공학박사 학위논문

교차로에서 자율주행 차량의 제한된 가시성과 불확실성을 고려한 종방향 거동 계획

Limited Visibility and Uncertainty Aware

Longitudinal Motion Planning of

Autonomous Vehicle at Intersection

2023년 2월

서울대학교 대학원

기계공학부

유 진 수

교차로에서 자율주행 차량의 제한된 가시성과 불확실성을 고려한 종방향 거동 계획

Limited Visibility and Uncertainty Aware Longitudinal Motion Planning of Autonomous Vehicle at Intersection

지도교수 이 경 수

이 논문을 공학박사 학위논문으로 제출함

2022년 10월

서울대학교 대학원

기계공학부

유 진 수

유진수의 공학박사 학위논문을 인준함

2022년 12월

위 원 장 _____ (인)

부위원장 _____ (인)

위   원 _____ (인)

위   원 _____ (인)

위   원 _____ (인)

Abstract

Jinsoo Yoo

Department of Mechanical Engineering

The Graduate School

Seoul National University

This dissertation presents a novel longitudinal motion planning of autonomous vehicle at urban intersection to overcome the limited visibility due to complicated road structures and sensor specification, guaranteeing the safety from the potential collision with vehicles appearing from the occluded region.

The intersection autonomous driving requires high level of safety due to congested traffics and environmental complexities. Due to complicated road structures and the detection range of perception sensors, the occluded region is generated in urban autonomous driving. The virtual target is one of the motion planning methods to react the sudden appearance of vehicles from the blind spot. The Gaussian Process Regression (GPR) is implemented to train the virtual target model to generate various future driving trajectories interacting with the motion of the ego vehicle. The GPR model provides not only the predicted trajectories of the virtual target but also the uncertainty of the future motion. Therefore, prediction results from GPR can be utilized to a position constraint for the Model Predictive Control (MPC), and the uncertainties are taken into account as a chance constraint in the MPC.

In order to comprehend the surrounding environment including dynamic

objects, a region of interest (ROI) is defined to determine targets of the interest. With the pre-determined driving route of the ego vehicle and the route information of the intersection, driving lanes intersecting with the ego driving lane can be determined, and the intersecting lanes are defined as ROI, reducing the computational load by eliminating targets of disinterest. Then the future motion of the selected target is predicted by a Long Short-Term Memory-Recurrent Neural Network (LSTM-RNN). Driving data for training are directly obtained with two different autonomous vehicles, providing their odometry information regardless to the limited field of view (FOV). For a widely known autonomous driving datasets such as Waymo and nuScenes, the vehicle odometry information are collected from the perceptive sensors mounted on the test vehicle. Thus, information of target that are out of the FOV of the test vehicle can't be obtained. The obtained training data are organized in the target centered coordinates for better input-domain adaptation and generalization. The mean squared error and the negative log likelihood loss functions are adapted to train and provide the uncertainty information of the target vehicle for the motion planning of the autonomous vehicle.

The MPC with a chance constraint is formulated to optimize the longitudinal motion of the autonomous vehicle. The dynamic and actuator constraints are designed to provide ride comfort and safety to drivers. The position constraint with the chance constraint guarantees the safety and prevent the potential collision with target vehicles. The position constraint for the travel distance over the prediction horizon time is determined based on the clearance between the predicted trajectories of the target and ego vehicle at every prediction sample time.

The performance and feasibility of the proposed algorithm are evaluated via computer simulation and test-data based simulation. The offline simulation

validates the safety of the proposed algorithm, and the suggested motion planner

has been implemented on an autonomous driving vehicle and tested in a real road.

Through the implementation of the algorithm to an actual vehicle, the suggested

algorithm is confirmed to be applicable in real life autonomous driving.

List of Figures

Table of Contents

Chapter 1. Introduction

1.1. Research Background and Motivation of Intersection Autonomous Driving
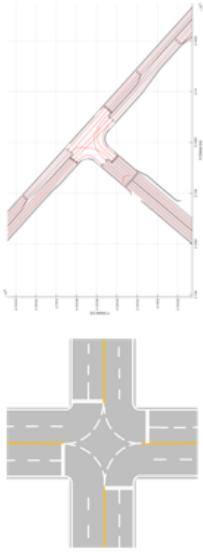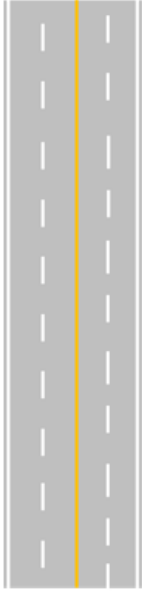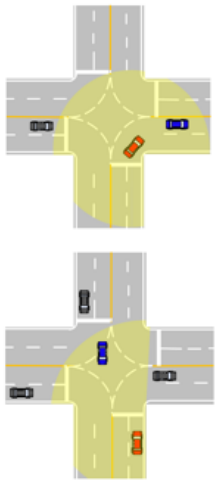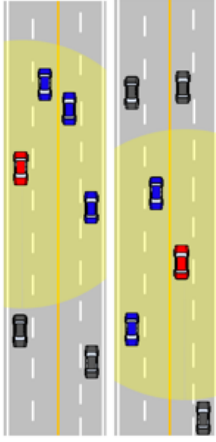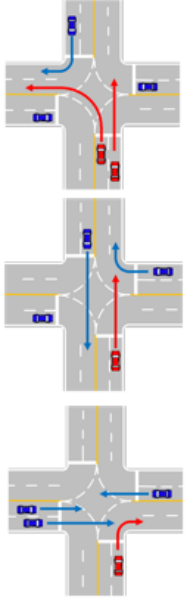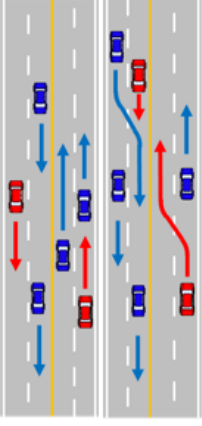
The autonomous driving has been rigorously studied and implemented in real vehicles. Leading automotive manufacturers such as Mercedes Benz, BMW, Volvo, Hyundai, Toyota, and Tesla have already adopted Advanced Driving Assistance System (ADAS) several years ago. Most companies developed highway autonomous driving system such as Highway Driving Assistance (HAD) and HDA2 from Hyundai Motor Company. Such highway autonomous driving system can be operated under the level 2 autonomous driving regulation due to simple road structures and simple driving patterns of surrounding vehicles.

As a higher level such as the level 3 or 4 of self-driving is being pursued these days, the Operational Design Domain (ODD) of autonomous driving has shifted to urban environment. Compared to the highway autonomous driving, the urban driving requires high level of safety due to the congested traffic and environmental complexity. The intersection is the most complicated urban road structures with the following challenges: traffic laws, dynamic surrounding objects, and complicated driving environments. The intersection traffic is controlled with various traffic laws such as stop and yield, and with different traffic signals. The dynamic surrounding objects require precise intention inference and their trajectory prediction to properly plan the motion of autonomous vehicle. The complex urban

environment can be categorized into adverse weather, light condition, and occlusion, which are critical in perceiving the surrounding.

The urban autonomous driving can be categorized into two: the normal autonomous driving, and the intersection autonomous driving. Although the intersection is part of the urban roads, significant differences exist between the urban intersection autonomous driving and normal self-driving. The differences between those two autonomous driving are described in the Table 1.

Table 1. Intersection vs. normal autonomous driving

	Urban Intersection	Urban Normal Road
Road Structure	- Multiple way (direction)	- Simple One-way or Two-way (direction)
Field of View	- **Corner blind spot** due to complex road structure	- No corner blind spot
Traffic Flow Direction & Interaction	- Target vehicle with **opposite** driving direction → **conflict region collision avoidance**	- Target vehicle with **same** driving direction → **clearance control**

The road structures of the urban intersection and normal road are different. The road structure of the intersection diverges to multiple directions or merges into one, but that of the normal road is a simple one-way or two-way direction. Due to the complex road structures and boundaries of the urban intersection, the field of view (FOV) is highly limited. Therefore, the autonomous vehicle should be extremely cautious when entering or changing the direction in the intersection by obtaining enough FOV for the safety. On the other hand, the FOV of normal autonomous driving is generally not obstructed by road structures other than severely curved road. The critical difference between the intersection and the normal urban autonomous driving is the traffic flow directions and traffic participants interactions. Because the roads of the intersection merges to one point from different directions, the traffic flows in the intersection variates in many directions, and major and minor traffic stream exist due to traffic rules. In extreme case, the target vehicle approaches to the intersection from the opposite direction of the ego vehicle. In the intersection, the conflict region is the critical index to maintain the safety and avoid collision. Therefore, in the intersection motion planning, it is significant to properly define the conflict region in the intersection. Unlike the intersection autonomous driving, the normal autonomous driving with simple road structures where target and ego vehicles driving in the same direction maintains the safety and prevent collision through clearance control.

Although different types of intersection exist, the major three different driving directions are defined in the intersection: left, straight, and right. The driving directions of the ego vehicle and other intersection traffic properties

depending on the ego direction are described in the table 2. When the ego vehicle is conducting the left turn, the ego vehicle has the right of way to cross the intersection, and detects target through frontal view. When the ego vehicle turns left, the vehicle may interact with the target vehicle turning right at the opposite road. During the left turn at intersection, no occlusion from static obstacles occurs. When the ego vehicle travels straight through the intersection, the ego vehicle also has the right of way in the traffic flow. The ego vehicle will interact with a target vehicle that turns right. The target vehicle can be detected with frontal view of the ego vehicle. The right turn in the intersection is the most difficult motion in the urban intersection. When the ego vehicle turns right, the ego has to yield to the vehicles that have the right of way in the intersection entering from the different directions. During the right turn, the ego vehicle should continuously observe the side directions to react to oncoming target vehicle with the right of way to cross the intersection. Due to the complex road structure of the intersection, the FOV of the ego vehicle is obstructed. The interaction between ego vehicle and target vehicle in the intersection is described in Figure 1.1.

Table 2. Ego vehicle direction and related properties in the intersection.

Ego Direction	Traffic Flow Priority	Target Detection Direction	Target Oncoming Direction	Occlusion from Road Boundary
Left	Right of way	Front	Right	No
Straight	Right of way	Front	Right	Yes
Right	Yield	Side	Straight	Yes
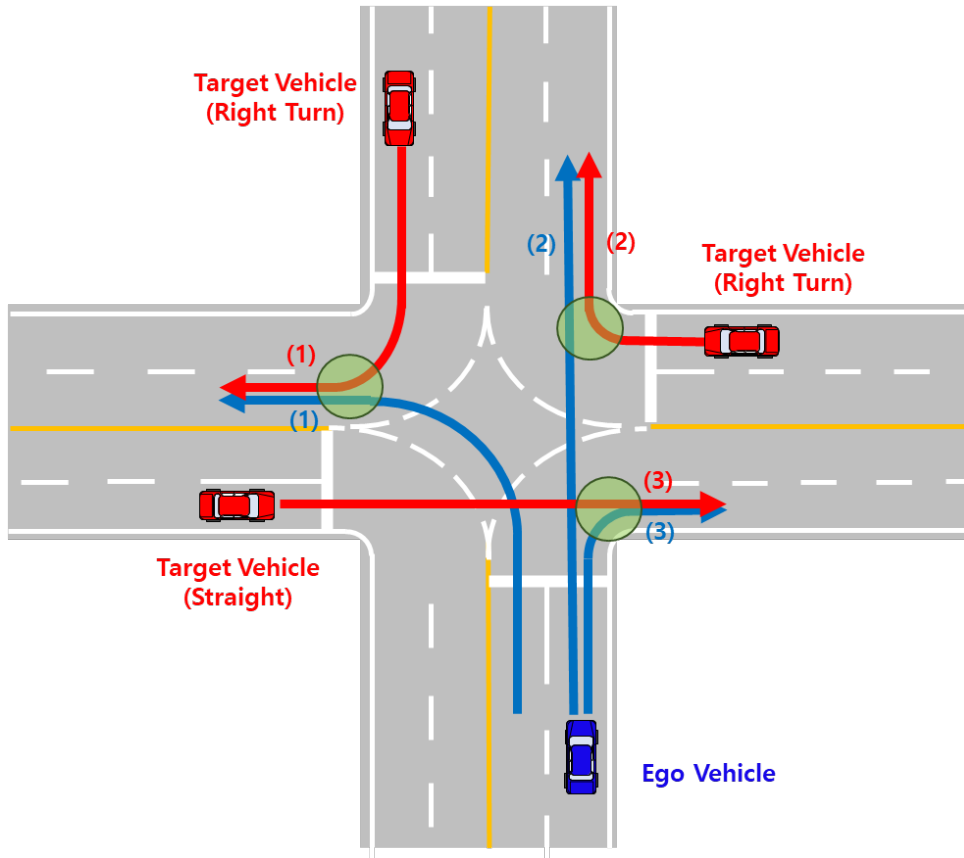
Figure 1.1. Possible vehicle interactions in the intersection

The most critical factor for not only the autonomous vehicle but also human driven automobile is the visibility. In the autonomous driving, the visibility of the self-driving vehicle is defined as FOV that how far and wide the automotive can perceive. Although human can acknowledge surrounding environment in wide range, the self-driving vehicles are limited in field of view when approaching to the intersection due to sensors' specification, mounting location of sensors on a vehicle, and road structures. Among various perceptive sensors providing a vision to an autonomous vehicle, a Lidar sensor is one of the most promising and commonly

adopted perceptive sensors. As shown in Table 3 below, channels of lidar sensor varies from 16 to 128 channels, and the 32 channel Lidar sensor is chosen by many research institutes and companies such as Google Waymo and General Motors Cruise. According to the sensor specifications, Lidar sensor provides the longitudinal measurement range from 80m to 300m. However, the maximum measurement range does not guarantee the detection and classification of road users such as vehicles, bicycles, and pedestrians. From Figure 1.1 below, the 16 channel Lidar sensor can detect the vehicle up to 25m, and the 32 channel Lidar sensor recognizes the vehicle up to 55m. Beyond the range, Lidar point clouds are sparsely detected, and it is hard to distinguish and classify as a specific object. Since such detection range of Lidar sensor is insufficient compared to that of human drivers, the autonomous vehicle motion is highly restricted in driving in intersection with complex road structures and blind spots. Such limitation can result in late detection of oncoming traffic participants, leading to a fatal accident.

From extensive reviews of preliminary studies, researchers have attempted various approaches to guarantee the safety of road users and provide comfortable riding quality by overcoming challenge of urban autonomous driving. The autonomous driving is categorized into four different major modules: the localization, the perception, the planning, and the control. Based on the odometry of ego and other traffic participants and local geometry information from upper modules, such as the localization and the perception, the planning module systematically predicts the surrounding situations and plans the proper ego behavior that guarantees safety in case of the unexpected obstacles and collisions

from the occlusion. Therefore, we aim to develop a motion planning algorithm for an autonomous vehicle in a complex urban environment, especially in the intersection, guaranteeing the safety and preventing collision from the sudden appearance of traffic participants from the occlusion.

Table 3. 3D Lidar Velodyne Sensor Specification

VELODYNE	Specification
Puck/Puck LITE	16 - channel Meas. : 80m FOV : +15~-15° Ang. Res : 2°
HDL-32E	32 - channel Meas. : 100m FOV : +11~-31° Ang. Res : 1.33°
Ultra Puck	32 - channel Meas. : 200m FOV : +15~-25° Ang. Res : 0.33°
HDL-64E	64 - channel Meas. : 120m FOV : +2~-24.8° Ang. Res : 0.4°
Alpha Prime	128 - channel Meas. : 180~300 FOV : +15~-25° Ang. Res : 0.11°
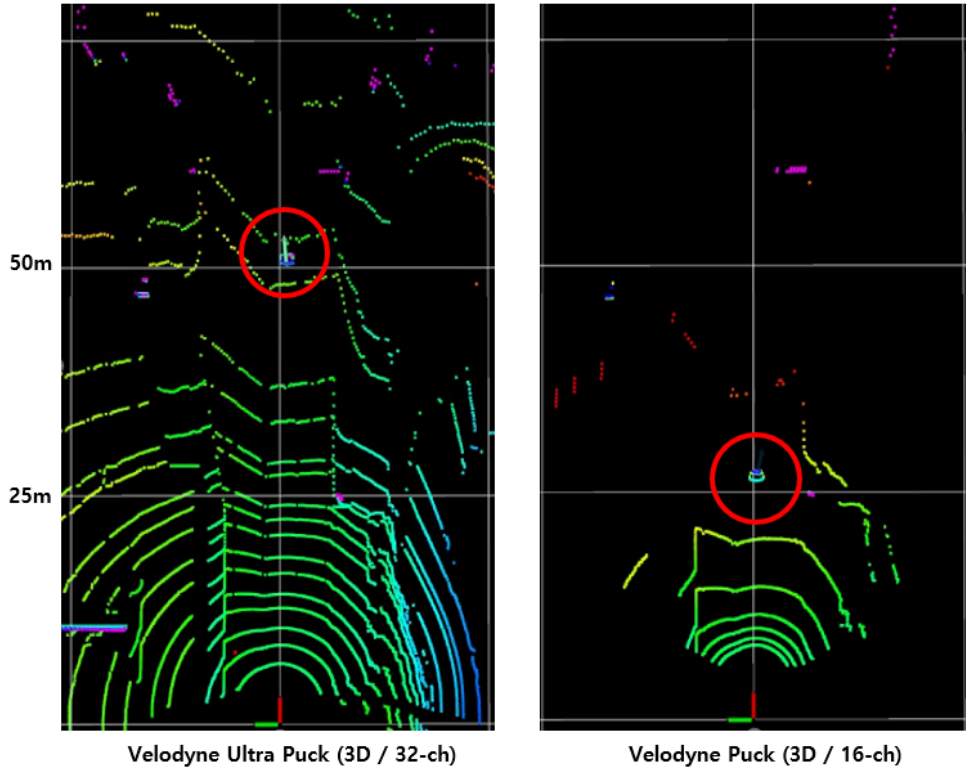
Figure 1.2. Detection range of Velodyne sensors.

1.2. Previous Researches on Intersection Autonomous Driving

Various methodologies of different autonomous driving modules have been employed to the self-driving vehicle to enhance ride comfort and guarantee the safety of self-driving at urban intersection. In motion planning, literature reviews have generally studied two specific autonomous driving technologies: Target trajectory prediction and ego motion planning. In the following sections 1.2.1 and 1.2.2., preliminary studies of surrounding vehicle motion prediction and motion planning for limited field of view are summarized.

1.2.1. Research on Trajectory Prediction and Intention Inference at Urban Intersection

In urban environment where road structures are complex, and velocity of traffic participants varies in wide ranges, a precise and accurate prediction of road users is critical in safety and collision prevention. Various trajectory prediction methods can be categorized into physics-based methods and learning-based methods. The most common physics-based methods that are widely utilized and compared as a basic prediction model are constant velocity (CV) and constant yaw rate and acceleration (CYRA) model [Prevost et al. 2007]. Jeong et al generated three different velocity profiles with Intelligent Driver Model (IDM) and predicted trajectories using Interactive Multiple Model (IMM) [Jeong et al 2019]. However, IMM based prediction can only estimate predefined predictive trajectories. Koschi et al utilized a reachable set for occupancy prediction of surrounding traffic participants instead of analyzing probability distributions of nearby vehicles, guaranteeing a full collision free safety [Koschi et al 2020]. However, such occupancy prediction does not reflect the future field of view as the ego vehicle proceeds and visibility changes.

In order to predict the future trajectory and contextual behaviors of surrounding vehicles delicately, learning-based approaches have been widely adopted in both a short-term and a long-term prediction horizon and proves enhance performance rather than physic-based prediction. Jeong et al implemented LSTM-RNN with predicted ego states from model predictive controller as additional inputs to predict interactive trajectories of vehicles at multi-lane turn

intersections [Jeong et al. 2020]. Deo et al proposed Convolutional Social Pooling LSTM model to predict the future trajectory of target vehicles where convolutional social pooling layer allows to capture the spatial interaction among vehicles [Deo et al. 2018]. Roy et al proposed Generative Adversarial Networks to predict the future trajectories of vehicles that captures the social interactions [Roy et al 2019]. Sheng et al proposed Graph-based Spatial Temporal Convolutional Network for trajectory prediction that graph convolutional network (GCN) defines the spatial interaction, convolutional neural network (CNN) captures the temporal features, and the features are encoded and decoded by a gaited recurrent unit (GRU) [Sheng et al. 2021]. Yoon et al implemented a Gaussian Process Regression (GPR) to predict lane change trajectories of nearby vehicles [Yoon et al. 2021]. Li et al designed a Graph-based Interaction aware Trajectory Prediction (GRIP) that incorporates the graph convolution extracts social interaction features and encodes and decodes features through LSTM to predict future trajectories [Li et al. 2019]. Most of these prediction results are computer simulated, and predictions are performed with environmental and dynamic features from space-fixed coordinate instead of coordinates from a self-driving car perspective. The space fixed coordinate is disadvantageous in vehicle implementation.

1.2.2. Research on Intersection Motion Planning

In urban autonomous driving, myriad researches have been developed to overcome the limited field of view due to occlusions. The approach can be subdivided into the two themes: communication and non-communication

autonomous systems. With the development of 5G network, the data transmission speed and load have enhanced dramatically, and the communication among autonomous vehicles and traffic infrastructures become possible, recently named as Cooperative-Intelligent Transport Systems (C-ITS). In C-ITS, Vehicle to Infrastructure (V2I), Vehicle to Vehicle (V2V), and Vehicle to Everything (V2X) communication systems have been introduced to overcome the sensor limitation and the blind spots in the intersection. Elleuch et al proposed a Cooperative Intersection Collision Avoidance Persistent system based on V2V and Real-time database system to reduce collision risks [I. Elleuch et al 2017]. In accordance with V2V, Sun et al introduced the Intersection Vehicle Fog (IVF) model to proactively plan motions at the intersection [G. Sun et al 2020]. Ci et al developed the vehicle following mode based V2I to enhance traffic efficiency and vehicle operation behaviors [Y.Ci et al 2019]. Duan et al designed the RGB Point-Voxel-Region based Convolutional Neural Network (PVRCNN) perception module to improve the acknowledgement of the surrounding object information comping with V2I system of road side Lidar sensor [X. Duan et al 2021]. Liu et al proposed the radar-based road side sensor communication to detect vehicles and vulnerable road users from the blind corner, and the support vector machine classifier is adopted to classify object from both sensors mounted on self-driving vehicle and road side radar sensor [W. Liu et al 2018]. The collision prediction of vehicles and vulnerable road users is measured based on their velocity, position, and heading. However, due to the characteristics of the radar, the multiple road users may cause large uncertainties and noise to detection and classification results. Jung et al

proposed V2X communication aided autonomous driving vehicle with three sub-modules: beyond line-of-sight (BLOS) perception, planning, and control [C. Jung et al 2020]. The BLOS perception module integrates local perception information from perceptive sensors mounted on vehicle and communication perception from V2X communication system and allows to overcome the limited visibility. Muller et al developed the motion planning framework with a sampling-based optimization method that adopts the external environmental model of V2I system to ego environmental model with perceptive sensors in prarallel instead of track to track (T2T) conventional fusion [J. Muller et al 2022]. The T2T fusion method is not effective by increasing the redundancy of detection and tracking number of targets in non-mutual field of view. However, these studies are only restricted to V2V and V2I environment, and most vehicles and road infrastructures in a real world are not equipped with such communication systems. Furthermore, different cities and countries do not offer the unified regulation and format of data transmission of the C-ITS for autonomous vehicles, causing interference when operating in different regions.

Other than communication-based solutions, a common strategy to overcome the occlusions that many preliminary studies implemented is to assume the virtual vehicle at the end of the field of view of autonomous vehicles. With a virtual vehicle coming from blind spots, the autonomous vehicle can properly plan the motion to prevent the potential collision. One of the most common methods to plan the motion with a phantom vehicle is the Partially Observable Markov Decision Process (POMDP). Because of the generality of POMDP, many

1 3

researches adopted the approach to optimize the occlusion-aware motion planning [P. Narksri et al 2021]. Hubmann et al proposed a POMDP based behavior planner with virtual vehicles that the probability of virtual vehicles represented in reachable set are determined with the traffic density at the occluded lane described as uniform probability distribution instead of assurance of existence of phantom vehicle at the frontier of the field of view [Hubmann et al. 2019]. Schorner et al designed the field of view calculated based on the vehicle sensor configuration accordance with the current and future environmental information and incorporate potential hidden traffic participants from the generated field of view with POMDP to plan the longitudinal motion [P. Schorner et al. 2019]. The probability of hidden vehicle is determined based on the traffic density of the occupying lane and distance to the vehicle. Brechtel et al. developed a continuous POMDP with a dynamic Bayesian network to determine a discrete finite set of acceleration and deceleration during the merging scenario at the intersection where the observation model of POMDP solved with the Monte Carlo Value Iteration algorithm treats the hidden vehicle probability as Boolean [S. Brechtel et al 2014]. However, when the observation model encounters the non-visibility, the optimized policy enforces the stop mode to look behind the blind area. Zhang et al proposed a POMDP based behavior planner with a phantom vehicle whose probabilities were inferred based on road structures and pre-defined map information [C. Zhang et al. 2021]. The proposed POMDP is solved online with Monte Carlo tree with reachable set analysis. Lin et al suggested a POMDP based decision making through occluded intersection accounting both static and dynamic obstacles [Lin et al. 2019]. Thornton proposed a longitudinal

and lateral controller to guarantee the safety at the occluded crosswalk formulating POMDP that dynamic programming is implemented for control policy to control the longitudinal motion [Thornton 2018]. The model predictive controller is applied to generate lateral motion with the velocity scalar from the POMDP. Bouton et al designed a POMDP formulated with scene decomposition method to manage multiple traffic participants independently at scene with occlusion [M. Bouton et al. 2018]. Although the POMDP is a very promising methodology to resolve the occluded driving scenario at the complex urban road structures, the computational burden can be high to be solved in real-time, and the method is difficult to be implemented in a real vehicle. Moreover, most of the POMDP based researches assumed over-simplified predicted action of virtual targets such as approaching with the maximum constant velocity. However, under such assumption, the self-driving vehicle can remain stopped indefinitely, causing a dead-lock situation.

Deep learning-based approaches have been studied to overcome autonomous driving scenarios with blind spots. Among various deep learning approaches such as Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) and Generative Adversarial Network (GAN), many preliminary studies utilize Deep Reinforcement Learning to resolve the occlusions at the intersection. Morales et al proposed a data driven motion planning with an inverse reinforcement learning to reflect expert driving characteristics, guaranteeing safety motion at occluded intersection [Morales et al. 2018]. Isele et al generated creeping behaviors when approaching the occluded intersections with Deep Q Networks to

improve the safety and achieve visibility [Isele et al. 2018]. Kamran et al proposed risk-based reward function instead of collision based in order to generate reliable policies and motions three Deep Q-Networks [Kamran et al. 2020]. Kamran et al proposed a distributional RL framework with dynamic programming to learn adaptive policies and consider the maximum uncertainty, guaranteeing the safety and ride comfort [Kamran et al. 2021]. Instead of trajectory prediction of vehicles, Ren at el. proposed a novel deep learning model with three different loss functions to predict the occupancy map to predict unseen vehicles [Ren et al 2021]. Although such learning-based approaches can easily reflect human driving patterns, some of the researches only conducted simulation studies, and the relationship between the input and the output is inexplicable.

Another method for motion planning to overcome the occlusion is to design the potential risk of unobserved oncoming vehicles and incorporate the risk to optimize the motion of self-driving vehicle. The advantage of such model-based approach is the explicit relationship between the risk and output motion of the vehicle. Tas et al proposed the motion planning based on the environmental model where the sampling preview point along a center of ego driving corridor is implemented to up-do date offline map, and the motion planner optimizes a motion stochastically according to different driving modes with Intelligent Driving Model [Tas et al 2018]. Morales et al designed a motion planning algorithm at blind cornered intersection with expert driving data [Morales et al 2017]. With a geometry of an intersection, Yoshihara et al developed a longitudinal model that computes the maximum safety velocity at possible proceeding points and critical

longitudinal point and incorporate the velocity to the Frenet frame path planner to generate the optimal motion [Yoshihara et al. 2017]. Hoermann et al. proposed a dynamic grid-map based motion planning that optimizes the trajectory with the minimum cost incorporated with occluded objects predicted with pre-defined map-based velocity [Hoermann et al. 2017]. Naumann et al considered a probabilistic trajectory prediction of the potential object at an edge of a sensing filed to plan a safe, comfortable, and not overcautious motion [Naumann et al. 2019]. Orzechowsk et al. estimated the occupancy of the over approximations of potential occluded traffic participants with Kamm's circle and a velocity interval [Orzechowsk et al. 2018]. Yu et al predicted the risk of unobserved vehicles with the pre-defined road structures and incorporate the risk to optimize the trajectory [Yu et al. 2019]. Zhang et al proposed a game-theoretic based framework that plans the future trajectory to prevent collision with hidden vehicles from the occlusion [Zhang et al 2021]. Wang et al proposed aDynamic Bayesian Network based model for real-time estimation of a potential risks at blind corners and incorporated the risk to a longitudinal and lateral motion planning [Wang et al 2022]. Narksri et al proposed a deadlock-free planner of autonomous vehicle at urban intersection that model multiple velocity profiles and visibilities of hidden vehicles from human driving data [Narksri et al. 2021]. Jeong et al proposed Approach Phase and Risk Management Phase to maintain the maximum safety velocity of self-driving vehicle when approaching the intersection with limited visibility [Jeong et al. 2019]. However, most of previous researches assumes the worst-case scenario with a virtual vehicle approaching with a constant velocity of road regulation. However,

the constant velocity assumption may enforce the ego vehicle remains stoped indefinitely, causing "Dead-lock" situation.

1.3. Thesis Objectives

The dissertation aims to propose the longitudinal motion planning of autonomous vehicle at urban intersection to improve safety and prevent potential collision due to occlusions, coping with limited visibility and uncertainty. A simple way to prevent the collision due to a sudden appearance of oncoming vehicle is "stop and go" that the autonomous vehicle must stop first and then accelerate to pass through the intersection. However, in such case, the vehicle stops at every intersection even if no other traffic participants exist, leading to an inexplicable behavior for a following vehicles. Therefore, the proposed thesis develops main two modules for intersection motion planning for autonomous vehicle: Surrounding vehicle prediction and longitudinal motion planning

The surrounding vehicle prediction module estimates the future trajectories of the oncoming vehicles. In order to overcome perceptive sensors limitations and prevent collisions due to occlusions, the surrounding vehicle prediction module is subdivided into two sub-modules: Virtual target and real target. The virtual target module anticipates the future trajectories of virtual targets from the occlusion. The GPR method is implemented to measure the mean and covariance of predicted trajectories. The visibility index of virtual targets based on the road structure is designed as one of input features of the GPR. The mean and covariance of future trajectories of virtual target are implemented in formulation of stochastic model

predictive controller of the longitudinal motion planning. The real target module predicts the future trajectories of detected vehicles with LSTM-RNN. The training and test data for both GPR and LSTM-RNN are obtained from actual driving at the intersection.

With the future trajectories from surrounding vehicle prediction module, the longitudinal motion planning proactively plans the ego vehicle motion with stochastic model predictive control. The stochastic model predictive control is formulated to consider prediction uncertainties, actuator limits, and jerk to provide ride comfort and guarantee the safety from collisions.

1.4. Thesis Outline

This dissertation is organized in the following manner. In Chapter 2, the overall architecture of the proposed algorithm is described. The overview of software configuration is summarized as different modules. The hardware configurations of the test vehicle are described. Further, the vehicle test environment for evaluation of the proposed algorithm is introduced. The test environment is occupied with unregulated human driven vehicles.

In Chapter 3, the methodology of virtual target modeling and prediction is described. The proposed algorithm generates virtual targets at the end of the field of view that encounters target vehicle's driving routes. The field of view is defined by configuration of sensors mounted on a vehicle. Virtual targets are modeled using Gaussian Process Regression to measure the future trajectories. A visibility index measurement, one of input features for GPR, based on the road structures is

described. Actual human driving data are collected and used to train and validate the GPR model. Different effects of kernel functions of GPR are defined.

In Chapter 4, the detected surrounding vehicle prediction is introduced. For the actual vehicles, historical trajectories of the detected vehicles can be observed whereas the historical paths of virtual targets don't exist due to its passive generation based on change in field of view. The Long Short-Term Memory based Recurrent Neural Network is constructed to predict the future trajectories of detected vehicles. LSTM-RNN structures and train setup for trajectory predictions are described.

In Chapter 5, the longitudinal motion planning of autonomous vehicle is defined. In order to incorporate the future trajectories of target vehicles to proactively plan the ego motion, the model predictive control is formulated. The ego vehicle state prediction and reference states for MPC are described. The conflict region is the most critical index in intersection motion planning to prevent collisions and generated based on future trajectories of target and ego vehicles. Constraints of MPC to provide driver acceptance and safety from collision are defined.

In Chapter 6, the performance evaluation of the proposed algorithm is summarized. The GPR based virtual target model prediction accuracy is evaluated with mean absolute error, root mean squared error, and standard deviation. The GPR trajectory prediction is compared with a constant velocity model to describe the efficiency. MATLAB simulation is conducted to describe overcoming the deadlock situation. The LSTM based surrounding vehicle prediction accuracy is

described and compared with a constant velocity model. The stochastic model predictive controller is performed under MATLAB simulation. The vehicle test result under real urban intersection is described.

In Chapter 7, the conclusion and summary of the proposed longitudinal motion planning algorithm at urban intersection is presented, and future works are described.

Chapter 2. Overall Architecture of Intersection Autonomous Driving System

2.1. Software Configuration of Intersection Autonomous Driving

The autonomous driving system largely consists of perception, localization, motion planning, and control. From upper modules, perception and localization, the motion planning obtains global routes, vehicle odometry, and surrounding vehicle information and determines the desired acceleration. The software architecture of overall autonomous driving motion planner is described in Figure 2.1.
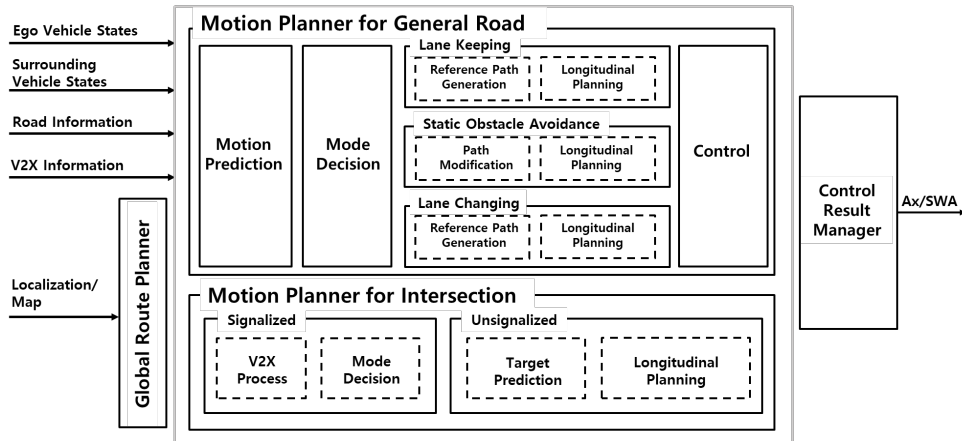
Figure 2.1. System overview of the autonomous driving motion planner.

As shown in Figure 2.1, the motion planner can be categorized into a general road and an intersection motion planner. Both two different motion planners

operate parallelly. The general road motion planner manipulates vehicle motions such as lane keeping, lane changes, overtaking, and biased driving. The detailed description of the intersection motion planning is presented in Figure 2.2.
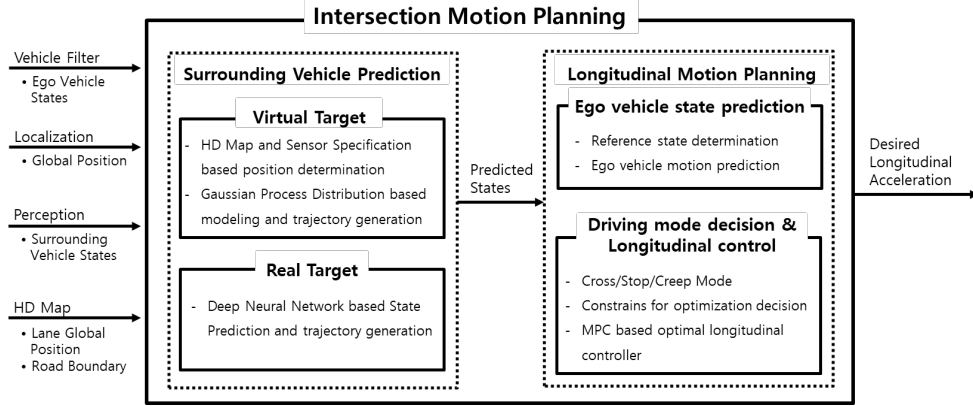
Figure 2.2. Intersection motion planning architecture

As described in Figure 2.2, the intersection motion planning is composed of surrounding vehicle prediction and longitudinal motion planning modules. In surrounding vehicle prediction module, two different target prediction sub-modules, virtual target and real target, perform parallelly. Based on the sensor specification and HD map information, a virtual target is generated, and future trajectories are predicted based on GPR. When real targets are observed, the future trajectories of the detected targets are calculated using LSTM-RNN. The virtual target submodule is activated when no oncoming target vehicle is observed, and deactivated when targets are detected in the intersection. The predicted future trajectories from surrounding vehicle prediction modules are utilized for the longitudinal motion planning. The ego vehicle longitudinal motion is optimized with a stochastic model

predictive control in the longitudinal motion planning module. The ego future trajectories and reference states are defined for the MPC. The future trajectories of both ego and target from preceding modules are computed to define the conflict region at the intersection and determine the position constraints of stochastic MPC to prevent collisions. The mean and covariance of virtual target future trajectories from GPR are implemented to manipulate the uncertainty of virtual target. Through the longitudinal motion planning, the desired longitudinal is determined to generate driving motion and ensure the safety at the intersection.

2.2. Hardware Configuration of Autonomous Driving and Test Vehicle

The test vehicle for the propose algorithm is a KIA Carnival, a multi-purpose vehicle. Two 32 channel 3-D Velodyne Lidars are mounted on the front and rear of the vehicle roof. The front vision camera is also mounted under a front glass for lane and traffic participants detection. OmniView around view camera for right and left lanes around the vehicle and Septentrio GPS are used to provide vehicle odometry through a localization module. Two industrial PCs, Nuvo 8108GC, are built in to operate perception and motion planning modules respectively. The desired motion calculated from the industrial PC is provided to a lower-level controller of a micro-autobox to operate actuators of the autonomous vehicle. Such hardware configurations for autonomous vehicle system and perceptive sensors detection ranges are depicted in figure 2.3.
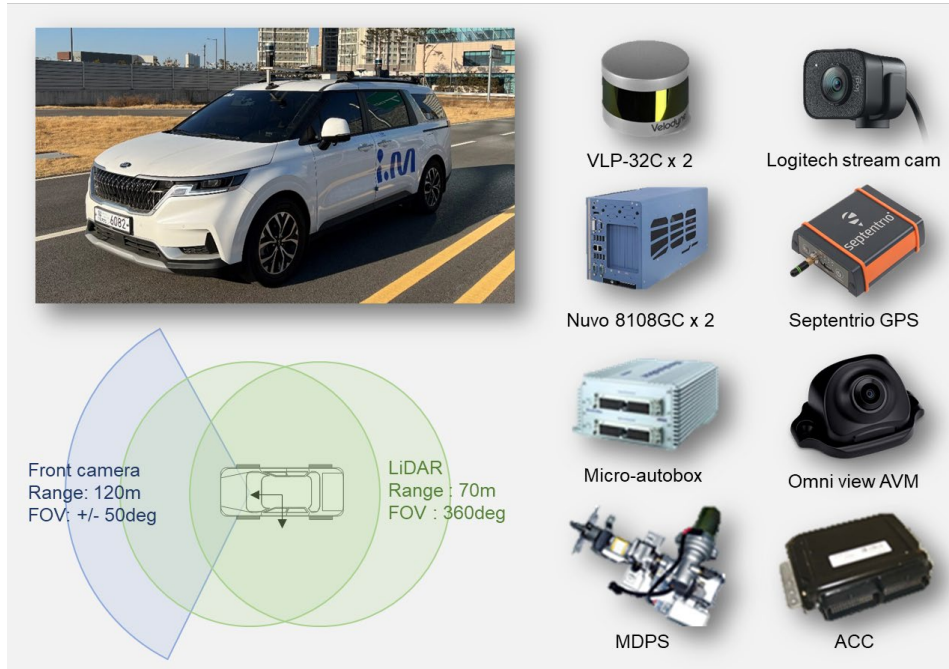
Figure 2.3. Configuration of sensors for autonomous vehicle.

2.3. Vehicle Test Environment for Intersection Autonomous Driving

The vehicle tests were performed in a real intersection. A full autonomous driving interacting with human drivers was conducted in Sangam, Seoul, South Korea. Sangam city is nominated as self-driving test city, and various autonomous vehicles from different research institutes conduct test driving and provide autonomous shuttle service. The driving course of a full autonomous driving is described in Figure 2.4.

Figure 2.4. Vehicle test course in Sangam, Seoul, South Korea

Chapter 3. Virtual Target Modelling for Intersection Motion Planning

3.1. Limitation of Conventional Virtual Target Model for Intersection

In many preliminary studies, the virtual targets are utilized as if a real vehicle is appearing from the occluded region at the intersections. Most conventional methods assume the virtual targets are approaching to the intersection with constant velocity and maximum road speed regulation. However, with the assumption of a constant velocity, the ego vehicle gradually decreases velocity to react to the virtual target and becomes stopped at the entering point of the intersection. However, when the ego vehicle reaches to a certain point at the intersection, the predicted trajectory of the virtual target enforces the ego remain stopped indefinitely, causing "Deadlock" situation as shown in Figure 3.1. In Figure 3.1, the predicted horizons of a virtual target and ego vehicles are assumed to be 3 seconds.

