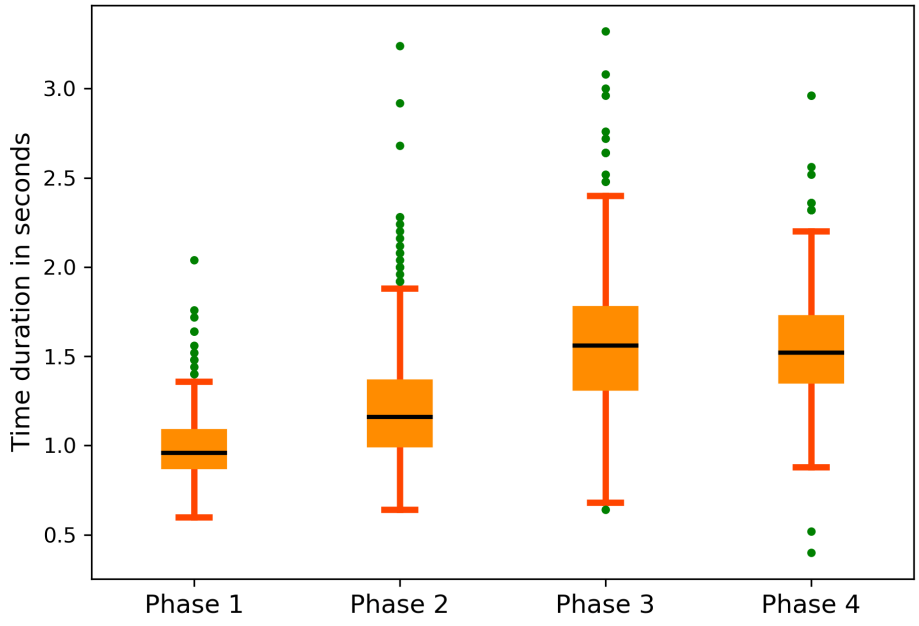


Figure 6.2: Box plot of phase duration

Features with ‘ ~ ’ (tilde sign) means that they are extracted from the predicted trajectory. Every feature is classified based on direction (including longitudinal, lateral, and yaw) and vehicle category (LCV-related, EV-related, interaction-related features). Within the extracted features, three interaction-related features are included. Since each extracted feature is a continuous time-series data, their maximum value, minimum value, and average values are calculated as the final set of features.

Table 6.2: Extracted features for deceleration analysis

	Longitudinal	Lateral	Yaw
LCV-related	V_x, a_x, $\widetilde{V}_x$, $\widetilde{a}_x$	V_y, a_y, $\widetilde{V}_y$, $\widetilde{a}_y$	ψ
EV-related	V_x, THW	\	\
Interaction-related	ΔX, ΔV, $\frac{\Delta X}{V_{x\ (\mathrm{EV})}}$	\	\

In this step, the features that can characterize the cut-in event are selected as the cut-in features for the later model design. Since the minimum acceleration of the EV (min $a_{\mathrm{x(EV)}}$) is an important factor to determine the cut-in event, this section shows the Pearson's correlation coefficient r of each feature in phase n and min $a_{\mathrm{x(EV)}}$ in phase *n+1*. The results are shown in Table 6.3. The signs ‘*’ and ‘**’ indicate that the significance (p-values) are at 0.05 and 0.01 level, respectively.

Among these 12 features, only eight features of phase n show the significant correlations with min $a_{\mathrm{x(EV)}}$ of phase *n+1*. These eight features are ranked out based on the average value of r: the average speed difference ($r = 0.3725$), the minimum value of speed difference over EV velocity ($r = 0.2582$), the minimum gap distance ($r = 0.2331$), the average LCV acceleration ($r = 0.2326$), the predicted average LCV velocity ($r = 0.2153$), the predicted minimum LCV acceleration ($r = 0.2075$), the average LCV velocity ($r = 0.1849$) and the minimum EV THW ($r = 0.1727$). These eight features are selected and going to be used for cut-in event prediction and risk estimation. The selected eight features are all longitudinal-based, including three interaction-related features. On the contrary, none of the lateral-based and yaw-based features are selected. This finding is consistent with the conclusion from [5], showing that the lateral features do not possess significant correlations with the future occurrences of the cut-in behavior.

Table 6.3: Pearson's correlation coefficients between the features in phase n and minimum $a_{\mathrm{x(EV)}}$ in phase *n+1*

	Average $a_{\mathrm{x(LCV)}}$	Minimum $\frac{\Delta X}{V_{\mathrm{x(EV)}}}$	Average ΔV	Minimum ΔX	Minimum $THW_{\mathrm{(EV)}}$
$P0-P1$	0.2295**	0.2339**	0.1490**	0.2372**	0.1934**
$P1-P2$	0.2555**	0.2607**	0.3958**	0.2193**	0.1438**
$P2-P3$	0.2019**	0.2730**	0.4717**	0.2307**	0.1531**
$P3-P4$	0.2436**	0.2653**	0.4735**	0.2450**	0.2213**
Average	0.2326	0.2582	0.3725	0.2331	0.1779

	Average $V_{\mathrm{x(EV)}}$	Average $V_{\mathrm{x(LCV)}}$	Maximum $V_{\mathrm{y(LCV)}}$	Maximum $a_{\mathrm{y(LCV)}}$	Maximum $\psi_{\mathrm{(LCV)}}$
$P0-P1$	-0.0502	0.1570**	0.0856	-0.0912	-0.0628
$P1-P2$	-0.1062	0.1515**	0.0554	0.0297	-0.0625
$P2-P3$	-0.1040	0.1931**	0.0217	0.0018	-0.1226*
$P3-P4$	-0.0662	0.2102**	-0.0784	-0.0798	-0.1953**
Average	-0.0817	0.1779	-0.0422	-0.0349	-0.1108

