



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사학위논문

**통합환경인지 및 모델예측제어기법
기반 안전자율주행시스템 개발**

**Automated Driving System with Guaranteed
Safety based on Generic Environment
Representation and Model Predictive Control**

2016년 2월

서울대학교 대학원

기계항공공학부

김 범 준

Abstract

Automated Driving System with Guaranteed Safety based on Generic Environment Representation and Model Predictive Control

Beomjun Kim

School of Mechanical and Aerospace Engineering

The Graduate School

Seoul National University

Recently, the interest of automotive researches changes from the passive safety system to the active safety system and, by extension, automated driving system due to advances in sensing technologies. For example, active safety applications, such as vehicle stability control (VSC), adaptive cruise control (ACC), lane keeping assistance (LKA) and lane change assistance (LCA) system), automated parking assist system (APA) and blind spot intervention (BSI), already have been commercialized by major automakers. Furthermore, there are various ongoing projects which are trying to achieve the zero fatality. Several research teams around the world are continuously advancing the field of autonomous driving. And some of major automakers have been researching to integrate individual active safety system for the enhancement of safety. GM is trying to develop and introduce ‘Super Cruise’ system which can drive on the highway without human driver’s intervention. Toyota has undertaken researches to develop ‘Automatic Highway Driving Assist’ technology. The BMW managed to drive 100% automated in real traffic on the freeway from Munich to Ingolstadt, showing a robust, comfortable, and safe driving behavior, even during multiple automated LC maneuvers and the Mercedes Benz

developed ‘Intelligent Drive’ system and followed the route from Mannheim to Pforzheim, Germany, in fully autonomous manner

From a careful review of considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption.

Therefore, this dissertation focused on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier. Mainly three research issues are considered: an environment representation, a motion planning, and a vehicle control.

In the remainder of this paper, we will provide an overview of the overall architecture of the proposed automated driving control algorithm and the experimental results which shown the effectiveness of the proposed automated driving algorithm. The effectiveness of the proposed automated driving algorithm is evaluated via vehicle tests. Test results show the robust performance on an inner-city street scenario.

Keywords: Automated driving vehicle, Model predictive control, Automated driving control algorithm, Environment representation, Safe driving envelope decision

Student Number: 2011-20689

List of Figures

Figure 2.1. System overview of the proposed automated driving system. The proposed algorithm consists of the following three steps: an environment representation, a motion planning, and a vehicle control.....	11
Figure 3.1. The experimental vehicle and its sensor-setup.....	13
Figure 3.2. Sequences of initial driving corridor decision.	18
Figure 3.3. An example of static obstacle map construction.....	20
Figure 4.1. Four coordinate systems used to derive standard process model and standard measurement models	25
Figure 4.2. Parameters to derive the standard form of the refined measurement model	30
Figure 4.3. Various driving patterns of multi target vehicles.	34
Figure 4.4. Various measurement patterns due to the target aspect with respect to the host vehicle’s radar sensor.....	34
Figure 4.5. Twelve modes of detected point of measurement model to describe various measurement patterns.....	35
Figure 4.6. Set of measurement model, their two subsets classified by validity (of each target vehicle in multi traffics), and eight events of target aspect.	38
Figure 4.7. Associated area of possible target aspect event on target-centered coordinate.....	39
Figure 4.8. Architecture of IMM/EKF multi target state estimation algorithm	40
Figure 4.9. Configurations of the host vehicle and the target vehicle	49
Figure 4.10. Test scenario: overtaking target vehicle	51
Figure 4.11. Test data: actual position (from RT-Range) and the associated radar signals of target vehicle	51
Figure 4.12. Three case examples of mode probability update	52