(a)

Target Vehicle

Virtual Target

*t = 1*s

*t = 2*s

*t = 3*s

Field of View

*t = 3*s

*t = 2*s

*t = 1*s

Ego Vehicle

(b)

*t = 3*s

Virtual Target

*t = 2*s

*t = 1*s

*t = 3*s

Field of View

Target Vehicle

*t = 2*s

*t = 1*s

Ego Vehicle
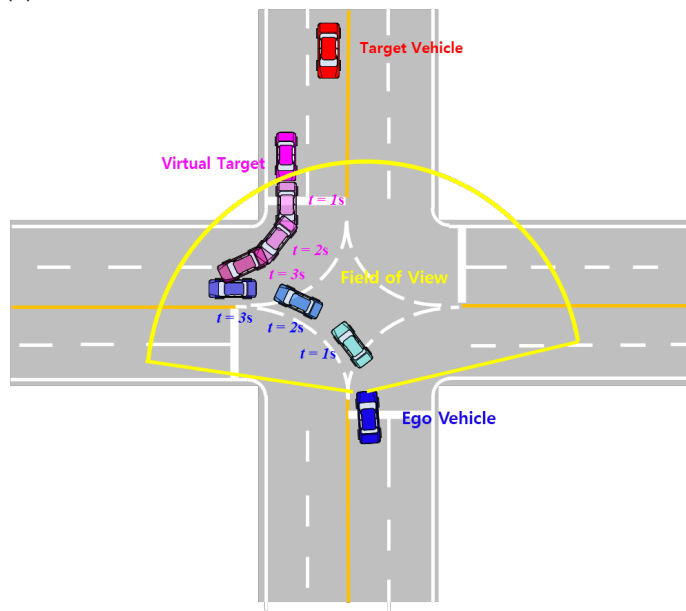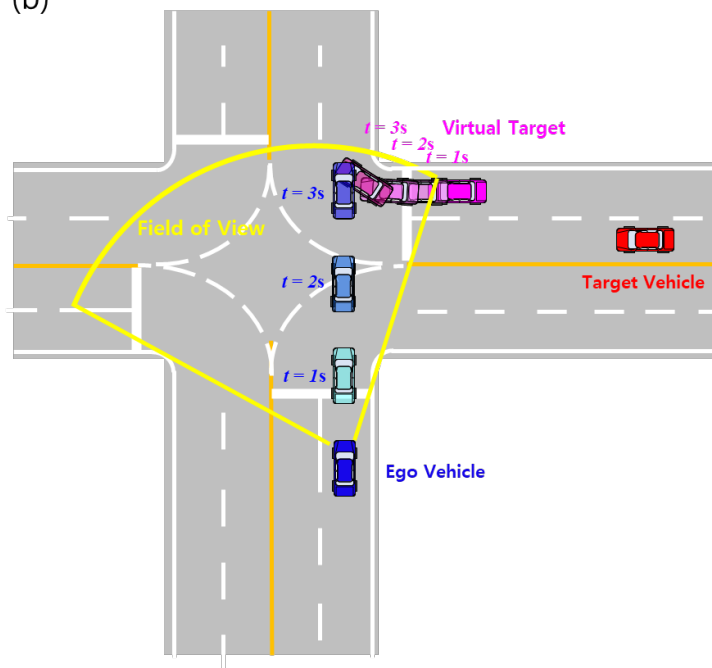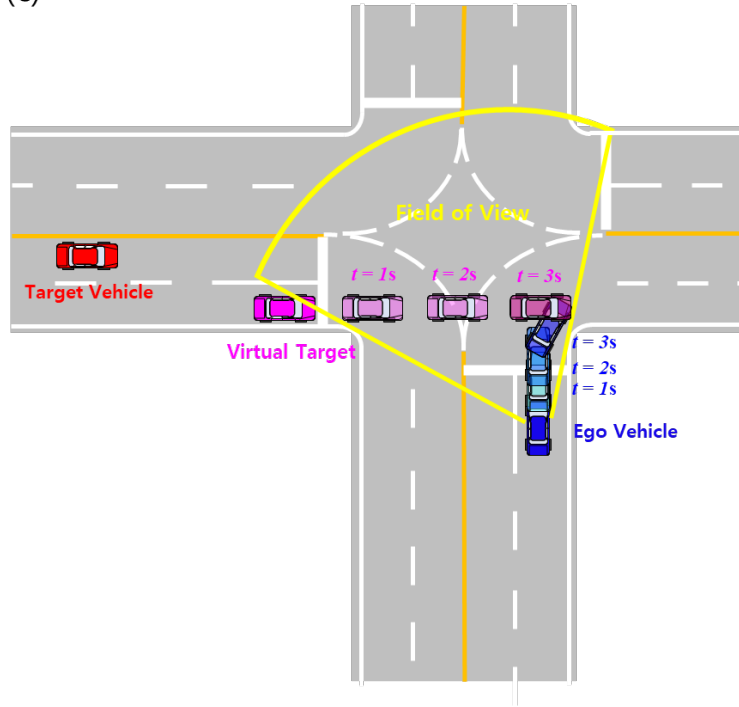
Figure 3.1. Predicted trajectories of virtual target with constant velocity in "dead-lock" situation (a) left turn (b) straight (c) right turn

Due to a constant velocity prediction of virtual target and fixed position from field of view of the ego vehicle, the conflict region is generated, and ego vehicle cannot cross the intersection. However, in real world driving, traffic participants interact with others and generate different motions such as yield motion instead of crossing, non-yield motion. Although a virtual target is not a real traffic participant, the interactive motion can be designed as a real vehicle. Human-driven data can be utilized to generate yielding motion trajectories as shown in Figure 3.2 below.

(a)

Target Vehicle

Virtual Target

*t = 1*s
*t = 2*s
*t = 3*s

Field of View

*t = 3*s *t = 2*s
*t = 1*s

Ego Vehicle

(b)

*t = 3*s Virtual Target
*t = 2*s
*t = 1*s

*t = 3*s

Field of View

Target Vehicle

*t = 2*s

*t = 1*s

Ego Vehicle

3 0

(c)
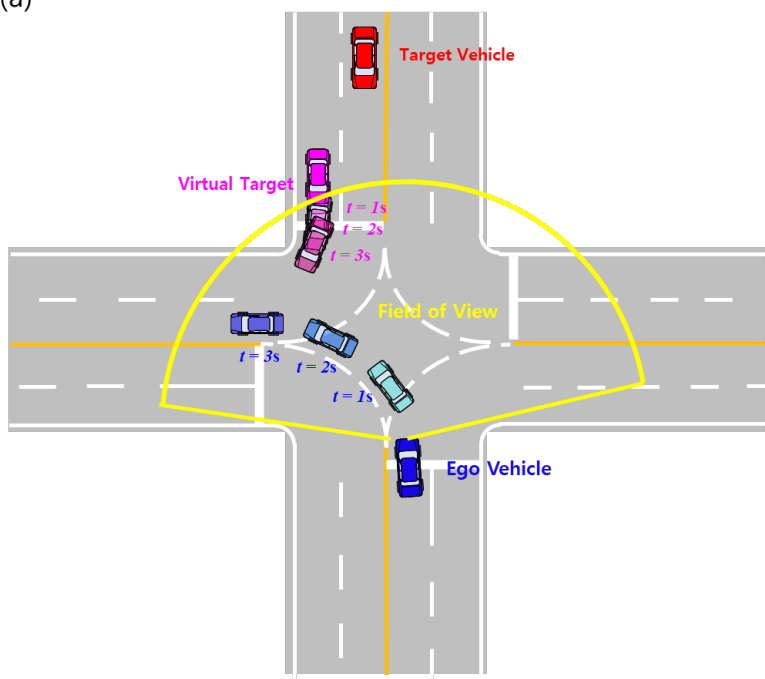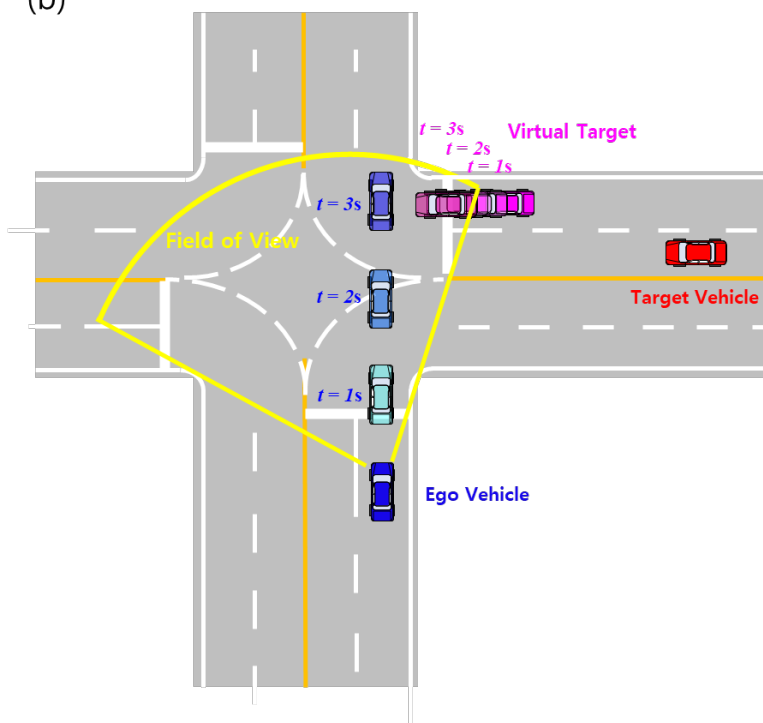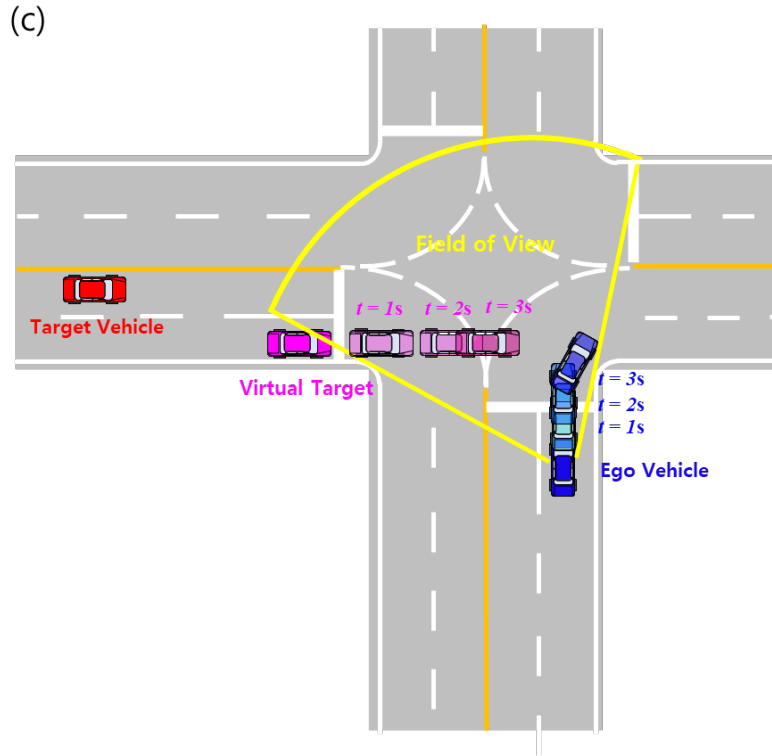
Figure 3.2. Predicted trajectories of virtual target with yield motion (a) left turn (b) straight (c) right turn

As shown in Figure 3.2, the trajectory of the virtual targets becomes shortened as ego vehicle approaches the intersection. Then, the sufficient clearance between future trajectories of the virtual target and ego vehicle allows the safe region for the self-driving vehicle to cross the intersection.

3.2. Virtual Target Generation for Intersection Occlusion

An autonomous vehicle based on local perceptive sensors is limited in motion planning due to blind spots caused by surrounding environments. Since an

autonomous vehicle perceives surrounding information by analyzing scene images and point clouds from the perceptive sensors, if rays from those sensors are obstructed by neighboring objects and road structures, it is difficult to fully identify the oncoming vehicles beyond the obstruction. The ego vehicle with limited perception range due to obstruction is difficult to properly plan ego motion in regard to an oncoming vehicle.

In this study, the concept of virtual target is introduced to overcome the limitation of perceptive range of sensors and reduce the potential collision risk. The virtual target generation based on HD map and cognitive sensor specifications are shown in Figure 3.3.
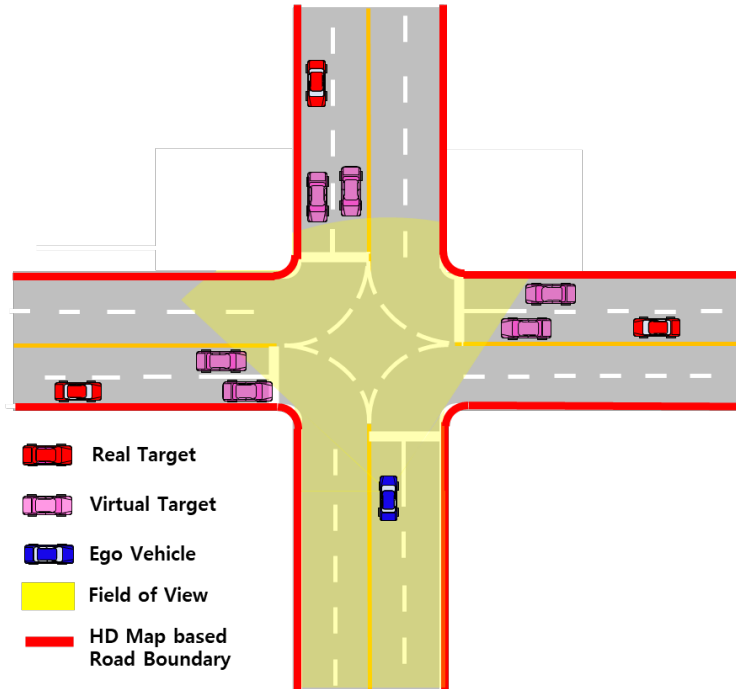
Figure 3.3. Virtual target generation with cognitive sensors and road boundary

Various perceptive sensors have different specifications in perception range. As shown in Figure 3.3, the field of view of the vehicle is defined based on perceptive sensor specifications such as resolution, point, density, and horizontal range. Without understanding the road boundary from the current position, the correct location of the occluded region cannot be recognized since it is ambiguous to recognize where the rays of LiDAR or Radar will be prevented. With the road boundary information from HD map, it is possible to define the correct FOV by exactly comprehending the borderline where sensor rays are blocked. When the FOV is generated based on road boundary, it is assumed that virtual targets exist at the crossing point between the limit of the generated FOV and the expected driving path of the virtual target. The autonomous ego vehicle assuming the existence of virtual target can proactively design motion planning and determine driving mode and control input, providing enhanced ride comfort.

As the ego vehicle proceeds toward the intersection, the virtual target is generated when no oncoming vehicles are detected, and ego vehicle has not yet reached to the point, where the full FOV for the target vehicle is allowed. Multiple virtual targets are generated as shown in Fig. 4 at the limit of the FOV obtained from the road boundary. The expected driving routes of each virtual target can be obtained with the geometric information of the intersection from HD map, and the primary virtual target for motion planning is determined by existence of a conflict point when interacted with the driving route of the ego vehicle. Among various virtual targets, the virtual target with the smallest TTCP is chosen as the primary to continuously plan and control the ego vehicle motion to either cross or stop while

maintaining the safety clearance until entering the intersection. For the safety driving of the ego vehicle, the virtual target is assumed to be approaching and crossing the intersection with the constant velocity of speed limit of the urban road.

3.3. Intersection Virtual Target Modeling

Various methods have been studied to model the vehicle motion in autonomous driving to mimic human driving behaviors. Rule-based and a learning-based methodologies are widely used two different categories to generate human-like driving patterns. Although a rule-based method can generate human-like driving motions, detailed and fine assumptions of behavior rules and parameters are required, and a generated motion is still inelegant and clumsy due to a complexity of other vehicle motion dynamics.

In this study, among various learning-based regression methods, the GPR is implemented to mimic human driving behaviors to generate "Yield" and "Cross" trajectories of virtual target vehicles as urban intersection, overcoming the deadlock situation. The LSTM-RNN is widely utilized when predicting sequential data such as future vehicle trajectories. Since LSTM-RNN method incorporates the historical data of the driving motion of the target vehicle, it provides accurate and precise predicted trajectories. However, the virtual targets in this study are generated based on the field of view of the sensors mounted on the vehicle and do not have historical trajectories due to a sensor field of view dependency instead of time historical sequential dependency. In other words, the virtual target does not move consecutively and is generated sensor and geometry dependent.

3.3.1. Gaussian Process Regression based Virtual Target Model at Intersection

The GPR is a powerful tool which is a data-driven non-parametric estimation method expressed in terms of mean and variance, providing the uncertainties of prediction results. A Gaussian process, by its widely known definition, is a collection of random variables, and those random variables have a characteristic of joint Gaussian distribution, multivariate Gaussian distribution [Rasmussen et al. 2006]. Gaussian distribution is a distribution of vectors and matrices, whereas Gaussian process is a distribution of functions. The Gaussian process is defined as mean and covariance functions, and mean and covariance functions are described as below,

$$m(x) = \mathrm{E}[f(x)] \tag{3.1}$$

$$\mathrm{cov}(x, x') = \mathrm{E}[(f(x) - m(x))(f(x') - m(x'))] \tag{3.2}$$

The covariance function of a gaussian process can be described as a kernel function as below,

$$\mathrm{cov}(x, x') = k(x, x') \tag{3.3}$$

The kernel function describes the similarity and correlation between two inputs as the covariance between outputs at the query inputs. According to the widely employed kernel functions, if the two input values are similar and close to each other, the covariance value of the outputs at the query inputs calculated becomes larger. However, for some kernel functions, the closer the input points are, the smaller the covariance is calculated between the outputs at the query inputs.

Therefore, the overall Gaussian process can be expressed as

$$f(x) \sim GP(m(x), k(x, x'))$$

(3.4)

The Gaussian Process Regression, GPR, is a GP based regression method and estimates the probability distribution of function value f(x*) at given input value x* and training dataset [Yoon et al. 2021]. The training dataset can be expressed as below,

$$D = \{(x^i, f^i) | i = 1, 2, \cdots, n\}$$

(3.5)

In the equation above, x and f denote the input and output of the dataset, and n represent the number of training samples. A joint Gaussian distribution of f and f(x*) with noise free observation can be described as follows,

$$\begin{bmatrix} f \\ f(x^*) \end{bmatrix} \sim N\left(\begin{bmatrix} m(X) \\ m(x^*) \end{bmatrix}, \begin{bmatrix} K(X,X) & K(X,x^*) \\ K(x^*,X) & k(x^*,x^*) \end{bmatrix} \right)$$

(3.6)

Introducing the noise term to the above equation, the joint Gaussian distribution can be re-written as below,

$$\begin{bmatrix} f \\ f(x^*) \end{bmatrix} \sim N\left(\begin{bmatrix} m(X) \\ m(x^*) \end{bmatrix}, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K(X,x^*) \\ K(x^*,X) & k(x^*,x^*) \end{bmatrix} \right)$$

(3.7)

where X, x*, and f are the corresponding training inputs, test/query inputs, and training outputs. From the above equation, the kernel matrix, K(X,X) can be expressed as,

$$K(X,X) = \begin{bmatrix} k(x_1, x_1) & \cdots & k(x_1, x_n) \\ \vdots & \ddots & \vdots \\ k(x_n, x_1) & \cdots & k(x_n, x_n) \end{bmatrix} \in \mathbb{R}^{n \times n}$$

(3.8)

It can be observed from the equation above, all the training input dataset pairs have corresponding covariance values. Therefore, the kernel matrix determines the predicted output mean and covariance value with the given query input. The predictive distribution of function value follows the Gaussian distribution, and mean and covariance can be described according to the conditional distribution of a multivariate normal distribution as below,

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim N\left(\begin{bmatrix} a \\ b \end{bmatrix}, \begin{bmatrix} A & C \\ C^T & B \end{bmatrix} \right) \Rightarrow x|y \sim N(a + CB^{-1}(y-b),\ A - CB^{-1}C^T) \quad (3.9)$$

The mean and covariance of the predictive distribution of the function value f(x*) are shown below,

$$\mu^* = m(x^*) + K(x^*, X)\left[K(X, X) + \sigma_n^2 I \right]^{-1} (f - m(X)) \quad (3.10)$$

$$\sigma^{*2} = k(x^*, x^*) - K(x^*, X)\left[K(X, X) + \sigma_n^2 I \right]^{-1} K(X, x^*) \quad (3.11)$$

With the equations above, the Gaussian process regression can conditionally determine the mean and covariance of output at given query test input using training datasets. In this study, a Gaussian process model is trained to estimate the distribution of behavior parameter, a distance from a conflict point to a target vehicle at predicted time of 5s.

$$f_{dtc_{tar,pred(5s)}} \sim GP_{dtc_{tar,pred(5s)}}\left(m_{dtc_{tar,pred(5s)}}(x), k_{dtc_{tar,pred(5s)}}(x, x') \right) \quad (3.12)$$

The predictive distribution of the distance to conflict point of target vehicle is measured as below,

$$dtc_{tar,pred}|D,\quad x^* \sim N(\hat{dtc}_{tar,pred}, \hat{\sigma}^2_{dtc_{tar,pred}}) \quad (3.13)$$

For Gaussian process regression five input features and one output feature is constructed. The input features are dtc_{tar} , v_{tar} , dtc_{ego} , v_{ego} , I_{vis}, which are distance to conflict point of target vehicle, velocity of target vehicle, distance to conflict point of ego vehicle, velocity of ego vehicle, and visibility index respectively. The output feature is $dtc_{tar,pred(t=5)}$ that is a distance to conflict point of target vehicle at predicted time of 5s.

3.3.2. Data Processing for Gaussian Process Regression based Virtual Target Model

The driver behavior modeling is significant to predict future motion accurately and prevent a potential collision. The driving dataset at the real urban intersection are collected and analyzed to construct the Gaussian Process model for a behavior parameter. The data are collected at two different urban intersections at Siheung, Gyeonggi-do, South Korea. The first intersection as shown in Figure 3.4 is at a test track of Future Mobility Test Center (FMTC). The intersection is a four-way intersection, and two ways have two entering lanes and two exiting lanes, and other two ways have one entering lane and one exiting lane. The distinct characteristic of the intersection at the test bed is that all traffic participants can have a full open field of view with no road boundary structures. The second intersection as shown in Figure 3.5 is a three-way T-shaped intersection outside the test track of FMTC. The all driving routes of intersection have two entering lanes and two exiting lanes. Compared to the first intersection at the open field, the second intersection is

located in the urban area with buildings and construction sites. Therefore, the vehicles approaching to the intersection with various driving routes have limited visibility due to road boundary with obstacles such as buildings and walls. Therefore, driving data from two different intersection affected by different field of view generate distinctive driving behaviors at the intersection. Two autonomous vehicles with perceptive sensors, industrial PCs, and GPS shown in Figure 3.6 are manually driven to collect the data and analyze the interactive motions. Each self-driving vehicle has four 16-channel Velodyne Lidar sensors, each mounted on each side of the vehicle. The vehicle states such as position and velocity are collected for data analysis.
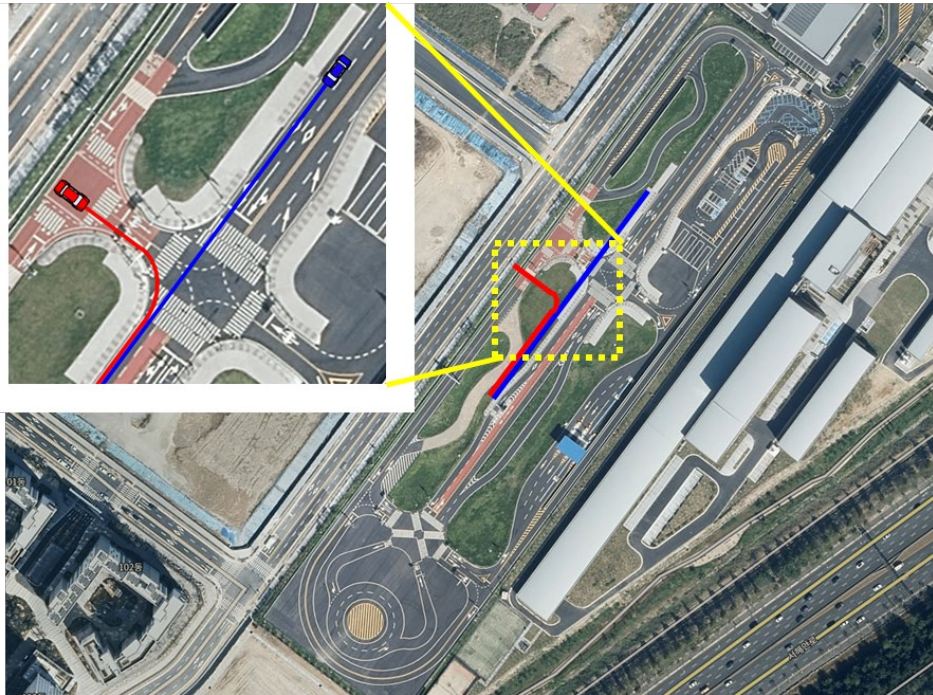
Figure 3.4. Satellite view of intersection in FMTC, Siheung, Korea
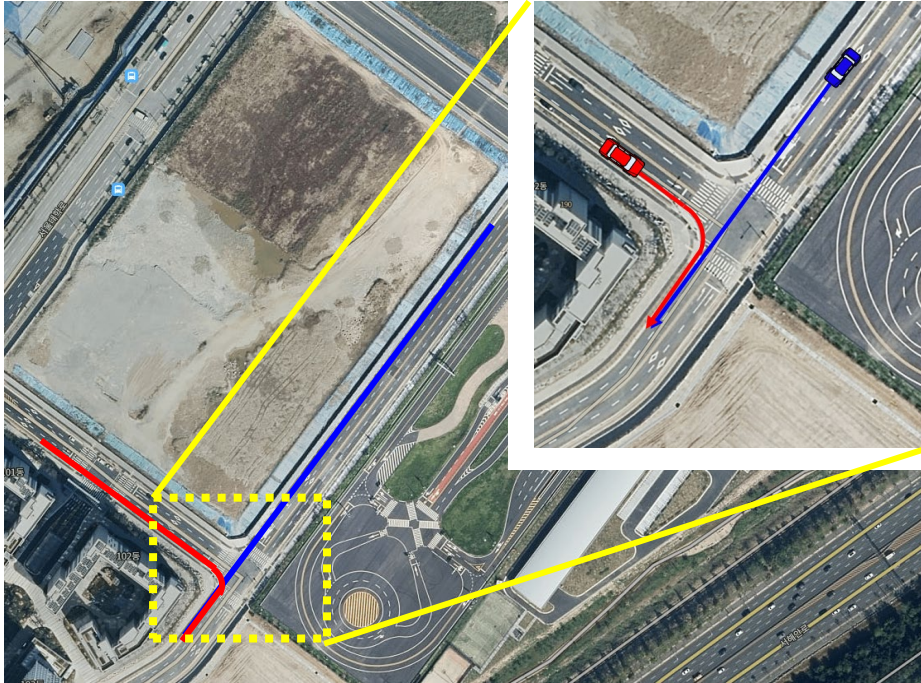
Figure 3.5. Satellite view of intersection outside FMTC, Siheung, Korea

Figure 3.6. Driving data collection vehicle sensor configuration

The recorded driving data has been analyzed in terms of distance to conflict point. In order to grasp all the interactive intersection crossing data of vehicles, the distance from the vehicle to the estimated conflict point is used to describe the position of the vehicles. The total 131 driving cases are obtained from 13 different drivers. The velocity profiles respect to the distance to the conflict point is shown in Figure 3.7. As shown in the obtained velocity profile, it is observed that oncoming target vehicle is approaching to the intersection with two different driving patterns: yield and cross. Since vehicles are driving at the urban intersection, it is assumed that target vehicle is driving at the maximum speed limit of 50km/h. From Figure 3.7, the initial velocity of the target vehicle variates within 40km/h to 60km/h. For the cross case, the vehicle tries to maintain the velocity and tends to decrease the speed to 40km/h at 20m before it reaches to the estimated conflict point. For the yield case, the vehicle starts to decelerate velocity to 10km/h around 50m before the conflict point and then accelerates after. There are no vehicles observed become full stop before the conflict point.
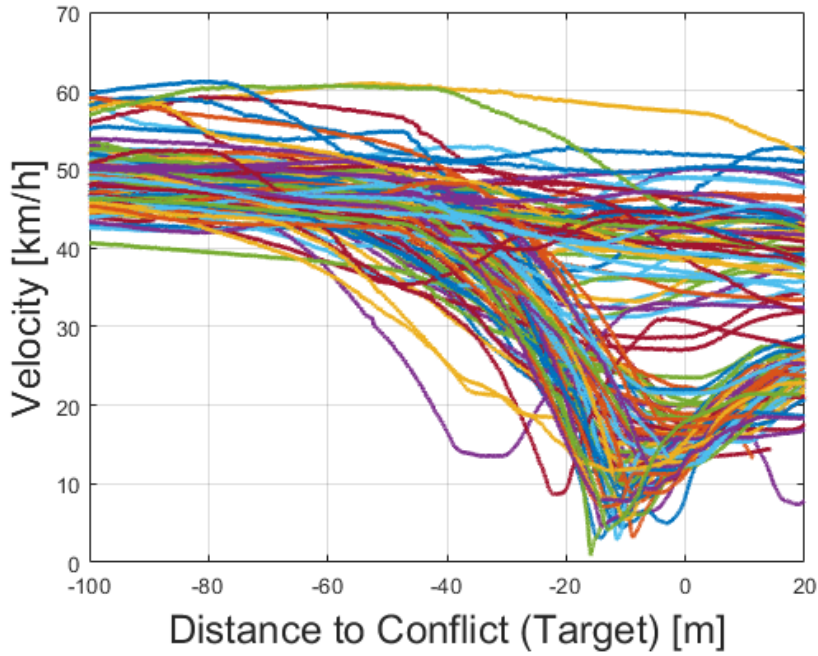
Figure 3.7. Velocity profile of the driving data at the intersection

The two different driving patterns, yield and cross, are plotted in 3-dimension of x and y domain along with time domain as depicted in Figure 3.8 and 3.9 in order to better describe the interaction between target and ego vehicle. In Figure 3.8, the target vehicle approaches to the intersection and crosses the conflict point without decreasing velocity, acknowledging the sufficient distance between target and ego vehicle. In Figure 3.9, the target vehicle gradually reduces the speed by acknowledging the potential risk of collision with the approaching ego vehicle from the other side.
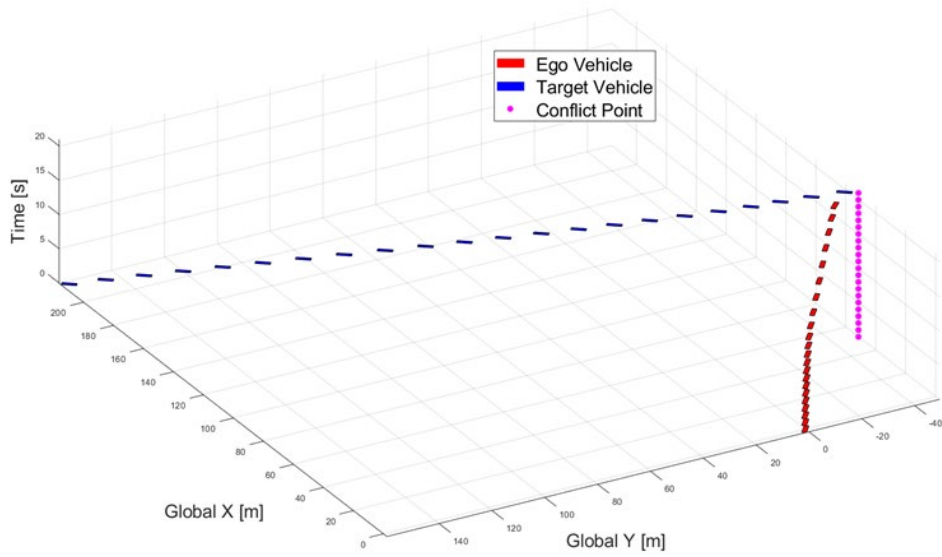
Figure 3.8. Driving trajectories of target vehicle crossing at intersection
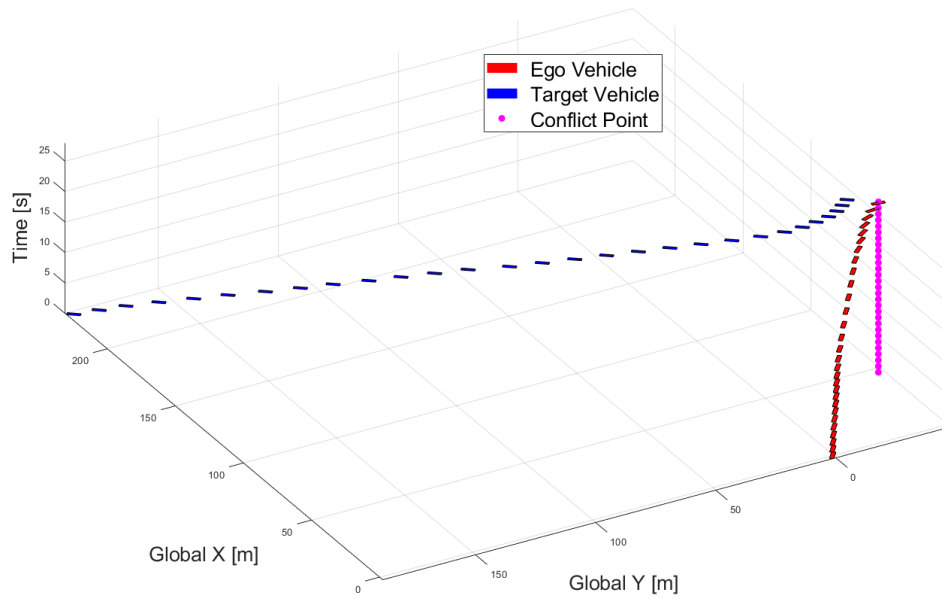
Figure 3.9. Driving trajectories of target vehicle yielding at intersection

Moreover, the driving data is analyzed in acceleration distribution in order to

4 3

check if driving dataset are obtained from drivers with different driving behaviors. The acceleration distribution is depicted as a histogram in Figure. 3.10. As shown in Figure 3.10, the diverse driving characteristics and no aggressive and naïve driving patterns are observed, and the mean and standard deviation of different drivers are described in Table 4.
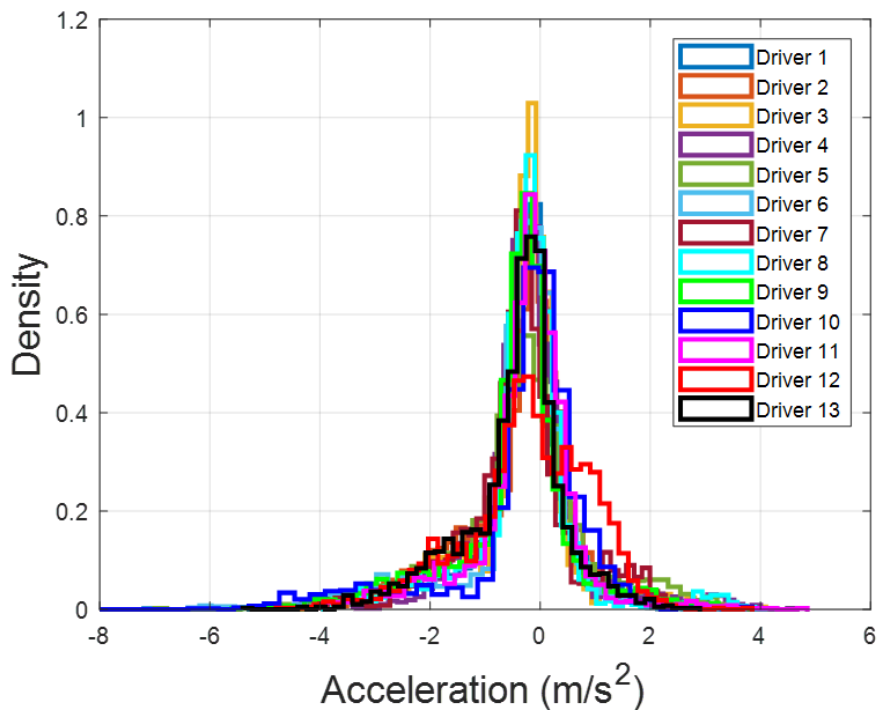
Figure 3.10. Acceleration histogram of 13 different human drivers

Table 4. The mean and standard deviation of acceleration of 13 different drivers

Driver	Mean [m/s²]	STD [m/s²]
1	-0.396	0.987
2	-0.376	1.012
3	-0.316	0.873
4	-0.180	0.919
5	-0.388	1.338
6	-0.455	1.103
7	-0.381	0.998
8	-0.292	1.001
9	-0.467	1.138
10	-0.326	1.281
11	-0.294	0.976
12	-0.214	1.147
13	-0.458	0.934

3.3.3. Definition of Visibility Index of Virtual Target at Intersection

The first step toward the prediction of virtual targets is to comprehend the characteristics of visibility generated by urban intersection road boundaries and obstacles. In order to cross an intersection safely, it is essential to secure the sufficient visibility and field of view. The wider a secured field of view is, the safer the autonomous vehicle can plan the motion in react to the other traffic participants within the visible area. From the perspective of target vehicle, the target vehicle

can also properly and interactively behave in response to the ego vehicle within the target vehicle's field of view. In urban driving, target vehicles driven by human have wider perceptive range than the autonomous vehicle. In this study, the idea that target vehicle has a wider field of view is utilized to develop a novel visibility index employed to design Gaussian process regression model.

The field of view of the autonomous vehicle at the urban intersection depends on various factors such as road boundary and buildings. In the intersection, traffic facilities such as sidewalks and crosswalks are constructed at the location where collisions among various traffic participants and vulnerable road users can be minimized. According to the Ministry of Land, Infrastructure and Transport (MOLIT), there exist a regulation that should be secured in designing and constructing an intersection. Since the visibility is the most significant factor for vehicles entering the intersection, the regulation is required for not only the road boundary size but also distance between road boundary and building construction as shown in Figure 3.11.

Figure 3.11. Safe intersection design from the Ministry of Land, Infrastructure and Transport

Considering the worst-case scenario, we designed the intersection with the regulation of road design parameters from the Ministry of Land, Infrastructure and Transportation for constructing visibility as shown in Figure 3.12.
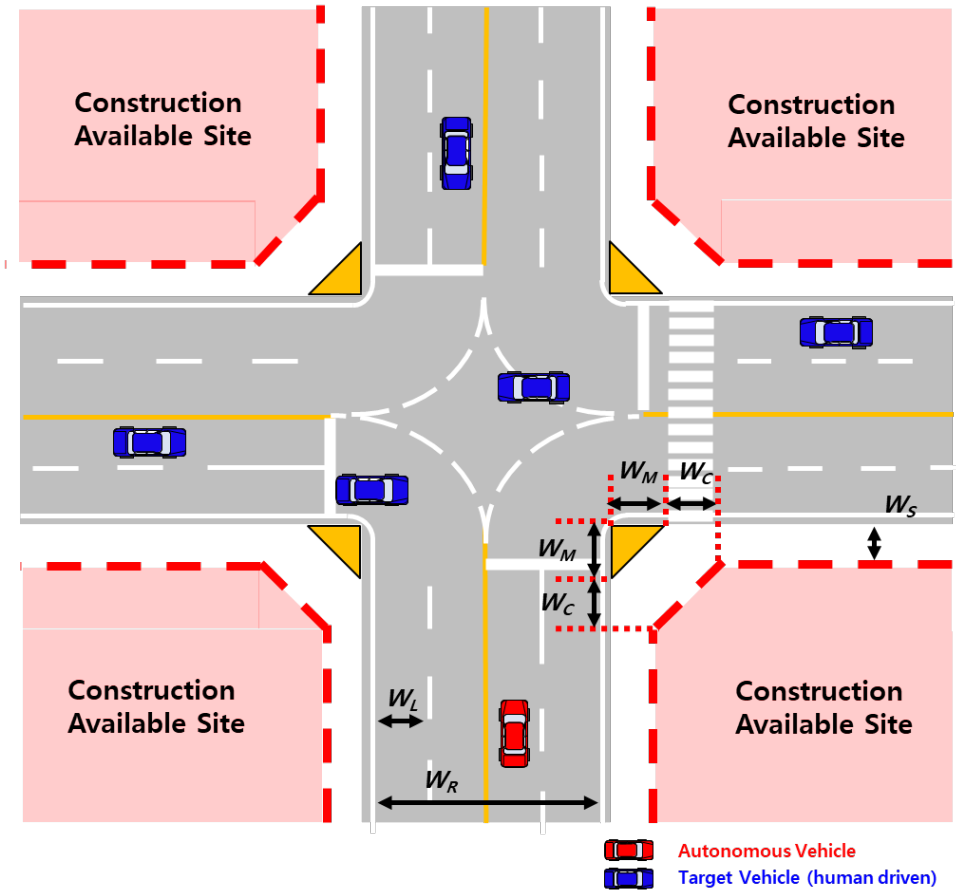
Figure 3.12. Design regulations for intersection and their parameters

According to the regulation by MOLIT, the road corner width, W_M, should be at least 6m in order for a vehicle to turn at the corner of the intersection. After the distance of a minimum road corner width, W_M, from the intersection corner, at least 2m of the crosswalk width, W_C, must be secured. A minimum 4m of a side walk width is required for pedestrians safely walking by the intersection. Buildings are not allowed to be constructed within the parameters of the road corner width, the crosswalk width, and the sidewalk width from the intersection road boundary. In

Figure 3.12, W_R is a road width, and W_L is a lane width. According to the MOLIT, the lane width should be at least 3m in not only intersection but also general urban road.