	Minimum $\widetilde{a}_{\mathrm{x(LCV)}}$	Minimum $\widetilde{a}_{\mathrm{x(LCV)}}$	Average $\widetilde{V}_{\mathrm{x(LCV)}}$	Minimum $\widetilde{V}_{\mathrm{x(LCV)}}$
$P0-P1$	0.2330**	-0.1243*	0.1786**	-0.1555**
$P1-P2$	0.1990**	-0.1378*	0.2284**	-0.1704**
$P2-P3$	0.1982**	0.0476	0.2214**	-0.1317*
$P3-P4$	0.2000**	0.0087	0.2327**	0.0058
Average	0.2075	-0.0515	0.2153	-0.1130

6.4 Online cut-in event predictor and risk estimator

To enable AVs to achieve online cut-in event prediction and risk estimation, this section presents the detailed design of the phase-based cut-in event predictor and the estimator. GMM is chosen as the machine learning model of the predictor and the estimator, due to its excellent representation properties.

6.4.1 Design of phase-based cut-in event predictor

To realize the phase-based online prediction, three sub-predictors are designed for predicting the cut-in event of phase 1, phase 2, phase 3, and phase 4 based on features of phase 0, phase 1, phase 2, and phase 3, respectively, with each constructed based on GMM. The GMM is a function that describes probability density parametrically. The structure of GMM is formed by an arbitrary number of weighted Gaussian probability density functions. The general form of GMM is shown as:

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) \ , \tag{6.1}$$

where $\mathbf{x}$ is a d-dimensional random variable, and $\mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)$ is a multivariate normal distribution with mean $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{\Sigma}_k$. K represents the number of mixed Gaussian kernels of GMM, and π_k is the weighting coefficient that satisfies:

$$\begin{cases} 0 \leq \pi_k \leq 1 \\ \\ \sum_{k=1}^{K} \pi_k = 1. \end{cases} \tag{6.2}$$

In the training phase, the EM algorithm iteratively consumes all pieces of training data and produces the model parameters [85]. The form of the input training data point is shown as:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_f & x_l \end{bmatrix}, \tag{6.3}$$

where

$$\mathbf{x}_f = \begin{bmatrix} a_{x(LCV)} & \frac{\Delta X}{V_{x_{(EV)}}} & \Delta V & \Delta X & THW_{(EV)} & V_{x(LCV)} & \widetilde{V}_{x(LCV)} & \widetilde{a}_{x(LCV)} \end{bmatrix} \tag{6.4}$$

is the vector, with including the features of the current phase n, and x_l is the predictor label of the next phase *n+1*, with the binary form of 1 for 'cut-in event' and 0 for 'normal lane change event'.

Before the training starts, there are three parameters need to be specified. The Gaussian kernel number K is to determine the complexity of GMM, while the convergence threshold h_{conv} and the maximum iteration J_{max} are to jointly determine the model accuracy and training speed.

The trained model with parameters π_k, $\boldsymbol{\mu}_k$, and $\boldsymbol{\Sigma}_k$, describes the joint mixture distribution of $\mathbf{x}_f$ and x_l. Cut-in event prediction is to compute the conditional distribution of estimated label $\widetilde{x}_l$ given a test feature vector $\mathbf{x}_f$ and trained GMM. The conditional distribution can also be written as:

$$\begin{aligned} p\left(\widetilde{x}_l|\mathbf{x}_f\right) &= \frac{p\left(\mathbf{x}_f, \widetilde{x}_l\right)}{\int p\left(\mathbf{x}_f, \widetilde{x}_l\right) d\widetilde{x}_l} \\ &= \sum_{k=1}^{K} \widetilde{\pi}_k \mathcal{N}\left(\widetilde{x}_l|\mathbf{x}_f, \widetilde{\mu}_k, \widetilde{\Sigma}_k\right), \end{aligned} \tag{6.5}$$

which is still the form of Gaussian mixture. The parameters of the conditional mixture are given as:

$$\widetilde{\pi}_k = \frac{\pi_k \mathcal{N}\left(\mathbf{x}_f|\boldsymbol{\mu}_{k,x_f}, \boldsymbol{\Sigma}_{k,x_f x_f}\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\mathbf{x}_f|\boldsymbol{\mu}_{j,x_f}, \boldsymbol{\Sigma}_{j,x_f x_f}\right)} \tag{6.6}$$

$$\widetilde{\mu}_k = \mu_{k,x_l} + \boldsymbol{\Sigma}_{k,x_l,x_f} \boldsymbol{\Sigma}_{k,x_f x_f}^{-1} \left(\mathbf{x}_f - \boldsymbol{\mu}_{k,x_f}\right) \tag{6.7}$$

$$\widetilde{\Sigma}_k = \Sigma_{k,x_l x_l} - \boldsymbol{\Sigma}_{k,x_l x_f} \boldsymbol{\Sigma}_{k,x_f x_f}^{-1} \boldsymbol{\Sigma}_{k,x_f x_l}, \tag{6.8}$$

where

$$\boldsymbol{\mu}_k = \begin{bmatrix} \boldsymbol{\mu}_{k,x_f} \\ \mu_{k,x_l} \end{bmatrix} \tag{6.9}$$

$$\boldsymbol{\Sigma}_k = \begin{bmatrix} \boldsymbol{\Sigma}_{k,x_f x_f} & \boldsymbol{\Sigma}_{k,x_f x_l} \\ \boldsymbol{\Sigma}_{k,x_l x_f} & \Sigma_{k,x_l x_l} \end{bmatrix} \tag{6.10}$$

is the partition of means and covariance of the GMM.