Figure 4.13. Mode probability update results.....	52
Figure 4.14. Estimated target states and the distributions of the estimation errors	53
Figure 4.15. Architecture of model-switching/EKF Algorithm.....	55
Figure 4.16. Multi-target state estimation in complex driving situation	60
Figure 4.17. Target state estimation results for Track 14 (a target cutting in front of the host vehicle)	61
Figure 4.18. Target state estimation results for Track 10 (a target vanishing to rear-right-blind area)	62
Figure 5.1. Procedure and Concept of Traffic Behavior Prediction	66
Figure 5.2. The procedure to calculate the parameter values: covariance of sensor noise, process noise, and modeling error	75
Figure 5.3. An example of moving object tracking and prediction.	76
Figure 5.4. Curvature measurements of the Mobileye system including 4 cases of driving scenarios	77
Figure 5.5. Density distribution of the prediction error of each method at $t_p=1.0s$ (i.e. short range prediction)	80
Figure 5.6. Density distribution of the prediction error of each method at $t_p=3.5s$ (i.e. long-range prediction).....	81
Figure 5.7. Prediction reliability evaluation along the prediction time axis ..	82
Figure 5.8. Safety envelop decision results at different time steps.	85
Figure 5.9. An example of MPC based motion planning at the instance of meeting an oncoming vehicle.	89
Figure 6.1. Normalized relative positions of human driver with respect to the safe driving rectangle which is defined 1 step (0.1s) ahead.....	93
Figure 6.2. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 1 step (0.1s) ahead.	94
Figure 6.3. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 3 steps (0.3s) ahead.	95
Figure 6.4. Histogram and contours of normalized human driver's relative	

positions with respect to the safe driving rectangle which is defined 5 steps (0.5s) ahead.....	96
Figure 6.5. Distribution of normalized human driver’s relative position with respect to the safe driving rectangle defined 1~5 step (0.1s~0.5s) ahead.	97
Figure 6.6. Configuration of test route in SNU campus (2km). The route contains a variety of different traffic situations as e.g. narrow passages, curved roads and inclined roads with other traffic participants such as oncoming vehicles, preceding vehicle, and parked busses and non-vehicle obstacles such as guardrails, pedestrian crossings, etc.	99
Figure 6.7. Test results with proposed automated driving algorithm	103
Figure 6.8. Histogram and contours of normalized relative position of ADS with respect to the safe driving rectangles which are defined 1~5-step ahead.	104
Figure 6.9. Distribution of normalized relative position of ADS with respect to the safe driving rectangles which are defined 1~5-step ahead.....	105

Contents

Chapter 1 Introduction	1
1.1. Background and Motivation	1
1.2. Previous Researches	4
1.3. Thesis Objectives	7
1.4. Thesis Outline	8
Chapter 2 Overview of an Automated Driving System	9
Chapter 3 Environment Representation	12
3.1. Driving Corridor Decision	14
3.2. Static Obstacle Map Construction	19
Chapter 4 Moving Object Tracking and Estimation	21
4.1. Problem Formulation	22
4.1.1. Stochastic hybrid system	22
4.1.2. Coordinate Systems	24
4.1.3. Standard Process Model	25
4.1.4. Standard Measurement Model.....	28
4.2. Selection of Multiple Model Set and Parameter Design.....	31
4.2.1. Set of Multiple Process Model	31
4.2.2. Set of Multiple Measurement Model.....	33
4.2.3. Event Dependent Transition Probability Matrix	35
4.3. IMM/EKF Multi Target State estimation	40
4.3.1. Host Vehicle Filter	41
4.3.2. IMM/EKF based Filtering	42

4.3.3. Track Management.....	45
4.4. Vehicle Tests based Performance Evaluation.....	47
4.4.1. Configuration of Vehicle Tests	47
4.4.2. Implementation and Evaluation.....	49
4.4.3. Comparison with Model-switching/EKF	54
4.4.4. Experimental Results with Multi-target Situation.....	57
 Chapter 5 . Safety Driving Envelope Decision and Motion Optimization.....	 63
5.1. Multi-traffic Prediction	64
5.1.1. Lane Keeping Behavior Model	66
5.1.2. Vehicle Predictor.....	68
5.1.3. Test Data based Implementation and Performance Evaluation	 72
5.2. Safety Driving Envelope Decision	83
5.3. Model Predictive Control based Motion Planning	86
 Chapter 6 Vehicle Tests based Performance Evaluation	 90
6.1. Test-Data based Simulation	91
6.2. Vehicle Tests: Automated Driving on Urban Roads	98
 Chapter 7 Conclusions	 106
 Bibliography.....	 107
 Abstract in Korean	 114