Considering the intersection and structures of traffic facilities, the difference of field of view between a human driven vehicle and an autonomous vehicle can be described as in Figure 3.13. In Figure 3.13, a red thick line and a translucent red colored area represents a visibility limit and field of view of an autonomous vehicle respectively. A blue thick border and a translucent blue colored region denotes a visibility limit and field of view of human driven target vehicle. In Figure 3.12, we will assume that pedestrians, street light, and street plants occupy sidewalks between thick red and blue colored border line. We can presume that the human driven vehicles are able to observe other traffic participants through the occupied region, and vehicles opposite the occupied area cannot be detected by an autonomous vehicle. In other words, if a target vehicle and an autonomous vehicle are positioned at the same distance from the intersection but at the different driving route, it is possible that the self-driving vehicle can be detected by the target vehicle and cannot observe the target vehicle. With such an assumption, a visibility distance shown in Figure 3.13 is determined depending on the location of oncoming target vehicle from the other driving routes of the intersection based on the road structure. In this study, the oncoming target vehicle will be a virtual target generated by field of view from an autonomous vehicle; therefore, the visibility of the virtual target is generated, and it is implemented as one of input features of the Gaussian process model for virtual target modeling.
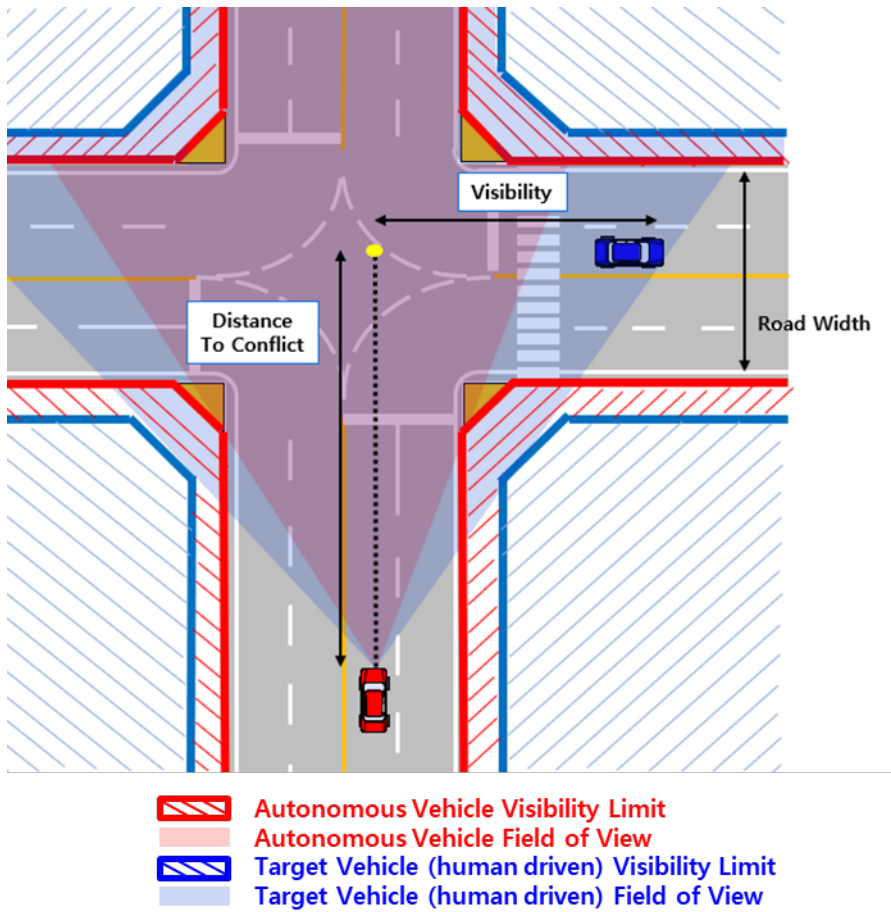
Figure 3.13. Field of view of autonomous vehicle and human driven vehicle

The proposed visibility profile can be determined in as a function of the road width and a distance to conflict point and is graphed in 3-dimension in Figure 3.14.

<Visibility Profile of Oncoming Vehicle>
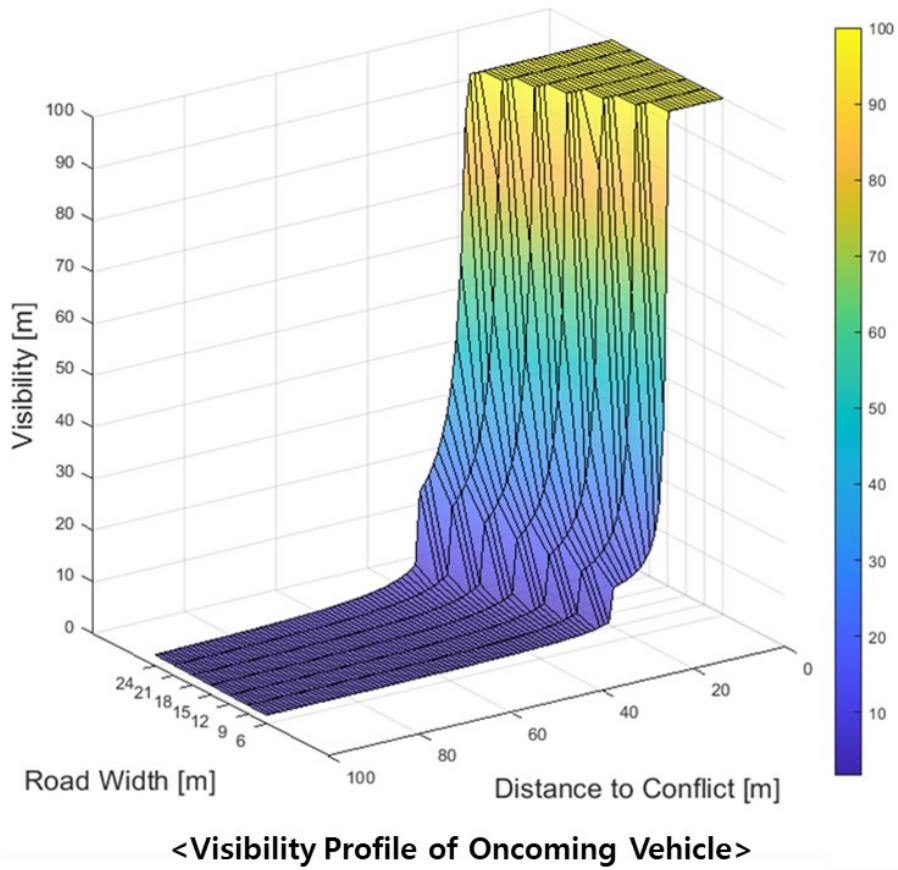
Figure 3.14. Visibility profile of oncoming target vehicle

3.3.4. Long Short-Term Memory based Virtual Target Model at Intersection

The Long Short-Term Memory Recurrent Neural Network is one of widely used learning methods to predict future trajectories of the moving objects. The LSTM-RNN utilizes the data within observation horizon as input. The characteristics of the LSTM is the sequential history input for prediction. On the

other hand, the GPR, one of supervised learning methods like LSTM, utilizes data at current/single time step as an input. The mean and variance of the prediction result can be obtained from the GPR method. As mentioned previously, the virtual target is generated at the edge of the FOV of the autonomous vehicle, and the position of the virtual target becomes further away as the FOV of the self-driving vehicle increases. Most importantly, since virtual target has no observed historical positions, it is advantageous to implement the GPR to predict future trajectory of the virtual target. In order to compare the effectiveness of the GPR and the LSTM based virtual target model, two different models are designed and described in the Table 5.

Table 5. GPR and LSTM based virtual target model

	GPR based Virtual Target	LSTM based Virtual Target
Input	$dtc_{tar}, \ v_{tar}, \ dtc_{ego}, \ v_{ego}, \ I_{vis}$	$p_{x,tar}, \ p_{y,tar}, \ v_{x,tar}, \ d_{x,tar}, \ \theta_{tar}$ $p_{x,ego}, \ p_{y,ego}, \ v_{x,ego}, \ \theta_{ego}$
Output	$dtc_{tar,pred(t=5)}$	$\hat{p}_{x,tar}, \ \hat{p}_{y,tar}$
Assumption	▪ Constant accel/deceleration over prediction horizon	▪ Constant velocity over observation horizon
Advantage	▪ No historical information required ▪ Variance of prediction	▪ Integrated with a real target prediction module → No need to separate virtual and real target modules ▪ Detailed position information (x & y)
Limitation	▪ Computational burden (Relatively compared to LSTM)	▪ No historical trajectory observed

The prediction results of the GPR based, LSTM based, CV based, and constant turn rate velocity (CTRV) based virtual target model are compared and analyzed in Table 6. From the Table 6, the GPR based virtual target model shows better prediction results over the prediction horizon. Figure 3.15 describes the histogram of prediction errors from the GPR based, LSTM based, CV based, and CTRV based models. From Figure 3.15, it is clearly observed that the histogram of the GPR based virtual target model is more centered at zero.

Table 6. Prediction errors over prediction horizons of different virtual target model

Case	Position Error at Predicted Steps (m) : MAE | RMSE | STD														
	1s			2s			3s			4s			5s		
GPR	1.46	2.06	1.62	2.20	3.15	2.46	2.36	3.36	2.68	2.03	2.91	2.57	1.71	2.80	2.80
CV	2.13	3.41	2.95	4.66	7.44	6.28	7.79	12.1	9.95	11.6	17.4	13.8	15.9	23.2	17.8
LSTM	2.55	3.52	2.42	4.63	6.54	4.62	6.34	9.28	6.78	7.57	11.4	8.55	8.45	13.0	9.88
CTRV	4.83	6.04	3.62	9.22	11.4	6.79	13.9	17.2	10.1	19.1	23.3	13.4	24.6	29.9	16.9
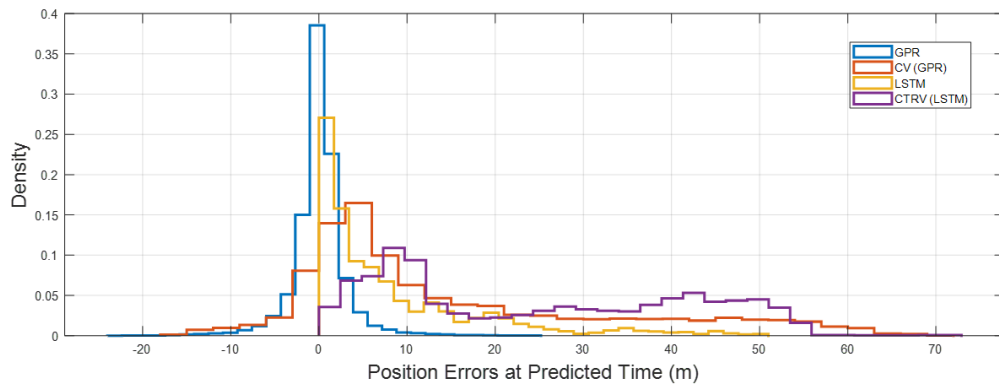
Figure 3.15. The histogram of prediction errors of four different virtual target models

Chapter 4. Surrounding Vehicle Motion Prediction at Intersection

4.1. Intersection Surrounding Vehicle Classification

In simple driving maneuvers such as lane change and lane keeping, the self-driving vehicle only needs to consider traffic participants driving in the same direction. In urban autonomous driving, especially in the intersection, traffic participants are driving in various directions. However, due to multiple crossing directions at the intersection, an autonomous driving at urban intersection should take account of greater degree of drivable area and field of view. However, with the HD map, it is possible to eliminate traffic participants not in interest. As shown in Figure 4.1, the HD map contains the information of path, node, and lane in the intersection. From the HD map, it is possible to define the enter and exit node and lane of the intersection, and future driving direction and routes of vehicles can be determined based on the current driving lane.
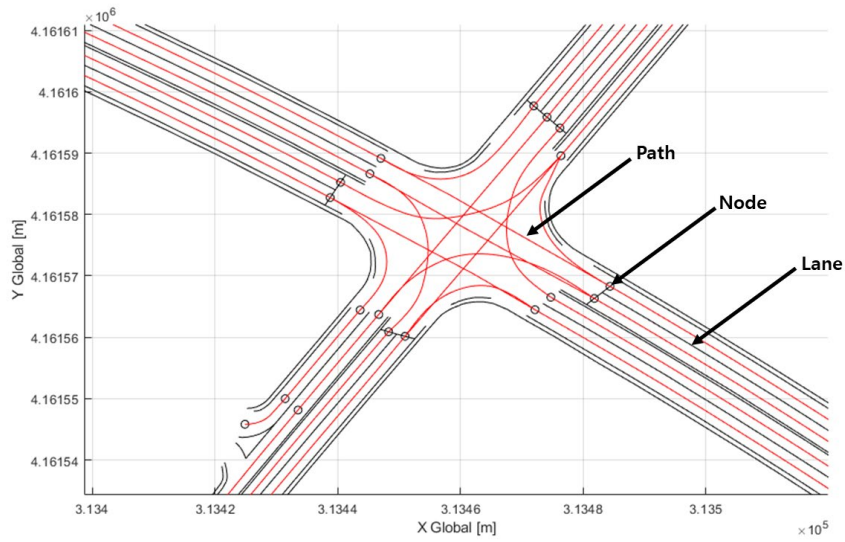
Figure 4.1. HD Map of intersection at Sangam, Seoul

For better comprehension, the raw HD map can be redesigned as in Figure 4.2. The green nodes represent the enter nodes, and red ones are exit nodes. Since the intersection networks are pre-defined from the HD map, if the vehicle enters to the intersection from the node 8, the possible exit nodes that the oncoming target can pass are the node 2 and 4. If the autonomous vehicle is conducting a right turn at the intersection from the enter node 3 to the exit node 4, the traffic participants to be considered for the interaction is ones crossing from the node 4. The target driving path and the ego driving route share the same exit node, the node 4 in Figure 4.2. From the HD map, the driving route of the autonomous vehicle, the block dotted path from Figure 4.2, is defined, and therefore it is only necessary to select the target path that shares the same exit node as the driving route of the autonomous vehicle. In Figure 4.2, the blue dotted route shares the same exit node

with the ego driving path. The computational burden of target vehicle prediction can be diminished with the reduced Region of Interest (ROI) and decreased number of interested traffic participants in the intersection. The surrounding vehicle selection methodology is summarized in Algorithm.
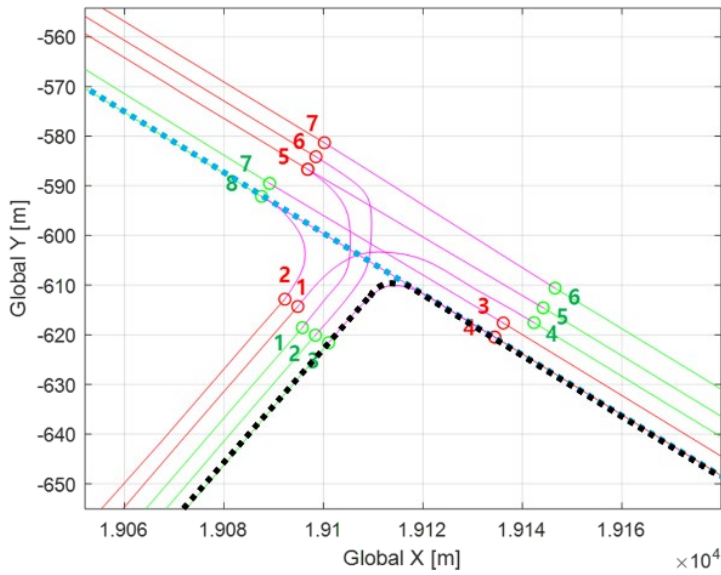
Figure 4.2. Driving lanes and nodes in intersection

Algorithm 1 : Surrounding Vehicle Selection

1:	Inputs: Estimate states (position, heading angle, velocity) of surrounding object, Route information from HD map
2:	**if** object is vehicle
3:	**if** object is on the lane sharing the same exit node
4:	**if** object heading within a certain degree
5:	Target of interest
6:	**else**
7:	Target of disinterest
8:	**else**
9:	Target of disinterest
10:	**else**
11:	Other moving object
12:	**end**

4.2. Data-driven Vehicle State based Motion Prediction at Intersection

4.2.1. Network Architecture of Motion Predictor

The driving behavior of a vehicle is sequential and interactive with other vehicles. Due to the continuous motion of vehicle, the current and future driving patterns are affected by past driving motions. Various methodologies have been conducted to predict the future motions of the vehicle, and rule-based and learning-based models are two widely adopted methods to anticipate the future trajectories and intentions of target vehicles. However, the rule-based methodologies are difficult to implement the historical driving information to infer the intention and predict trajectories. Especially in the intersection where the road structure is complicated and traffic participants are approaching from various direction with unexpected driving patterns, it is complicated to predict future driving motion.

In this study, the LSTM-RNN is utilized to incorporate the historical data of the driving motion of the target vehicle, predicting the accurate and precise predicted trajectories. The RNN is one of various deep learning technologies and enables to model time dependent and sequential data problem such as stock market prediction and text generation. Since vehicle behavior can be interpreted in a sequential way, many researchers started implementing the RNN to predict future driving motion of vehicles. However, the RNN suffers from the vanishing gradient problem which fails to learn and update weights of hidden states. In other words,

the RNN can easily fail to learn long sequential data. The gradient carries the information used to update the RNN parameters, and gradients becomes reduced through activation functions in hidden layers of sequential inputs. Thus, due to small value of gradient, the parameter update becomes insignificant and no actual training takes place. Due to a vanishing gradient problem, the very past hidden states will not be updated, and the RNN is also named as short-term.

In order to overcome the vanishing gradient problem, the LSTM is adopted. In the LSTM, an additional internal hidden state called cell state is added to manage the long-term memory for a gradient update. Thus, the LSTM can preserve long-range information better than the RNN can. The LSTM adopted in this study to predict the future trajectories of target vehicles is shown in Figure 4.3.
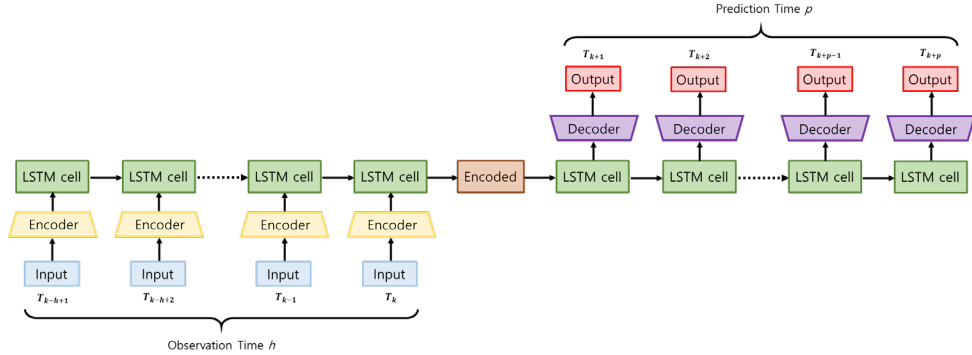
Figure 4.3. Diagram of the proposed LSTM-RNN

We determined the input features and a length of a horizon history through various combinations and error analysis. The candidates of the combination of input features and the horizon history are summarized in the Table 7. Different

types of odometry are analyzed to determine the optimal prediction results along with two different history horizons. History horizons of 1 second and 2 second are chosen to compare the prediction results. Due to the limited perception range of the perceptive sensors, history horizons longer than 2 second are not considered. If we consider 3 second of a history horizon of target vehicles, the predicted trajectories of target vehicle can be obtained after 3 second from the initial detection of the target vehicle. In other words, the target vehicle will be positioned in the middle of the intersection when the autonomous vehicle is able to incorporate the future trajectories of the target vehicle. The target vehicle has already crossed the significant distance in the intersection, and the missing trajectory of 3 second is critical and fatal for the autonomous vehicle to properly plan the ego motion to maintain the safety during intersection crossing.

Table 7. Candidates of Input and Output Feature Combinations for the Proposed

LSTM-RNN

Case	Input	Output	Hist. Horizon
1	$p_{x,tar}, p_{y,tar}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	1s
2	$p_{x,tar}, p_{y,tar}, v_{x,tar}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	1s
3	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	1s
4	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}, \theta_{tar}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	1s
5	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}, \theta_{tar}, p_{x,ego}, p_{y,ego}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	1s
6	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}, \theta_{tar}, p_{x,ego}, p_{y,ego}, v_{x,ego}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	1s
7	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}, \theta_{tar}, p_{x,ego}, p_{y,ego}, v_{x,ego}, \theta_{ego}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	1s
8	$p_{x,tar}, p_{y,tar}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	2s
9	$p_{x,tar}, p_{y,tar}, v_{x,tar}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	2s
10	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	2s
11	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}, \theta_{tar}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	2s
12	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}, \theta_{tar}, p_{x,ego}, p_{y,ego}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	2s
13	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}, \theta_{tar}, p_{x,ego}, p_{y,ego}, v_{x,ego}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	2s
14	$p_{x,tar}, p_{y,tar}, v_{x,tar}, d_{tar}, \theta_{tar}, p_{x,ego}, p_{y,ego}, v_{x,ego}, \theta_{ego}$	$\hat{p}_{x,tar}, \hat{p}_{y,tar}$	2s

$* \, d_{tar} : travel\ distance\ between\ sampling\ steps$

In order to enhance the prediction accuracy, the prediction results from input features of two different coordinates are compared. For the first case, the input features are measured in the ego vehicle centered coordinate. If the input features from the ego vehicle centered coordinates are trained for the model, the autonomous vehicle can directly utilize the environmental information obtained from the local perceptive sensors mounted on the vehicle. The input features of ego vehicle centered coordinate is advantageous in real-time calculation. The candidates of input features of the ego vehicle centered coordinate are depicted in Figure 4.4. The $p_{x,tar,k}$, $p_{y,tar,k}$, $v_{x,tar,k}$, $\theta_{tar,k}$ represents a position x, a position y, a longitudinal velocity, and a heading angle of target vehicle at time k, which is a

current time in the scenario in Figure 4.4. The d_k is the travel distance between the position at time k and time k-1. The $p_{x,ego,k}$, $p_{y,ego,k}$, $v_{x,ego,k}$, $\theta_{ego,k}$ represents a position x, a position y, a longitudinal velocity, and a heading angle of the ego vehicle at time k.
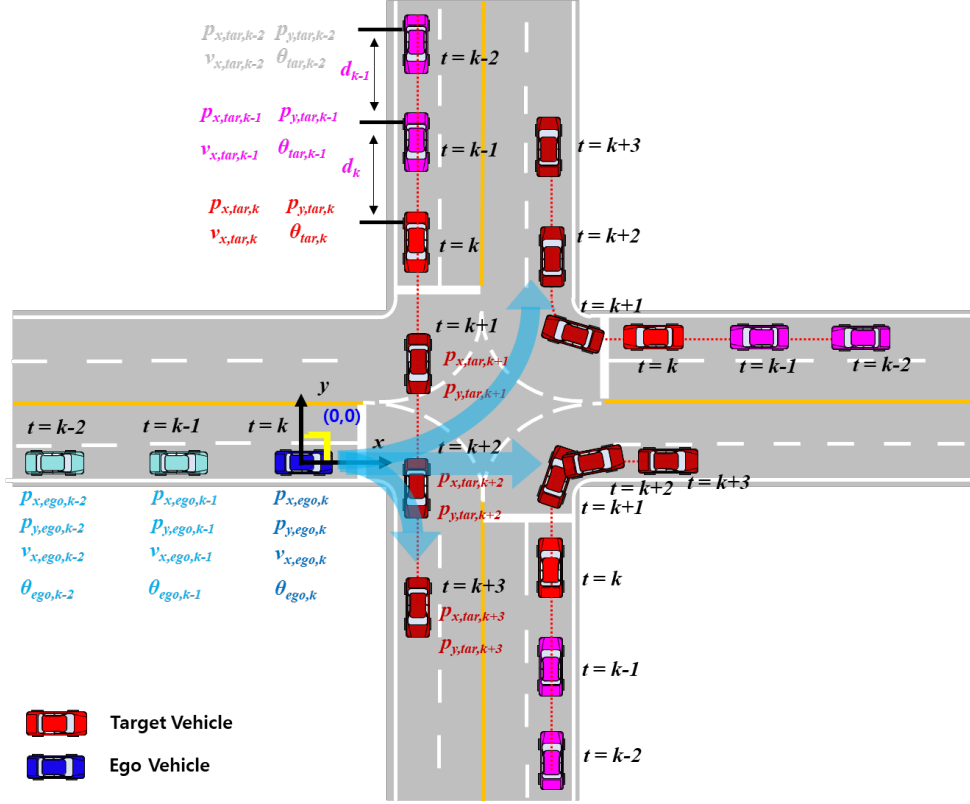
Figure 4.4. Input feature candidates in ego vehicle centered coordinate.

For the second case, the input features are rearranged in the detected target vehicle centered coordinate. If the input features are organized in the target vehicle centered coordinate, the training parameters of the model can be more easily converged. Since the current position of the target vehicle is centered at (0,0), input

features from different training data share the common and similar odometry at the current time, allowing better input-domain adaptation and generalization. The position of the target vehicle at current time k is (0,0). The candidates of input features are depicted in Figure 4.5.
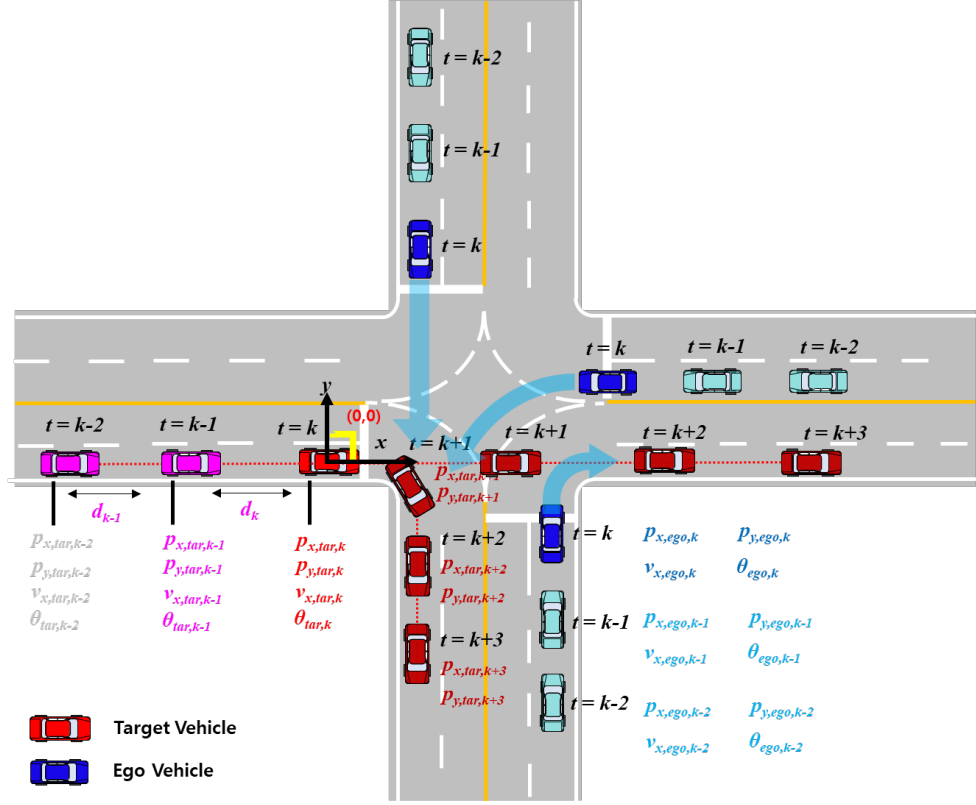
Figure 4.5. Input feature candidates in target vehicle centered coordinate.

The optimal network configuration is determined by comparing the accuracy of prediction results. The prediction errors between the true and predicted trajectories of the target vehicle over the prediction horizon are analyzed for various input feature candidates. The position error measurement of the predicted

trajectories is depicted in Figure 4.6.
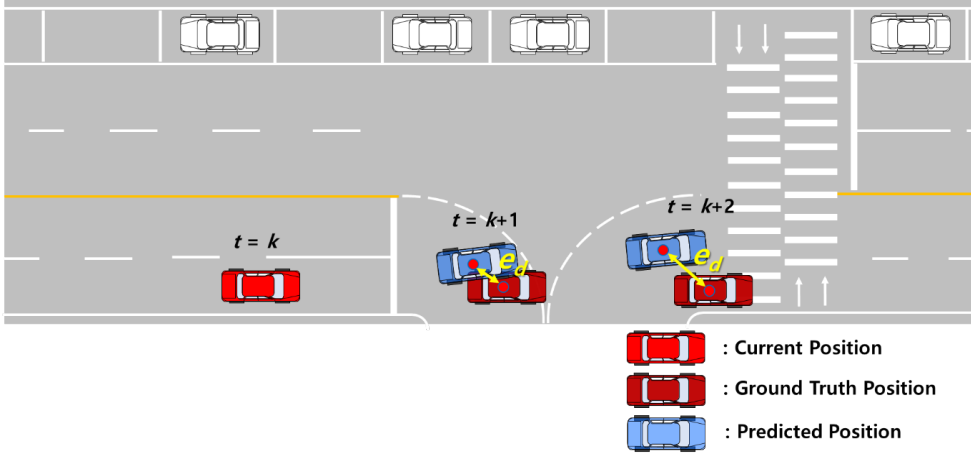
Figure 4.6. Definition of prediction errors over a prediction horizon

A mean squared error and a negative log likelihood are used for the loss function for training the model. In addition to the mean squared error loss function, the negative log likelihood function is utilized for the probabilistic prediction. The negative log likelihood loss function minimizes the negative log likelihood. From the bivariate gaussian distribution, the negative log likelihood function can be described as below,

$$Loss = \frac{1}{2(1-\rho^2)} \times \left[\sigma_x^2(x-\mu_x)^2 + \sigma_y^2(y-\mu_y)^2 - 2\rho\sigma_x\sigma_y(x-\mu_x)(y-\mu_y)\right] - log\left(\sigma_x\sigma_y\frac{1}{(1-\rho^2)}\right) \tag{4.1}$$

The outputs of the network consist of mean, sigma, and rho, and the equation 4.1 is formulated with those parameters to train the model.

4.2.2. Dataset Processing of the Network

The training dataset obtained for the surrounding target vehicle prediction is the same dataset utilized for the virtual target modeling. As described in the section 3.3.2. Gaussian Process Training, the training dataset for the network is obtained from two different intersections with 13 different drivers. The dataset of two different driving scenes is achieved: the ego vehicle crosses straight and the target vehicle turns right, and the ego vehicle turns right and the target vehicle drives straight from the right. The total 196 different driving scenarios are obtained where 98 cases are target vehicle yielding scenario and the other 98 cases are non-yielding scenarios. The total dataset is 141,032, and the training and test dataset are 105,774 and 35,258 respectively. The odometry information of ego and target vehicle are obtained from the sensors built in autonomous vehicles. Among the total driving datasets, randomly selected 350 data are plotted in the ego vehicle centered coordinate as shown in Figure 4.7.
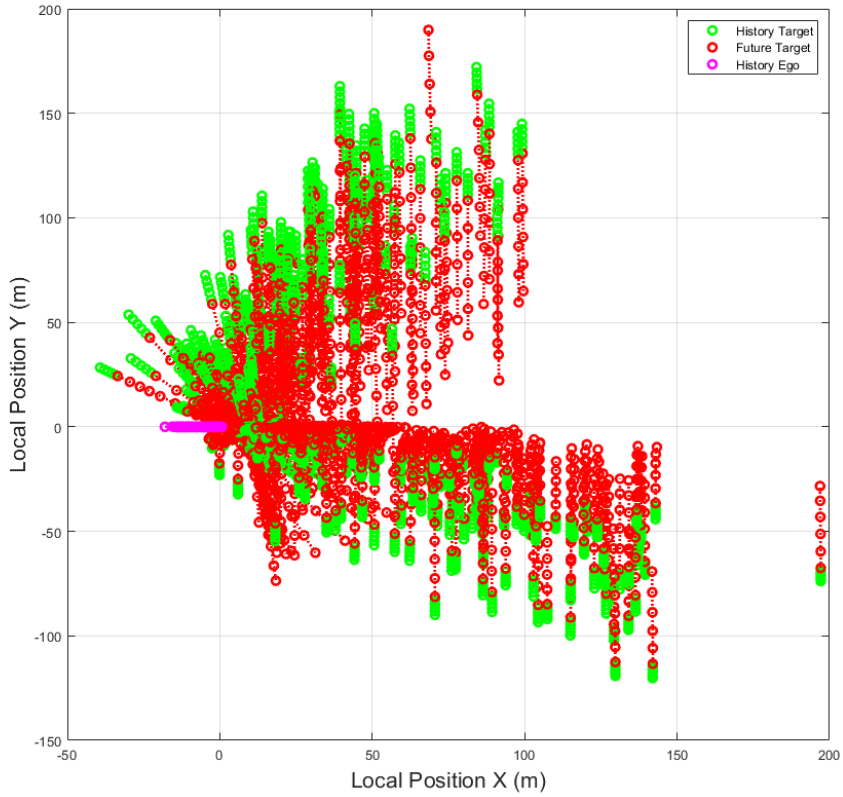
Figure 4.7. Trajectories of ego and target vehicle in ego vehicle centered coordinate.

The green circles of figure represent history trajectories of target vehicles, and red circles describe future trajectories of target vehicles. The magenta-colored circles are trajectories of the ego vehicle. It is observed that the trajectories of target vehicles are scattered randomly from the center coordinate (0,0) of the ego vehicle. Also, randomly selected 350 data are plotted in the target vehicle centered coordinate as shown in Figure 4.8. Compared to trajectories of target vehicles in the ego vehicle centered coordinate, the those in the target vehicle centered

coordinate are gathered, and future trajectories of the target vehicle projects from the shared common position (0,0) at current time.
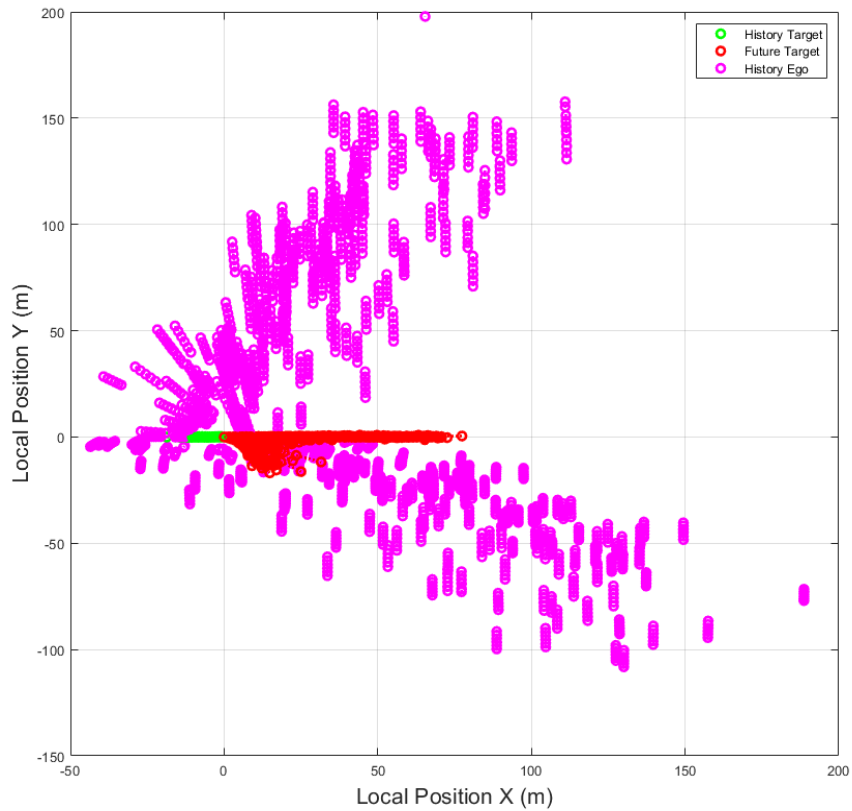
Figure 4.8. Trajectories of ego and target vehicle in target vehicle centered coordinate.

Chapter 5. Intersection Longitudinal Motion Planning

5.1. Outlines of Longitudinal Motion Planning with Model Predictive Control

The longitudinal motion planning of the proposed algorithm implements the model predictive control to optimize and obtain the desired control input, a desired longitudinal acceleration. The model predictive control is formulated with the system dynamics model, reference states, constraints, and input control. The MPC is one of the optimal control methods and widely implemented in planning and control in various fields such as robotics and autonomous driving. With the MPC, system dynamics and constraints are employed to a cost function to generate and optimized control input suitable for the current situation, thereby enabling a stable autonomous driving.

The MPC with the state variable at k-1 step plans the control input from k to predefined prediction horizon in advance and feedbacks the states at the current k-step with the planned control input. The control input is evaluated by comparing the response, and the const function is updated accordingly to calculate the control input from k+1 to the prediction horizon. In other words, with the given constraints, the optimal control input is obtained with the repetition of plan and evaluation. Such methodology of MPC is known as a Receding Horizon Control. It is a significant difference between MPC and Linear Quadratic Regulator that optimizes

across the entire time horizon.

5.2. Stochastic Model Predictive Control of Intersection Motion Planner

5.2.1. Definition of System Dynamics Model

For the vehicle dynamics model of the proposed longitudinal motion planning, a kinematic model with the first-order input delay has been employed for the vehicle model of the ego vehicle. The continuous-time dynamics model is shown in equation below,

$$\dot{x}_{long}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1/\tau \end{bmatrix} x_{long}(t) + \begin{bmatrix} 0 \\ 0 \\ 1/\tau \end{bmatrix} u_{long}(t) \tag{5.1}$$

Where x(t) represents the state vector of travel distance, p(t), longitudinal velocity, v(t), and longitudinal acceleration, a(t). The control input u(t) represents the desired longitudinal acceleration. The time constant, τ, is assumed to be 0.33s in this study. Using the Euler discretization method, the continuous-time model can be re-written as a discrete-time model.

$$x_{long}(k+1|t) = \begin{bmatrix} 1 & dt & 0 \\ 0 & 1 & dt \\ 0 & 0 & 1-dt/\tau \end{bmatrix} x_{long}(k|t) + \begin{bmatrix} 0 \\ 0 \\ dt/\tau \end{bmatrix} u_{long}(k|t) \tag{5.2}$$

5.2.2. Ego Vehicle Prediction and Reference States Definition

The MPC is aimed to optimize the reference motion incorporating the constraints. In the proposed longitudinal motion planning, the reference states are a travel distance, a longitudinal velocity, and a longitudinal acceleration of the predicted states of the ego vehicle and are based on the curvilinear velocity when no target vehicle is approaching at the intersection. In other words, they are the predicted ego states during crossing the intersection without any interference from the target vehicle.

The pre-defined driving route of the ego vehicle can be expressed in the 2nd order curve fitting equation as below,

$$y_k = a_2 x_k^2 + a_1 x_k + a_0 \tag{5.3}$$

The radius, r, of the given ego driving path during a prediction horizon can be computed using the 2nd order curve fitting equation and expressed as

$$r_k = \frac{\left(1 + \left(dy_k/dx\right)^2\right)^{3/2}}{d^2 y_k/dx^2} \tag{5.4}$$

With the calculated radius and maximum lateral acceleration that is defined as 2m/s² in this study, the longitudinal curvilinear velocity is determined as shown in an equation below,

$$v_{x,k} = \sqrt{a_{y,\max} \left| \frac{\left(1 + \left(dy_k/dx\right)^2\right)^{3/2}}{d^2 y_k/dx^2} \right|} \tag{5.5}$$

The predicted longitudinal velocities can be implemented to a virtual yaw rate at predicted steps as shown below,

$$\gamma_k = v_{x,k} \frac{d^2 y_k / dx^2}{\left(1 + \left(dy_k / dx\right)^2\right)^{3/2}} \tag{5.6}$$

Based on the computed virtual yaw rates, the process update model for ego vehicle state prediction is defined as follows,

$$p_{x,k+1} = p_{x,k} + v_{x,k} \cos\theta_k \cdot \Delta t + (a_{x,k} \cos\theta_k - \gamma_k v_{x,k} \sin\theta_k) \cdot \frac{(\Delta t)^2}{2}$$

$$p_{y,k+1} = p_{y,k} + v_{x,k} \sin\theta_k \cdot \Delta t + (a_{x,k} \sin\theta_k - \gamma_k v_{x,k} \cos\theta_k) \cdot \frac{(\Delta t)^2}{2}$$

$$\theta_{k+1} = \theta_k + \gamma_k \cdot \Delta t \tag{5.7}$$

$$v_k = \min\left(v_{x,\max}, \sqrt{a_{y,\max} \left| \frac{\left(1 + \left(dy_k / dx\right)^2\right)^{3/2}}{d^2 y_k / dx^2} \right|}\right)$$

The predicted reference states from the path following model are incorporated to formulate the model predictive control of the proposed longitudinal motion planning.

5.2.3. Safety Clearance Decision for Intersection Collision Avoidance

In autonomous driving at urban intersection, various safety indices such as time to conflict region, clearance, and distance to conflict region are used to plan the safe motion at the intersection. Among those criteria, a conflict region is the most significant factor to consider in motion planning. Though various methodologies exist to determine the conflict region, we estimated the conflict

region based on the predicted trajectories of the ego and target vehicles. The scheme of defining the conflict region is depicted in Figure 5.1. In the 4-way intersection, the red and blue vehicles are target and ego vehicles. In Figure 5.1, the prediction horizon is assumed to be 5s, and predicted trajectories and positions of target and ego vehicles are illustrated with translucent red and blue colors respectively. Predicted positions are located at intervals of 1s. In order to determine the conflict region, the clearance, c, between future trajectories at every predicted step is compared. The clearance at every predicted step is compared with a predefined clearance threshold to determine time to predicted collision.

In order to mimic human driving turn-motion at the intersection, human driven data at the intersection is collected to analyze and define the conflict region threshold. Total 131 driving cases of data are obtained from 13 different drivers and graphed in Figure 3.7. In Figure 5.2, the clearance between a target and ego vehicle is described in a displacement domain, a distance to conflict. In Figure 5.3, the clearance between a target and ego vehicle is described in a velocity domain, a velocity of ego vehicle. It is inferred that as the velocity decreases the clearance between two vehicles diminishes. The minimum clearance from two different domain analysis is both 5m. Human drivers tend to maintain at least the minimum clearance of 5m.