The mean and the covariance of the estimated label are specified as the full conditional distribution, which are given as:

$$\widetilde{\mu} = \sum_{k=1}^{K} \widetilde{\pi}_k \widetilde{\mu}_k \tag{6.11}$$

$$\widetilde{\Sigma} = \sum_{k=1}^{K} \widetilde{\pi}_k^2 \widetilde{\Sigma}_k \ . \tag{6.12}$$

To recover the predicted results as the categorical form instead of distribution, a threshold value of 0.5 is set and compared with the output mean value. If the estimated mean value $\widetilde{\mu}$ satisfies $\widetilde{\mu} \geq 0.5$, the prediction result is a cut-in event. r

6.4.2 Design of phase-based cut-in event risk estimator

The configuration of the phase-based cut-in event risk estimator is similar to the design of the aforementioned predictor, except that the label is risk scores, which is designed based on the human driver's cut-in response.

From the training data, the EV minimum acceleration min $a_{(\mathrm{EV})}$ of the next phase is extracted as the factor of the risk scores. The idea of the risk estimator is essentially to estimate 'what would be the acceleration of the human drivers applied if they encountered the same cut-in situation'. Showing as the human driver's longitudinal response, the minimum acceleration of EV has two merits as the risk scoring factor. First, it can imitate the human sense of the risk levels. When the AV considered interactions with the surrounding human-driven vehicles, the human-like risk outcomes would potentially assist the AVs to achieve better decision-making strategies. Second, the driver's response acceleration can also objectively reflect the risk level. According to the statistics of '100-Car Naturalistic Driving Study' by NHTSA [88], the extreme deceleration of the rear vehicle possesses a positive correlation with the rear-end accident rate. In summary, this scoring factor can reflect not only the subjective human-like risk level but also the objective risk level due to the correlation with the statistical accidental rate.

After the extraction, min $a_{(\mathrm{EV})}$ is normalized by a mapping function, shown as:

$$\mathit{Risk}\ \% = 1 - \frac{1}{1 + e^{-\alpha(m-\beta)}} \ , \tag{6.13}$$

where

$$m = \min\, a_{(\mathrm{EV})}, \tag{6.14}$$

and α, β are the parameters. It normalizes the minimum acceleration min $a_{(\mathrm{EV})} \in (-\infty, \infty)$ to the range (0, 1), with 0 for ideally no risk and 1 for the theoretically maximum risk. Based on the cut-in data, the values of α and β are determined as $\alpha = 2.031$ and $\beta = -0.92$, so that outcome risk scores of the least to the riskiest cut-in events are mapped from the value of 0.5 to 0.95. The score below 0.5 represents normal lane change events.

Due to the GMM, the estimated output is also a distribution form, so the output mean value is extracted as the estimated risk score.

6.5 Results and discussions

For the evaluation of the proposed predictor and the estimator, a Constant Acceleration(CA) model is introduced as the baseline cut-in event predictor and the baseline risk estimator for comparison. By assuming the acceleration constant, it estimates phase *n+1* as the cut-in event if the phase n is a cut-in event, with the equivalent risk scores.

6.5.1 Phase-based cut-in event predictor

To evaluate the phase-based cut-in event predictor, 478 numbers of lane change data, with the cut-in event labels included in each phase, are utilized for training and testing. Training for the GMM is done with $K = 75$ components, by using the aforementioned EM algorithm with numbers of iteration $J_{max} = 1000$ and convergence threshold value $h_{conv} = 0.001$.

The results of the proposed method and the baseline method are shown in Table 6.4. The better results are marked in bold. Comparing with the baseline, the proposed cut-in event predictor shows overall higher accuracy, precision, recall, and F1-scores. However, the recall values of 'phase 3 to phase 4' prediction are not ideal (88.9%), since the cut-in data is not balanced for training.

In Phase 0 to 1, both predictors show the perfect results. This does not approach to realistic because the cut-in cases in Phase 1 have only three cases out of 370. If the cut-in cases could be increased, the results would be more realistic.

Table 6.4: Phase-based cut-in prediction results of the proposed method and the baseline method

Phase 0 to phase 1

			Predicted results					
			Normal	Cut-in	Accuracy	Precision	Recall	F1-score
Proposed method	Ground truth	Normal	367	0	**100.0%**	**100.0%**	**100.0%**	**100.0%**
		Cut-in	0	3				
Baseline method	Ground truth	Normal	367	0	**100.0%**	**100.0%**	**100.0%**	**100.0%**
		Cut-in	0	3				

Phase 1 to phase 2

			Predicted results					
			Normal	Cut-in	Accuracy	Precision	Recall	F1-score
Proposed method	Ground truth	Normal	354	1	**99.5%**	**93.3%**	**93.3%**	**93.3%**
		Cut-in	1	14				
Baseline method	Ground truth	Normal	354	1	98.4%	91.0%	66.7%	76.8%
		Cut-in	5	10				