Chapter 1 Introduction

1.1. Background and Motivation

Since traffic on the road becomes more and more congested over the last decade, a large number of traffic accidents have occurred. According to the previous researches, traffic accidents caused by inattention or fatigue of the driver have been accounted for 90 percent of all traffic accidents [Huang'00]. For instance, according to the previous research [Pohl'07], unintended lane departure accounted for 15 percent of all traffic accidents occurred over the last 10 years in German roadways, and 35 percent of those have been fatal. Furthermore, traffic accidents during a lane change maneuver accounted for about 12 percent of all traffic accidents. Moreover, during many types of collision accidents, most drivers do not attempt to avoid crashes due to unawareness of collision risks [Tideman'07].

Recently, the interest of automotive researches changes from the passive safety system to the active safety system and, by extension, automated driving system due to advances in sensing technologies. For example, active safety applications, such as vehicle stability control (VSC), adaptive cruise control (ACC), lane keeping assistance (LKA) and lane change assistance (LCA system), automated parking assist system (APA) and blind spot intervention (BSI), already have been commercialized by major automakers

[Hoedemaeker'98, Bishop'00, Tingvall'00, Kato'02, Netto'04, Tideman'07, Naranjo'08, Moon'09, Gordon'10, Kastner'11, Zhang'11].

In recent years, the automotive industry is trying to achieve the zero fatality. For instance, there is an ongoing project called 'Vision Zero' in Sweden to reduce fatalities to zero by 2020. In order to achieve the zero fatality, many researches have been undertaken to integrate individual active safety systems for the development of an automated driving system [Bishop'00, Eskandarian'12]. Furthermore, some of major automakers have been researching to integrate individual active safety system for the enhancement of safety. For instance, GM is trying to develop and introduce 'Super Cruise' system which can drive on the highway without human driver's intervention. In this system, adaptive cruise control technology and active lane keeping control technology would be combined. Toyota has undertaken researches to develop 'Automatic Highway Driving Assist' technology. The objective of this system is to maintain safety of the subject vehicle and reduce fatigue on the driver. This system is developed by integrating cooperative adaptive cruise control technology, which communicates with surrounding vehicles wirelessly, and lane trace control technology.

Nowadays many major automakers have undertaken researches to develop an automated driving system. Moreover Google is active in developing technology for automated driving vehicle. The automated driving system is required to detect surrounding environment, recognize traffic situation surrounding the subject vehicle and travel autonomously without human driver's support.

From a considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption [Furgale'13]. And in order to develop a highly automated driving control algorithm, mainly three research issues should be considered: an environment representation, a motion planning, and a vehicle control.

Therefore, this dissertation focuses on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier.

1.2. Previous Researches

A number of studies have been introduced for the development of an automated driving control algorithm. Rossetter et al. designed an active lane-keeping assistance system to apply an additional guidance force by using a potential field approach [Rossetter'06]. Sattel et al. proposed a path planning algorithm to keep the lane and avoid a collision based on the potential field approach [Sattel'08]. Naranjo et al. designed a lane change control algorithm which mimics human behaviors during over taking maneuvers using fuzzy logic for autonomous vehicles [Naranjo'08]. Talvala et al. designed an autonomous driving control algorithm combining look-ahead controller and longitudinal controller to travel the reference path [Talvala'11].

In developing an automated driving system, it is required to operate in a wide operating region and limit the set of permissible states and inputs. Therefore MPC approach has been used widely because of its capability to handle system constraints in a systematic way and adaptability for various driving scenarios [Anderson'10, Falcone'07a, Falcone'08, Gray'12, Gray'13].