(a)

Target Vehicle

t = 1s
t = 2s
t = 3s
t = 4s
t = 5s
c(t=3s)
c(t=2s)
t = 5s   t = 4s
t = 3s
Conflict Region
t = 2s
c(t=1s)
t = 1s

Ego Vehicle

(b)

t = 5s
t = 5s
t = 4s
t = 3s   t = 2s   t = 1s
Target Vehicle
t = 4s
Conflict Region
c(t=3s)
t = 3s
c(t=2s)
t = 2s
c(t=1s)
t = 1s
Ego Vehicle
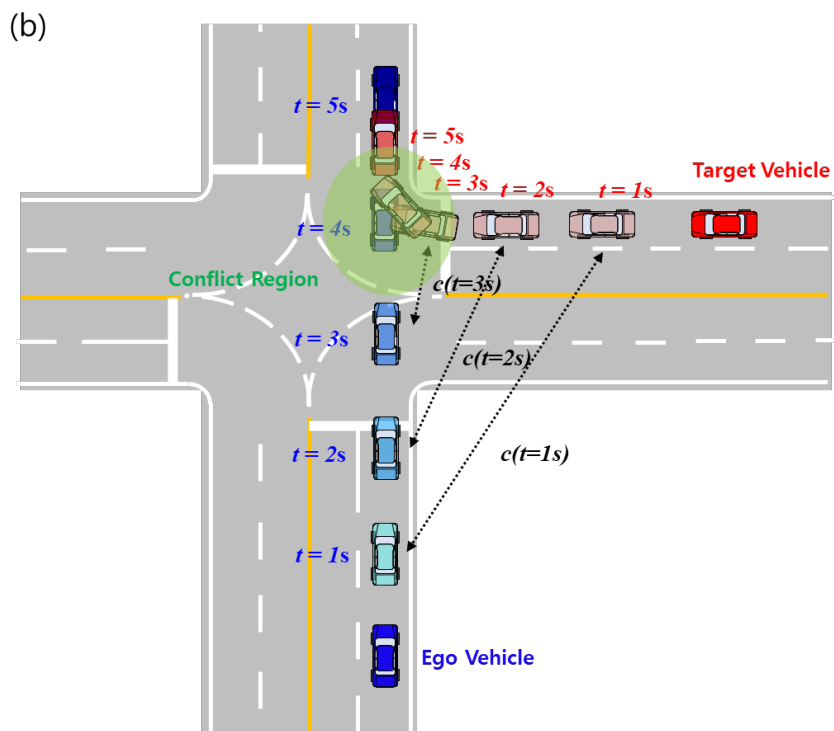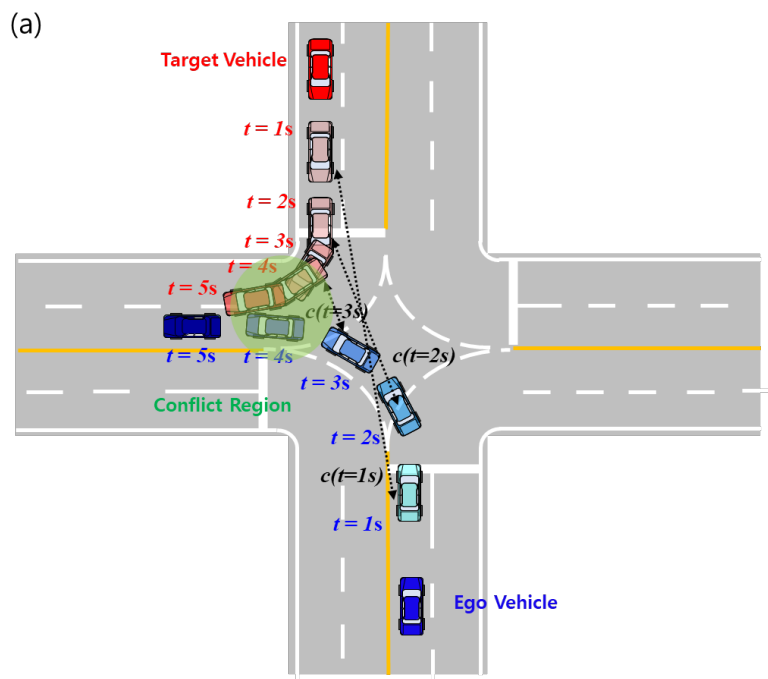
(c)
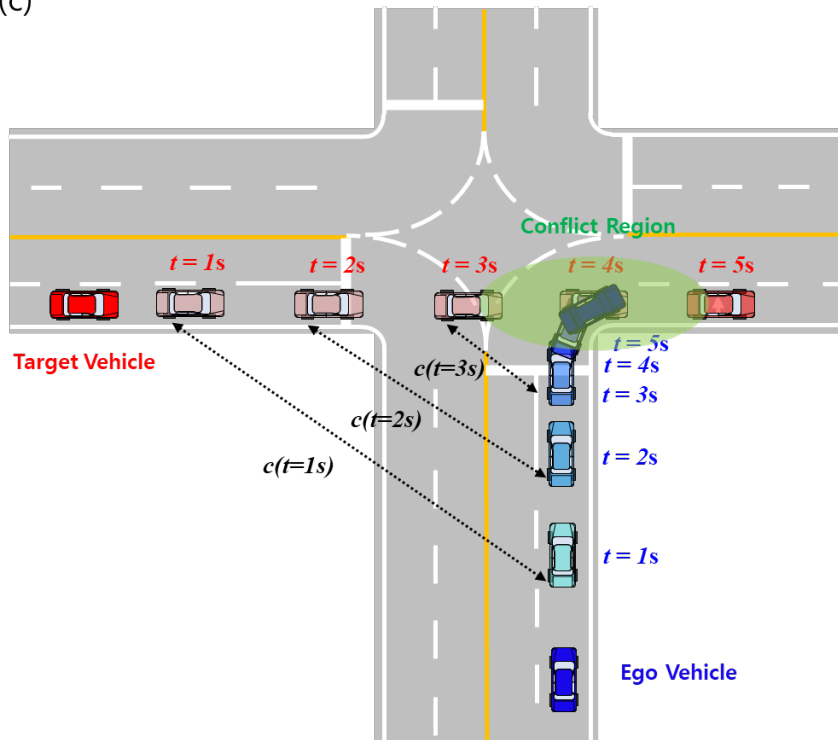
Figure 5.1. A conflict region from predicted trajectories of target and ego vehicles

(a) left turn (b) straight (c) right turn
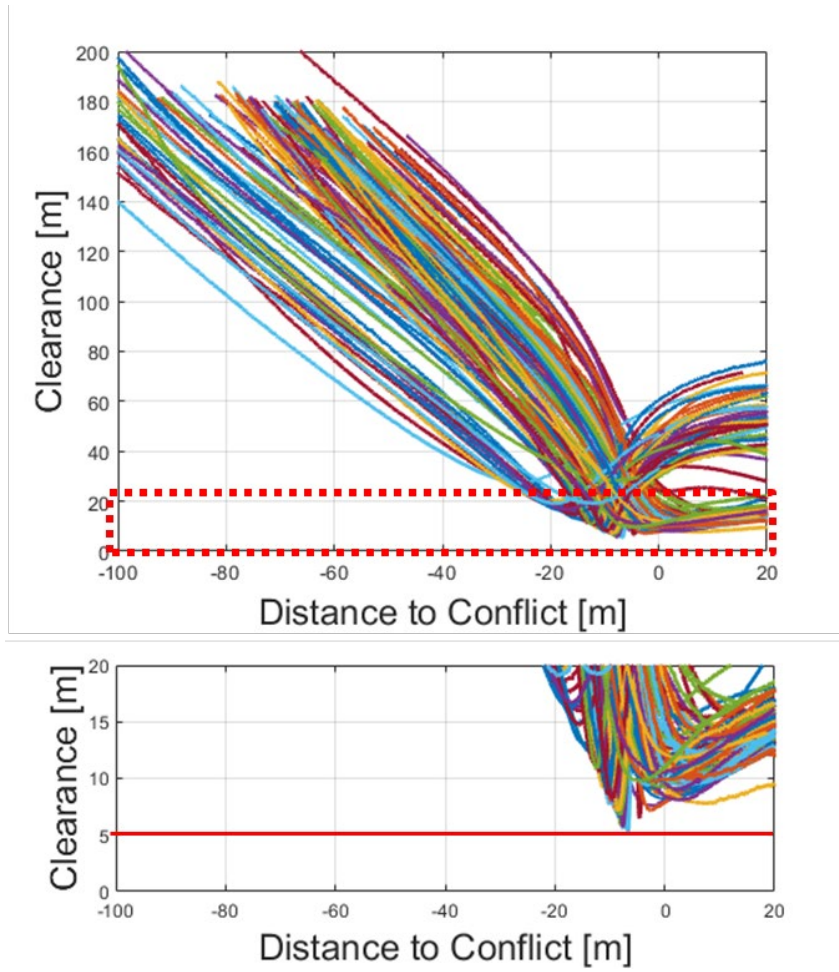
Figure 5.2. Change in clearances between target and ego vehicles in a position domain.
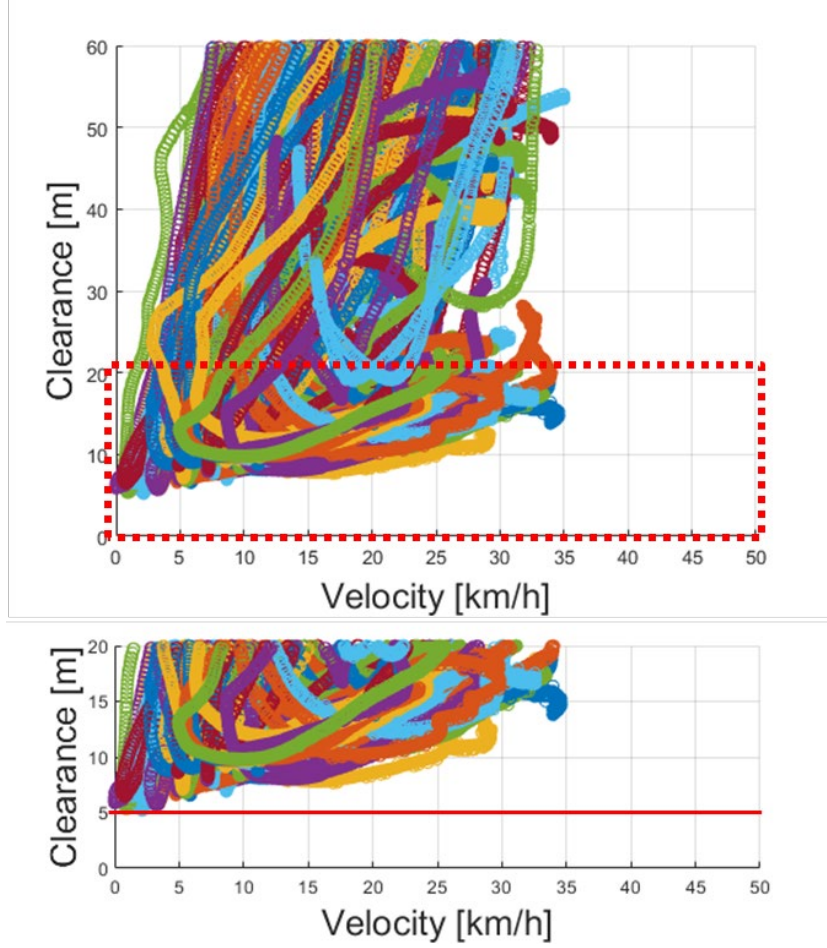
Figure 5.3. Change in clearances between target and ego vehicle in a velocity domain.

With the desired clearance threshold, the predicted time steps where conflict regions are generated can be determined as in Figure 5.4. In Figure 5.4, it is assumed that clearances during predicted time steps from 3s to 5s are less than clearance threshold and depicted with magenta color. The estimated travel distances of the ego vehicle during the predicted time steps of the conflict region will be employed as position constraints for MPC formulation. In order to ensure

7 6

the safety from the collision, the gain is applied to the position constraint, and the position safety constraint is defined as follows,

$$p(k) \leq p_{upper}(k) = k_{TTC} \cdot DTC$$
$$where \ k_{TTC} = f(TTC)$$

(5.8)

The safety gain, k_{TTC}, is described in Figure 5.5, and the shorter the predicted time step of the collision is, the smaller safety gain will be applied to guarantee the safety margin.

(a)
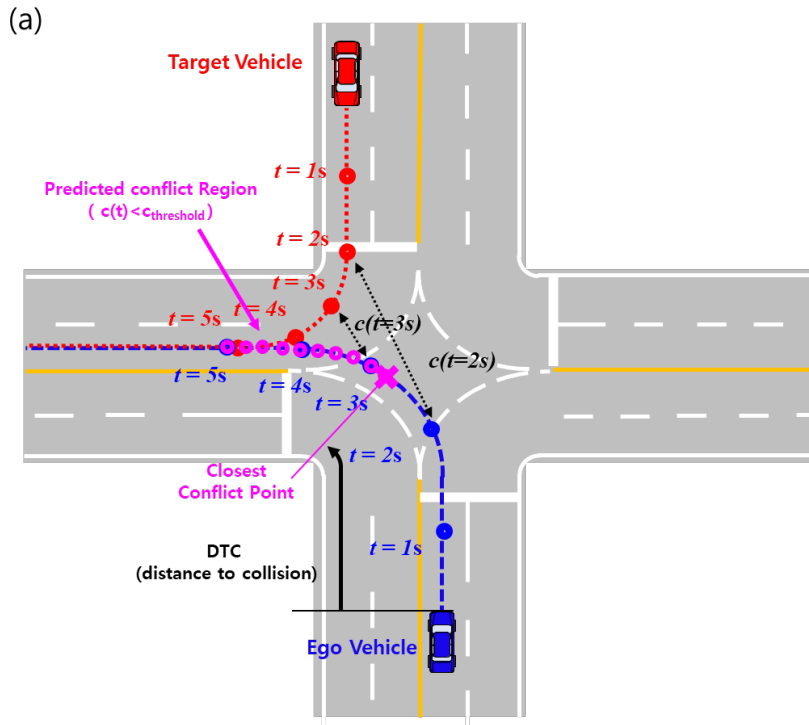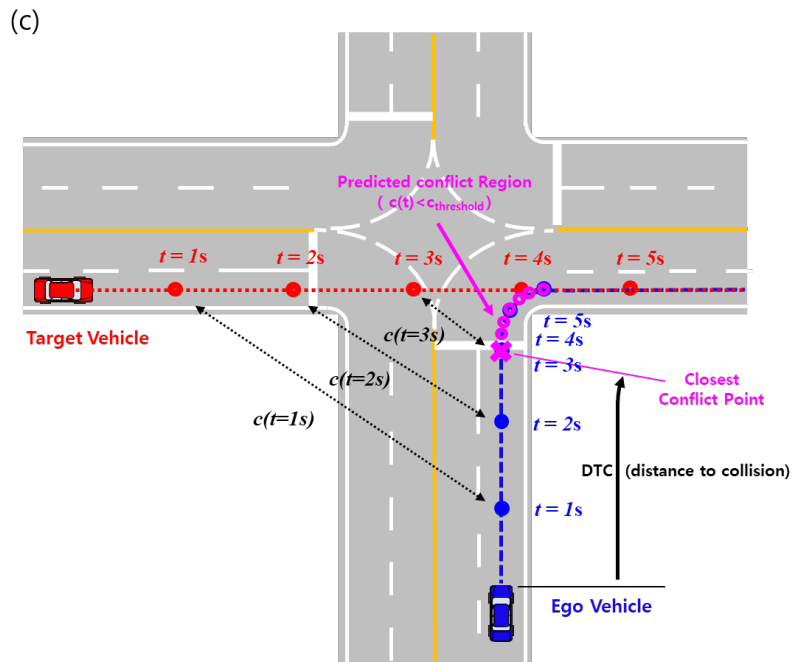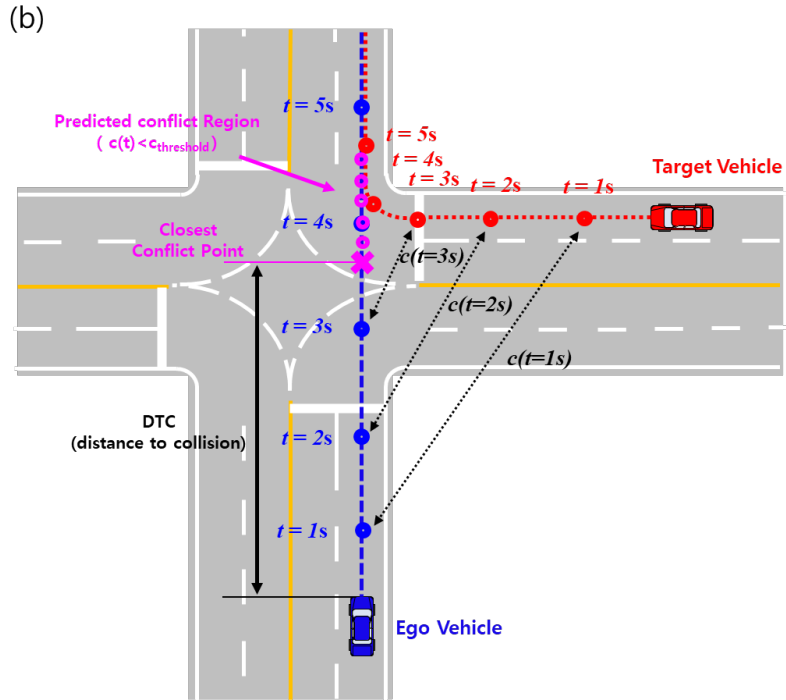
Figure 5.4. Closest conflict point from the predicted conflict region in the intersection (a) left turn (b) straight (c) right turn
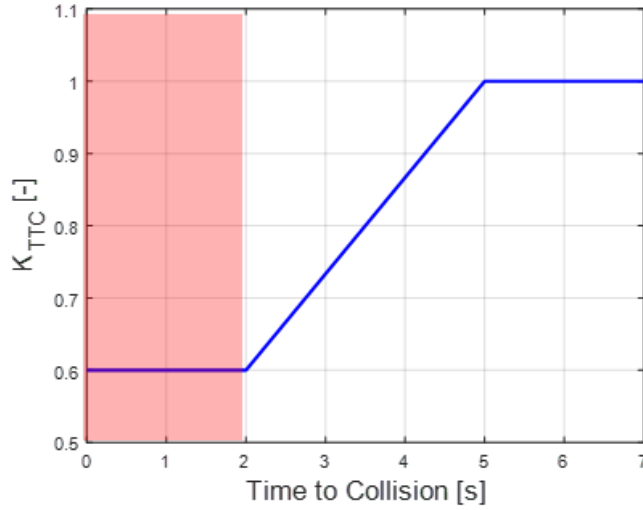
Figure 5.5. The safety gain profile for position upper bound

5.2.4. Driving Mode Decision of Intersection Motion Planning

The driving mode of the longitudinal motion planning of the intersection is composed of cross, stop, and creep mode. The cross and stop mode are determined based on the intersection of predicted trajectories of target and ego vehicles. As shown in Figure 5.4, if the conflict region is generated based on the future trajectories of the target and ego vehicles, the position constraint is determined to decelerate the ego vehicle to avoid collision, naturally activating the stop mode. On the other hand, as shown in Figure 5.6, if the conflict region is not generated based on the predicted trajectories of vehicles, the position constraint is defined to allow ego vehicle to cross the intersection, leading to the cross mode.

(a)

Target Vehicle

$t = 1$s

$t = 2$s

$t = 3$s

$t = 4$s

$t = 5$s

$c(t=3s)$

$c(t=2s)$

$t = 5$s $t = 4$s $t = 3$s

$t = 2$s

$t = 1$s

Ego Vehicle

(b)

$t = 5$s

$t = 4$s

$t = 3$s

$t = 2$s

$t = 1$s

Ego Vehicle

Target Vehicle

$t = 5$s $t = 3$s $t = 1$s

$c(t=3s)$

$c(t=2s)$

$c(t=1s)$
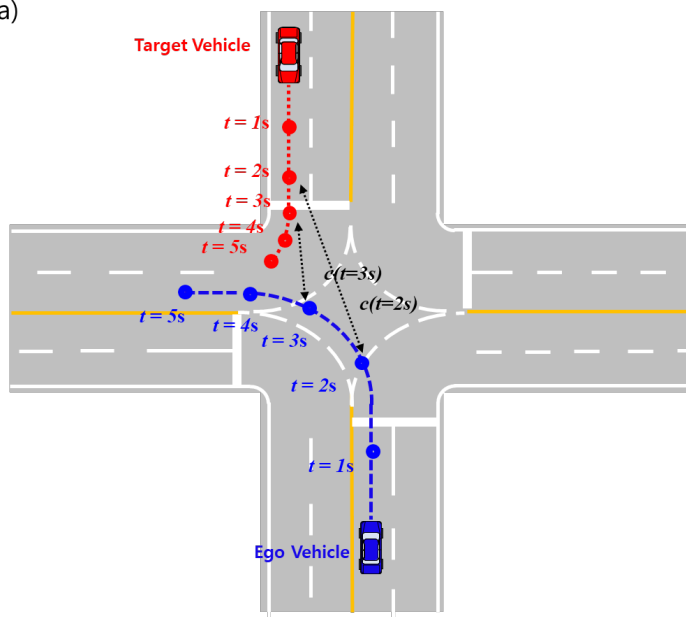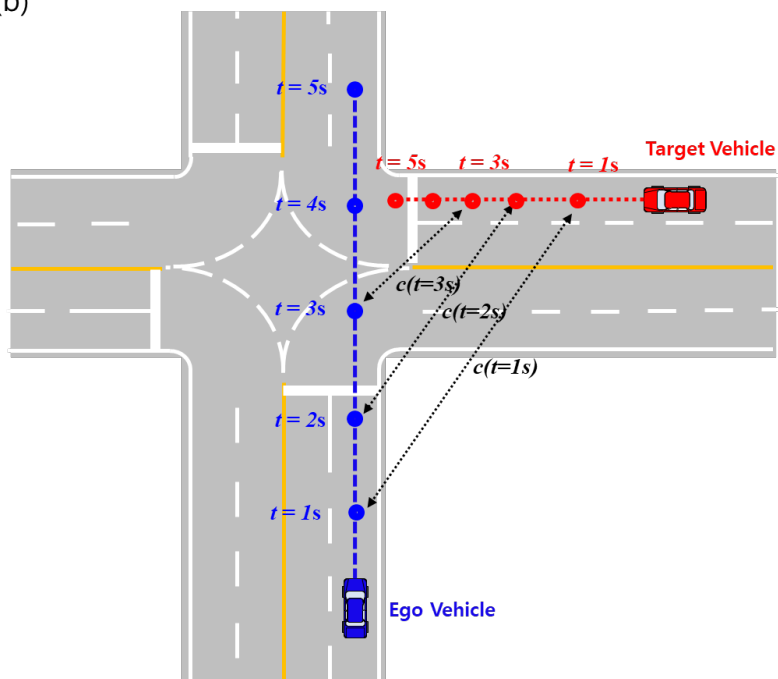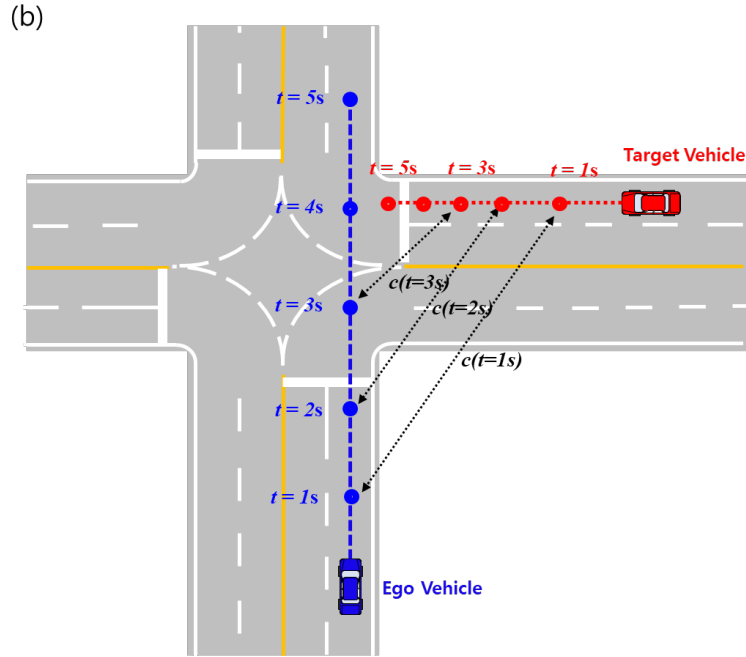
Figure 5.6. Cross mode activated situation during intersection crossing (a) left turn (b) straight (c) right turn

The creep mode is sub-mode of the stop mode. The creep mode is designed to overcome the dead-lock situation and mimic the human driving pattern that a human driver slowly drives the vehicle forward until enough visibility is achieved. The creep mode algorithm is described in Figure 5.7.
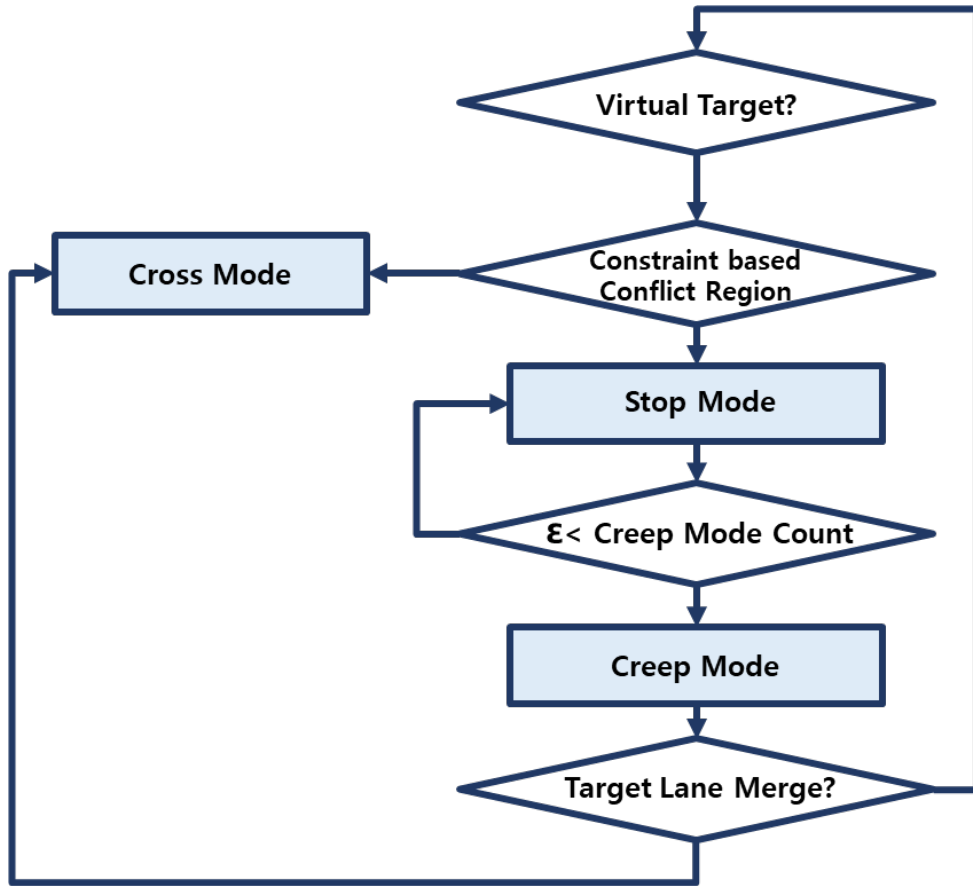
Figure 5.7. Diagram of the creep mode algorithm

The creep mode is activated after the stop mode. The stop mode should begin under generation of conflict region and position constraint from virtual target trajectory. During the stop mode, the creep mode count increases. The creep mode is activated if the creep mode count becomes greater than ε. The epsilon in the creep mode count is 2 seconds. During the creep mode, the maximum acceleration is 1.0m/s^2 and maximum velocity is 10km/h.

5.2.5. Formulation of Model Predictive Control with the Chance Constraint

The MPC is implemented to solve for the desired states and minimize control efforts while satisfying various constraints. MPC problem is formulated with vehicle dynamics model, constraints, and input limits. The quadratic const function of the MPC problem is defined as follows.

$$\min J_{long} = \sum_{k=1}^{N_p} \left\| x_{long}(k|t) - x_{ref}(k|t) \right\|_Q^2 + \sum_{k=0}^{N_p-1} \left\| u_{long}(k|t) \right\|_R^2 + \sum_{k=0}^{N_p-2} \left\| \Delta u_{long}(k|t) \right\|_{\mathbf{R}_{\Delta u}}^2$$
$$+ \sum_{k=1}^{N_p} \left\| q_{u,\min}(k|t) \right\|_{\mathbf{R}_u}^2 + \sum_{k=1}^{N_p} \left\| q_j(k|t) \right\|_{\mathbf{R}_j}^2 \tag{5.8}$$

Where x_{long} is the states that are distance travelled, velocity, and acceleration, and x_{ref} is the reference motions of travel distance, p_x, and longitudinal velocity, v_x. The u_{long} is the control input, the desired longitudinal acceleration. The Q and R are the weighting matrix to penalize the reference tracking error and control effort respectively. Since the field of view while driving at the intersection is limited and sudden appearance of traffic participants from blind spots is unexpected, the $q_{u,\min}$ and q_j are the penalties when violating the minimum control input, a longitudinal acceleration, and a lower bound of a change in control input, a jerk. R_u and R_j are the weight matrix for penalizing. Though the minimum hard constraints are defined in order to guarantee both safety and ride comfort, it is sometime necessary to give up a certain degree of the ride comfort to secure the safety from the collision. Since a crossing the intersection is unpredictable unlike other driving behaviors such as lane change, lane keeping, and clearance control, the prediction horizon is

determined as 5s, and a sampling time is 0.2s.

In the MPC formulation, the constraints are defined to consider the system dynamics, actuator limits, ride comfort, and safety. As explained in chapter 5.2.1, the kinematic model with the first order input delay is implemented to define the dynamic constraints.

$$x_{long}(k+1|t) = \begin{bmatrix} 1 & dt & 0 \\ 0 & 1 & dt \\ 0 & 0 & 1-dt/\tau \end{bmatrix} x_{long}(k|t) + \begin{bmatrix} 0 \\ 0 \\ dt/\tau \end{bmatrix} u_{long}(k|t) \qquad (5.9)$$

where τ is the time constant of actuator delay. The time constant reflects the vehicle characteristics and is 0.33s in this model. The control input constraints are described as below,

$$
\begin{aligned}
& a_{\min} - q_{u,\min}(k|t) \le u(k+1|t) \le a_{\max} \\
& |u(k+1|t) - u(k|t)| \le (a_{jerk} \_ q_j(k|t))\Delta t_{\text{logic}} \\
& |u(0|t) - u_{pre}| \le (a_{jerk} \_ q_j(k|t))\Delta t_{\text{logic}}
\end{aligned}
\qquad (5.10)
$$

Where $a_{\min}$ and $a_{\max}$ are the minimum and maximum control input boundary and are -3m/s^2 and 2m/s^2 respectively. The a_{jerk} is a slew rate of the control input and 5m/s^2. The u_{pre} represents the control input from the previous step. These parameters are defined to provide comfortable ride comfort to drivers. The excessive use of control input will cause the discomfort to users.

The chance constraint is implemented to consider the uncertainty of the virtual target and detected targets. The chance-constrained upper bounds of distance traveled is defined as below,

$$\Pr(g^T(k|t)x(k|t) \le \mathrm{p}_{upper}) \ge 1 - \varepsilon$$

$$where, \quad p_{upper}(k|t) = \begin{cases} \text{base constraint (stop mode)} \\ \qquad 100 \text{ (cross mode)} \end{cases} \qquad (5.11)$$

$$g^T = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

The Casadi solver is used to solve the MPC formulation for simulation studies and implementation in test vehicle.

Chapter 6. Performance Evaluation of Intersection Longitudinal Motion Planning

6.1. Performance Evaluation of Virtual Target Prediction at Intersection

6.1.1. GPR based Virtual Target Model Prediction Results

The prediction errors of the distance to a conflict point at prediction time 5s with 10 different kernel functions are compared to determine the optimal kernel function of Gaussian Process Regression based virtual target model. The 10 different kernel functions are the Squared Exponential, Exponential, Matern 3/2, Matern 5/2, Rational Quadratic, ARD Squared Exponential, ARD Exponential, ARD Matern 3/2, ARD Matern 5/2, and ARD Rational Quadratic. The errors are evaluated in terms of mean absolute error (MAE), root mean squared error (RMSE), and standard deviation (STD). The errors of different kernel functions of GPR model for the virtual target crossing straight is shown in the Table 8. The ARD Exponential kernel computes the smallest errors of training GPR. As shown in the Table 8, the MAE, RMSE, and STD are 1.7099, 2.7958, and 2.7956. The GPR based virtual target model with the optimal kernel function is also compared with the constant velocity virtual target model. The errors of distance to conflict point are analyzed with MAE, RMSE, and STD as shown in Table 9 and depicted with

histogram as in Figure 6.1. The GPR predictor depicts errors more distributed near zero and expresses the observable improvement in prediction results in a long-term prediction time.

Table 8. Prediction Errors of the GPR Model with Different Kernels (Virtual Target Crossing Straight).

Error Metric	DTC at 5 sec prediction									
	a	b	c	d	e	f	g	h	i	J
MAE	3.9453	2.6657	2.7395	3.6675	3.3907	3.4436	**1.7099**	2.9398	3.0403	2.5799
RMSE	5.9161	4.1741	4.3168	5.5886	5.2230	5.2687	**2.7958**	4.6082	4.7641	4.1319
STD	5.9160	4.1740	4.3168	5.5885	5.2230	5.2686	**2.7956**	4.6082	4.7641	4.1319

a. Squared Exponential
b. Exponential
c. Matern 3/2
d. Matern 5/2
e. Rational Quadratic
f. ARD Squared Exponential
g. **ARD Exponential**
h. ARD Matern 3/2
i. ARD Matern 5/2
j. ARD Rational Quadratic

Table 9. Prediction Errors between the CV Model and the GPR Model (Virtual Target Crossing Straight).

Distance to Conflict Error at 5s of prediction time (m)					
CV			Proposed		
MAE	RMSE	STD	MAE	RMSE	STD
7.93	12.14	10.34	2.67	2.80	2.80
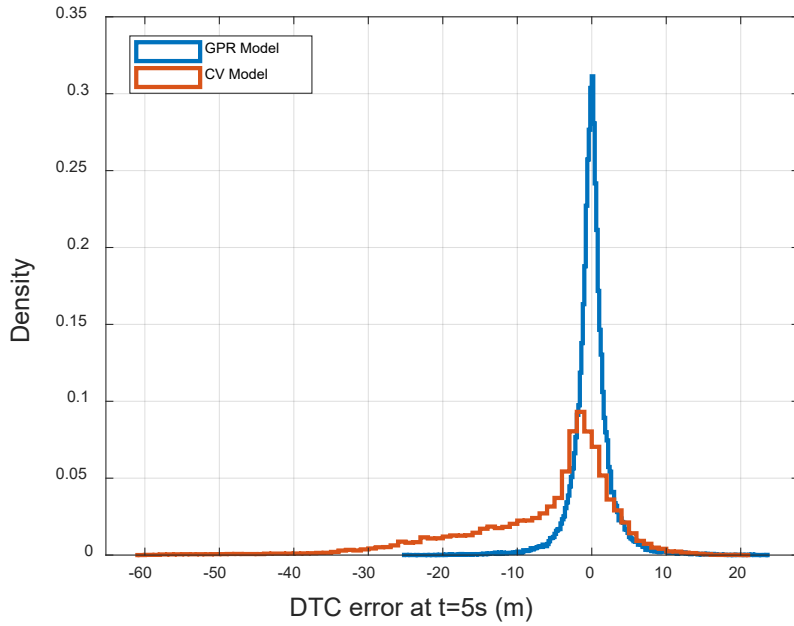
Figure 6.1. Histogram of errors of distance to conflict point between the GPR model and the CV model (virtual target straight cross).

The errors of different kernel functions of the GPR model for the virtual target turning right at the intersection is shown in the Table 10. The Exponential kernel computes the smallest errors of training the GPR. As shown in the Table 10, the MAE, RMSE, and STD are 1.7742, 2.8901, and 2.8901. The GPR based virtual target model with the optimal kernel function is also compared with the constant velocity virtual target model. The errors of distance to conflict point are analyzed with MAE, RMSE, and STD as shown in Table 11 and depicted with histogram as in Figure 6.2. The GPR predictor depicts errors more distributed near zero and expresses the observable improvement in prediction results in a long-term prediction time.

Table 10. Prediction Errors of the GPR Model with Different Kernels

(Virtual Target Turning Right)

Error Metric	DTC at 5 sec prediction									
	a	b	c	d	e	f	g	h	i	J
MAE	2.2347	1.7742	2.0507	2.1821	2.1617	2.0965	1.7862	2.0031	2.0400	2.1178
RMSE	3.5178	2.8901	3.3211	3.4679	3.4469	3.3678	2.9671	3.2755	3.3168	3.3971
STD	3.5178	2.8901	3.3211	3.4679	3.4468	3..3678	2.9670	3.2754	3.3168	3.3970

a. Squared Exponential   e. Rational Quadratic   h. ARD Matern 3/2
b. **Exponential**   f. ARD Squared Exponential   i. ARD Matern 5/2
c. Matern 3/2   g. ARD Exponential   j. ARD Rational Quadratic
d. Matern 5/2

Table 11. Prediction Errors between the CV Model and the GPR Model

(Virtual Target Turing Right).

Distance to Conflict Error at 5s of prediction time (m)					
CV			Proposed		
MAE	RMSE	STD	MAE	RMSE	STD
7.03	8.43	7.71	1.77	2.89	2.89
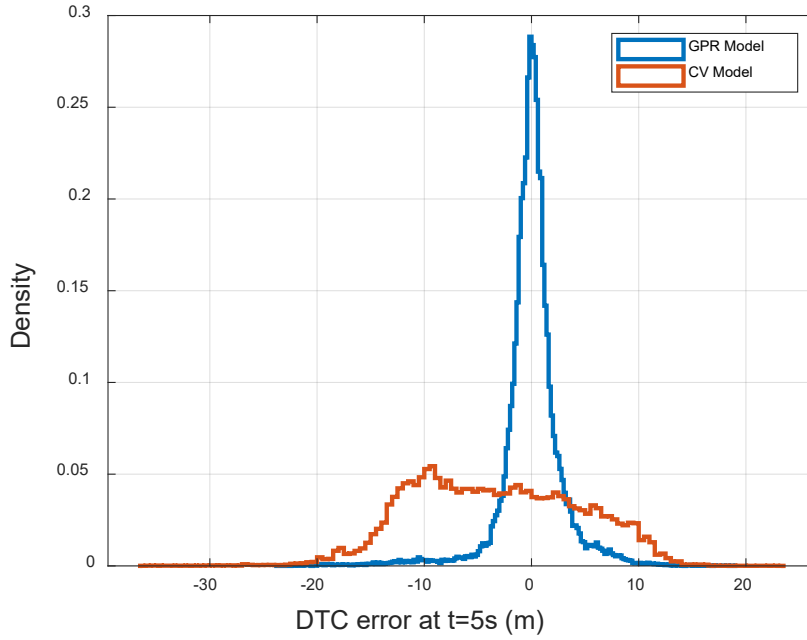
Figure 6.2. Histogram of errors of distance to conflict point between the GPR model and the CV model (virtual target turning right).

6.1.2. Intersection Autonomous Driving Computer Simulation Environment

The simulation studies for the virtual target prediction are performed to evaluate the feasibility of the proposed algorithm to overcome the dead-lock situation at the intersection. The simulation study is essential before the vehicle test since the simulation can generate various scenarios to validate the algorithm regardless of a dangerous situation, such as collision, and reproduce the same situation to compare different algorithms. The kinematic model is chosen for the vehicle model for the ego and target vehicles. The simulation environment is a 4-

way intersection where each two two-lane roads intersect at 90 degrees as shown in Figure 6.3. Road length before and after the intersection is 300m each, road width is 3.5m, and the radius of road curvature is 12m to reflect the actual urban road geometry of Seoul, Korea.
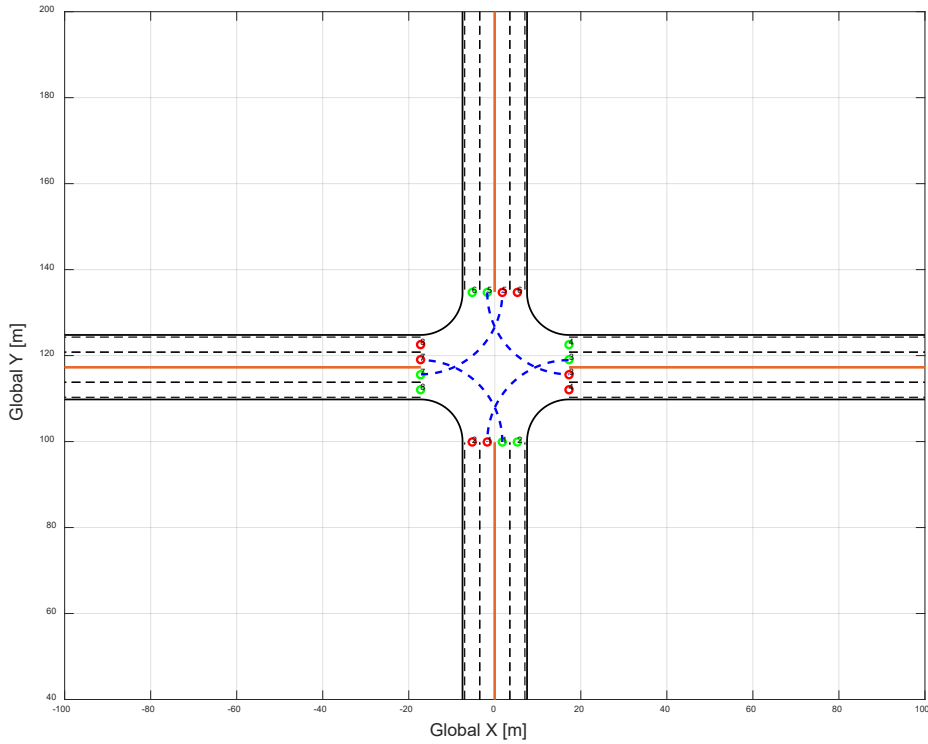
Figure 6.3. 4-way intersection offline simulation map

The actual specification of perceptive sensors equipped in the autonomous vehicle 'KIA Carnival' is used to construct FOV with a road boundary as illustrated in the section 2.2. Hardware Configuration and Test Vehicle. Two 3-D Velodyne LiDAR sensors are mounted on the top roof of the vehicle, but only one 3-D

Velodyne LiDAR sensor is considered for the offline simulation. Since two 3-D Velodyne LiDAR sensors are closely installed on the roof, it is assumed to be irrelevant to consider two sensors as one. The LiDAR sensor specification of distance range of 45m and horizontal FOV of 360 degree is selected.