Phase 2 to phase 3

			Predicted results					
			Normal	Cut-in	Accuracy	Precision	Recall	F1-score
Proposed method	Ground truth	Normal	340	0	**99.7%**	**100.0%**	**96.7%**	**98.3%**
		Cut-in	1	29				
Baseline method	Ground truth	Normal	334	6	94.6%	72.2%	53.3%	61.5%
		Cut-in	14	16				

Phase 3 to phase 4

			Predicted results					
			Normal	Cut-in	Accuracy	Precision	Recall	F1-score
Proposed method	Ground truth	Normal	343	0	**99.5%**	**100.0%**	**88.9%**	**94.1%**
		Cut-in	3	24				
Baseline method	Ground truth	Normal	339	4	97.0%	83.3%	74.1%	78.4%
		Cut-in	7	20				

The false positive of the proposed predictor is small, meaning that it seldom fails to predict the cut-in scene as the normal lane change scene. It shows the conservative property of the predictor, as it has a small probability underestimating the risk. By comparison, the baseline shows larger false negative values of 'phase 1 to phase 2', 'phase 2 to phase 3', and 'phase 3 to phase 4', meaning that the proposed cut-in event predictor is proven to have better safety qualification in the real application.

6.5.2 Phase-based cut-in risk estimator

The phase-based cut-in risk estimator is trained by the GMM model that shares the same configuration as the predictor, with $K = 75$, $h_{conv} = 0.001$ and $J_{max} = 1000$. The metrics of the proposed method and the baseline method are measured by mean absolute error (MAE), as shown in Table 6.5. The better results are marked in bold under the same conditions. Generally, the performance of the proposed estimator outperforms the baseline. Under different conditions, the performance of the proposed method in the normal lane change event is better than that in the cut-in event because the training data for normal lane change event is more sufficient than the cut-in ones.

Next is to measure the performances of the proposed estimator at different risk levels. The scenes are separated into 15 subsamples based on the ground truths of their risk scores. Each subsample goes through the risk estimator and the corresponding MAE is calculated. Fig. 6.3 shows the MAE in different ranges of the ground truth in P0-P1, P1-P2, P2-P3, and P3-P4 respectively. The model shows better performance under the conditions ranging from 0 to 0.3, and after 0.3 the results become oscillating. One possible reason is that the data is imbalanced and more than half of the training data is focusing on that range. If more training data with balanced samples were offered, the performance of the risk estimator would be thereby improved.

Table 6.5: Mean absolute error of the risk estimator

		Mean absolute error (MAE)		
		Overall	Cut-in	Normal lane change
P0-P1	Proposed method	**0.0081**	0.0130	**0.0082**
	Baseline method	0.0246	**0.0064**	0.0247
P1-P2	Proposed method	**0.0104**	**0.0365**	**0.0093**
	Baseline method	0.0288	0.0553	0.0277
P2-P3	Proposed method	**0.0126**	**0.0263**	**0.0113**
	Baseline method	0.0517	0.1226	0.0455
P3-P4	Proposed method	**0.0140**	**0.0197**	**0.0135**
	Baseline method	0.0365	0.0884	0.0324

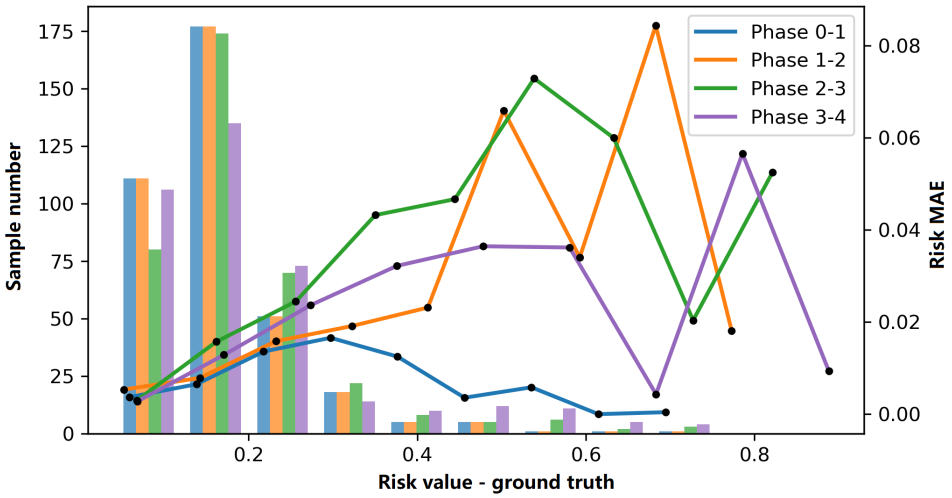


Figure 6.3: MAE in different ranges of the ground truth

Chapter 7

Conclusion and Future Work

The AVs are the future of transportation as they have the promising effect of increasing traffic safety and commuting efficiency. However current AVs can hardly understand the driving intention of the nearby human-driven vehicles in the mixed traffic, and it is difficult for them to foresee the risky driving behaviors, like cut-in behavior, from the nearby vehicles. To deal with the problem, this thesis proposes an interaction-aware cut-in event prediction and risk assessment framework based on naturalistic driving data. Moreover, it enhanced the application performances by achieving online prediction and risk assessment, considering the interaction between vehicles. The original contributions of this work are made and summarized as followings:

1. A novel phase-based method for cut-in analysis is proposed and implemented for event prediction and risk estimation. This novel method is used in online applications and is demonstrated to improve the safety of AV.