Borrelli et al. proposed a MPC based steering control algorithm to track a predefined desired path while satisfying constraints and evaluated this approach via simulation studies [Borrelli'05]. Falcone et al. presented a MPC based active steering controller for tracking the desired trajectory as close as possible while satisfying various constraints. In this research, the performance of the control algorithm using a nonlinear vehicle model as a predictive model is compared

with that using successive linearized vehicle model via simulation and experiment test. Then, Falcone et al. presented a sufficient stability condition for the predictive active steering control algorithm based on linear time-varying model. These approaches mentioned above only focused on the development of a lateral control law. However, there could be a limitation to follow the desired path only with a steering control. Falcone et al. designed a MPC based automated driving control algorithm to control the front steering angle and independent tractive torque at each wheel and evaluated this approach via simulation studies [Borrelli'05, Falcone'07a, Falcone'08, Falcone'07b].

In order to compensate the effect on the control performance by model uncertainties and exogenous disturbances, robust MPC approach which adds a linear feedback control input to the nominal control inputs based on the analysis of robust invariant sets have been introduced and used to design an autonomous control algorithm [Gao'14, Gray'13, Kothare'96, Mayne'00].

In most of these researches mentioned above, it is assumed that the desired trajectory over a finite horizon is known. However, in order to design an automated driving system, we should decide a drivable area or desired path in real time. Erlien et al. use a safe driving envelope which means a safe region of states in which the system should be constrained. In this research, the safe driving envelope consists of a stable handling envelope to ensure vehicle stability and an environmental envelope to constrain the position states for the collision avoidance. The environmental envelope is defined based on the current states of surrounding environment of the subject vehicle. Carvalho et al. presented a control algorithm for an autonomous ground vehicle to follow the

centerline while avoiding collisions with obstacle and evaluate this approach via experiment test. In experiment test, it is assumed that an obstacle is moving at constant speed [Carvalho'13, Erlie'13, Erlie'14].

However, according to the patent of Google at 2013, an autonomous vehicle should be controlled based on the current state of the subject vehicle, the current state of surrounding vehicles and predicted behaviors of surrounding vehicles [Ferguson'14]. Furthermore, according to previous researches, a probabilistic prediction of probable behaviors of surrounding vehicles is required to prevent a potential collision accident in the foreseeable future [Althoff'09, Kim'14, Polychronopoulos'07].

From a considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption [Furgale'13].

Therefore, in this research, we focus on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier.

1.3. Thesis Objectives

This dissertation focuses on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier.

From a considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption [Furgale'13].

Mainly three research issues are considered: an environment representation, a motion planning, and a vehicle control. In the remainder of this paper, we will provide an overview of the overall architecture of the proposed automated driving control algorithm and the experimental results which shown the effectiveness of the proposed automated driving algorithm. The effectiveness of the proposed automated driving algorithm is evaluated via vehicle tests. Test results show the robust performance on an inner-city street scenario.

1.4. Thesis Outline

This dissertation is structured in the following manner. An overall architecture of the proposed automated driving control algorithm is described in Chapter 2. In Chapter 3 and 4, an environment representation algorithm are introduced. The main objectives of the environment representation algorithm are a lane-level localization, a static obstacle detection, and moving object tracking and estimation. In Chapter 5, the concept of the probabilistic prediction of surrounding vehicle's behavior and a determination of the safe driving envelope based on the probabilistic prediction are described. Then an algorithm for a motion optimization and a vehicle control is designed based on MPC approach. Chapter 6 shows the experiment results for the evaluation of the performance of the proposed algorithm. Then the conclusion which describes the summary and contribution of the proposed automated driving control algorithm and future works is presented in Chapter 7.

Chapter 2 Overview of an Automated Driving System

From a considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption. Therefore, in this research, we focus on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier.