6.1.2.1. Simulation Result of Effect of Virtual Target in Intersection Autonomous Driving

In order to evaluate the effect of the virtual target, the longitudinal motion planning with the virtual target has been compared with the based algorithm with no virtual target. The target vehicle is driving at a constant velocity of 50km/h, which is the standard speed limit of urban road in Seoul, Korea, and an initial velocity of 30km/h of ego vehicle is used for the simulation. In order to simulate the most dangerous collision situation during the right turn crossing at the intersection, the initial positions of target vehicle and ego vehicle are 90m and 30m away from the intersection, respectively. The ego vehicle approaches the intersection from south to north, and the target vehicle drives from west to east toward the intersection. In Figure 6.4, the change in clearance and TTCP of ego vehicle has been analyzed to show effect of the virtual target. The Clearance is the distance between target vehicle and ego vehicle to show risk potential of collision during intersection crossing. In Figure 6.4(a), The minimum Clearance of the proposed and base algorithm are 13.08m and 6.62m respectively. For the proposed algorithm, the Clearance decreases from 45m to 26m as the ego vehicle approaches 16m before the conflict point. At 16m before the conflict point, the actual target

vehicle within the FOV is detected, and the ego vehicle plans the motion in regard to the target vehicle, decreasing the Clearance to 13m without change in DTCP. For the base algorithm with no virtual target, since there is no virtual target, the actual target has been detected at about 13m before the conflict point, and the Clearance has diminished to 6m. In Figure 6.4(b), TTCP of ego vehicle of the proposed and base algorithm has been compared. The minimum TTCP of the proposed and the conventional are 0.94 and 0.44s respectively. From the slope of the clearance curve in Figure 6.4, it can be inferred that the proposed algorithm with the virtual target enables the ego vehicle to proactively decelerate and stop with sufficient clearance when encountered with the actual target vehicle.
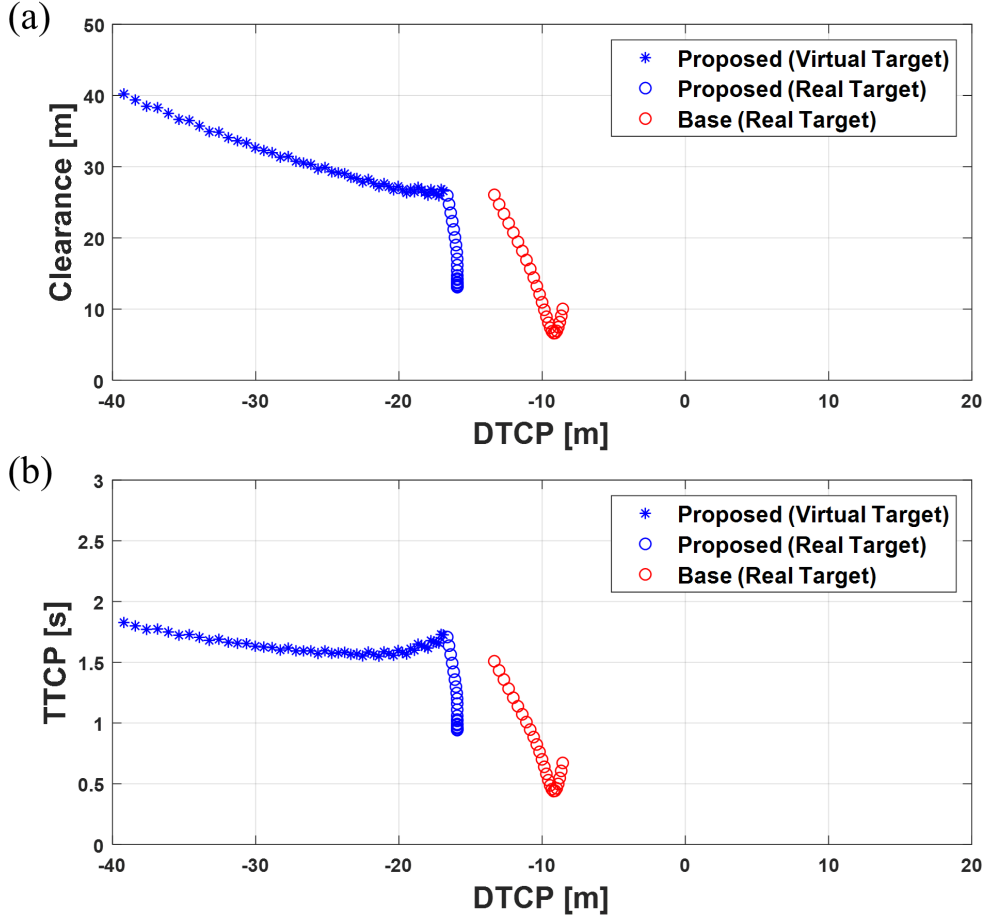
Figure 6.4. Change in safety indices during right turn at intersection (a) Clearance with and without virtual target (b) Time to Conflict Point with and without virtual target

The Monte-Carlo simulation has been conducted 100 times to evaluate the proposed algorithm in various states of target and ego vehicle. The initial velocity of target and ego vehicle has been generated using the normal distribution. The initial velocity of target and ego vehicle is determined as $N(50,10)$ and $N(40,10)$ respectively. The algorithm with no virtual target shows mean minimum clearances

of 0.69m with standard deviation of 1.56m. The minimum and maximum are 0m and 3.13m respectively. The 5 randomly selected simulation cases out of 100 are shown in Figure 6.5. The proposed algorithm with virtual target shows mean minimum clearance of 11.66m with standard deviation of 1.94m. The minimum and maximum clearances are 6.90m and 13.57m, respectively. The 5 randomly selected simulation cases with virtual target out of 100 are shown in Figure 6.6. The simulation results successfully show that the proposed algorithm can reduce the collision risk caused by a sudden appearance of the target vehicle from the occluded region by proactively planning the ego vehicle velocity motion based on the target vehicle.
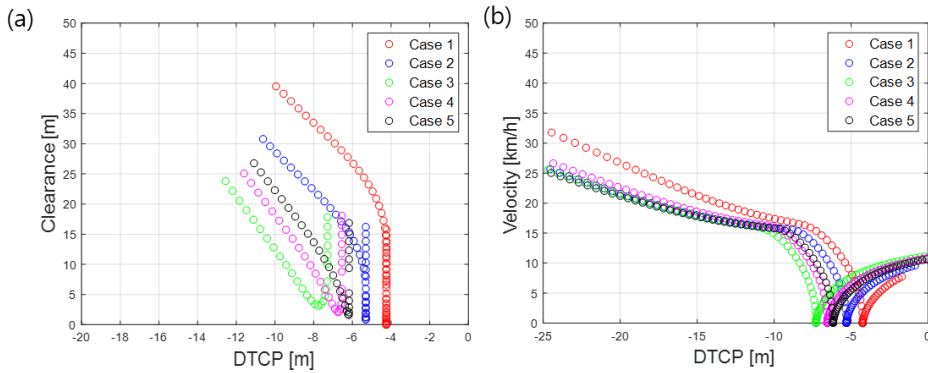
Figure 6.5. Monte Carlo simulation result of intersection crossing with non-virtual target.
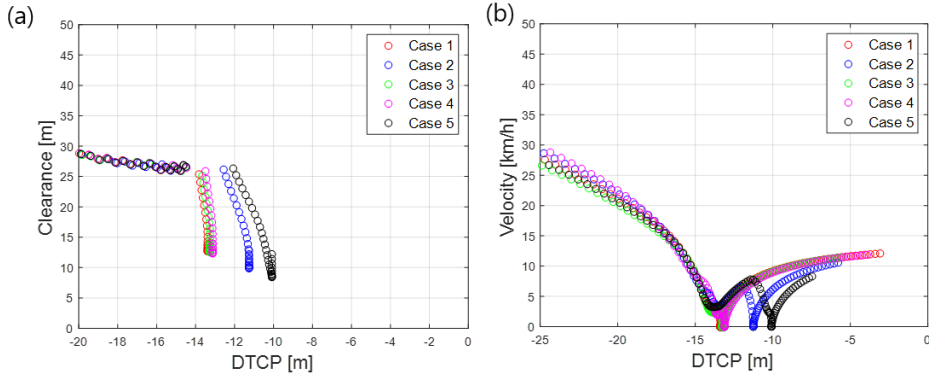
Figure 6.6. Monte Carlo simulation result of intersection crossing with virtual target

The post encroachment time (PET) is another safety index for intersection crossing. PET is the time between the moment that the first vehicle leaves the potential conflict point and the moment that the second vehicle enters the conflict point. The PET is calculated for the Monte Carlo simulation. The mean PET is 3.64s, the standard deviation is 1.66s, the minimum PET is 1.3s, and the maximum PET is 6.1s.

6.1.2.2. Virtual Target Simulation Result of the Right Turn Across Path Scenario in the Intersection

The offline simulation study of the Right Turn Across Path (RTAP) scenario at the intersection is conducted to evaluate the feasibility of human data driven virtual target model to overcome the dead-lock situation. In the signalized intersection, the maneuver of the ego vehicle is often not regulated by traffic signal in the RTAP

case. For the RTAP case, even if the traffic signal is red, if the pedestrian traffic signal is red or no pedestrian is crossing the crosswalk, the vehicle is allowed to make a right turn at the intersection.

For the RTAP scenario, the ego vehicle is driving with an initial velocity of 20km/h and starts at 30m away from the intersection. The offline simulation result of the RTAP scenario with the CV virtual target model is illustrated in Figure 6.7. As shown in Figure 6.7, the ego vehicle decreases a velocity as the predicted trajectory conflicts with the future trajectory of the constant velocity virtual target. However, as shown in Figure 6.7(c), the ego vehicle stops permanently as the predicted trajectory of the constant velocity model virtual target remains the same. The ego vehicle eventually becomes to a 'dead-lock' situation.
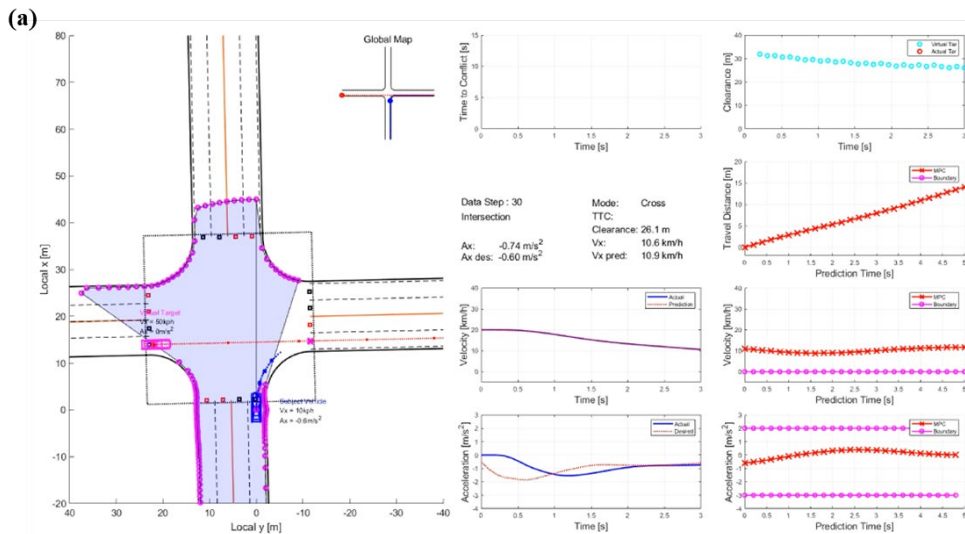
Figure 6.7. Snapshots of the RTAP offline simulation study with the CV based virtual target model (a) t=3s (b) t=5s (c) t=12s.

The offline simulation result of the RTAP scenario with the GPR based virtual target model is illustrated in Figure 6.8. The ego vehicle approaches and decreases the velocity as the predicted ego vehicle trajectory conflicts with future trajectories

of the target vehicle. Then, it is observed that the future trajectory of the virtual target decreases by interacting with the ego vehicle in Figure 6.8 (c). With shortened future trajectory of the virtual target, the ego vehicle manages to turns right at the intersection safely without collision, resolving the 'dead-lock' situation that happened with constant velocity virtual target model.

Figure 6.8. Snapshots of the RTAP offline simulation study with the GPR based virtual target model (a) t=3s (b) t=5s (c) t=9s (d) t=11s.

In Figure 6.9, the simulation results with the GPR based and constant velocity based virtual target model are described over displacement and time domain. For the acceleration, the ego vehicle with the GPR based virtual target model uses -

1.5m/s² acceleration to decrease the velocity and to maintain a distant clearance for the safety. However, the ego vehicle with the constant velocity based virtual target model uses over -2.0m/s² acceleration to lowers the velocity. Such difference in degree of acceleration is probably due to change in future trajectory of GPR based virtual target which interacts with the ego vehicle motion.

Figure 6.9. Comparisons of the RTAP offline simulation results between the GPR based and the CV based virtual target model.

6.1.2.3. Virtual Target Simulation Result of the Straight Across Path Scenario in the Intersection

Like the RTAP scenario that ego vehicle can sometimes ignore the traffic signal to proceed, for the SAP scenario, even if the ego vehicle crosses the intersection in a straight direction with a green light, the ego vehicle should plan the motion in react to the target vehicle turning right from the right side since the target vehicle can make a right turn with a red light.

The offline simulation of the SAP scenario with CV model virtual target is simulated. The ego vehicle is driving with an initial velocity of 40km/h and starts at 60m away from the intersection. The snapshots of offline simulation results of the SAP scenario with virtual target with a constant velocity model are illustrated in Figure 6.10. The virtual target appears at time t=3s and starts to decrease a velocity as the motion planner predicts the possible collision due to the intersection of predicted trajectories from the ego vehicle and the virtual target. From the simulation, the desired acceleration becomes up to $-4m/s^2$ and the actual acceleration is used up to about $-3m/s^2$. Such a degree of acceleration is large enough for drivers and passengers to feel discomfort in ride comfort. Unlike the ego vehicle during the RTAP scenario, the ego does not remain stopped permanently in the SAP scenario since target vehicle, attempting to turn right at the intersection, approaches with relatively low velocity compared to target crossing the intersection straight.
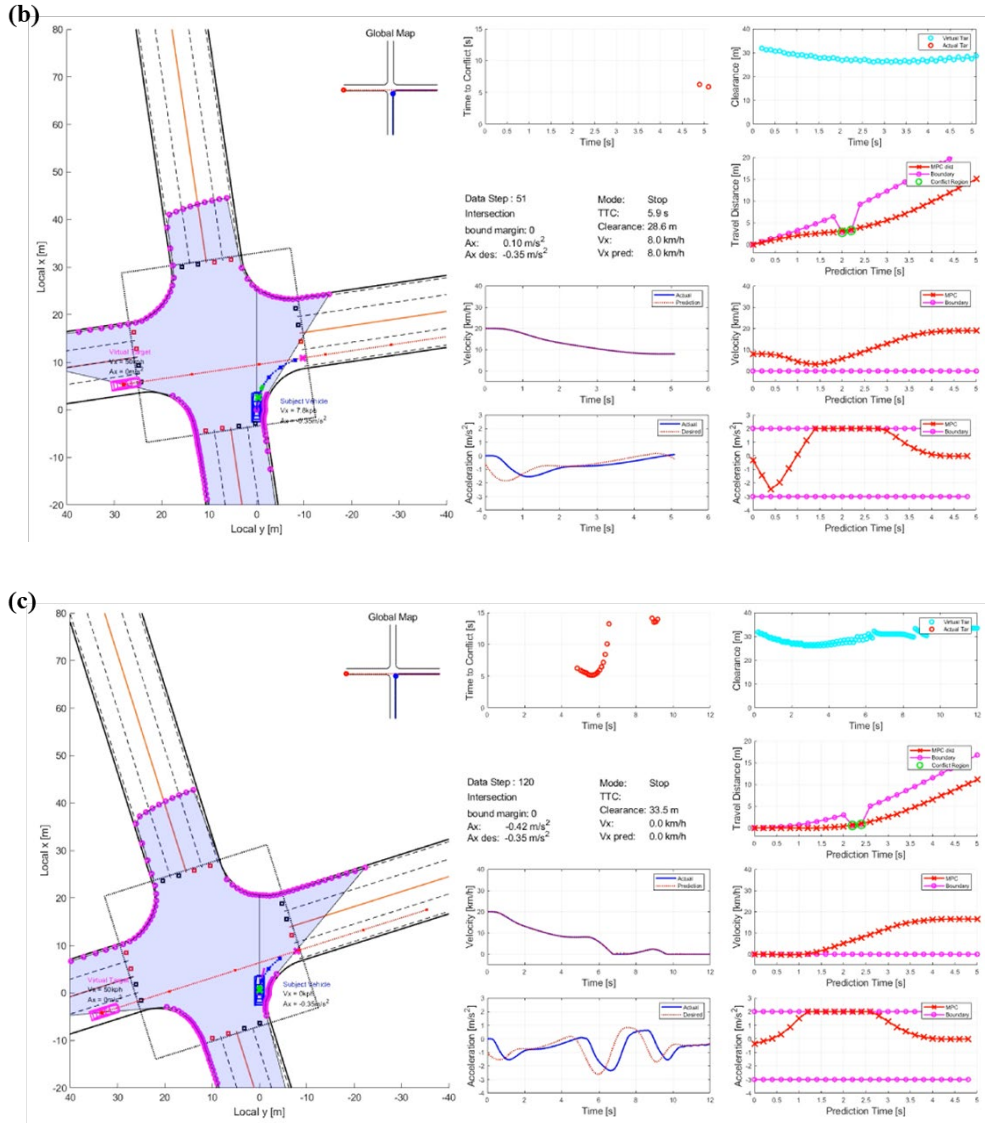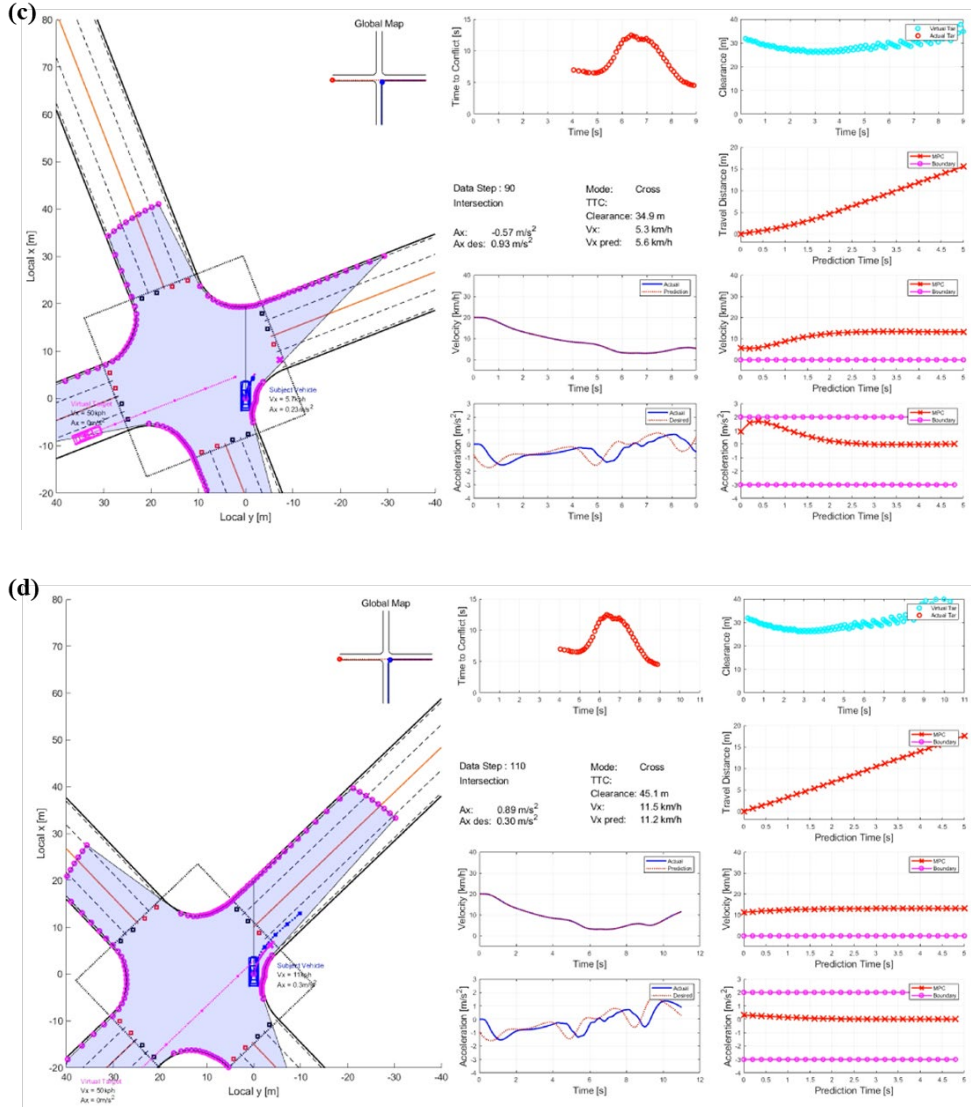
Figure 6.10. Snapshots of the SAP offline simulation study with the CV based virtual target model (a) t=1s (b) t=4s (c) t=5s.

The offline simulation of the SAP scenario with the GPR based model virtual target is simulated. Like simulation study of the virtual target with constant velocity model, the ego vehicle is driving with an initial velocity of 40km/h and 60m away from the intersection. The snapshots of offline simulation results of ego vehicle motion interacting with GPR based virtual target are described in Figure 6.11. The virtual target appears at time t=3s and starts to decrease a velocity as the motion planner predicts the possible collision due to the intersection of predicted trajectories from the ego vehicle and the virtual target. However, the GPR based predicted trajectory becomes shortened and resolves the collision risk quickly, managing the ego vehicle to use less degree of acceleration. From the simulation, the desired acceleration becomes up to -3m/s² and the actual acceleration is used up to about -2m/s². Such a degree of acceleration is much less than the one from the

CV model virtual target simulation, providing the enhanced ride comfort to drivers.

(a)

(b)

Figure 6.11. Snapshots of the SAP offline simulation study with the GPR based virtual target model (a) t=1s (b) t=3s (c) t=4s.

Simulation results with different parameters over distance to the conflict point and time with the GPR based and the CV based virtual target model are described in Figure 6.12. For the acceleration, the ego vehicle with the GPR based virtual target model uses -2.0m/s² acceleration to decrease the velocity and to manage a safety clearance. However, the ego vehicle with the constant velocity based virtual target model uses over -3.0m/s² acceleration to lowers the velocity. From the time domain graph, it is observed that the ego vehicle interacting the GPR based virtual target starts to decelerate earlier than the ego interacting the virtual target with CV model. The early initial deceleration provides and ensures drivers and passengers safety when crossing the intersection. From Figure 6.12 (e) and (f), it is cleared that the predicted trajectory of ego vehicle interacts earlier about 10m in distance domain and 1s in time domain. Since the ego vehicle approaches to the intersection

with 40km/h which is near speed limit of the urban road and crosses the intersection in short period of time, the difference in clearance change between GPR model and CV model is negligible and similar.
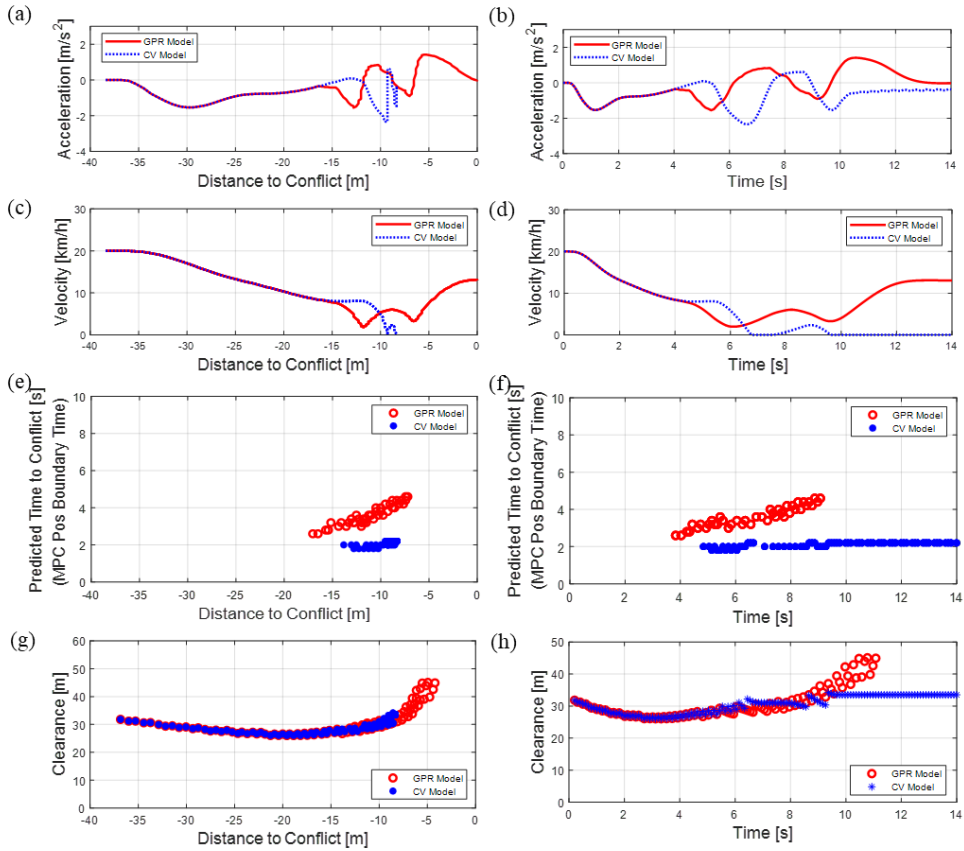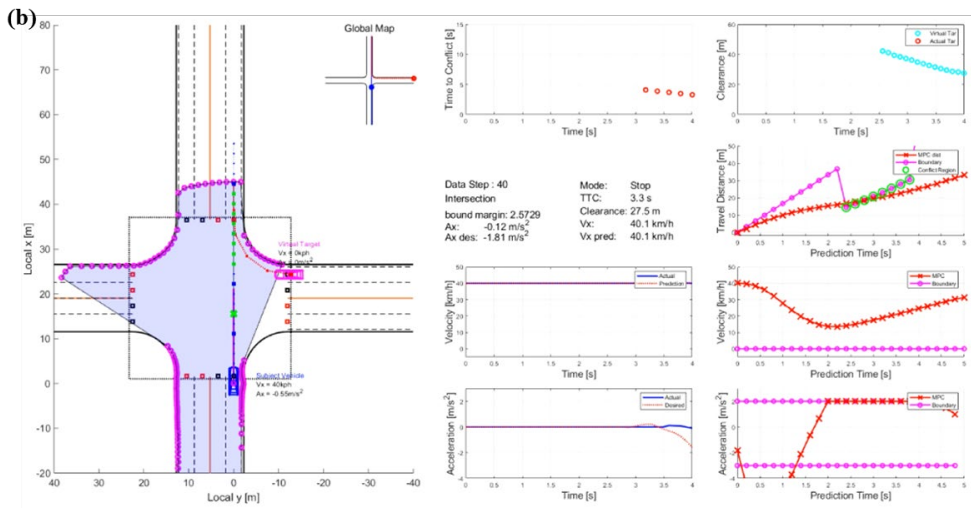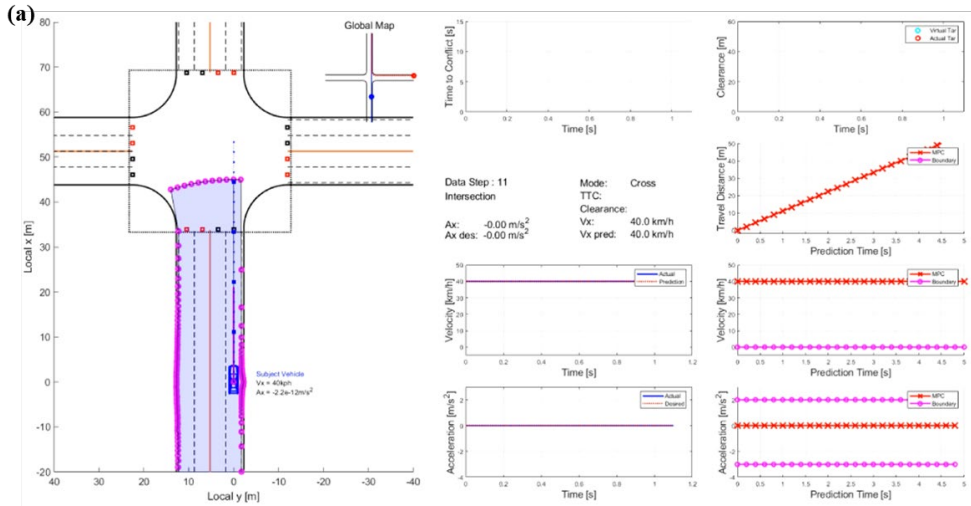

Figure 6.12. Comparisons of the SAP offline simulation results between the GPR based and the CV based virtual target model.

6.1.2.4. Virtual Target Simulation Result of the Left Turn Across Path Scenario in the Intersection

The offline simulation of the LTAP scenario with CV model virtual target is simulated. The ego vehicle is driving with an initial velocity of 40km/h and starts at 60m away from the intersection. The snapshots of offline simulation results of the LTAP scenario with virtual target with a constant velocity model are illustrated in Figure 6.13. The virtual target appears at time t=2s and the deceleration command for autonomous vehicle is applied from time t = 4 to time t = 6. Then, since the clearance between the autonomous vehicle and the virtual target at the initial appearance of virtual target is near the maximum distance of FOV, as the autonomous enters the intersection, the clearance remains constant at the maximum distance during the scenario. The maximum clearance from the beginning to the end of the autonomous vehicle crossing the intersection rapidly discards the conflict region between the virtual target and the autonomous vehicle.
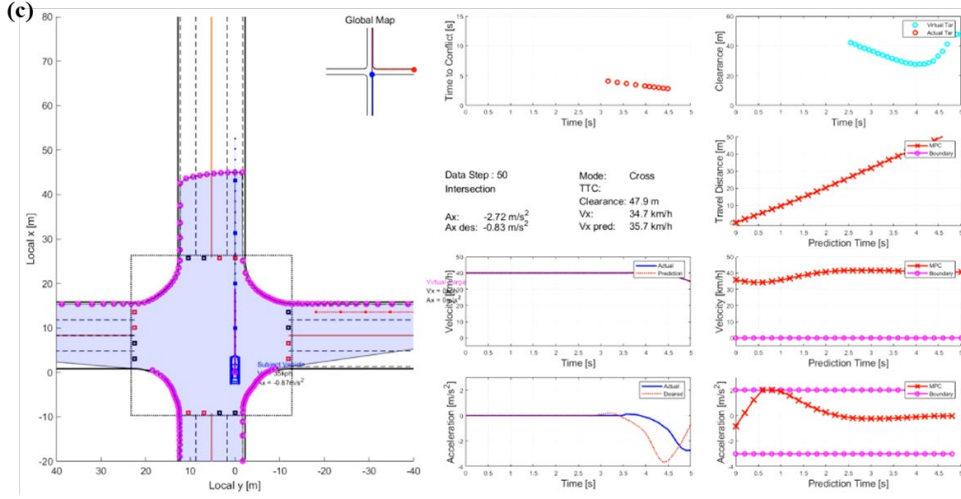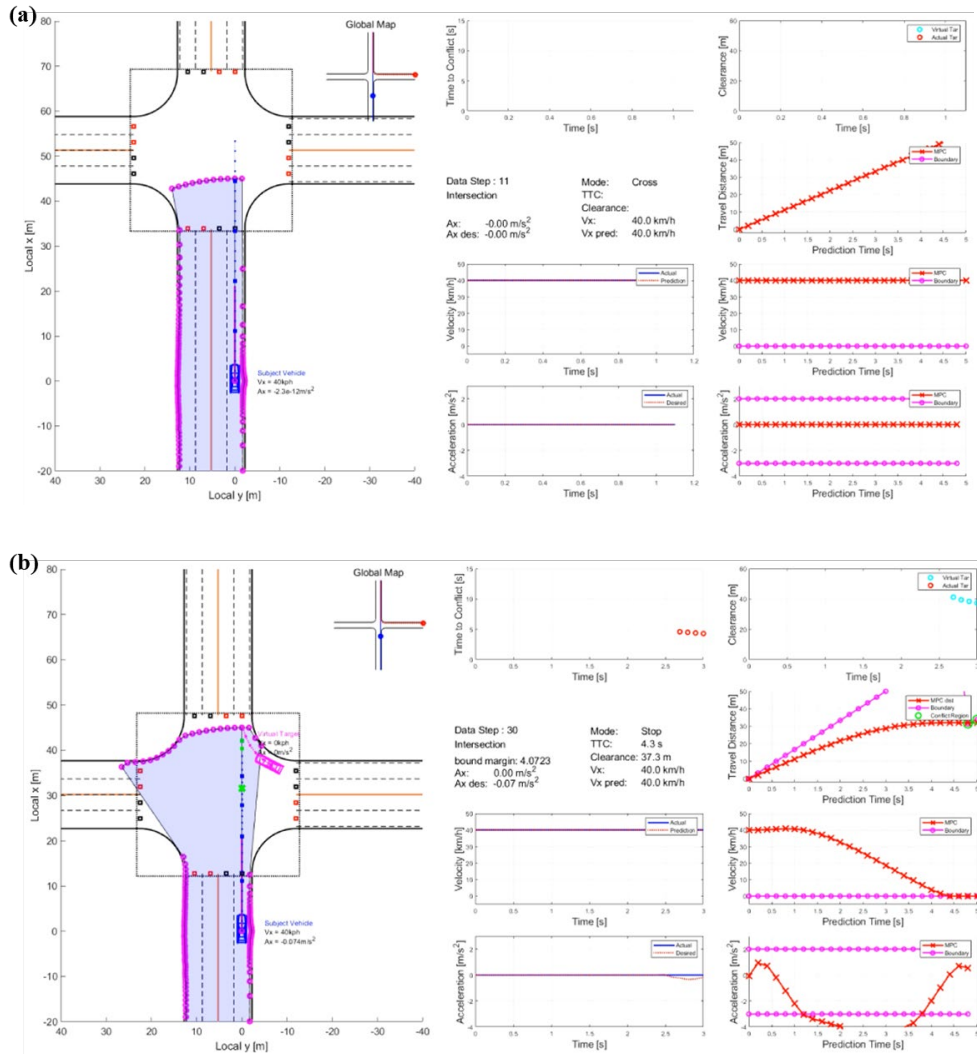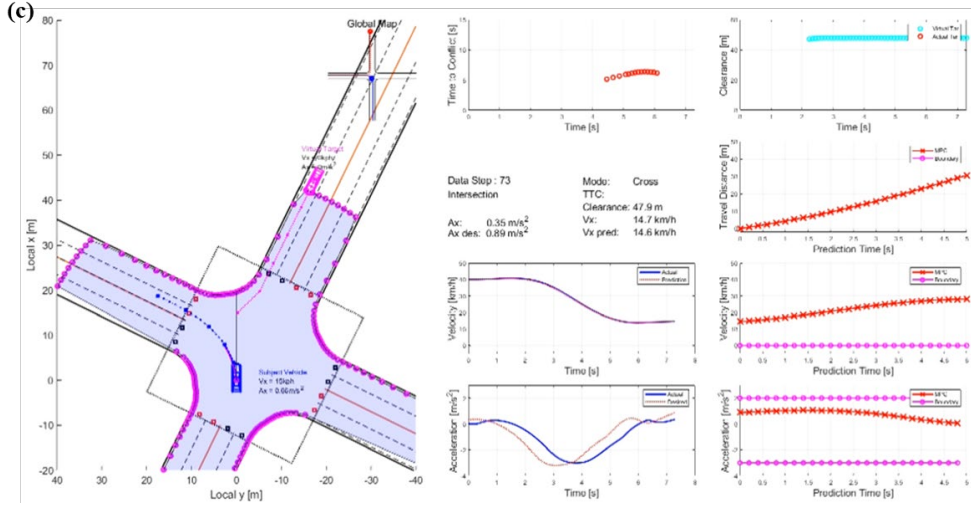
Figure 6.13. Snapshots of the LTAP offline simulation study with the CV based virtual target model (a) t=2s (b) t=5s (c) t=7s.

The offline simulation of the LTAP scenario with the GPR based model virtual target is simulated. Like simulation study of the virtual target with a constant velocity model, the ego vehicle is driving with an initial velocity of 40km/h and 60m away from the intersection. The snapshots of offline simulation results of ego vehicle motion interacting with GPR based virtual target are described in Figure 6.14. The virtual target appears at time t=2s and it is observed that the predicted trajectory of the virtual target is shorter than that of constant velocity modeled virtual target. Due to the predicted short trajectory of GPR based virtual target, the collision region is not generated during the LTAP scenario, and the autonomous vehicle is able to cross the intersection without any deceleration command.
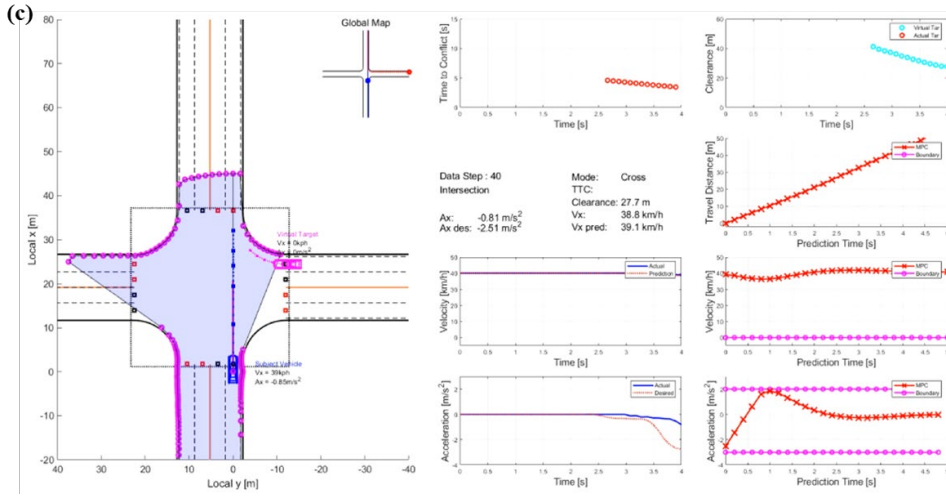
(a)

(b)

111
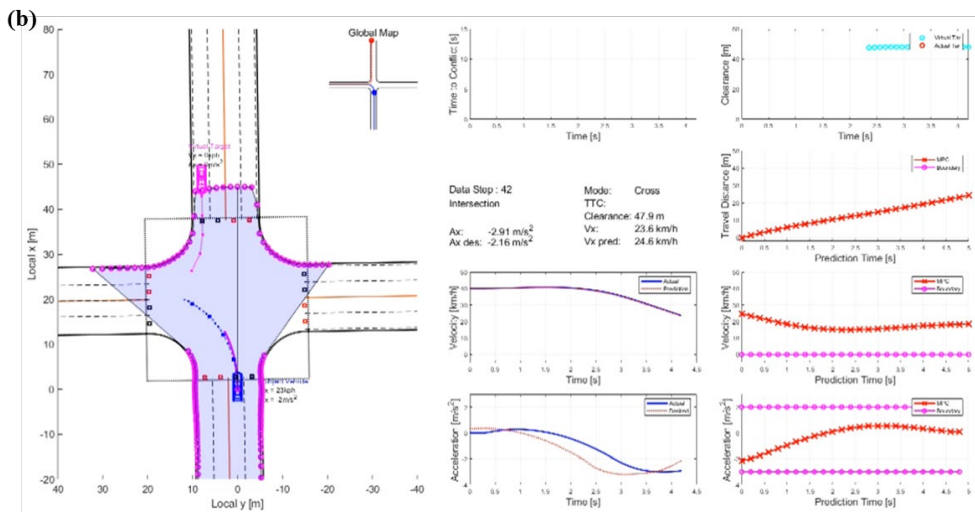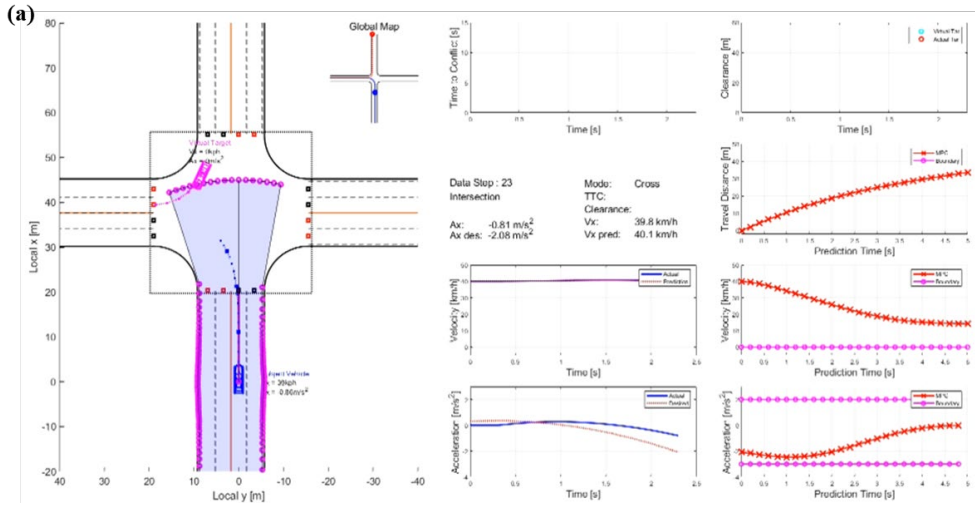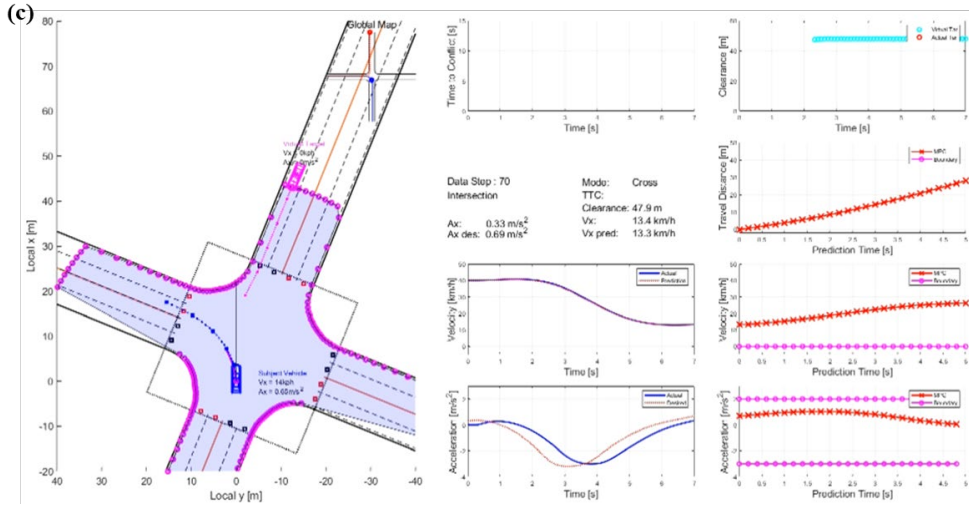
Figure 6.14. Snapshots of the LTAP offline simulation study with the GPR based virtual target model (a) t=2s (b) t=4s (c) t=7s.