2. A Gaussian mixture model-based interaction-aware motion prediction algorithm is applied for predicting the future trajectory of lane change vehicles. The predicted outcome provides more features for the later cut-in event prediction and risk assessment, thus increasing the prediction accuracy.

3. Interaction-aware features extracted from the past trajectories and predicted future trajectories are selected for cut-in event prediction and risk assessment, and enhanced prediction and estimation performances are achieved compared to the baseline methods without considering the interaction. This extends the previous cut-in studies from local features of a single-vehicle to interaction-aware features that comprehensively considering the interaction process between vehicles, which is closer to reality.

To the best of the authors' knowledge, the proposed work is the first attempt so far to realize online cut-in event prediction and its risk assessment considering interactions. The proposed design enables autonomous vehicles to foresee the risky cut-in behavior of the front vehicles, and accordingly offers more reaction time to avoid or mitigate potential traffic accidents.

Despite the landmark achievements listed above, the research can be extended in three aspects in the future as followings:

1. The prediction and risk assessment performances can be improved by providing a larger volume of the naturalistic driving data. For motion prediction of lane change vehicles, we prove that the accuracy can be further increased using a larger amount of lane changing data. For the cut-in event prediction, the amount of cut-in scenes and normal lane change are not balanced, thus the results do not reach the optimal. Hence, providing a larger amount of lane change data and cut-in events data is the most straightforward way to enhance performances.

2. The proposed cut-in prediction and risk assessment approach can be applied to more complex scenarios. It can be the same lane change scene with more surrounding vehicles involved. It can also be similar scenes with more complex road geometry, such as multi-lanes highway or with merging ramp. The approach can also be further developed to handle the cut-in events on the urban driving scenes.

3. A lower-layer planning and vehicle control module of the AV can be expanded into the proposed framework, so that the risky cut-in scenes can not only be predicted but also can be actuated and avoided eventually for driving safety.

References

[1] National Highway Traffic Safety Administration. *Traffic Safety Facts: A Compilation of Motor Vehicle Crash Data from the Fatality Analysis Reporting System and the General Estimates System*. USDOT/National Highway Traffic Safety Administration, 2012.

[2] Transport Canada. Canadian motor vehicle traffic collision statistics: 2018, Jul 2020.

[3] Vijay Gill, Kirk Barrie, Godsmark Paul, and Flemming Brian. *Automated Vehicles: The Coming of the Next Disruptive Technology*. 2015.

[4] David Schrank, Bill Eisele, and Jim Bak. *2015 Urban Mobility Scorecard*. 2015.

[5] Chunqing Zhao, Wenshuo Wang, Shaopeng Li, and Jianwei Gong. Influence of Cut-In Maneuvers for an Autonomous Car on Surrounding Drivers: Experiment and Analysis. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–11, 2019.

[6] James J Gibson and Laurence E Crooks. A Theoretical Field-Analysis of Automobile-Driving. *The American Journal of Psychology*, 51(3):453–471, 1938.

[7] Society of Automotive Engineers. *Operational Definitions of Driving Performance Measures and Statistics*. 2016.

[8] Luca Mengani and Prashanth Dhurjati. Hazard Analysis and Risk Assessment and Functional Safety Concept. *D2.11 of H2020 project ENSEMBLE*, 2019.

[9] Xuesong Wang, Minming Yang, and David Hurwitz. Analysis of cut-in behavior based on naturalistic driving data. *Accident Analysis and Prevention*, 124(January):127–137, 3 2019.

[10] Yimin Chen, Chuan Hu, and Junmin Wang. Human-Centered Trajectory Tracking Control for Autonomous Vehicles with Driver Cut-In Behavior Prediction. *IEEE Transactions on Vehicular Technology*, 68(9):8461–8471, 2019.

[11] Maytheewat Aramrattana, Tony Larsson, Cristofer Englund, Jonas Jansson, and Arne Nabo. Simulation of cut-in by manually driven vehicles in platooning scenarios. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, 2018.

[12] Vicente Milanés and Steven E. Shladover. Handling Cut-In Vehicles in Strings of Cooperative Adaptive Cruise Control Vehicles. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, 20(2):178–191, 2016.

[13] Floris Remmen, Irene Cara, Erwin De Gelder, and Dehlia Willemsen. Cut-in Scenario Prediction for Automated Vehicles. *IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, pages 1–7, 2018.

[14] Xuehan Ma, Zhixiong Ma, Xichan Zhu, Jianyong Cao, and Feng Yu. Naturalistic Driving Behavior Analysis under Typical Normal Cut-In Scenarios. In *SAE Technical Paper*, pages 1–8, 4 2019.

[15] Guotao Xie, Hongmao Qin, Manjiang Hu, Daiheng Ni, and Jianqiang Wang. Modeling discretionary cut-in risks using naturalistic driving data. *Transportation Research Part F: Traffic Psychology and Behaviour*, 65:685–698, 2019.

[16] Sehwan Kim, Junmin Wang, Dennis Guenther, Gary Heydinger, Joshua Every, M Kamel Salaani, and Frank Barickman. Analysis of Human Driver Behavior in Highway Cut-in Scenarios. *SAE Technical Paper*, 2018.

[17] Jinwei Zhou, Pavlo Tkachenko, and Luigi Del Re. Gap acceptance based safety assessment of autonomous overtaking function. *IEEE Intelligent Vehicles Symposium (IV)*, pages 2113–2118, 2019.