As aforementioned, mainly three research issues are considered: an environment representation, a motion planning, and a vehicle control. The system architecture of the algorithm is outlined in Figure 2.1. The environment representation system consists of three main modules: object classification, vehicle/non-vehicle tracking and map/lane based localization. All system modules make use of information from equipped various sensors. The main sensing components are a vision, radars and vehicle sensors. The chosen sensor

configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous robots presented earlier. The objective of the motion planning modules is to derive an optimal trajectory as a function of time, from the environment representation results. A safety envelope definition module determines the complete driving corridor that leads to the destination while assigning all objects to either the left or right corridor bound. In the case of moving objects such as other traffic participants, their behaviors are anticipated in the near future. An optimal trajectory planner uses the safety envelop as a constraint and computes a trajectory that the vehicle stays in its bounds. The vehicle control module feeds back the pose estimate of the localization module to guide the vehicle along the planned trajectory.

In the remainder of this paper, we will provide an overview of the overall architecture of the proposed automated driving control algorithm and the experimental results which shown the effectiveness of the proposed automated driving algorithm.

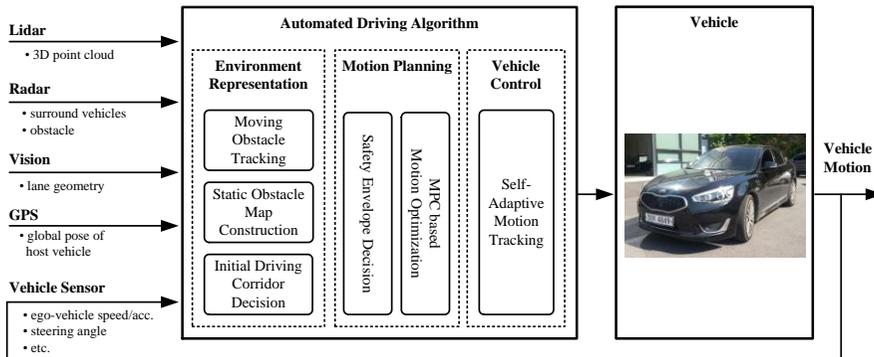


Figure 2.1. System overview of the proposed automated driving system. The proposed algorithm consists of the following three steps: an environment representation, a motion planning, and a vehicle control.

Chapter 3 Environment Representation

Precise and comprehensive environment perception is the basis for safe and comfortable autonomous driving in complex traffic situations such as busy cities [Vanholme'13, Ziegler'14]. We modified the serial-production sensor setup already available in our test vehicles as follows: A multilayer laser scanner was added for monitoring static obstacles with increased precision. For lane detection, an additional monocular vision system was mounted on the windshield. And a low-cost-GPS was equipped for a rough-precision ego-localization which is used for initial corridor decision. The complete sensor setup is shown in Figure 3.1. Depicted in yellow is a monocular vision system for lane detection. Front radar system is depicted in blue and two rear-side radars are depicted in green. A multilayer laser scanner for obstacle monitoring is shown in red and low-cost GPS is depicted in purple signal. The main objectives of these sensors are lane-level localization, static/moving obstacle detection, and drivable area representation for safe and comfortable autonomous driving.