Simulation results with various parameters over distance to the conflict point and time with the GPR based and the CV based virtual target model are depicted in Figure 6.15. For the acceleration, the ego vehicles with both the GPR based virtual target model and the CV based virtual target model use -3.0m/s2 acceleration to decrease the velocity and to manage a safety clearance. From both the distance and the time domain graph, the acceleration and velocity profiles from two different virtual target models turn out to be similar. From Figure 6.15 (e), it is observed that although the conflict region is generated from the CV based virtual target, the conflict region disappeared rapidly and the effect on the degree of acceleration is negligible.
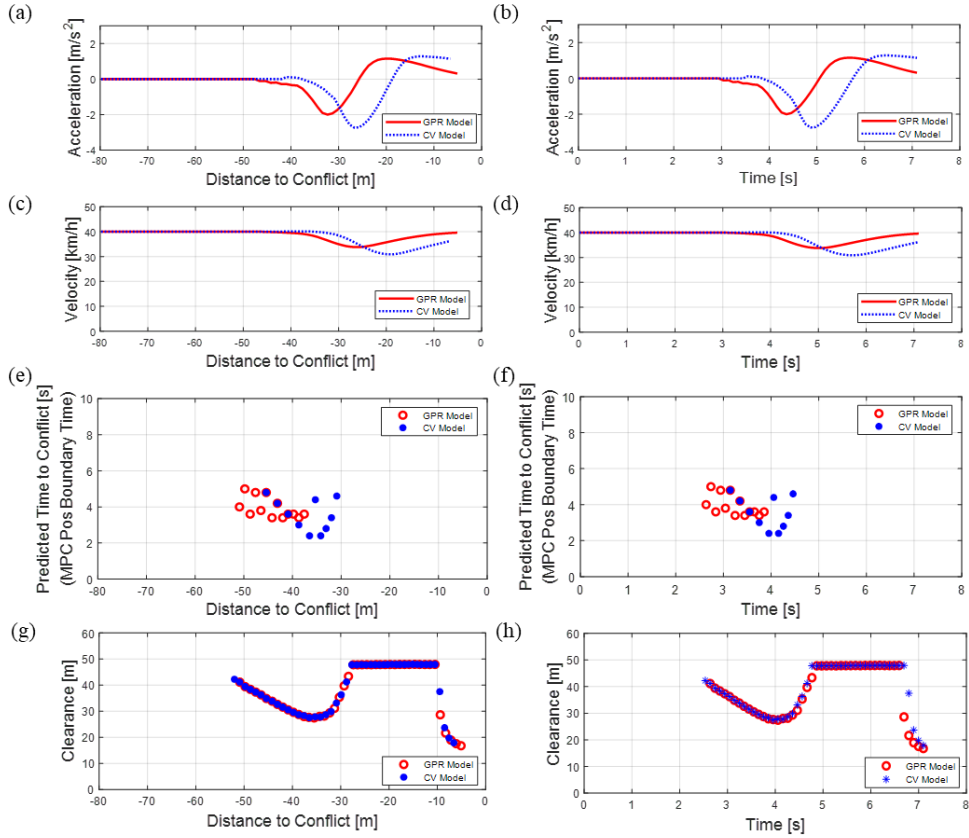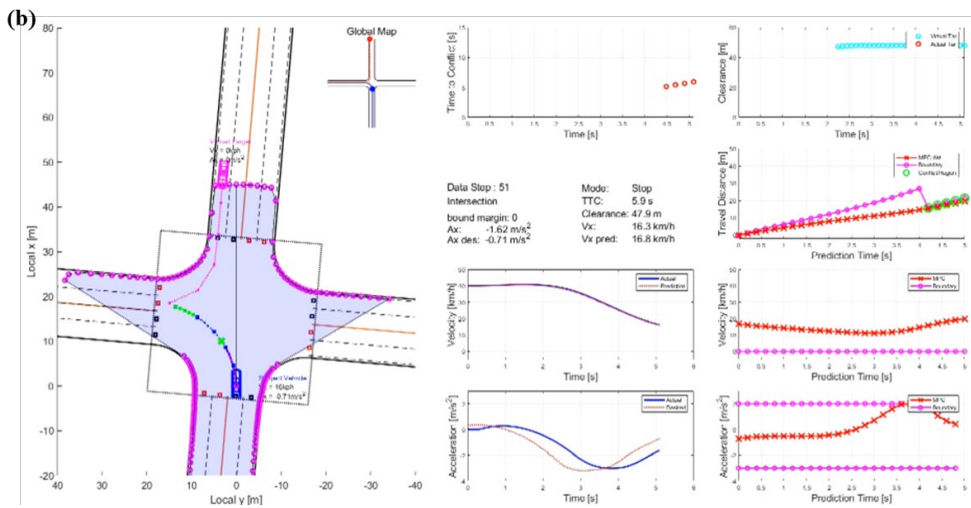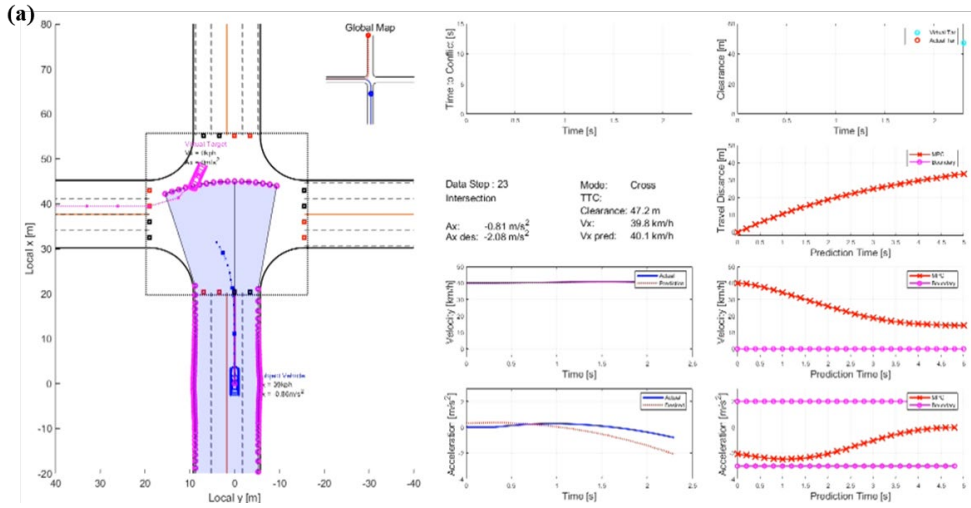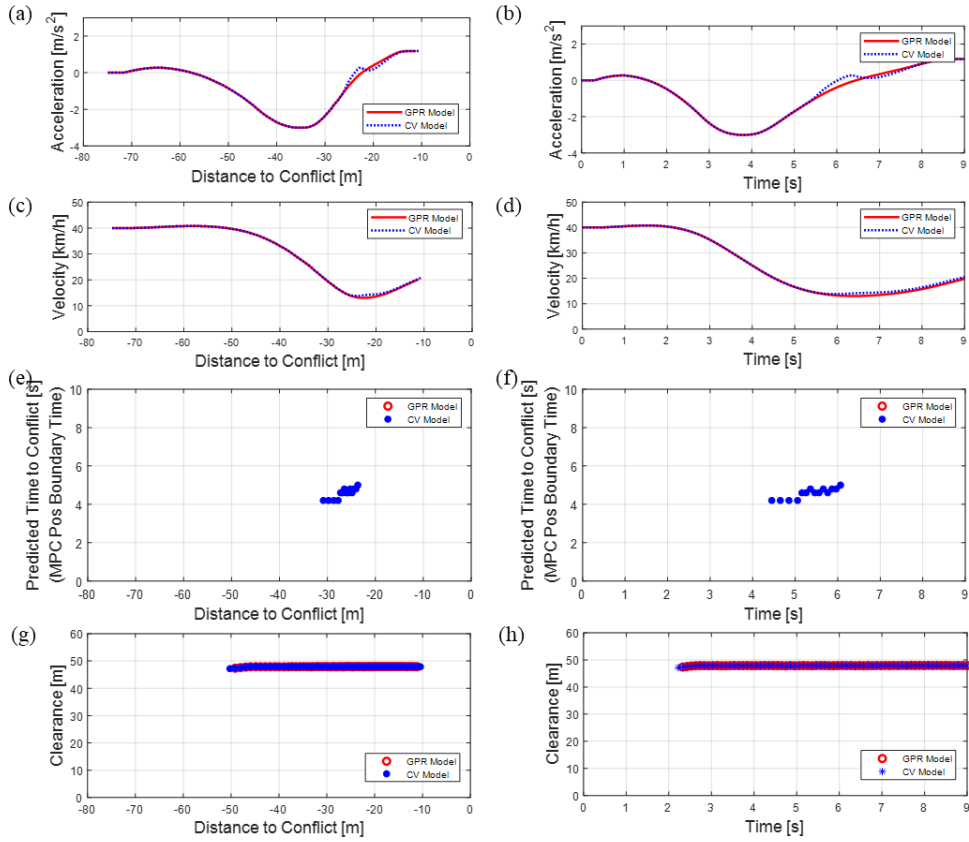
Figure 6.15. Comparisons of the LTAP offline simulation results between the GPR based and the CV based virtual target model.

6.1.2.5. Virtual Target Simulation Result of Crooked T-shaped Intersection

The simulation studies for the virtual target prediction are performed to evaluate the feasibility of the proposed algorithm to overcome the dead-lock situation at severely crooked T-shaped intersection. According to the Ministry of Land, Infrastructure and Transportation, the two roads intersecting at the intersection should maintain the angle between 75 and 105 degrees. In order to

evaluate the effectiveness of the virtual target model at the complex intersection, the T-shaped intersections with the intersecting angle of 75 degree are constructed. The simulation environment is a 3-way intersection where each two two-lane roads intersect at 75 degrees as shown in Figure 6.16. Road length before and after the intersection is 100m each, road width is 3.5m, and the radius of road curvature is 10m to reflect the actual urban road geometry of Seoul, Korea. Figure 6.16 (a) shows the intersection with the rounded corner, and Figure 6.16 (b) describes the intersection with the sharp corner that obstruct the FOV of the approaching vehicles.

(a)

(b)
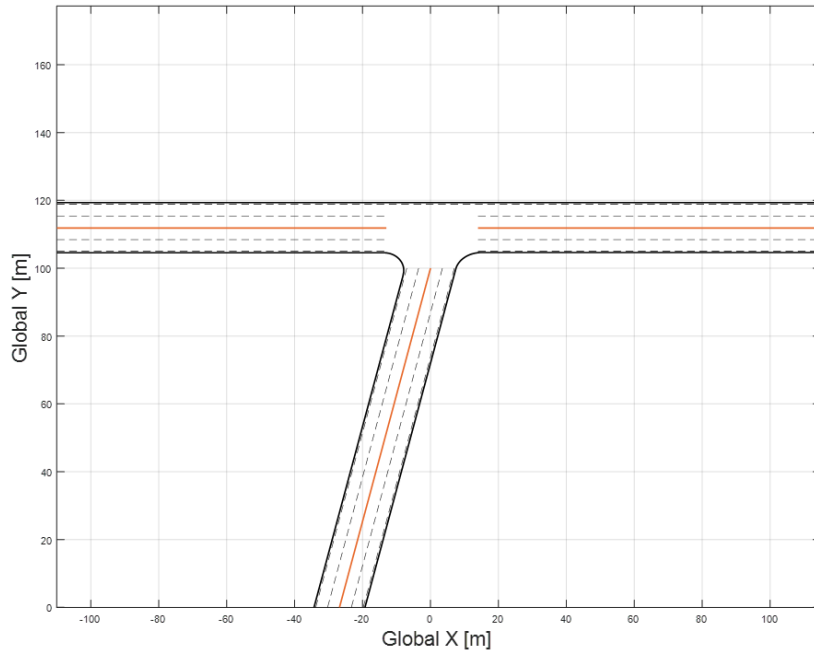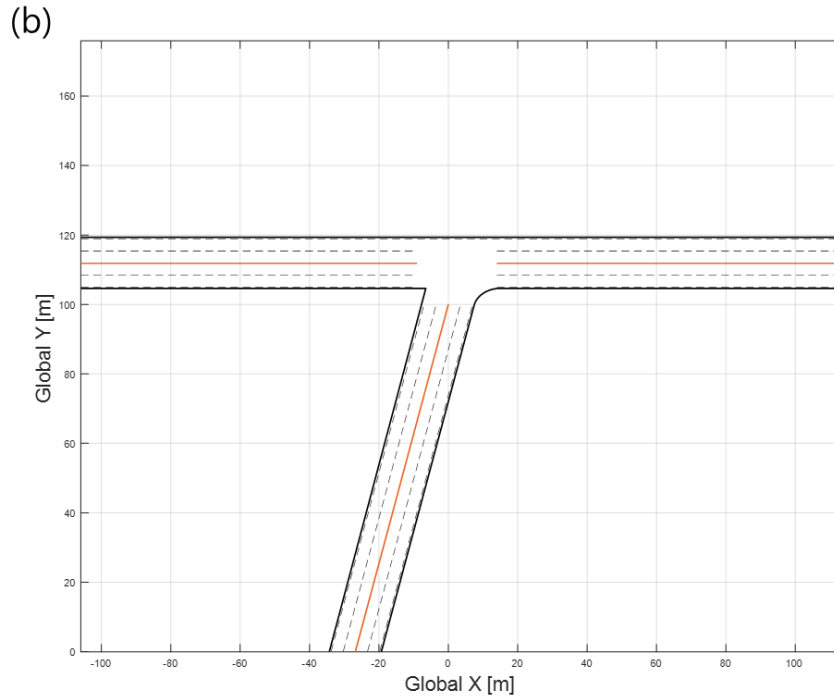
Figure 6.16. T-shaped intersection with intersecting angle of 75 degree (a) rounded corner (b) sharp corner.

The offline simulation of the RTAP scenario with CV model virtual target is simulated at the intersection in Figure 6.16 (a). The ego vehicle is driving with an initial velocity of 30km/h and starts at 30m away from the intersection. The snapshots of offline simulation results of the RTAP scenario with virtual target with a constant velocity model are illustrated in Figure 6.17. The virtual target appears, and the ego vehicle starts to decrease a velocity as the motion planner predicts the possible collision due to the intersection of predicted trajectories from the ego vehicle and the virtual target. From the simulation, the desired acceleration becomes up to -4m/s2 and the actual acceleration is used up to about -3m/s2. Such

a degree of acceleration is large enough for drivers and passengers to feel discomfort in ride comfort. As shown in Figure 6.17 (b), the ego vehicle stops permanently as the predicted trajectory of the constant velocity model virtual target remains the same. The ego vehicle eventually becomes to a 'dead-lock' situation.

Figure 6.17. Snapshots of the RTAP T-shaped intersection offline simulation study with the CV based virtual target model (a) t=4s (b) t=10s.

The offline simulation result of the RTAP scenario with the GPR based virtual target model at T-shaped intersection with smooth corner is illustrated in Figure 6.18. The ego vehicle approaches and decreases the velocity as the predicted ego vehicle trajectory conflicts with future trajectories of the target vehicle. Then, it is observed that the future trajectory of the virtual target decreases by interacting with the ego vehicle in Figure 6.18 (b). With shortened future trajectory of the virtual target, the ego vehicle manages to turn right at the intersection safely without collision, resolving the 'dead-lock' situation that happened with constant velocity virtual target model.

Figure 6.18. Snapshots of the RTAP T-shaped intersection offline simulation study with the GPR based virtual target model (a) t=3s (b) t=5s (c) t=10s.

In Figure 6.19, the simulation results with the GPR based and constant velocity based virtual target model in a T-shaped intersection are described. For the acceleration, the ego vehicle with the GPR based virtual target model uses -3m/s² acceleration to decrease the velocity and to maintain a distant clearance for the

safety. However, the ego vehicle with the constant velocity based virtual target model uses over -3.5m/s^2 acceleration to lowers the velocity. Such difference in degree of acceleration is probably due to change in future trajectory of GPR based virtual target which interacts with the ego vehicle motion. Moreover, the decrease in the future trajectory of GPR based virtual target allows the self-driving vehicle to overcome the dead-lock situation.
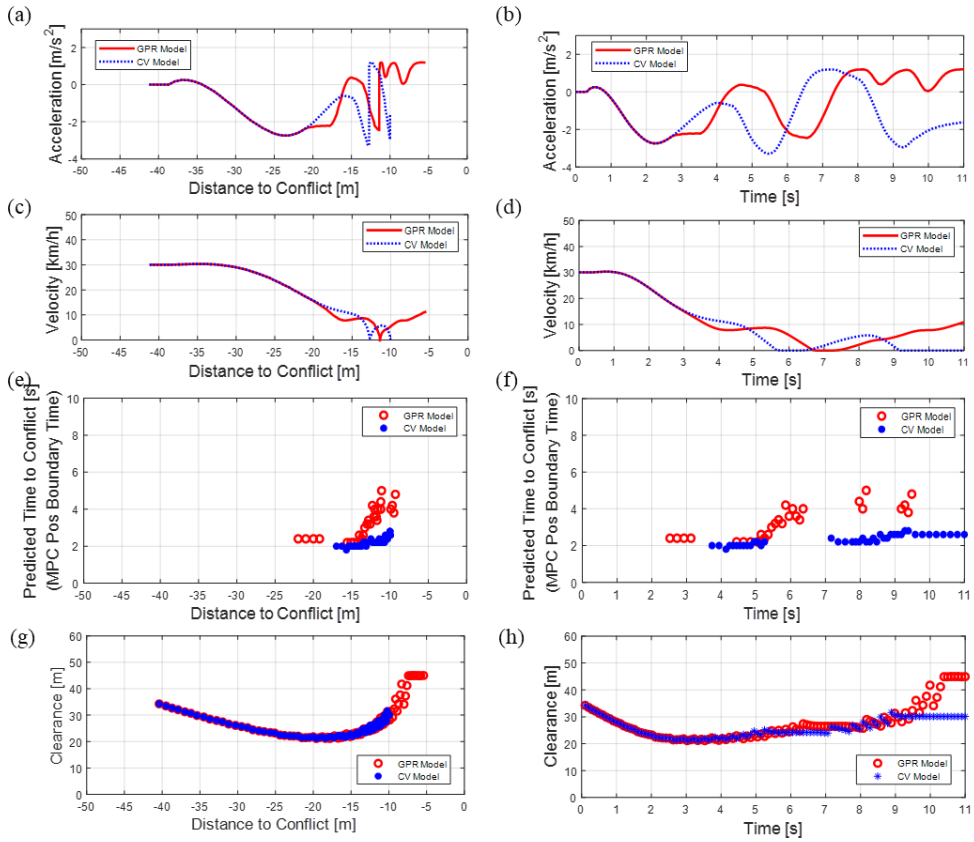
Figure 6.19. Comparisons of the RTAP T-shaped intersection offline simulation results between the GPR based and the CV based virtual target model.

The offline simulation of the RTAP scenario with GPR model virtual target is simulated at the T-shaped intersection with severe corner in Figure 6.16 (b). The ego vehicle is driving with an initial velocity of 30km/h and 30m away from the intersection. The snapshots of offline simulation results of the RTAP scenario with virtual target with a constant velocity model are illustrated in Figure 6.20. The virtual target appears, and the ego vehicle starts to decrease a velocity as the motion planner predicts the possible collision due to the intersection of predicted trajectories from the ego vehicle and the virtual target. From the simulation, the desired acceleration becomes up to -4m/s^2 and the actual acceleration is used up to about -3m/s^2. Since the intersection corner is sharp, the insufficient FOV is achieved for the autonomous vehicle to predict proper future trajectory of virtual target vehicle. As shown in Figure 6.20 (b), the ego vehicle stops permanently as the predicted trajectory of the GPR model virtual target remains the same. The ego vehicle eventually becomes to a 'dead-lock' situation even with GPR based virtual target model.
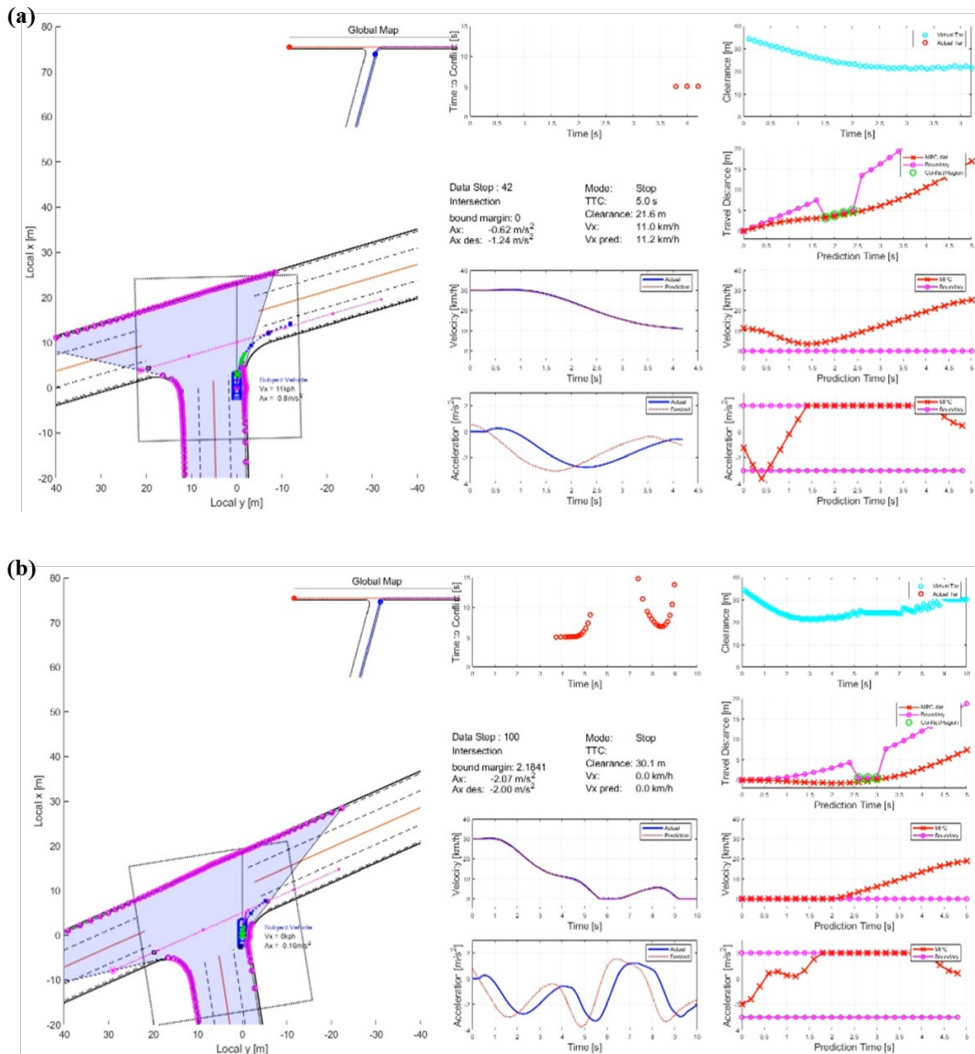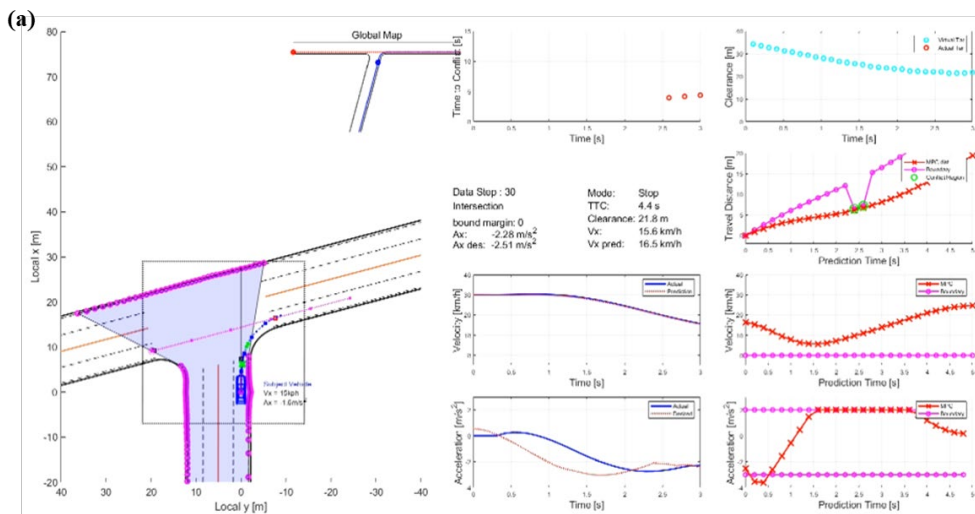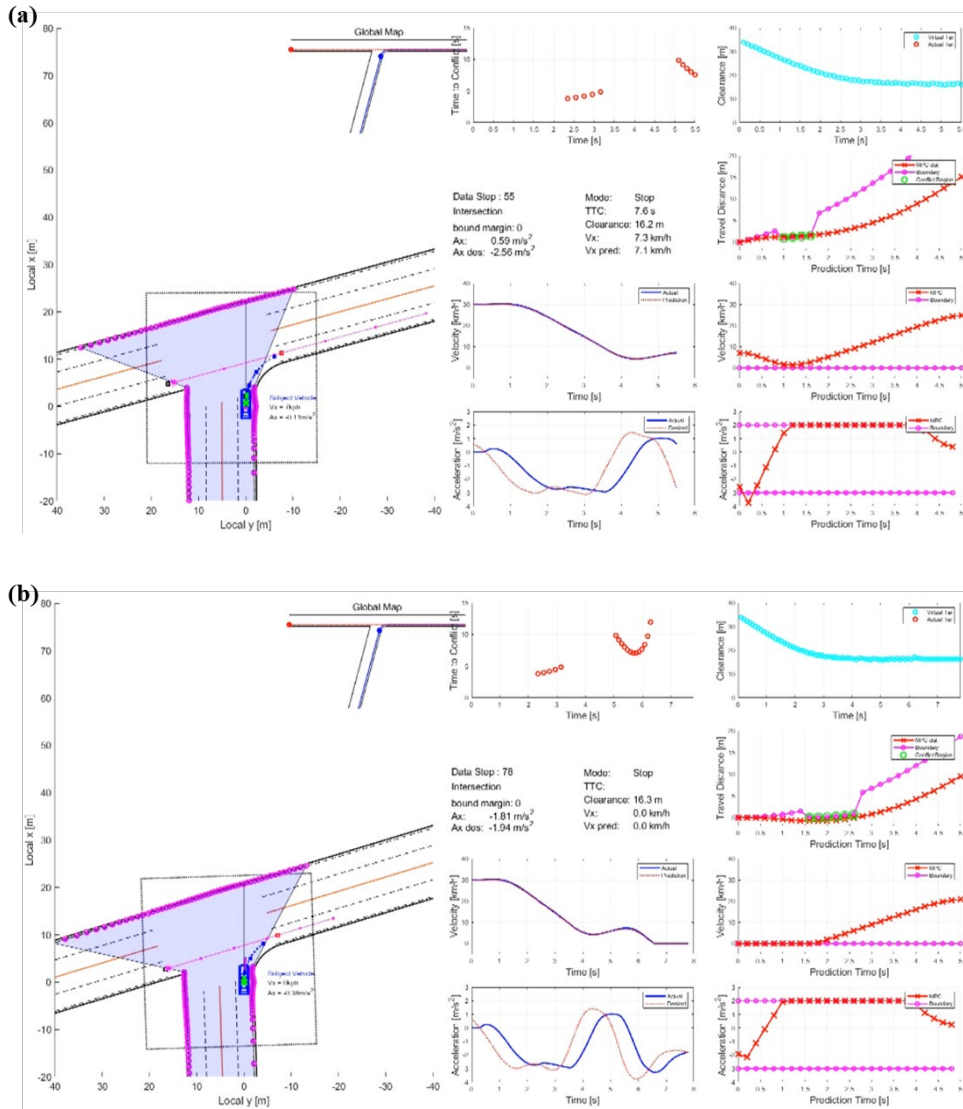
Figure 6.20. Snapshots of the RTAP severe T-shaped intersection offline simulation study with the GPR based virtual target model (a) t=5s (b) t=8s.

In order to overcome the dead-lock situation that can't be resolved with GPR based virtual target model, the creep mode is implemented in addition to the GPR based virtual target model. The effectiveness and snapshots of offline simulation

results of the proposed approaches at the RTAP scenario are illustrated in Figure 6.21. The virtual target appears, and the ego vehicle starts to decrease a velocity as the motion planner predicts the possible collision due to the intersection of predicted trajectories from the ego vehicle and the virtual target. In Figure 6.38, the ego vehicle becomes dead-lock situation and remains stopped permanently. However, in Figure 6.21 (b), the creep mode is activated to allow the autonomous vehicle to move forward slowly. In Figure 6.21 (c), the ego vehicle enters to the target lane, and the cross mode is activated to give "Right-of-way" to the ego vehicle
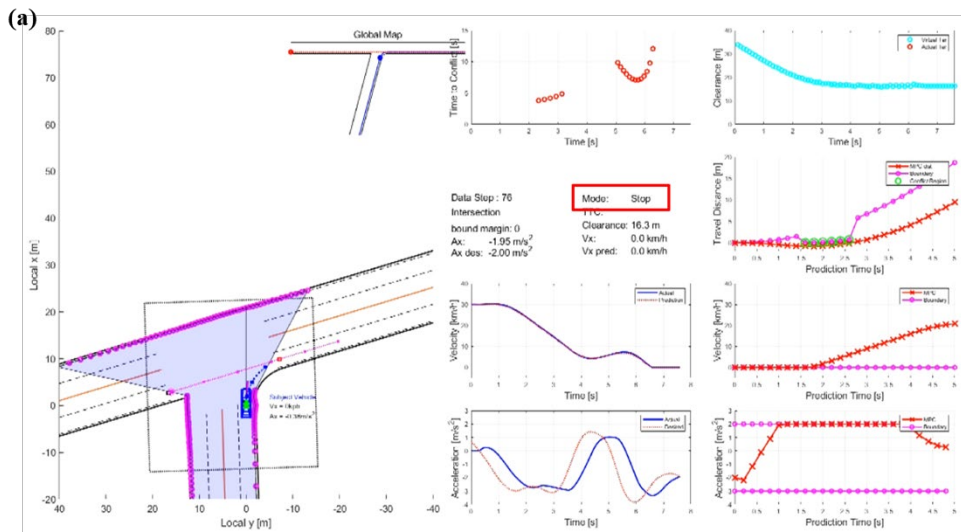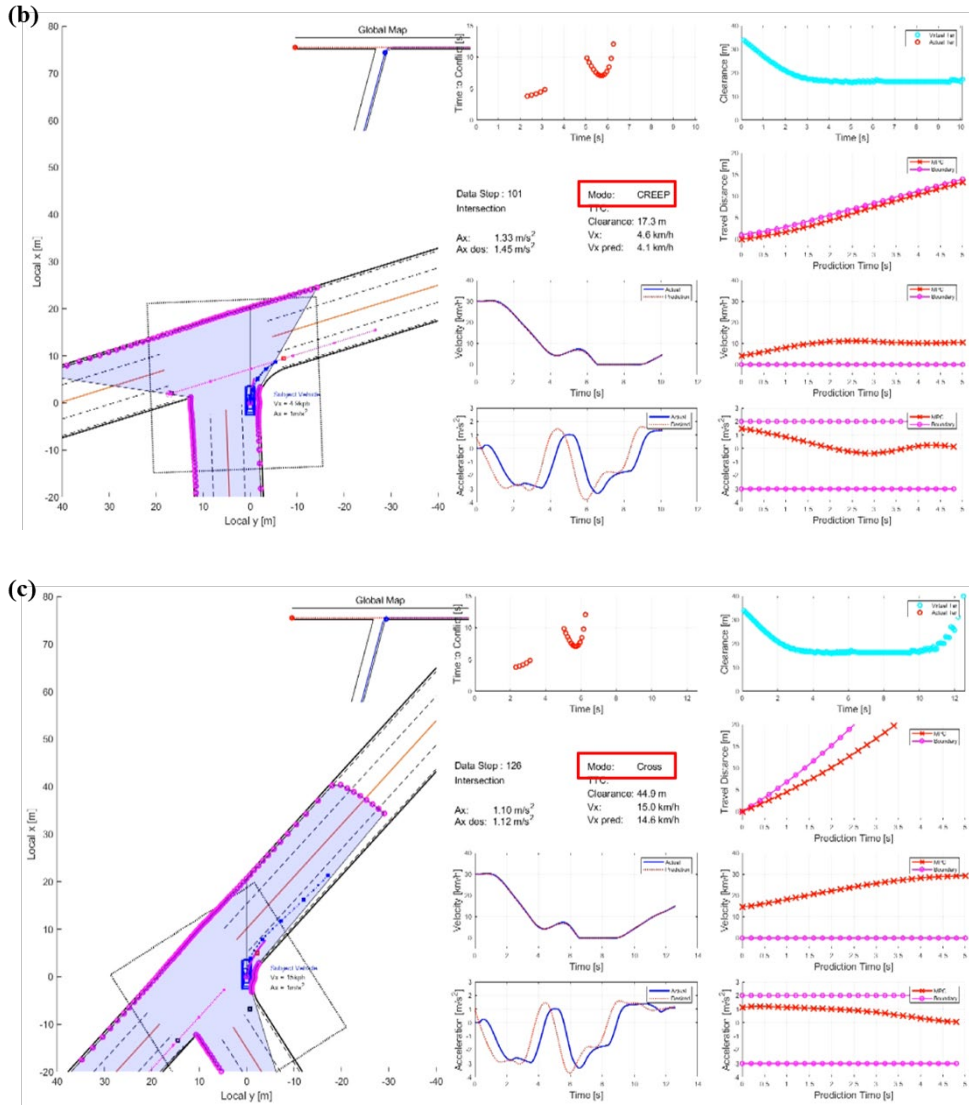
Figure 6.21. Snapshots of the RTAP severe T-shaped intersection offline simulation study with the GPR based virtual target model and the creep mode (a) t=7s (b) t=10s (c) t=12s.

6.2. Performance Evaluation of Data-driven Vehicle State based Motion Prediction at Intersection

6.2.1. Data-driven Motion Prediction Accuracy Analysis

The prediction analysis for the LSTM training result with different input feature candidate is performed to determine the optimal combination of input features. Moreover, the LSTM training results with input features in ego vehicle centered coordinate and target vehicle centered coordinate are analyzed to examine the effect of the coordinates. The total of 14 different input features and history horizon combinations are described in Table 12. The prediction errors of each input feature candidate are computed in the ego coordinates and shown in the table. The error results are analyzed in prediction time of 1s, 2s, 3s, 4s, and 5s, and MAE, RMSE, and STD are computed. The case 1 through 7 represent the input feature candidates with history horizon of 1s, and the case 8 through 15 express the same input feature combination from the case 1 through 7 but only different history horizon of 2s. For the input feature candidates with the history horizon of 1s, the case 6 shows the best prediction performance in prediction time of 1s, and the case 7 shows the best prediction results after the prediction time of 2s. For the input feature candidates with the history horizon of 2s, the case 14 shows the best prediction performance. It can be observed that the effect of difference in prediction horizon of 1s and 2s are negligible in prediction result. Both prediction results from input feature candidate with history horizon of 1s and 2s show the

similar results. The propose LSTM model is also compared with the Constant Turn Rate Velocity (CTRV) model. The prediction result of CTRV model at prediction time of 1s show the similar performance when compared with that of the proposed LSTM model. However, the error becomes greater as the prediction horizon increases.

Table 12. Prediction Errors of the Future Trajectories from the Model Trained in Ego Centered Coordinate

Case	Position Error at Predicted Steps (m) : MAE | RMSE | STD														
	1s			2s			3s			4s			5s		
1	0.58	0.80	0.56	0.85	1.27	0.94	1.20	1.80	1.34	1.67	2.53	1.90	2.30	3.45	2.58
2	0.32	0.39	0.23	0.41	0.51	0.32	0.59	0.77	0.49	0.80	1.10	0.75	1.19	1.70	1.21
3	0.45	0.63	0.44	0.52	0.66	0.40	0.68	0.88	0.55	0.90	1.22	0.83	1.26	1.80	1.28
4	0.60	0.71	0.38	0.67	0.79	0.42	0.77	0.94	0.55	0.91	1.18	0.75	1.18	1.57	1.03
5	0.47	0.60	0.36	0.60	0.72	0.40	0.72	0.88	0.51	0.85	1.06	0.63	1.09	1.38	0.84
6	0.33	0.43	0.26	0.40	0.49	0.28	0.50	0.61	0.35	0.60	0.75	0.45	0.81	1.04	0.65
7	0.36	0.45	0.27	0.36	0.48	0.31	0.43	0.56	0.35	0.53	0.68	0.43	0.71	0.89	0.53
8	0.49	0.65	0.43	0.76	1.02	0.69	1.05	1.49	1.06	1.43	2.17	1.64	1.96	3.14	2.46
9	0.36	0.44	0.25	0.45	0.53	0.29	0.61	0.76	0.46	0.83	1.10	0.72	1.17	1.65	1.16
10	0.37	0.48	0.30	0.42	0.55	0.35	0.58	0.74	0.47	0.78	1.06	0.71	1.12	1.57	1.11
11	0.50	0.63	0.37	0.57	0.70	0.40	0.64	0.77	0.44	0.71	0.88	0.51	0.89	1.11	0.65
12	0.31	0.40	0.25	0.36	0.44	0.25	0.51	0.62	0.35	0.65	0.82	0.49	1.00	1.29	0.81
13	0.38	0.46	0.26	0.47	0.56	0.31	0.57	0.69	0.39	0.64	0.79	0.47	0.74	0.94	0.58
14	0.30	0.36	0.19	0.33	0.39	0.20	0.44	0.53	0.29	0.62	0.76	0.44	0.83	1.05	0.63
CTRV	0.48	0.73	0.56	1.52	2.32	1.75	3.26	4.76	3.47	5.71	7.98	5.57	8.84	11.9	7.96

* CTRV : Constant Turn Rate Velocity Model

The prediction results of LSTM model with the optimal input feature candidates with different history horizon and CTRV model are depicted as a histogram in Figure 6.22. From the histogram, it can be inferred that the prediction errors from the LSTM model with history horizon of 1s are more clustered to the

zero compared to the errors from the model with history horizon of 2s and the CTRV model. Histograms of two different models and CTRV model at each predicted time horizon are depicted in Figure 6.23. The LSTM (A) is with history horizon of 1s, and the LSTM (B) is with history horizon of 2s.

Figure 6.22. Histogram of prediction errors over prediction horizons of models trained in ego centered coordinate (a) LSTM with history horizon 1s (b) LSTM with history horizon 2s (c) CTRV model.

Figure 6.23. Histogram of prediction errors of models with different history horizons trained in ego centered coordinate over different prediction horizon times (a)1s (b)2s (c)3s (d)4s (e)5s.

The prediction errors of each input feature candidate are computed in the target vehicle coordinates and shown in Table 13. As already done with input feature candidates measured in ego vehicle centered coordinate, the error results from input features with target vehicle centered coordinate are analyzed in prediction time of 1s, 2s, 3s, 4s, and 5s, and MAE, RMSE, and STD are computed. The total of 14 different input feature and history horizon combinations are described in the previous Table 7. For the input feature candidates with both history horizon of 1s and 2s, the case 7 and 14 which are composed of the same input feature combinations show the best prediction performance. It can be inferred that the difference in prediction horizon between 1s and 2s is negligible in prediction result. The prediction result of CTRV model at prediction time of 1s show the similar performance when compared with that of the proposed LSTM model, and the error becomes greater as the prediction horizon increases.