[18] Zhiwei Feng, Xuehan Ma, Xichan Zhu, and Zhixiong Ma. Analysis of Driver Brake Behavior under Critical Cut-in Scenarios. *IEEE Intelligent Vehicles Symposium (IV)*, pages 2054–2059, 2018.

[19] Biao Wu, Zhiwei Feng, Xichan Zhu, and Zhixiong Ma. Analysis of Boundary Condition Model under Cut-in Scenarios Based on Logistical Regression Method. *SAE Technical Paper*, 2019-Novem(November):1–6, 2019.

[20] Kazi Iftekhar Ahmed. Modeling Drivers' Acceleration and Lane Changing Behavior. *PhD thesis*, page 189, 1999.

[21] Chunqing Zhao, Shaopeng Li, Wenshuo Wang, Student Memeber, Fenggang Liu, and Jianwei Gong. Influence Analysis of Autonomous Cars' Cut-In Behavior on Human Drivers in a Driving simulator. *IEEE Intelligent Vehicles Symposium (IV)*, pages 1–8, 2018.

[22] Stéphanie Lefèvre, Dizan Vasquez, and Christian Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH Journal*, 1:1, 2014.

[23] Jae Hwan Kim and Dong Suk Kum. Threat prediction algorithm based on local path candidates and surrounding vehicle trajectory predictions for automated driving vehicles. *IEEE Intelligent Vehicles Symposium (IV)*, pages 1220–1225, 2015.

[24] Seungje Yoon and Dongsuk Kum. The multilayer perceptron approach to lateral motion prediction of surrounding vehicles for autonomous vehicles. *IEEE Intelligent Vehicles Symposium (IV)*, pages 1307–1312, 2016.

[25] Dewei Yi, Jinya Su, Cunjia Liu, and Wen Hua Chen. Data-driven situation awareness algorithm for vehicle lane change. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 998–1003, 2016.

[26] Andreas Eidehall, Lars Petersson, and Monte Carlo. Threat assessment for general road scenes using Monte Carlo sampling. *IEEE Transactions on Intelligent Transportation Systems*, pages 1173–1178, 2008.

[27] Nachiket Deo and Mohan M. Trivedi. Multi-Modal Trajectory Prediction of Surrounding Vehicles with Maneuver based LSTMs. *IEEE Intelligent Vehicles Symposium (IV)*, pages 1179–1184, 2018.

[28] Dietmar Kasper, Galia Weidl, Thao Dang, Gabi Breuel, Andreas Tamke, and Wolfgang Rosenstiel. Object-oriented Bayesian networks for detection of lane change maneuvers. *IEEE Intelligent Vehicles Symposium (IV)*, pages 673–678, 2011.

[29] Thierry Wyder, Georg Schildbach, Stephanie Lefevre, and Francesco Borrelli. A Bayesian filter for modeling traffic at stop intersections. *IEEE Intelligent Vehicles Symposium (IV)*, pages 1252–1257, 2015.

[30] Stéphanie Lefèvre, Ashwin Carvalho, Yiqi Gao, H. Eric Tseng, and Francesco Borrelli. Driver models for personalised driving assistance. *Vehicle System Dynamics*, 53(12):1705–1720, 12 2015.

[31] Susanne Ernst, Jens Rieken, and Markus Maurer. Behaviour recognition of traffic participants by using manoeuvre primitives for automated vehicles in urban traffic. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 976–983, 2016.

[32] Stéphanie Lefèvre, Christian Laugier, and Javier Ibañez-guzmán. Risk Assessment at Road Intersections : Comparing Intention and Expectation. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 165–171, 2012.

[33] Junxiang Li, Bin Dai, Xiaohui Li, Xin Xu, and Daxue Liu. A Dynamic Bayesian Network for Vehicle Maneuver Prediction in Highway Driving Scenarios: Framework and Verification. *Electronics*, 8(1):40, 1 2019.

[34] Long Xin, Pin Wang, Ching Yao Chan, Jianyu Chen, Shengbo Eben Li, and Bo Cheng. Intention-aware Long Horizon Trajectory Prediction of Surrounding Vehicles using Dual LSTM Networks. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1441–1446, 2018.

[35] Aida Khosroshahi, Eshed Ohn-Bar, and Mohan Manubhai Trivedi. Surround vehicles trajectory analysis with recurrent neural networks. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 2267–2272, 2016.

[36] Alex Zyner, Stewart Worrall, and Eduardo Nebot. A Recurrent Neural Network Solution for Predicting Driver Intention at Unsignalized Intersections. *IEEE Robotics and Automation Letters*, 3(3):1759–1764, 2018.

[37] Donghan Lee, Youngwook Paul Kwon, Sara Mcmains, and J. Karl Hedrick. Convolution neural network-based lane change intention prediction of surrounding vehicles for ACC. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, 2018.

[38] Jürgen Wiest, Matthias Höffken, Ulrich Kreßel, and Klaus Dietmayer. Probabilistic trajectory prediction with Gaussian mixture models. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 141–146, 2012.

[39] Julian Schlechtriemen, Florian Wirthmueller, Andreas Wedel, Gabi Breuel, and Klaus Dieter Kuhnert. When will it change the lane? A probabilistic regression approach for rarely occurring events. *IEEE Intelligent Vehicles Symposium (IV)*, pages 1373–1379, 2015.

[40] Jurgen Wiest, Matthias Karg, Felix Kunz, Stephan Reuter, Ulrich Kresel, and Klaus Dietmayer. A probabilistic maneuver prediction framework for self-learning vehicles

with application to intersections. *IEEE Intelligent Vehicles Symposium (IV)*, pages 349–355, 2015.