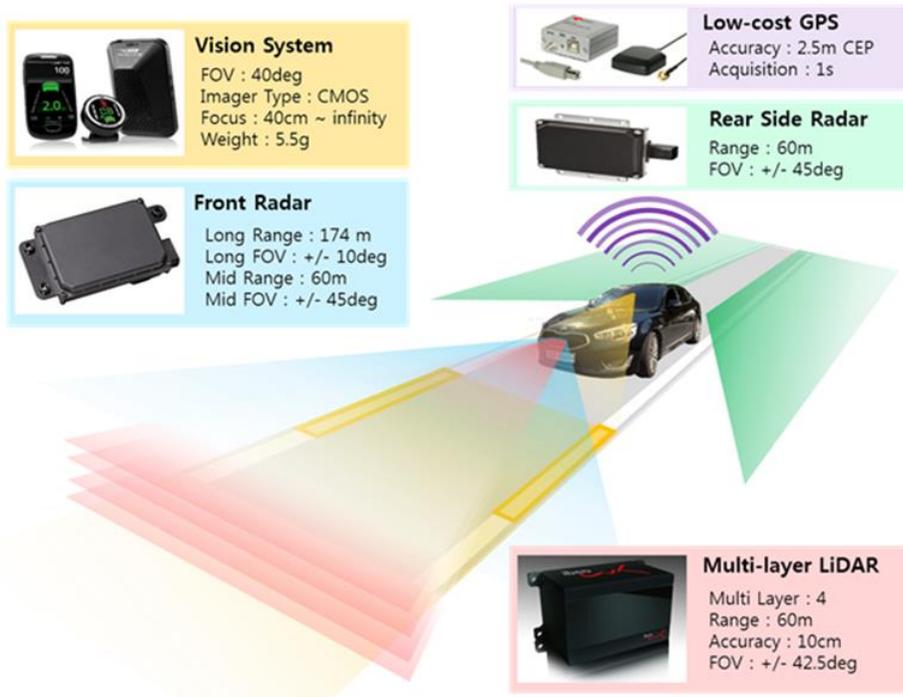


Figure 3.1. The experimental vehicle and its sensor-setup.

3.1. Driving Corridor Decision

An initial driving corridor is determined from detected lanes, rough-precision localization, and digital map which is stored in advance. The map contains properties of the environment which are necessary for driving, but cannot be reliably detected by sensors such as road width, and global waypoint of centerline of total route.

The rough-precision localization is accomplished with ego chassis sensor signals and low-cost GPS. The state vector for localization are defined as follows:

$$\mathbf{x} = [p_x \quad p_y \quad \psi \quad v_x \quad \gamma \quad a_x]^T \quad (3.1)$$

where the subscript x and y denote x-axis and y-axis of each frame, p denotes the relative position, θ denotes the relative yaw angle, v denotes the velocity, γ denotes the yaw rate, and a denotes the acceleration. The process model is discretized based on no-slip assumption as follows:

$$\begin{aligned} \mathbf{x}[k] &= \mathbf{f}(\mathbf{x}[k-1]) + \mathbf{w}[k] \\ &= \mathbf{x}[k-1] + \Delta T \cdot (\mathbf{a}(\mathbf{x}[k-1])) + \mathbf{w}[k] \end{aligned} \quad (3.2)$$

where

$$\begin{aligned} \mathbf{a}_1 &= v_x \cos \psi & \mathbf{a}_2 &= v_x \sin \psi & \mathbf{a}_3 &= \gamma & \mathbf{a}_4 &= a_x & \mathbf{a}_5 &= 0 & \mathbf{a}_6 &= 0 \\ \mathbf{w}[k] &\sim (\mathbf{0}, \mathbf{W}[k]) \end{aligned}$$

Where \mathbf{a} is the nonlinear process vector equation, and \mathbf{w} is the process noise vector which is the white noise with a covariance matrix, \mathbf{W} . The measurement model for the dead-reckoning is derived as follows:

$$\mathbf{z}_{DR}[k] = \mathbf{h}_{DR}(\mathbf{x}[k]) + \mathbf{v}_{DR}[k] \quad (3.3)$$

where

$$\begin{aligned} \mathbf{h}_{DR,1} &= v_x - \frac{w}{2} \cdot \gamma & \mathbf{h}_{DR,2} &= v_x + \frac{w}{2} \cdot \gamma & \mathbf{h}_{DR,5} &= \gamma & \mathbf{h}_{DR,6} &= a_x \\ \mathbf{h}_{DR,3} &= \begin{pmatrix} \cos \delta \left(v_x - \frac{w}{2} \cdot \gamma \right) \\ + \sin \delta (l \cdot \gamma) \end{pmatrix} & \mathbf{h}_{DR,4} &= \begin{pmatrix} \cos \delta \left(v_x + \frac{w}{2} \cdot \gamma \right) \\ + \sin \delta (l \cdot \gamma) \end{pmatrix} \\ \mathbf{v}_{DR}[k] &\sim (0, \mathbf{V}_{DR}[k]) \end{aligned}$$