Table 13. Prediction Errors of the Future Trajectories from the Model Trained in

Target Centered Coordinate

Case	Position Error at Predicted Steps (m) : MAE | RMSE | STD														
	1s			2s			3s			4s			5s		
1	0.28	0.55	0.48	0.87	1.57	1.31	1.82	3.00	2.38	3.14	4.76	3.58	4.68	6.74	4.85
2	0.19	0.47	0.43	0.70	1.39	1.20	1.50	2.55	2.06	2.56	4.02	3.09	3.93	5.75	4.20
3	0.18	0.47	0.44	0.63	1.33	1.17	1.36	2.38	1.95	2.46	3.82	2.92	3.74	5.54	4.09
4	0.16	0.23	0.18	0.45	0.62	0.43	0.93	1.30	0.90	1.60	2.28	1.63	2.51	3.54	2.50
5	0.12	0.26	0.23	0.28	0.52	0.44	0.46	0.73	0.57	0.67	1.00	0.74	1.02	1.45	1.04
6	0.10	0.13	0.09	0.20	0.25	0.16	0.31	0.40	0.26	0.44	0.57	0.37	0.65	0.83	0.52
7	0.09	0.13	0.09	0.18	0.25	0.17	0.27	0.37	0.26	0.38	0.52	0.35	0.55	0.74	0.50
8	0.24	0.53	0.48	0.78	1.51	1.30	1.63	2.88	2.38	2.81	4.53	3.56	4.40	6.45	4.72
9	0.15	0.42	0.39	0.54	1.05	0.90	1.20	2.05	1.66	2.14	3.43	2.68	3.27	5.09	3.91
10	0.16	0.46	0.43	0.57	1.22	1.07	1.26	2.29	1.91	2.21	3.66	2.92	3.43	5.31	4.06
11	0.14	0.22	0.17	0.41	0.58	0.41	0.85	1.22	0.88	1.43	2.10	1.54	2.26	3.27	2.36
12	0.12	0.16	0.10	0.26	0.34	0.21	0.41	0.55	0.36	0.59	0.82	0.57	0.84	1.20	0.85
13	0.09	0.13	0.09	0.20	0.25	0.16	0.31	0.41	0.26	0.44	0.58	0.38	0.65	0.86	0.56
14	0.11	0.14	0.09	0.18	0.23	0.14	0.27	0.36	0.23	0.37	0.49	0.32	0.55	0.74	0.49
CTRV	0.45	0.71	0.55	1.50	2.31	1.75	3.25	4.75	3.47	5.69	7.98	5.59	8.82	11.9	8.00

* CTRV : Constant Turn Rate Velocity Model

The prediction outcomes of LSTM model with the optimal input feature candidates with different history horizon and CTRV model are described as a histogram in Figure 6.24. The histograms from both model with history horizon of 1s and 2s show a similar shape, and the prediction errors from the LSTM model with history horizon of 1s are slightly more gathered around the zero compared to those from the model with history horizon of 2s. Histograms of two different models and CTRV model at each predicted time horizon are depicted in Figure 6.25. The LSTM (A) is with history horizon of 1s, and the LSTM (B) is with history horizon of 2s.
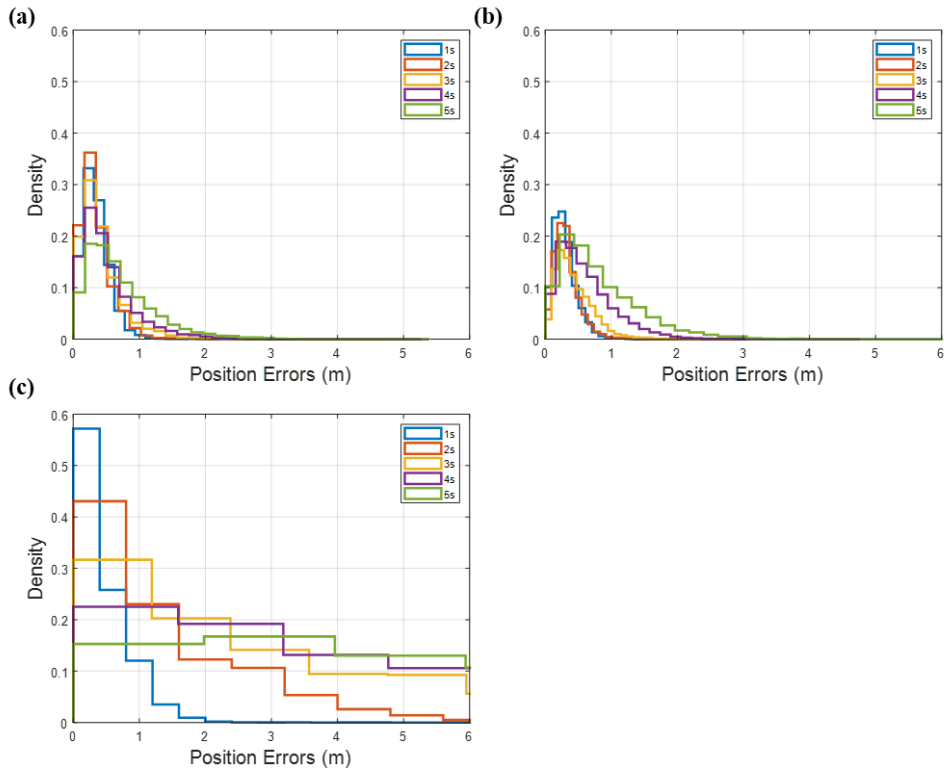
Figure 6.24. Histogram of prediction errors over prediction horizons of models trained in target centered coordinate (a) LSTM with history horizon 1s (b) LSTM with history horizon 2s (c) CTRV model.
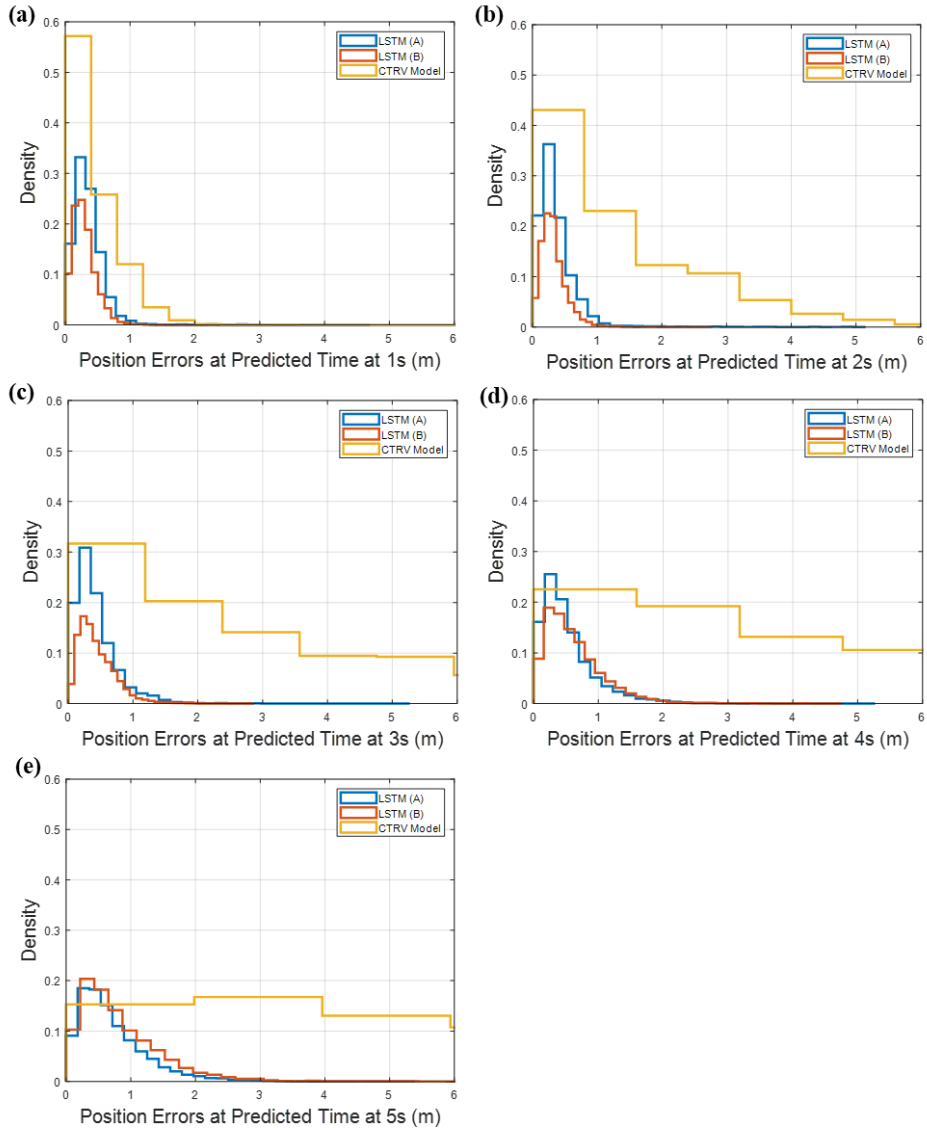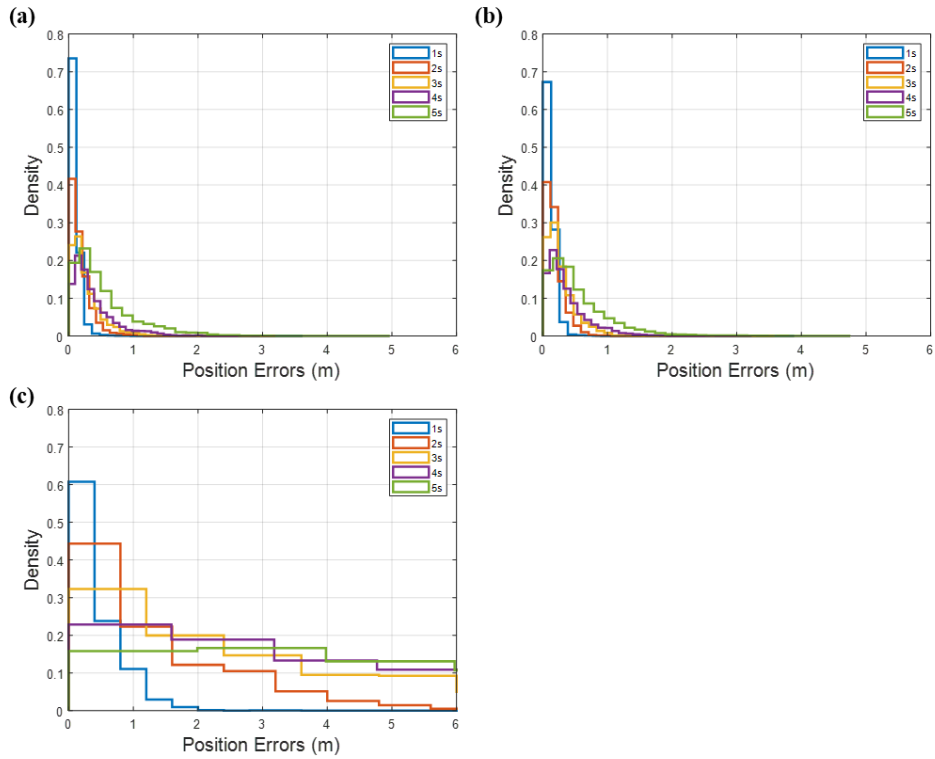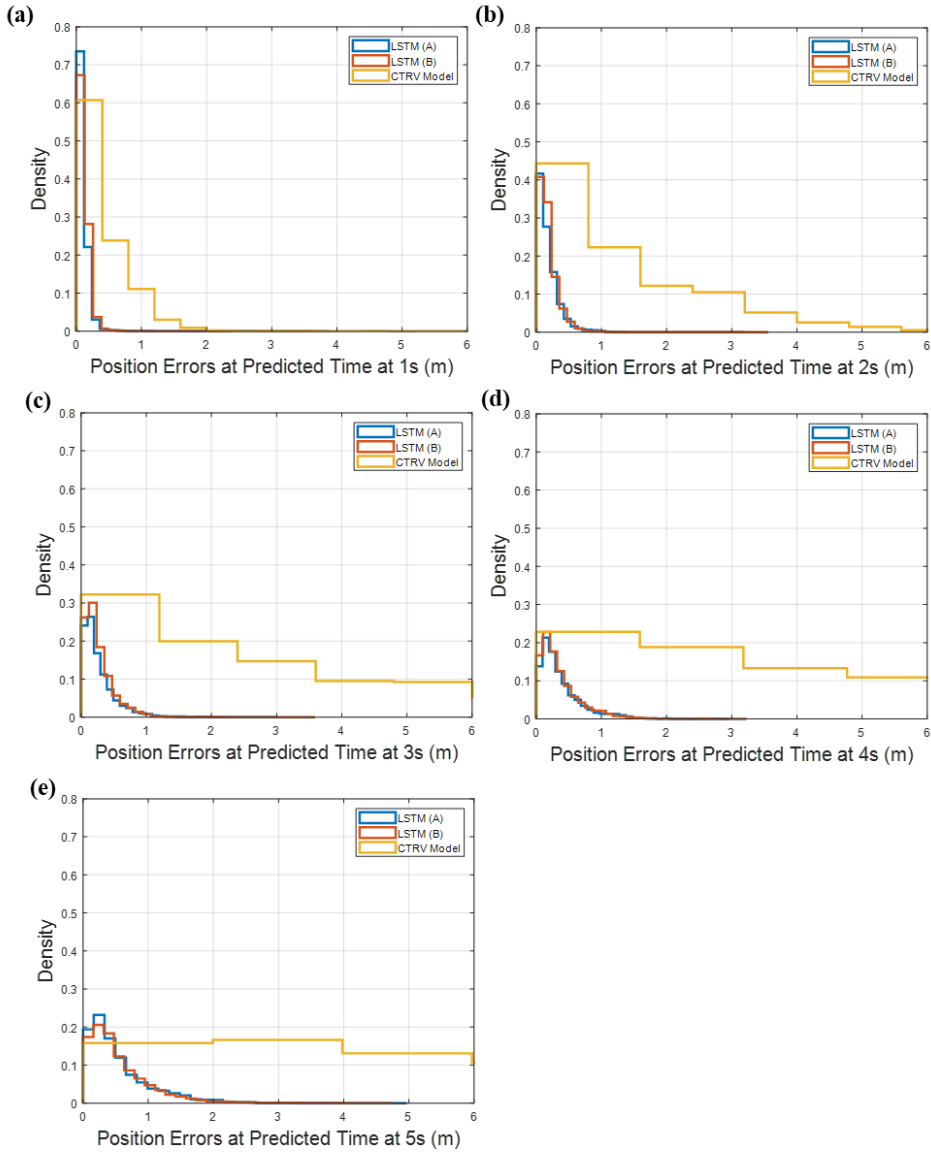
Figure 6.25. Histogram of prediction errors of models with different history horizons trained in target centered coordinate over different prediction horizon times (a)1s (b)2s (c)3s (d)4s (e)5s.

The prediction results of the model with the optimal input features in ego and target centered coordinates are compared and analyzed in Table 14. From the Table 14, it is observed that the model with input features of target centered coordinates is advantageous in shorter prediction horizon. The prediction errors of the model with target centered coordinates are less than half of those of the model with ego centered coordinates. Figure 6.26 describes the histogram of prediction errors from the LSTM models and the CTRV model. From Figure 6.26, it is clearly observed that histogram of the model with target vehicle coordinate is more centered at zero. Histograms of two different LSTM models and CTRV model at each predicted time horizon are depicted in Figure 6.27. The model with target vehicle centered coordinate performs better than the model with ego vehicle centered coordinate.

Table 14. Prediction Errors of the Trajectory Prediction from Models Trained in Different Coordinates.

Case	Position Error at Predicted Steps (m) : MAE | RMSE | STD														
	1s			2s			3s			4s			5s		
A	0.36	0.45	0.27	0.36	0.48	0.31	0.43	0.56	0.35	0.53	0.68	0.43	0.71	0.89	0.53
B	0.09	0.13	0.09	0.18	0.25	0.17	0.27	0.37	0.26	0.38	0.52	0.35	0.55	0.74	0.50
CTRV	0.45	0.71	0.55	1.50	2.31	1.75	3.25	4.75	3.47	5.69	7.98	5.59	8.82	11.9	8.00

* A: LSTM RNN with Ego Vehicle Coordinate
* B: LSTM RNN with Target Vehicle Coordinate
* CTRV: Constant Turn Rate Velocity Model
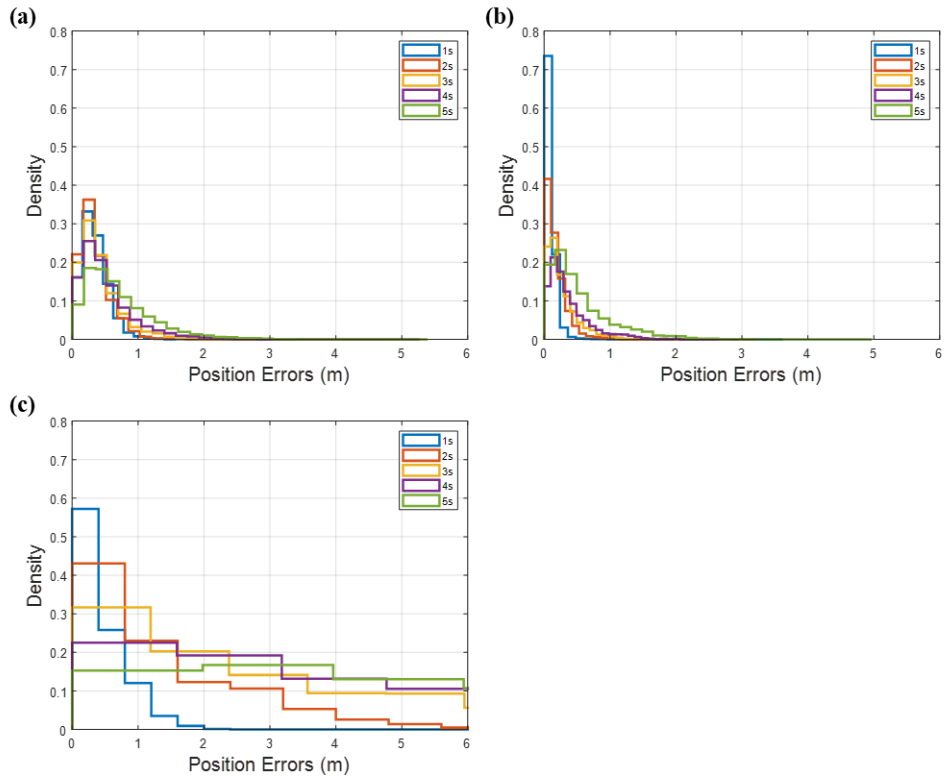
Figure 6.26. Histogram of prediction errors over prediction horizons (a) LSTM with ego centered coordinate (b) LSTM with target centered coordinate (c) CTRV model.
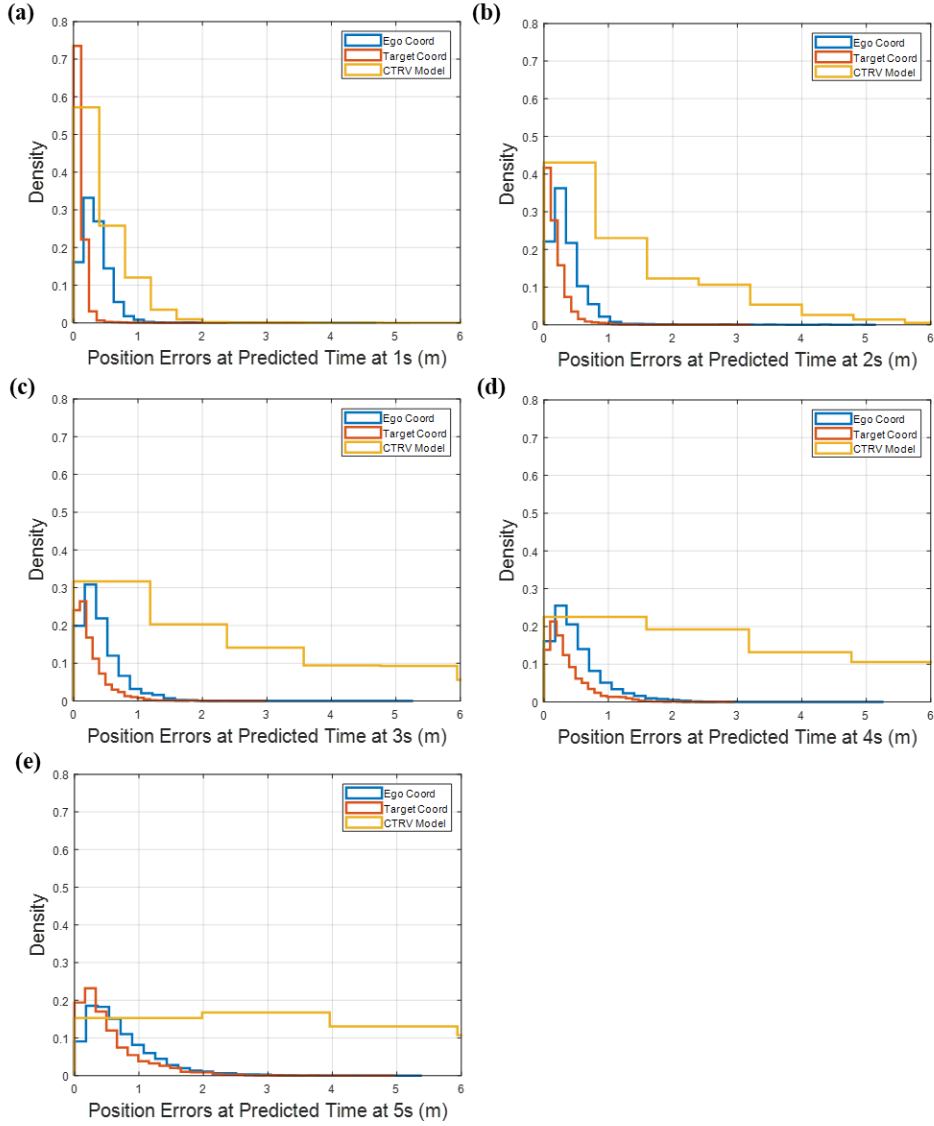
Figure 6.27. Histogram of prediction errors of LSTM and CTRV models over different prediction horizon times (a)1s (b)2s (c)3s (d)4s (e)5s.

6.2.2. Prediction Trajectory Accuracy Analysis

The target trajectory prediction results are analyzed in several driving scenarios such as the SAP with ego vehicle turning right and the RTAP with ego

vehicle crossing straight. The ground-truth trajectories, predicted trajectories of the proposed LSTM and the CTRV model, and history trajectories of target and ego vehicles are visualized to analyze the feasibility of the prediction model in the following figures. The black colored line is a ground truth of future trajectory of the target vehicle, and the red and blue lines are the predicted trajectories of target vehicle from the proposed LSTM model and CTRV model respectively. The green circles and magenta circles are history trajectories of target vehicle and ego vehicle. Figure 6.28 depicts the target vehicle crossing the intersection straight, interacting with the ego vehicle turning right. The predicted trajectory of the LSTM and the ground truth trajectory are heading straight, but the predicted trajectory from the CTRV model is turning left due to the error in the yaw rate. The predicted positions errors are analyzed for both models using MAE, RMSE, and STD, and shown in Table 15.

Figure 6.28. Trajectory prediction results of the RTAP with ego centered coordinate (lateral prediction error of the CTRV model)

Table 15. Prediction Errors of RTAP Scenario with Ego Centered Coordinate

	Position Errors (m)		
	MAE	RMSE	STD
LSTM	0.62	0.74	0.42
CTRV	1.77	2.57	1.90

Figure 6.29 also describes the target vehicle crossing the intersection straight, interacting with the ego vehicle turning right. Like the ground truth trajectory of the target vehicle, the predicted trajectory from the LSTM model becomes shortened over the prediction horizon. However, the future trajectory from the CTRV model describes the equidistant displacement over the prediction horizon. The CTRV model is unable to describe the slowing down motion of the vehicle over the prediction horizon. The predicted positions errors are analyzed in MAE, RMSE, and STD in Table 16, and from the Table 15 and 16, it can be inferred that the degree of error becomes greater when the vehicle changes the velocity over the future horizon.
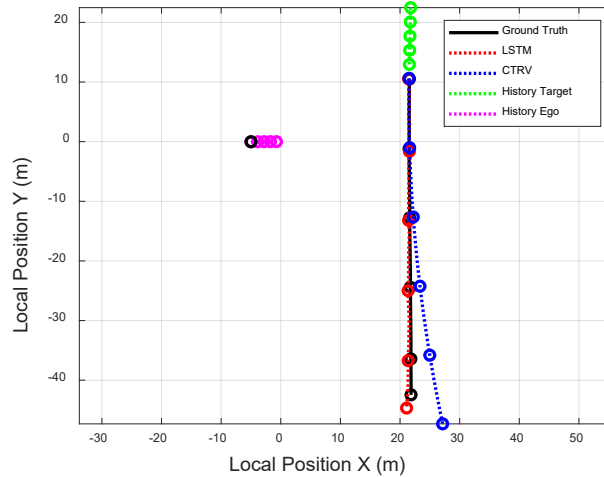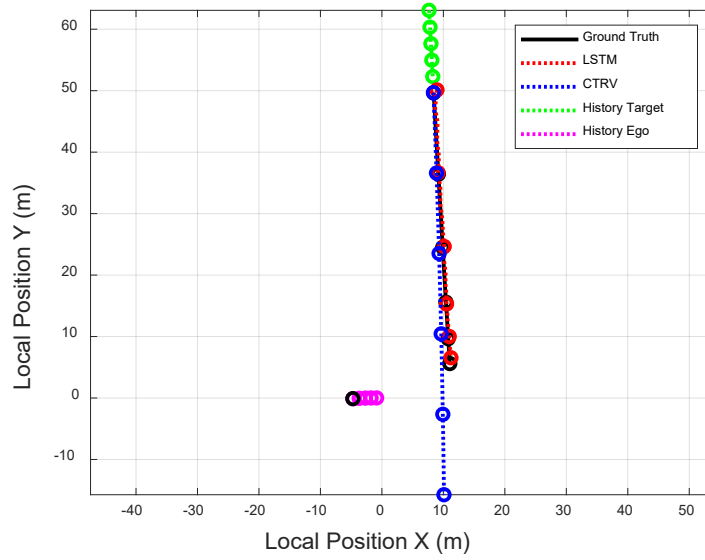
Figure 6.29. Trajectory prediction results of the RTAP with ego centered coordinate (longitudinal prediction error of the CTRV model)

Table 16. Prediction Errors of RTAP Scenario Ego Centered Coordinate

	Position Errors (m)		
	MAE	RMSE	STD
LSTM	0.36	0.42	0.22
CTRV	5.98	9.02	6.88

In Figure 6.30, the target vehicle turns right, interacting with the ego vehicle crossing straight at the intersection. The ego vehicle is driving from 50m away from the intersection at constant velocity. The proposed model appropriately predicts the target vehicle making a right turn. However, the CTRV model predicts the target vehicle driving straight through the intersection. Since the CTRV model

predicts the future trajectory of the target vehicle with the yaw and the yaw rate value at the initial guess point, the model fails to predict the turning driving behavior of the target vehicle as shown in Figure 6.30. The prediction errors of MAE, RMSE, and STD are described in the Table 17, and the degree of error is large due to a failure of prediction in future driving direction.

Figure 6.30. Trajectory prediction results of the SAP with ego centered coordinate (constant zero yaw rate in CTRV model)

Table 17. Prediction Errors of SAP Scenario with Ego Centered Coordinate

	Position Errors (m)		
	MAE	RMSE	STD
LSTM	0.56	0.63	0.29
CTRV	5.42	7.14	4.75

In Figure 6.31, the SAP scenario of the ego vehicle is described. The ego vehicle is driving 15m away from the intersection and slowly decreases the velocity.

The proposed model appropriately predicts the target vehicle making a right turn. The CTRV model correctly estimate the future driving direction, a right turn, of the target vehicle, yet fails to predict the degree of turning. The actual driving motion of the target vehicle in Figure 6.31 is that the target vehicle turns right and shows the straight driving motion at the end of the turning, and the LSTM model successfully generates such motion. It is observed that the LSTM model predicts the trajectory near perfect in longitudinal direction, but noticeable lateral errors exist. The CTRV model with its constant yaw rate predicts the target vehicle behaves like a circular motion. The predicted position errors are described as MAE, RMSE, and STD in Table 18.
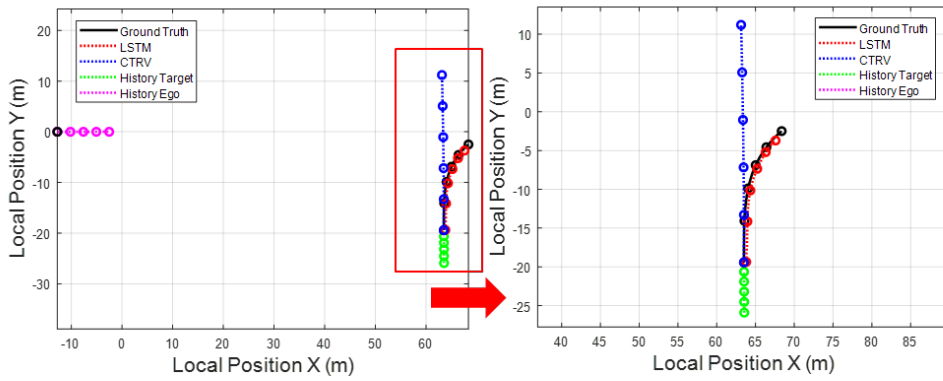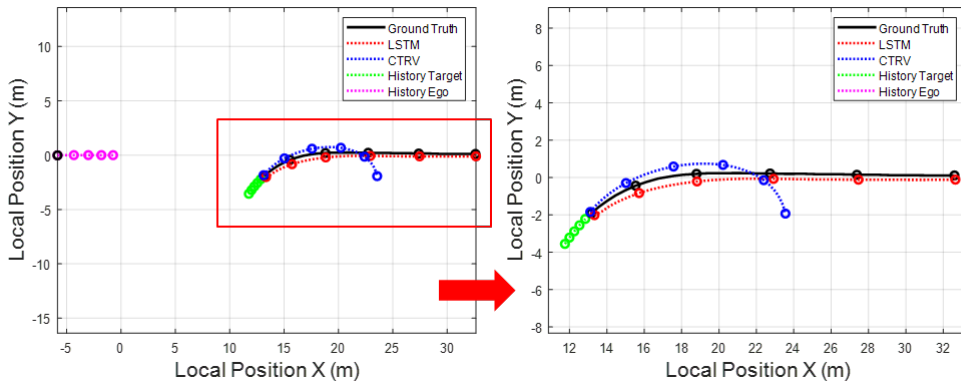

Figure 6.31. Trajectory prediction results of the SAP with ego centered coordinate (constant non-zero yaw rate in CTRV model)

Table 18. Prediction Errors of SAP Scenario with Ego Centered Coordinate

	Position Errors (m)		
	MAE	RMSE	STD
LSTM	0.34	0.35	0.08
CTRV	2.81	3.85	2.68

The LSTM model with the optimal input feature with 1s history horizon at the target vehicle centered coordinate is utilized to analyze the accuracy of the predicted trajectories. The target trajectory prediction results are analyzed in driving scenarios such as the SAP with ego vehicle turning right and the RTAP with ego vehicle crossing straight. The true trajectory, predicted trajectories of the proposed LSTM and the CTRV model, and history trajectories of target and ego vehicles are visualized to analyze the feasibility of the prediction model in the following figures. Each trajectory is colored differently to distinguish the prediction accuracy. In Figure 6.32, the target vehicle crosses the intersection straight, interacting with the ego vehicle approaching the intersection. Since the vehicles are oriented in the target vehicle coordinate, the current position of the target vehicle is located at the position (0,0). The predicted trajectory of the LSTM and the ground truth trajectory are heading straight, but the predicted trajectory from the CTRV model is predicted correctly in longitudinal direction but not in lateral direction due to the error in the yaw rate, slanted towards left. The predicted positions errors are analyzed for both models using MAE, RMSE, and STD, and shown in Table 19.

Figure 6.32. Trajectory prediction results of the RTAP with target centered coordinate (lateral prediction error of the CTRV Model)

Table 19. Prediction Errors of RTAP Scenario with Target Centered Coordinate

	Position Errors (m)		
	MAE	RMSE	STD
LSTM	0.73	1.00	0.70
CTRV	2.58	3.48	2.38

In Figure 6.33, the target vehicle crosses the intersection straight, interacting with the ego vehicle turning right, yet the targe vehicle in this scene gradually decreases the velocity. The LSTM model successfully predicts and describes the decreasing velocity driving behavior of the target. However, the CTRV model

predicts the target vehicle driving with the constant velocity, predicting the equidistant trajectory over the prediction horizon. The CTRV model is unable to describe the slowing down motion of the vehicle over the prediction horizon. The predicted positions errors are analyzed in MAE, RMSE, and STD in Table 20.
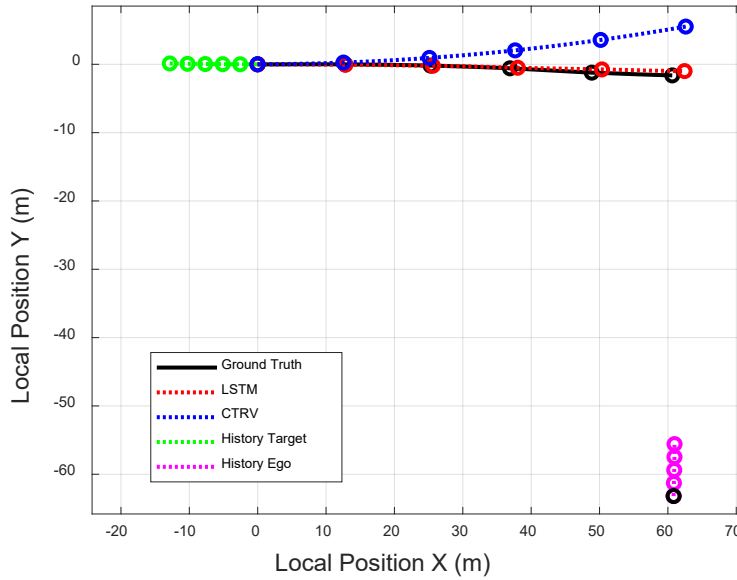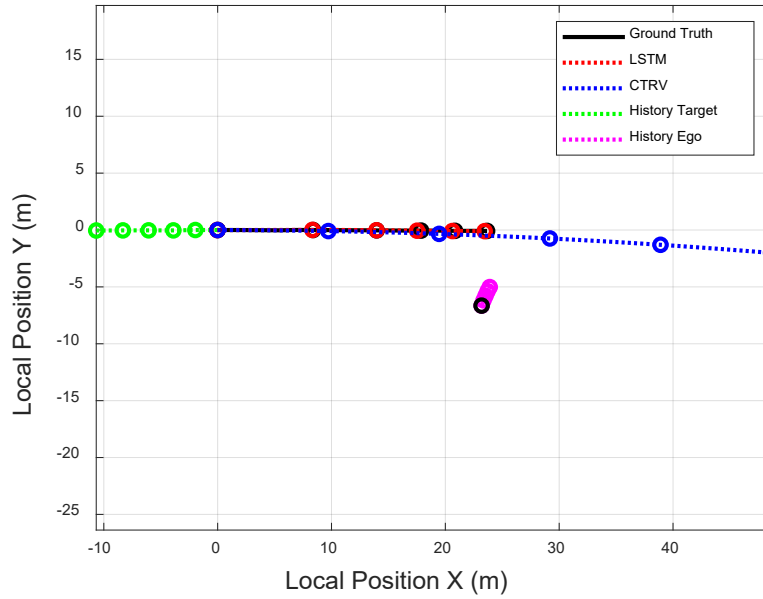

Figure 6.33. Trajectory prediction results of the RTAP with target centered coordinate (longitudinal prediction error of the CTRV model)

Table 20. Prediction Errors of the RTAP Scenario with Target Centered Coordinate

	Position Errors (m)		
	MAE	RMSE	STD
LSTM	0.16	0.20	0.12
CTRV	9.75	12.7	8.26

In Figure 6.34, the target vehicle turns right, interacting with the ego vehicle crossing straight at the intersection. The ego vehicle is driving from 15m away from the intersection. The proposed LSTM model predicts the proper target vehicle's right turn motion. The CTRV model correctly estimate the future driving direction, a right turn, of the target vehicle, yet the degree of yaw rate is not great enough to successfully predict the amount of turning motion. The radius of curvature of the predicted trajectory of the CTRV model is less than that of the predicted path of the LSTM model. The predicted position errors are described as MAE, RMSE, and STD in Table 21.
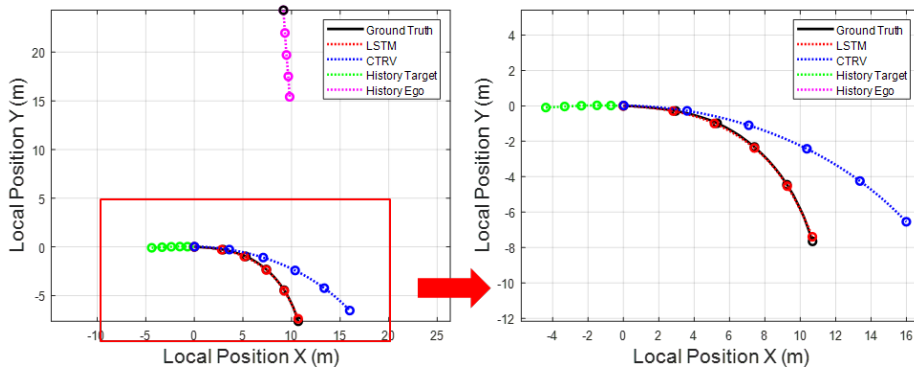
Figure 6.34. Trajectory prediction results of SAP with target centered coordinate (less constant yaw rate in CTRV model)

Table 21. Prediction Errors of SAP Scenario with Target Centered Coordinate

	Position Errors (m)		
	MAE	RMSE	STD
LSTM	0.10	0.12	0.06
CTRV	2.44	2.97	1.72

Another SAP scenario of intersection is described in Figure 6.35. The ego vehicle is driving approximately 10m away towards the intersection. The proposed model and the CTRV model appropriately predict the driving motion of the target vehicle, a right turn. Although the CTRV model correctly estimates the future driving direction, it overpredicts the degree of turning with greater trajectory curvature. In Figure 6.54, the target vehicle turns right in the beginning, and drives straight through the intersection at the end of the prediction horizon. The LSTM model successfully produces such future driving trajectories. The CTRV model with its constant yaw rate predicts the target vehicle behaves like a circular motion.

The predicted position errors are described as MAE, RMSE, and STD in Table 22.
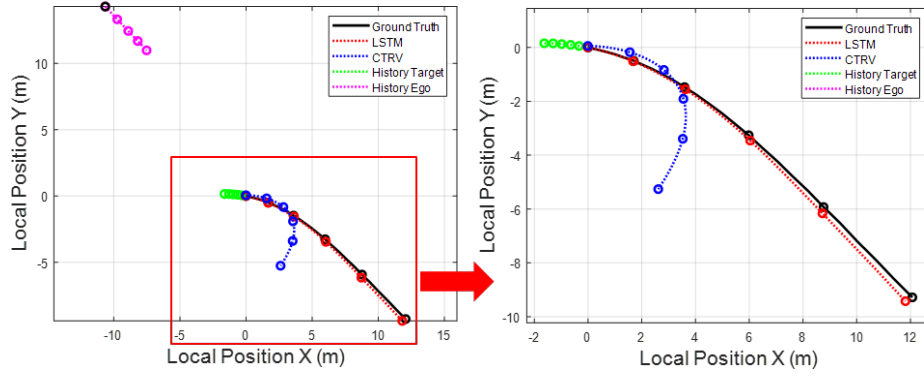

Figure 6.35. Trajectory prediction results of SAP with target centered coordinate (large constant yaw rate in CTRV model)

Table 22. Prediction Errors of SAP Scenario with Target Centered Coordinate

	Position Errors (m)		
	MAE	RMSE	STD
LSTM	0.14	0.17	0.10
CTRV	3.03	4.33	3.16

6.3. Vehicle Test for Intersection Autonomous Driving

6.3.1. Test Vehicle Configuration for Intersection Autonomous Driving

KIA Carnival vehicle is used for the test vehicle for the propose the intersection motion planning algorithm. Two 32 channel 3-D Velodyne Lidars are mounted front and rear roof of the vehicle to obtain the maximum visibility. The front vision camera is also mounted under a front glass for lane and traffic participants detection. In order to overcome weak GPS signal, the localization module implements Omni view around view camera for right and left lanes around the vehicle and estimates the odometry of the ego vehicle. Septentrio GPS is used to provide vehicle odometry. Two industrial PCs, Nuvo 8108GC, are built in to compute perception and motion planning respectively. The desired motion calculated from the industrial PC is provided to a lower-level controller in a micro-autobox to operate actuators of the autonomous vehicle. The detailed hardware configurations for autonomous vehicle system and perceptive sensors detection ranges are depicted in Figure 6.36.
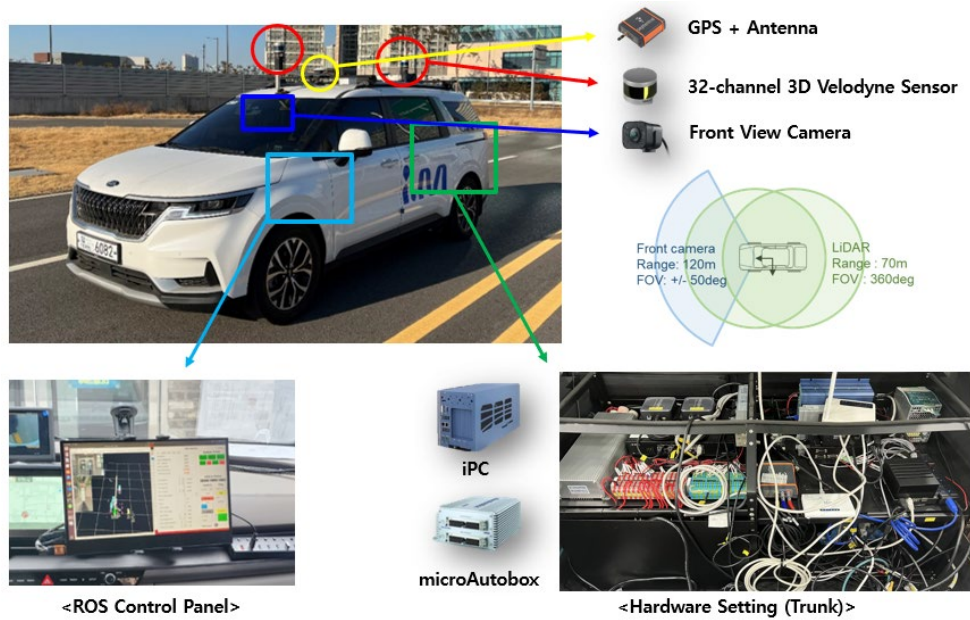
Figure 6.36. Test vehicle configuration for autonomous vehicle.

6.3.2. Software Configuration for Autonomous Vehicle Operation.

The Robot Operating System (ROS) is used as an environment for developing autonomous driving algorithm to operate the autonomous vehicle. In ROS system of autonomous vehicle, each automated driving algorithm modules are constructed as a node. Each node is communicating with different nodes through publishing and subscribing messages. The GPR based virtual target module, driving data-driven target prediction module, and stochastic MPC based longitudinal motion panning module represent different nodes. The control panel from the ROS allows to send the desired acceleration and steering angle to operate the autonomous vehicle. The control panel and example of ROS nodes for the autonomous driving
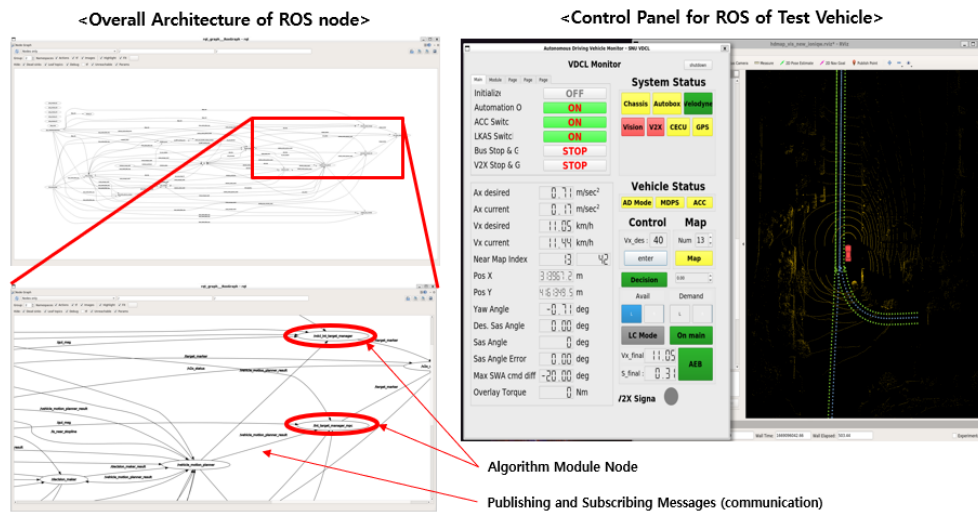
are described in Figure 6.37.

Figure. 6.37. Robot operating system for autonomous driving

6.3.3. Vehicle Test Environment for Intersection Autonomous Driving

The proposed algorithm is implemented in an autonomous vehicle and has been evaluated in urban intersection with real traffic participants. The vehicle tests have been conducted at the intersection in Mapo-gu, Sangam, a city nominated as an official autonomous driving testbed in Seoul, Korea. Many autonomous driving institutes and companies develop and test their logics and conduct the vehicle test in Sangam testbed. As previously mentioned, the proposed algorithm has been developed based on driving data obtained at Siheung, Gyeonggi-do, Korea. However, the motion planner has been tested in the different intersection in

Sangam if the algorithm can be applicable to various intersections other than intersections where data were obtained. The intersection in Sangam is a three-way signalized intersection. The vehicle test course and satellite view are described in Figure 6.38 and Figure 6.39.   The HD map of the vehicle test intersection is described in Figure 6.40. The road view image of the intersection is shown in Figure 6.41.

Figure 6.38. Vehicle test course in Sangam.