[41] Jiachen Li, Hengbo Ma, Wei Zhan, and Masayoshi Tomizuka. Generic Probabilistic Interactive Situation Recognition and Prediction: From Virtual to Real. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 3218–3224, 2018.

[42] Erik Ward and John Folkesson. Multi-classification of Driver Intentions in Yielding Scenarios. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 678–685, 2015.

[43] Christian Wissing, Till Nattermann, Karl Heinz Glander, and Torsten Bertram. Probabilistic time-To-lane-change prediction on highways. *IEEE Intelligent Vehicles Symposium (IV)*, pages 1452–1457, 2017.

[44] R. Izquierdo, I. Parra, J. Munoz-Bulnes, D. Fernandez-Llorca, and M. A. Sotelo. Vehicle trajectory and lane change prediction using ANN and SVM classifiers. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, 2018.

[45] Guotao Xie, Hongmao Qin, Manjiang Hu, Daiheng Ni, and Jianqiang Wang. Modeling discretionary cut-in risks using naturalistic driving data. *Transportation Research Part F: Traffic Psychology and Behaviour*, 65:685–698, 8 2019.

[46] Mathieu Barbier, Christian Laugier, Olivier Simonin, and Javier Ibanez-Guzman. Classification of drivers manoeuvre for road intersection crossing with synthethic and real data. *IEEE Intelligent Vehicles Symposium (IV)*, pages 224–230, 2017.

[47] Friedrich Kruber, Jonas Wurst, and Michael Botsch. An Unsupervised Random Forest Clustering Technique for Automatic Traffic Scenario Categorization. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 2811–2818, 2018.

[48] Nima Mohajerin and Mohsen Rohani. Multi-Step Prediction of Occupancy Grid Maps with Recurrent Neural Networks. *arXiv*, 12 2018.

[49] Mohammad Bahram, Constantin Hubmann, Andreas Lawitzky, Michael Aeberhard, and Dirk Wollherr. A Combined Model- and Learning-Based Framework for Interaction-Aware Maneuver Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 17(6):1538–1550, 2016.

[50] Wenchao Ding and Shaojie Shen. Online Vehicle Trajectory Prediction using Policy Anticipation Network and Optimization-based Context Reasoning. *arXiv*, 2019.

[51] Nachiket Deo, Akshay Rangesh, and Mohan M. Trivedi. How would surround vehicles move? A Unified Framework for Maneuver Classification and Motion Prediction. *IEEE Transactions on Intelligent Vehicles*, 3(2):129–140, 2018.

[52] Jens Schulz, Constantin Hubmann, Julian Lochner, and Darius Burschka. Interaction-Aware Probabilistic Behavior Prediction in Urban Environments. *IEEE International Conference on Intelligent Robots and Systems*, pages 3999–4006, 2018.

[53] Frederik Diehl, Thomas Brunner, Michael Truong Le, and Alois Knoll. Graph Neural Networks for Modelling Traffic Participant Interaction. *arXiv*, 2019.

[54] Jiachen Li, Wei Zhan, and Masayoshi Tomizuka. Generic Vehicle Tracking Framework Capable of Handling Occlusions. *IEEE Intelligent Vehicles Symposium (IV)*, pages 936–942, 2018.

[55] Nemanja Djuric, Vladan Radosavljevic, Henggang Cui, Thi Nguyen, Fang-Chieh Chou, Tsung-Han Lin, and Jeff Schneider. Short-term Motion Prediction of Traffic Actors for Autonomous Driving using Deep Convolutional Networks. *arXiv*, 2018.

[56] Yeping Hu, Wei Zhan, and Masayoshi Tomizuka. Probabilistic Prediction of Vehicle Semantic Intention and Motion. *IEEE Intelligent Vehicles Symposium (IV)*, (Iv):307–313, 2018.

[57] Shengzhe Dai, Li Li, and Zhiheng Li. Modeling Vehicle Interactions via Modified LSTM Models for Trajectory Prediction. *IEEE Access*, 7:38287–38296, 2019.

[58] Florent Altché and Arnaud de La Fortelle. An LSTM Network for Highway Trajectory Prediction. *International Conference on Intelligent Transportation Systems*, pages 353–359, 2017.

[59] Byeoungdo Kim, Chang Mook Kang, Jaekyum Kim, Seung Hi Lee, Chung Choo Chung, and Jun Won Choi. Probabilistic Vehicle Trajectory Prediction over Occupancy Grid Map via Recurrent Neural Network. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 399–404, 2017.

[60] Seong Hyeon Park, Byeongdo Kim, Chang Mook Kang, Chung Choo Chung, and Jun Won Choi. Sequence-to-Sequence Prediction of Vehicle Trajectory via LSTM Encoder-Decoder Architecture. *IEEE Intelligent Vehicles Symposium (IV)*, pages 1672–1678, 2018.

[61] Xinli Geng, Huawei Liang, Biao Yu, Pan Zhao, Liuwei He, and Rulin Huang. A Scenario-Adaptive Driving Behavior Prediction Approach to Urban Autonomous Driving. *Applied Sciences*, 7(4):426, 4 2017.

[62] Hyo Sang Shin, Dario Turchi, Shaoming He, and Antonios Tsourdos. Behavior Monitoring Using Learning Techniques and Regular-Expressions-Based Pattern Matching. *IEEE Transactions on Intelligent Transportation Systems*, 20(4):1289–1302, 2019.