Where \mathbf{h} is nonlinear measurement vector where the elements are rear-left wheel speed, rear-right wheel speed, front-left wheel speed, front-right wheel speed, yaw rate, and longitudinal acceleration in numeric order. When the GPS is updated, an augmented measurement model is modified as follows:

$$\begin{aligned} \mathbf{z}_{+GPS}[k] &= \mathbf{h}_{+GPS}(\mathbf{x}[k]) + \mathbf{v}_{+GPS}[k] \\ &= \left[\mathbf{h}_{DR}(\mathbf{x}[k]) \quad \mathbf{h}_{+GPS,7} \quad \mathbf{h}_{+GPS,8} \quad \mathbf{h}_{+GPS,9} \right]^T + \mathbf{v}_{+GPS}[k] \end{aligned} \quad (3.4)$$

where

$$\begin{aligned} \mathbf{h}_{+GPS,7} &= p_x & \mathbf{h}_{+GPS,8} &= p_y & \mathbf{h}_{+GPS,9} &= \psi \\ \mathbf{v}_{+GPS}[k] &\sim \left(0, \begin{bmatrix} \mathbf{V}_{+DR}[k] & 0 \\ 0 & \mathbf{V}_{+GPS}[k] \end{bmatrix} \right) \end{aligned}$$

From the global pose estimate of the ego vehicle, a nearest segment is selected from the map and transformed to local coordinate. From the detected lane information, distances and angle deviation between the vehicle's pose and the centerline of the local segment are updated with improved accuracy.

It is common practice to describe the forward road geometry by a 2nd-order polynomial [Swartz'03]. The relation between the ego-vehicle and the road

center line can be described by two factors: the relative lateral position and the relative heading angle. With these two factors, the road geometry, which has a curvature radius R , can be approximated by [Swartz'03]:

$$\begin{aligned} y_r(x) &= \frac{1}{2R}x^2 - \tan e_\theta \cdot x - e_y \\ &= a_2 \cdot x^2 + a_1 \cdot x + a_0 \end{aligned} \quad (3.5)$$

where x is the down-range distance, and y_r is the lateral position of the corresponding road center in the current body coordinates. As the vehicle drives with velocity v and yaw rate γ , the coefficients describing the road geometry change according to the motion of the vehicle. If the state vector are defined as these coefficients, the process model and the measurement model can be derived by:

$$\begin{aligned} \dot{\mathbf{x}}_r &= \mathbf{A}_r \mathbf{x}_r + \mathbf{B}_r \mathbf{u}_r + \mathbf{w}_r \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 2v_x & 0 & 0 \\ 0 & v_x & 0 \end{bmatrix} \begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \gamma + \mathbf{w}_r \end{aligned} \quad (3.6)$$

$$\mathbf{w}_r \sim (\mathbf{0} \quad \mathbf{Q}_r)$$

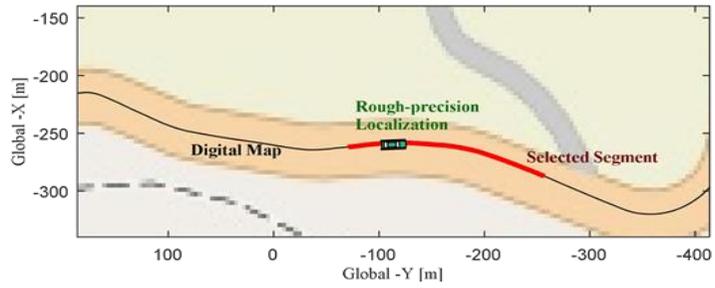
$$\begin{aligned} \mathbf{z}_r[k] &= \mathbf{H}_r \mathbf{x}_r[k] + \mathbf{v}_r[k] \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}_r[k] + \mathbf{v}_r[k] \end{aligned} \quad (3.7)$$