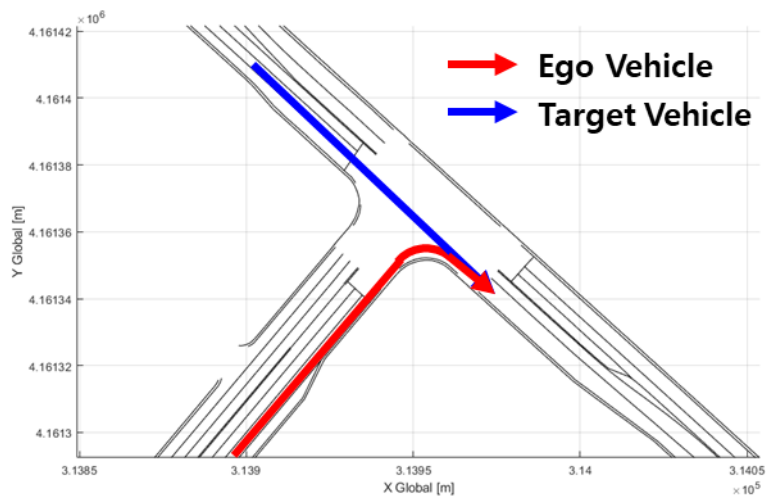Figure 6.39. Satellite view of intersection for the vehicle test in Sangam

Figure 6.40. HD map of Sangam intersection for the vehicle test

Figure 6.41. Road view of Sangam intersection for the vehicle test

6.3.4. Vehicle Test Result of Intersection Autonomous Driving

The proposed algorithm is evaluated in the RTAP scenario at the intersection. The two different driving conditions are tested at the intersection. For the first case, the autonomous vehicle turns right at the intersection with no traffic participants approaching from the opposite direction. For the second case, the autonomous vehicle turns right and interacts with oncoming vehicles from the right. The vehicle test result of no traffic participants scenario is described in Figure 6.42. The ego vehicle enters, turns right, and pass through the intersection. The time history of trajectory, acceleration, velocity, virtual target mode activation, clearance, and position upper bound for the MPC is depicted in Figure 6.42. In Figure 6.42 (f), the clearance between the virtual target and ego vehicle over the prediction time is described. It is observed that the ego vehicle interacts with the virtual target during time t =13s to 15s. As shown in Figure 6.42 (e), during time t=13s to 15s, at prediction time of 4s to 5s, the motion planner expects the predicted trajectory of

the ego vehicle and the future trajectory of the virtual target collides, and the negative acceleration command is ordered for deceleration. From Figure 6.43, the snapshots of vehicle travelling through the intersection is described, the predicted trajectory of the virtual target vehicle is depicted as green circles, and the FOV of the autonomous vehicle is drawn with cyan colored circles. The virtual target allows the ego vehicle to decelerate early to prevent potential collision with the traffic participants appearing from the occluded region.

(c)

(d)

(e)
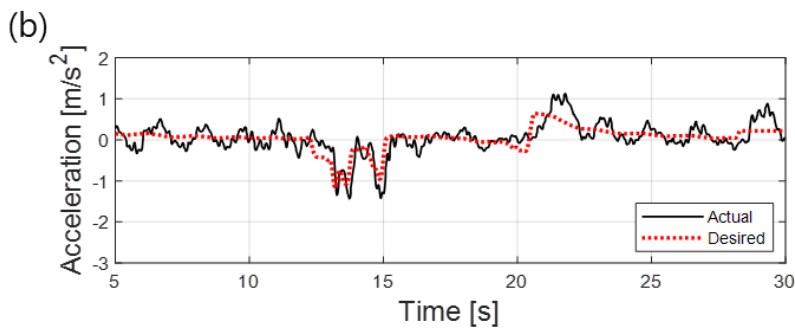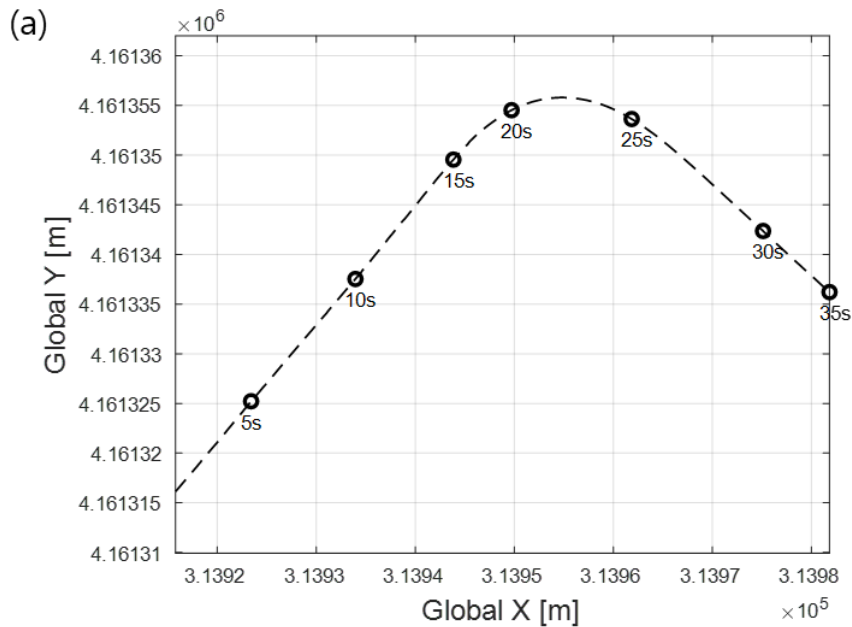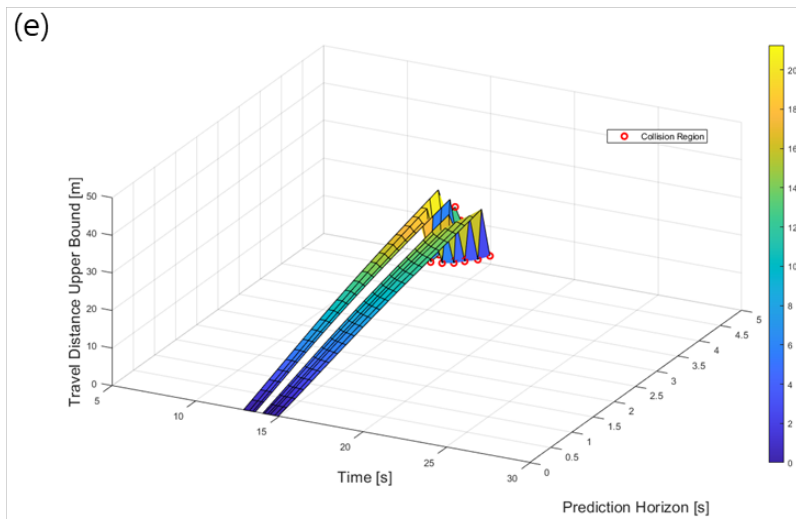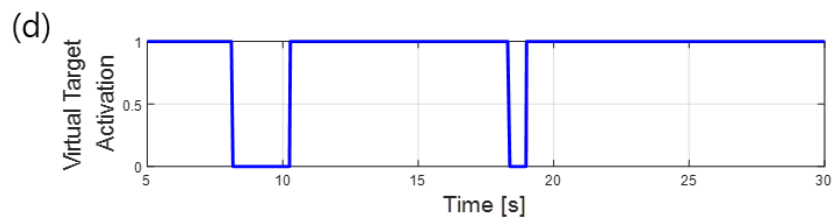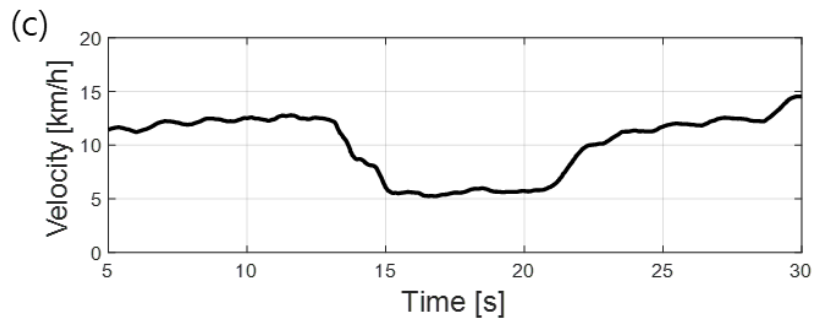
Figure 6.42. Time history of the vehicle test results of no vehicle approach scenario (a) trajectory history (b) acceleration (c) velocity (d) virtual target mode (e) clearance (f) travel distance upper bound.

(a)
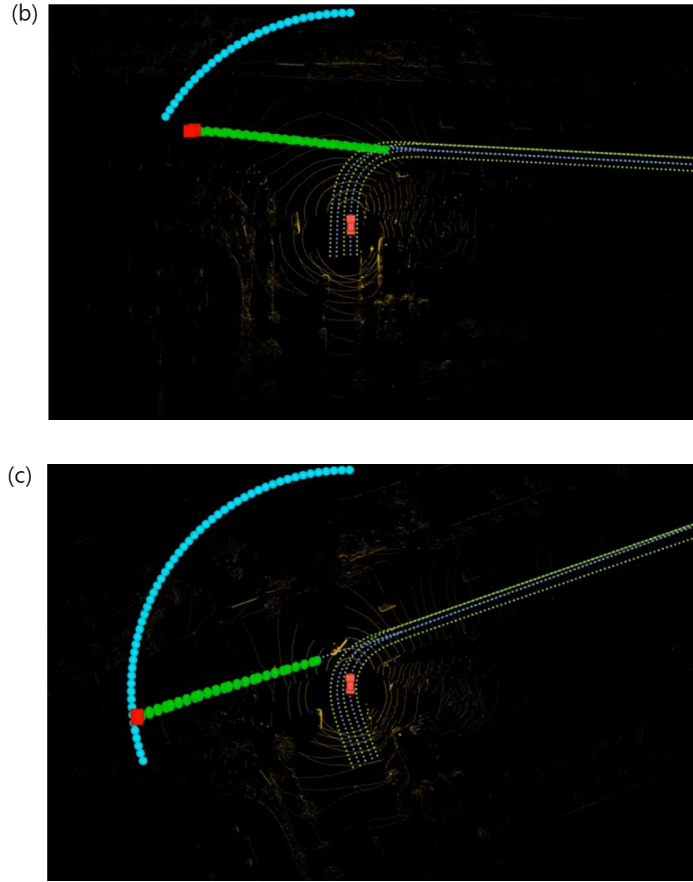
Figure 6.43. Snapshots of the vehicle test results of no vehicle approach scenario (a)t=10s (b)t=14s (c)t=20s.

The proposed algorithm is evaluated under the scene where the autonomous vehicle interacts with multiple traffic participants crossing the intersection. The vehicle result is summarized in Figure 6.44. The time history of trajectory, acceleration, velocity, virtual target mode activation, clearance, and position upper bound for the MPC is depicted in Figure 6.44. In Figure 6.44 (b), it is observed at the ego vehicle interacts with oncoming vehicles at time t=9s, 13s, 15s, and 18s

and decelerates. In Figure 6.44 (f), the clearance between the virtual target and ego vehicle over the prediction time during the RTAP scenario is described. As shown in Figure 6.44 (e), during time t=8s, the motion planner expects the predicted trajectory of the ego vehicle and the future trajectory of the target vehicles at prediction time of 4s to 5s. During time t= 13s, 15s, and 18, the ego vehicle enters the intersection and becomes close to the oncoming traffic participants. In Figure 6.44 (e), during those time periods, the proposed longitudinal algorithm predicts the potential collisions at prediction horizon of t=1s and 2s. Unlike the desired acceleration from the potential collision at time t=9s, since the potential collisions at time t=13s,15s, and 18s are expected to occur within prediction time of 2s, the degree of the desired control input is greater. With the proper desired acceleration command from the algorithm, the velocity profile in Figure 6.44 shows the autonomous vehicle successfully manages to stop and prevents the possible collision with oncoming vehicles. In Figure 6.45, the snapshots of the ego vehicle interacting oncoming vehicles at the intersection are described. The cyan circles describe the FOV of the autonomous vehicle. The green and white circles represent the predicted trajectories of the virtual target from GPR model and detected vehicles from the proposed LSTM model, respectively. The post encroachment time (PET) is also calculated for this multi vehicle approaching scenario. The time crossing conflict point of target vehicles are 4.19s, 11.70s, 14.02s, 16.79s, and 20.93s. The ego vehicle enters the conflict point at time of 26.78s. The PET is 5.85s for this vehicle test.
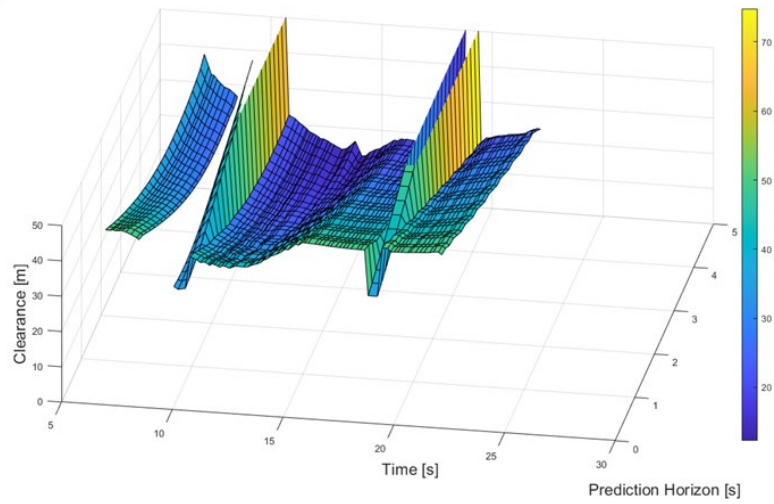
(a)

(b)

(c)

Figure 6.44. Time history of the vehicle test results of multiple vehicles approach scenario (a) trajectory history (b) acceleration (c) velocity (d) virtual target mode (e) clearance (f) travel distance upper bound.
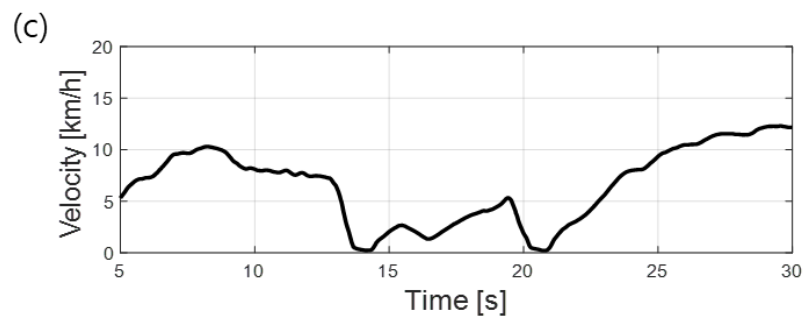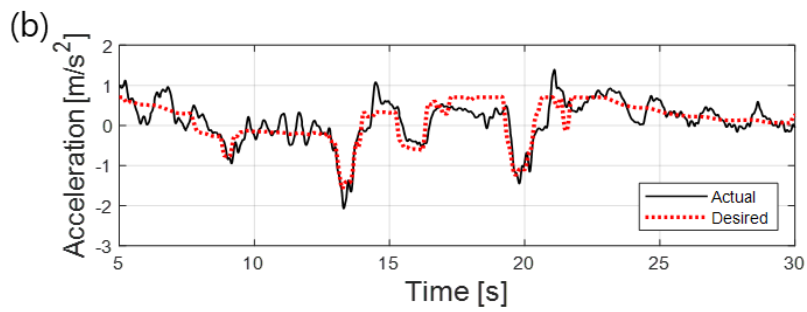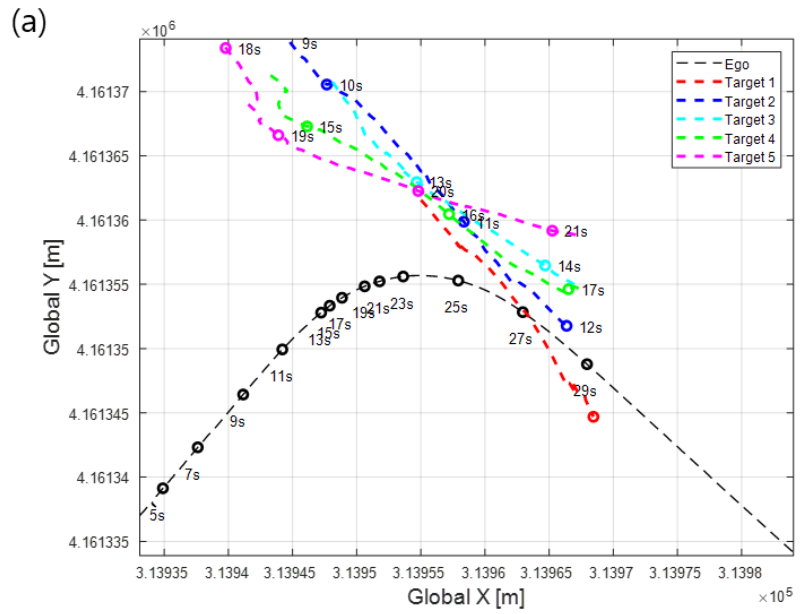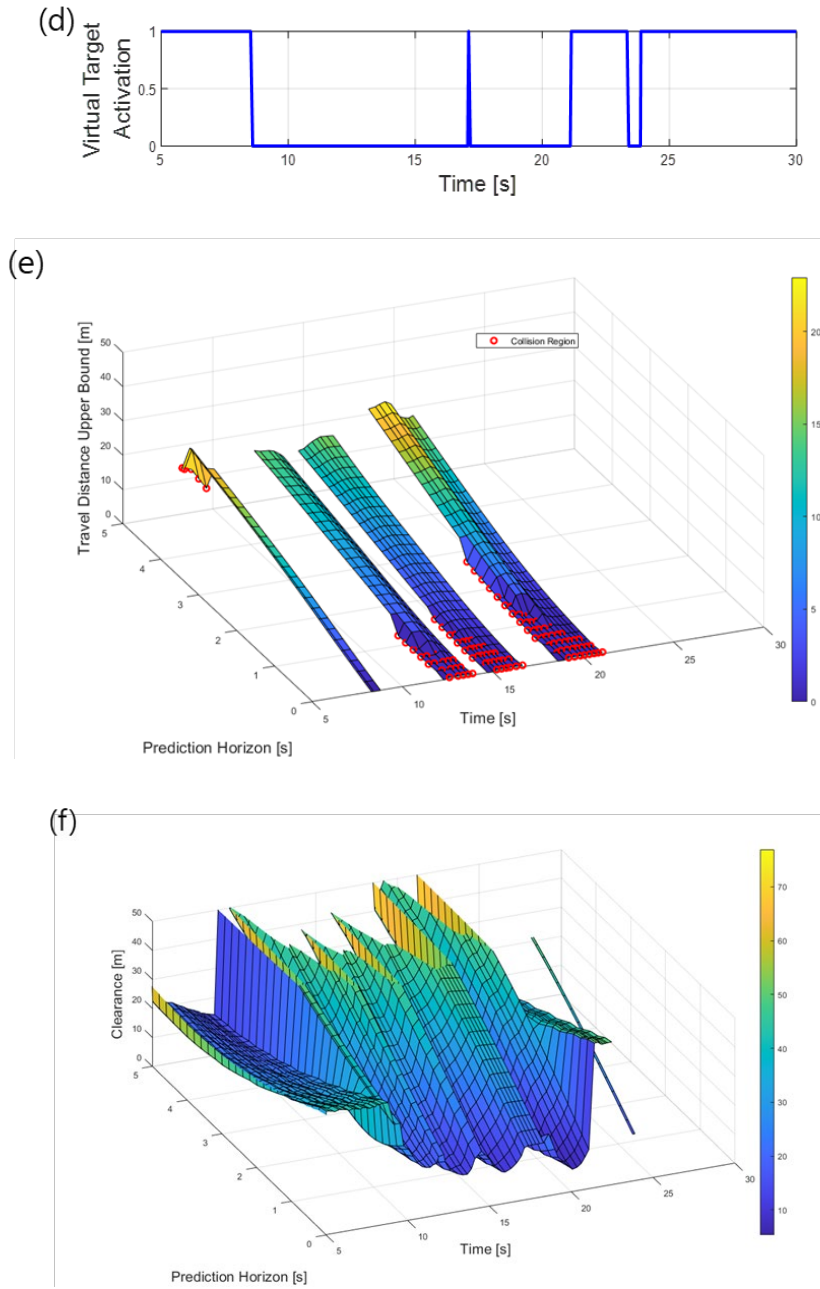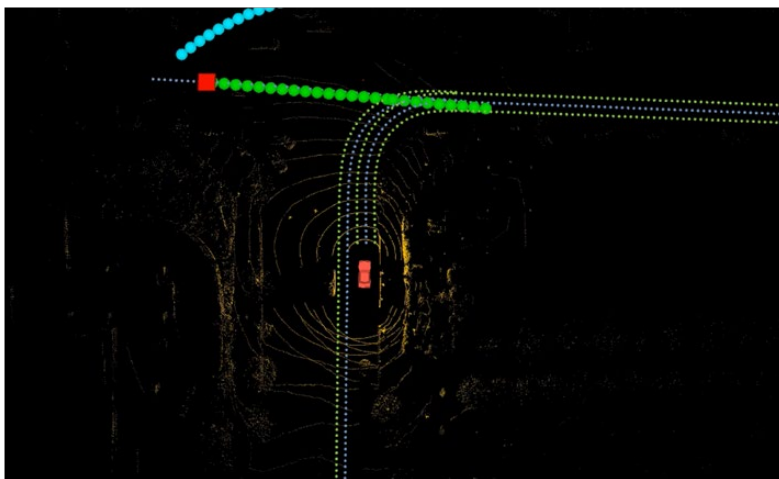
(a)

(b)

(c)

(d)
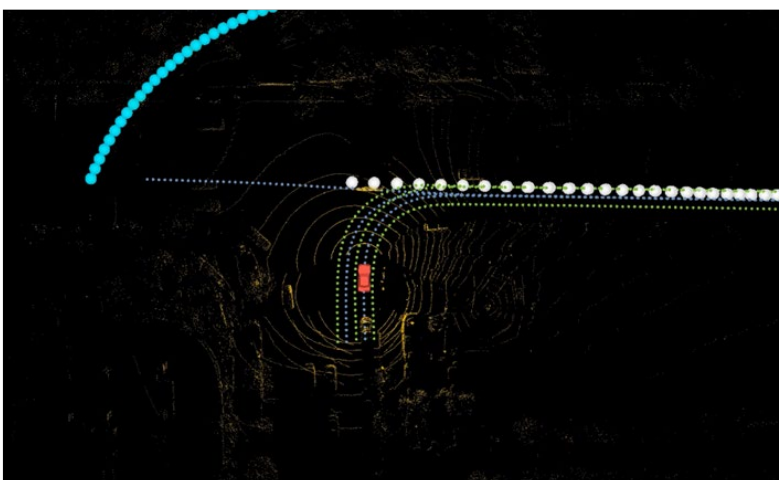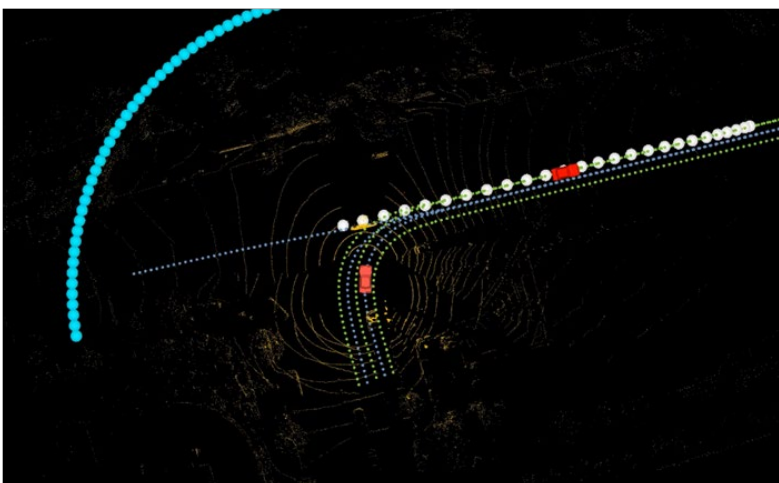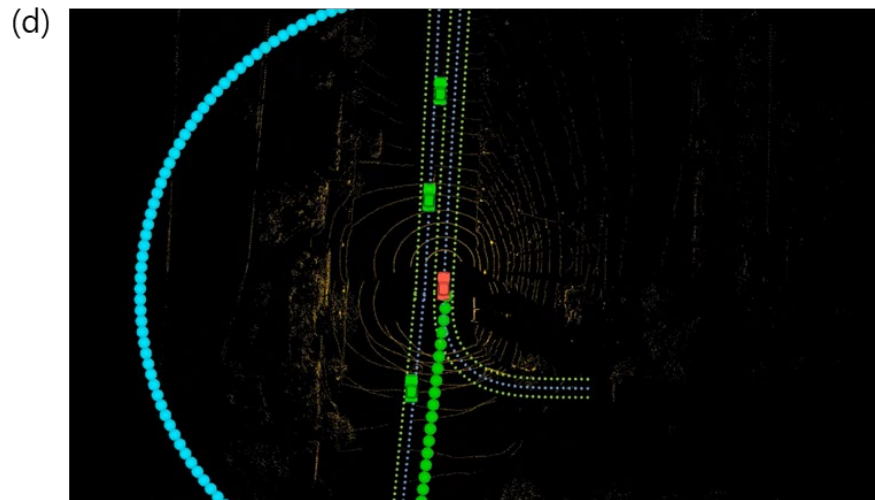
Figure 6.45. Snapshots of the vehicle test results of multiple vehicles approach scenario (a)t=9s (b)t=19s (c)t=22s (d)t=28s.

Chapter 7. Conclusion and Future Work

7.1. Conclusion

This thesis proposes a longitudinal motion planning algorithm in urban intersection with the limited visibility for fully automated driving system with open-market perceptive sensors. Three challenges in autonomous driving in complex urban intersection are identified: virtual target modelling, surround vehicle prediction, and longitudinal motion planning.

The virtual target method is utilized to ensure the safety and prevent the potential collision with targets from the blind spot. Human driving data are collected to model the virtual target and mimic driving patterns. The virtual target generated based on the field of view of the autonomous vehicle may cause the "dead-lock" situation that the ego vehicle remains stopped at the intersection indefinitely. Implementing the human driven data, the virtual target model interacting with the ego vehicle motion can generate the yield driving behavior that allows the ego vehicle to pass through the intersection. The Gaussian Process Regression is implemented to design the virtual target model with human driving data. The trained GPR model for the virtual target provides the predicted travel distance and its uncertainty, which can be utilized in the longitudinal motion planning of the autonomous vehicle as a position (travel distance) constraint and a chance constraint. The virtual target module is evaluated with the offline simulation studies in two different driving scenarios: the SAP and the RTAP in the intersection.

With the data-driven virtual target model, the autonomous vehicle decelerates with less control input at earlier time in the intersection.

The deep learning method, a LSTM-RNN, is implemented to predict the surround vehicle motion. The train and test data are collected at the actual urban intersection with two different autonomous vehicles. Two different autonomous vehicles can provide their odometry information regardless to the limitation in field of view. Popular autonomous driving datasets from Waymo open datasets, nuScenes, and Kitti obtain the target odometry information from the perceptive sensors mounted on the autonomous vehicle, and such vehicular configuration is limited in providing wide range of target odometry information due to limited field of view. The obtained training datasets and their parameters are reorganized in the detected target centered coordinates for better input-domain adaptation and generalization. The mean squared error and negative log likelihood loss functions are adapted to train and provide the uncertainty information of the target vehicle for the motion planning of the autonomous vehicle. The prediction accuracy with different input features is analyzed to determine the optimal input features.

The model predictive control with a chance constraint is used to plan the longitudinal motion at the urban intersection. The dynamic constraints and actuator constraints are designed to provide ride comfort and safety to drivers and passengers. The position constraint with the chance constraint guarantees the safety and prevent the potential collision with target vehicles. The position constraint for the travel distance over the prediction horizon time is determined based on the clearance between the predicted trajectories of the target and ego vehicle at every

prediction sample time.

The feasibility and effectiveness of the proposed automated driving motion planning algorithm has been evaluated through offline simulation, test-data based simulation, and vehicle test in a real road of an official autonomous driving test bed. The proposed motion planner with predictor has been tested with over 35,000 test data set. The offline simulation verifies the safety of the algorithm from the potential collision of a sudden appearance of traffic participants from the occlusion. Through the vehicle test in the urban test road, the autonomous vehicle successfully passes through the intersection by maintaining safety clearance with other oncoming traffic participants under no traffic regulation other than traffic signals. Through the implementation of the algorithm to actual vehicle, it is confirmed that the algorithm is not just validated in simulation studies, but also can be applicable in real life right away.

The main contributions of this thesis are as follows. First, the virtual target model is implemented to overcome the limited FOV of autonomous sensors by local cognitive sensors and blind spots. Instead of simple assumption of virtual target driving behavior, the virtual target motion is generated with a data driven method. The safety boundary of the predicted trajectory of virtual target is implemented to compensate the prediction uncertainty. Second, the safe trajectory prediction of detected target vehicles is determined for driver acceptance using rule-based and learning-based methods. Data collection from two autonomous vehicles for training a model allows to consider the wide range of interaction between human driving vehicles and autonomous vehicle. Third, the efficient and

effective decision-making and motion control are achieved using driving mode and optimal constraints based on learning based prediction and stochastic MPC. Finally, the efficacy and feasibility of the proposed algorithm are evaluated through both simulation and actual vehicle tests in real road.

7.2. Future Work

The proposed longitudinal motion planning at the urban intersection enables the autonomous vehicle to safely cross the intersection from oncoming target vehicles from the occluded region due to limited field of view and sensor configuration. The future works in the intersection motion planning can be categorized in three aspects. The first is comprehension of different types of intersection and environment. Although the motion planning algorithm can be applicable to the general shape of the intersection, the more complicated intersection such as the y-junction, staggered intersection, and roundabouts that drivers often encounter should be considered. The second is the prediction of the dynamic motion of target vehicles. The proposed motion planner can react to the target vehicle in the pre-defined ROI, and the algorithm may be vulnerable to target vehicle suddenly conducting lane changes in the middle of intersection and entering the ROI. The third is the human driven data-based learning model predictive control with optimal path planning. As the autonomous driving has been actively studied and implemented in a vehicle for real-life use, many autonomous driving related companies provide automated shuttle services to people. The scope of an autonomous vehicle is being extended to commercial vehicles such as a full-

sized bus and yard trucks. The exploration of these research scopes can substantially develop urban autonomous driving in complex road environment with safety and ride comfort.

Bibliography

PREVOST, C. G., DESBIENS, A. & GAGNON, E. Extended Kalman filter for state estimation and trajectory prediction of a moving object detected by an unmanned aerial vehicle, 2007. Proceeding to American Control Conference. 1805-1810.

JEONG, Y. & YI, K. 2019. Target vehicle motion prediction-based motion planning framework for autonomous driving in uncontrolled intersections. *IEEE Transactions on Intelligent Transportation Systems*, 22 (1), 168-177.

KOSCHI, M. & ALTHOFF, M. 2020. Set-based prediction of traffic participants considering occlusions and traffic rules. *IEEE Transactions on Intelligent vehicles*, 6(2), 249-265.

JEONG, Y., KIM, S. & YI, K. 2020a. Surround vehicle motion prediction using LSTM-RNN for motion planning of autonomous vehicles at multi-lane turn intersections. *IEEE Open Journal of Intelligent Transportation Systems*, 1, 2-14.

DEO, N. & TRIVEDI, M. M. Convolutional social pooling for vehicle trajectory prediction. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018. 1468-1476

ROY, D., ISHIZAKA, T., MOHAN, C. K. & FUKUDA, A. Vehicle trajectory prediction at intersections using interaction based generative adversarial networks. 2019 IEEE Intelligent Transportation Systems Conference, 2019. IEEE, 2318-2323.

SHENG, Z., XU, Y. & XUE, S. Graph-based spatial-temporal convolutional

network for vehicle trajectory prediction in autonomous driving. 2022. *IEEE Transactions on Intelligent Transportation Systems*, 23 (10), 17654-17665.

YOON, Y., KIM, C., LEE, J. & YI, K. 2021. Interaction-aware probabilistic trajectory prediction of cut-in vehicles using gaussian process for proactive control of autonomous vehicles. *IEEE Access*, 9, 63440-63455.

LI, X., YING, X. & CHUAH, M. C. GRIP: Graph-based interaction aware trajectory prediction. 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019. IEEE, 3960-3966.

ELLEUCH, I., MAKNI, A. & BOUAZIZ, R. Cooperative intersection collision avoidance persistent system based on V2V communication and real-time databases. Proceeding 14th IEEE/ACS International Conference on Computer System and Applications (AICCSA), 2017. 1082-1089.

SUN, G., ZHANG, Y., YU, H. DU, X. & GUIZANI, M. 2020. Intersection fog-based distributed routing for V2V communication in urban vehicular ad hoc networks. *IEEE Transactions on Intelligent Transport Systems*, 21(6), 2409-2426.

CI, Y., WU, L., ZHAO, J., SUN, Y. & ZHANG, G. 2019. V2I-based car-following modeling and simulation of signalized intersection. *Physica A: Statistical Mechanics and Its Applications*, 525(1), 672-679.

DUAN, X., JIANG, H., TIAN, D., ZOU, T., ZHOU, J. & CAO, Y. 2021. V2I based environment perception for autonomous vehicles at intersectiosn. *China Communications*, 18(7), 1-12.

LIU, W., MURAMATSU, S. & OKUBO, Y. Cooperation of V2I/P2I

communication and roadside radar perception for the safety of vulnerable road users. 2018 16th International Conference on Intelligent Transportation Systems Telecommunications (ITST), 2018.

JUNG, C., LEE, D., LEE, S. & SHIM, D. H. 2020. V2X-communication-aided autonomous driving: system design and experimental validation. *Sensors*, 20(10), 1-21.

MULLER, J., STROHBECK, J., HERRMANN, M. & BUCHHOLZ, M. 2022. Motion planning for connected automated vehicles at occluded intersections with infrastructure sensors. *IEEE Transactions on Intelligent Transportation Systems*, 23(10), 17479-17490.

NARKSRI, P., TAKEUCHI, E., NINOMIYA, Y. & TAKEDA, K. 2021. Deadlock-free planner for occluded intersections using estimated visibility of hidden vehicles. *Electronics*, 10(4), 1-28.

HUBMANN, C., QUETSCHLICH, N., SCHULZ, J., BERNHARD, J., ALTHOFF, D. & STILLER, C. A POMDP maneuver planner for occlusions in urban scenarios. 2019 IEEE Intelligent Vehicles Symposium (IV), 2019. 2172-2179.

SCHORNER, P., TOTTEL, L., DOLL, J. & ZOLLNER, J. Predictive trajectory planning in situations with hidden road users using partially observable Markov decision processes. 2019 IEEE Intelligent Vehicles Symposium (IV), 2019. 2299-2306.

BRECHTEL, S., GINDELE, T. & DILLMANN, R. Probabilistic decision-making under uncertainty for autonomous driving using continuous POMDPs. 2014 IEEE 17th International Conference on Intelligent Transportation Systems

(ITSC), 2014. 392-399.

ZHANG, C., STEINHAUSER, F., HINZ, G. & KNOLL, A. Improved occlusion scenario coverage with a POMDP-based behavior planner for autonomous urban driving. 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), 2021. 593-600.

LIN, X., ZHANG, J., SHANG, J., WANG, Y., YU, H. & ZHANG, X. Decision making through occluded intersections for autonomous driving. 2019 IEEE Intelligent Transportation System Conference (ITSC), 2019. 2449-2455.

THORNTON, S. M. 2018. Autonomous vehicle speed control for safe navigation of occluded pedestrian crosswalk. *arXiv:1802.06314*.

BOUTON, M., NAKHAEI, A., FUJIMURA, K. & KOCHENDERFER, M. Scalable decision making with sensor occlusions for autonomous driving. 2018 IEEE International Conference on Robotics and Automation (ICRA), 2018. 2076-2081.

MORALES, L. Y., NAOKI, A., YOSHIHARA, Y. & MURASE, H. Towards predictive driving through blind intersections. 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2018. 716-722.

ISELE, D., RAHIMI, R., COSGUN, A., SUBRAMANIAN, K. & FUJIMURA, K. Navigating occluded intersections with autonomous vehicles using deep reinforcement learning. 2018 IEEE International Conference on Robotics and Automation (ICRA), 2018.

KAMRAN, D., LOPEZ, C. F., LAUER, M. & STILLER, C. Risk-aware high-level decisions for automated driving at occluded intersections with reinforcement

learning. 2020 IEEE Intelligent Vehicles Symposium (IV), 2020.

KAMRAN, D., ENGELGEH, T., BUSCH, M., FISCHER, J. & STILLER, C. 2021. Minimizing safety interference for safe and comfortable automated driving with distributional reinforcement learning. *arXiv:2107.07316.*

REN, X., YANG, T., LI, L. E., ALAHI, A. & CHEN, Q. 2021. Safety-aware motion prediction with unseen vehicles for autonomous driving. *arXiv:2109.01510.*

TAS, O. S. & STILLER, C. Limited visibility and uncertainty aware motion planning for automated driving. 2018 IEEE Intelligent Vehicles Symposium (IV), 2018. 1171-1178.

MORALES, Y., YOSHIHARA, Y., AKAI, N., TAKEUCHI, E. & NINOMIYA, Y. Proactive driving modeling in blind intersections based on expert driver data. 2017 IEEE Intelligent Vehicles Symposium (IV), 2017. 901-907.

YOSHIHARA, Y., MORALES, Y., AKAI, N., TAKEUCHI, E. & NINOMIYA, Y. Autonomous predictive driving for blind intersections. 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017. 3452-3459.

HOERMANN, S., KUNZ, F., NUSS, D., RENTER, S. & DIETMAYER, K. Entering crossroads with blind corners. A safe strategy for autonomous vehicles. 2017 IEEE Intelligent Vehicles Symposium (IV), 2017. 727-732.

NAUMANN, M. KONIGSHOF, H., LAUER, M. & STILLER, C. Safe but not overcautious motion planning under occlusions and limited sensor range. 2019 IEEE Intelligent Vehicles Symposium (IV), 2019. 140-145

ORZECHOWSKI, P. F., MEYER, A. & LAUER, M. Tackling occlusions & limited

sensor range with set-based safety verification. 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2018. 1729-1736.

YU, M. Y., VASUDEVAN, R. & JOHNSON-ROBERSON, M. 2019. Occlusion-aware risk assessment for autonomous driving in urban intersection. *IEEE Robotics and Automation Letters*, 4(2),2235-2241.

ZHANG, Z. & FISAC, J. C. 2021. Safe occlusion-aware autonomous driving via game-theoretic active perception. *arXiv:2105.08169*.

WANG, D., FU, W., SONG, Q. & ZHOU, J. 2022. Potential risk assessment for safe driving of autonomous vehicles under occluded vision. *Scientific Report*, 12, 4981.

Williams, C. K. & Rasmussen, C. E. 2006. Gaussian processes for machine learning, Cambridge, MA: MIT press

Abstract in Korean

본 논문은 복잡한 도로 구조와 센서 사양으로 인한 시야 제한을 극복하며 사각지대에서 등장하는 차량과의 잠재적인 충돌로부터 안전을 보장하기 위한 도심 교차로에서의 자율주행차의 새로운 종방향 거동 계획을 제시한다.

도심 자율주행은 교통체증과 환경의 복잡성으로 인해 높은 수준의 안전성이 요구됩니다. 복잡한 도로 구조와 인지 센서의 인지 범위로 인해 도심 자율주행에서는 사각지대가 발생한다. 가상 타겟은 사각지대에서 차량의 갑작스러운 출현에 대응하기 위한 거동 계획 방법 중 하나입니다. 자차량의 거동과 상호작용하는 다양한 미래 주행 궤적을 생성하는 가상 타겟 모델을 구현하기 위하여 Gaussian Process Regression (GPR) 방법을 사용합니다. GPR 모델은 가상 표적의 예측된 궤적뿐만 아니라 미래 궤적에 대한 불확실성도 제공합니다. 따라서 GPR의 예측 결과는 Model Predictive Control (MPC)에 대한 위치 제약 조건으로 활용될 수 있으며 불확실성은 MPC에서 기회 제약 조건으로 고려됩니다.

동적 객체를 포함한 주변 환경을 파악하기 위해 관심영역을 정의하여 목표 대상을 결정합니다. 미리 결정된 자차량의 주행경로와 교차로의 경로정보를 통하여 자차량의 주행차로와 교차하는 다른 차선을 판단하여 관심영역으로 정의함으로써 관심영역 밖의 차량을 제외하여

연산량을 감소시킬 수 있다. 다음으로 인지된 차량의 미래 이동 궤적은 LSTM-RNN (Long Short-Term Memory Recurrent Neural Network)에 의해 예측됩니다. 훈련을 위한 주행 데이터는 두 대의 자율주행 차량에서 직접 획득하여 제한된 시야에 관계없이 차량의 상태 정보를 제공합니다. 구글 Waymo 및 nuScenes와 같이 널리 알려진 자율주행 데이터의 경우 차량 상태 정보는 테스트 차량에 장착된 인지 센서에서 수집됩니다. 따라서 테스트 차량의 시야에서 벗어나 있는 차량 정보는 얻을 수 없습니다. 취득한 주행 데이터는 더 나은 입력 데이터 적응 및 일반화를 위해 자차가 아닌 타겟차량 중심 좌표로 구성됩니다. 손실함수로 평균 제곱 오차 및 음의 로그 우도함수를 사용하였고 음의 로그 우도함수는 자율주행 차량의 거동계획에 사용될 수 있게 타겟차량의 미래 궤적에 대한 불확실성 정보를 제공한다.

기회 제약 조건이 있는 MPC는 자율차량의 종방향 거동을 최적화하도록 구현됩니다. 동적 제약 조건 및 구동기 제약 조건은 운전자에게 승차감과 안전을 제공하도록 설계되었습니다. 기회 제약 조건은 위치 제약 조건을 강건하게 하여 안전을 보장하고 대상 차량과의 잠재적인 충돌을 방지합니다. 예측 시간동안 이동 거리에 대한 위치 제약 조건은 각 예측시간의 타겟과 자차량의 예측된 궤적 간의 거리 차이에 의해 결정된다.

제안한 알고리즘의 성능과 타당성은 컴퓨터 시뮬레이션과

테스트 데이터 기반 시뮬레이션을 통해 평가된다. 오프라인 시뮬레이션을 통해 제안한 알고리즘의 안전성을 검증하였으며 제안한 거동계획 알고리즘을 자율주행차에 구현하여 실제 도로에서 테스트하였다. 제안한 알고리즘을 실제 차량에 구현하여 실제 자율주행에 적용할 수 있음을 확인하였다.