[63] Mohammad Bahram, Andreas Lawitzky, Jasper Friedrichs, Michael Aeberhard, and Dirk Wollherr. A Game-Theoretic Approach to Replanning-Aware Interactive Scene Prediction and Planning. *IEEE Transactions on Vehicular Technology*, 65(6):3981–3992, 2016.

[64] Katja Vogel. A comparison of headway and time to collision as safety indicators. *Accident Analysis and Prevention*, 35(3):427–433, May 2003.

[65] John C Hayward. Near miss determination through use of a scale of danger. 1972.

[66] Erik Coelingh, Andreas Eidehall, and Mattias Bengtsson. Collision warning with full auto brake and pedestrian detection - a practical example of automatic emergency braking. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2010.

[67] Jrg Hillenbrand, Andreas M. Spieker, and Kristian Kroschel. A multilevel collision mitigation approach—its situation assessment, decision making, and performance tradeoffs. *IEEE Transactions on Intelligent Transportation Systems*, 7(4):528–540, 2006.

[68] Masumi Nakaoka, Pongsathorn Raksincharoensak, and Masao Nagai. Study on forward collision warning system adapted to driver characteristics and road environment. *2008 International Conference on Control, Automation and Systems*, 2008.

[69] Georges S. Aoude, Brandon D. Luders, Kenneth K. H. Lee, Daniel S. Levine, and Jonathan P. How. Threat assessment design for driver assistance system at intersections. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2010.

[70] A Doi. Development of a rear-end collision avoidance system with automatic brake control. *JSAE Review*, 15(4):335–340, 1994.

[71] Jonas Jansson. *Collision Avoidance Theory: with Application to Automotive Collision Mitigation*. PhD thesis, 2005.

[72] Mattias Brannstrom, Jonas Sjoberg, and Erik Coelingh. A situation and threat assessment algorithm for a rear-end collision avoidance system. *IEEE Intelligent Vehicles Symposium (IV)*, 2008.

[73] A. Lambert, D. Gruyer, and G. Saint Pierre. A fast monte carlo algorithm for collision probability estimation. pages 406–411, 2008.

[74] Guotao Xie, Xinyu Zhang, Hongbo Gao, Lijun Qian, Jianqiang Wang, and Umit Ozguner. Situational assessments based on uncertainty-risk awareness in complex traffic scenarios. *Sustainability*, 9(9):1582, 2017.

[75] Matthias Althoff and Alexander Mergel. Comparison of markov chain abstraction and monte carlo simulation for the safety assessment of autonomous cars. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1237–1247, 2011.

[76] Jianqiang Wang, Yang Zheng, Xiaofei Li, Chenfei Yu, Kenji Kodaka, and Keqiang Li. Driving risk assessment using near-crash database through data mining of tree-based model. *Accident Analysis and Prevention*, 84:54–64, 2015.

[77] Donghoun Lee and Hwasoo Yeo. Real-time rear-end collision-warning system using a multilayer perceptron neural network. *IEEE Transactions on Intelligent Transportation Systems*, 17(11):3087–3097, 2016.

[78] Gregory Kahn, Adam Villaflor, Vitchyr Pong, Pieter Abbeel, and Sergey Levine. Uncertainty-aware reinforcement learning for collision avoidance. *arXiv*, 2017.

[79] Yang Li, Yang Zheng, Jianqiang Wang, and Keqiang Li. Threat assessment techniques in intelligent vehicles: A comparative survey. *IEEE Intelligent Transportation Systems Magazine*, pages 1–1, 2020.

[80] Peng Liu, Arda Kurt, Keith Redmill, and Umit Ozguner. Classification of Highway Lane Change Behavior to Detect Dangerous Cut-in Maneuvers. *Transportation Research Board, 95th Annual Meeting*, (January):1–14, 2016.

[81] Robert Krajewski, Julian Bock, Laurent Kloeker, and Lutz Eckstein. The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems. In *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 2118–2125. IEEE, 11 2018.

[82] U.S. Department of Transportation Federal Highway Administration. Next generation simulation (NGSIM) vehicle trajectories and supporting data: Department of transportation, Aug 2018.

[83] New York DMV. *Chapter 8: Defensive Driving*. New York DMV, Dec 2019.

[84] *The two-second rule*. Ireland Road Safety Authority, 2007.

[85] Jürgen Wiest, Matthias Höffken, Ulrich Kreßel, and Klaus Dietmayer. Probabilistic trajectory prediction with Gaussian mixture models. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 141–146, 2012.

[86] Wen Yao, Qiqi Zeng, Yuping Lin, Donghao Xu, Huijing Zhao, Franck Guillemard, Stephane Geronimi, and Francois Aioun. On-Road Vehicle Trajectory Collection and Scene-Based Lane Change Analysis: Part II. *IEEE Transactions on Intelligent Transportation Systems*, 18(1):206–220, 2017.

[87] Dario D. Salvucci and Andrew Liu. The time course of a lane change: Driver control and eye-movement behavior. *Transportation Research Part F: Traffic Psychology and Behaviour*, 5(2):123–132, 2002.

[88] Suzanne E Lee, Eddy Llaneras, Sheila Klauer, and Jeremy Sudweeks. *Analyses of rear-end crashes and near-crashes in the 100-car naturalistic driving study to support rear-signaling countermeasure development*. USDOT/National Highway Traffic Safety Administration, 2007.