$$\mathbf{v}_r[k] \sim (\mathbf{0} \quad \mathbf{V}_r)$$

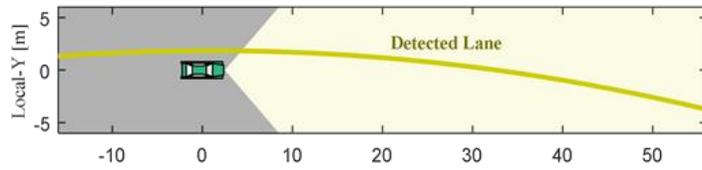
where the subscript 'r' is used to denote relation to the road geometry states. The Kalman filter is used for the estimation of road geometry coefficients. The best estimate result of the localization are used for the longitudinal velocity and the yaw rate. Hence, the covariance of the process noise should be well defined so that it can represent the effect of the estimate error of the vehicle motion

filtering. As a result, the estimate of the initial driving corridor coefficients is recursively estimated using the Kalman filter [RE'60]. Both detected lane and selected local segment are used for measurement update. Note that the covariance of the measurement noise should be well defined so that it can represent each of two errors: the localization error and lane detection error.

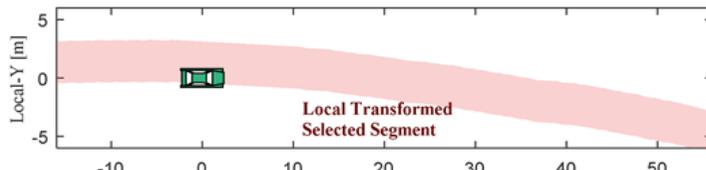
Figure 3.2 shows an example of sequence of initial driving corridor decision. Figure 3.2-(a) shows a nearest segment selection from the map data. Digital map is depicted as black line, rough-precision global pose estimate result is depicted as green vehicle, and the selected nearest segment is depicted as red-thick line. Figure 3.2-(b) shows a lane detection results. From the figure, we can see that yellow centerline is detected. Figure 3.2-(c) shows transformed segment to local coordinate. Due to the error of the global pose estimate, it seems that the ego-vehicle deviated from the lane (actually not). Figure 3.2-(d) shows lane-level segment matching result. Because the locally transformed segment is updated with the detected lane, the updated local segment has more improved accuracy.



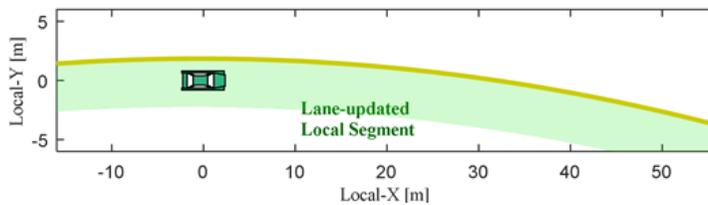
(a) Nearest segment selection



(b) Lane detection



(c) Segment transformation to local coordinate



(d) Lane-level segment matching

Figure 3.2. Sequences of initial driving corridor decision.

3.2. Static Obstacle Map Construction

A grid map representation was used for the static obstacle recognition. To cope with noisy measurement of laser scanner, we accumulate multiple observations to obtain reliable map information. Each grid of the map has the counter which indicates how many times obstacles have been observed in that grid. At each observation, the counter of each grid where obstacles exist is incremented. If the counter value of a grid is higher than a certain threshold, the grid is considered unsafe. The set of unsafe grids constitutes the current static obstacle map.

In the case of the candidates for moving objects, they are detected by comparing the current observation with estimated moving objects from radar. If the point is completely inside the region of estimated moving object, the point is considered as a part of a moving object.

Since the vehicle generates a map while it moves, we first transform the existing map to the current local position (including orientation) of the vehicle. And then current measured data is integrated to the transformed map. The position deviation of the vehicle is estimated from the odometry. To reduce the effect of accumulated error when the vehicle moves by a long distance, only thirteen latest observations are used for making the static obstacle map construction.

Figure 3.3 shows an example of static obstacle map construction. The sensing area of laser scanner is depicted as yellow region. Blue dots are current

observation while the red squares are unsafe-grids of the constructed static obstacle map. We can see that parked-buses on the right side and left-barrier are well tracked though they are hidden in blind area. And the part of a moving object is recognized very well.

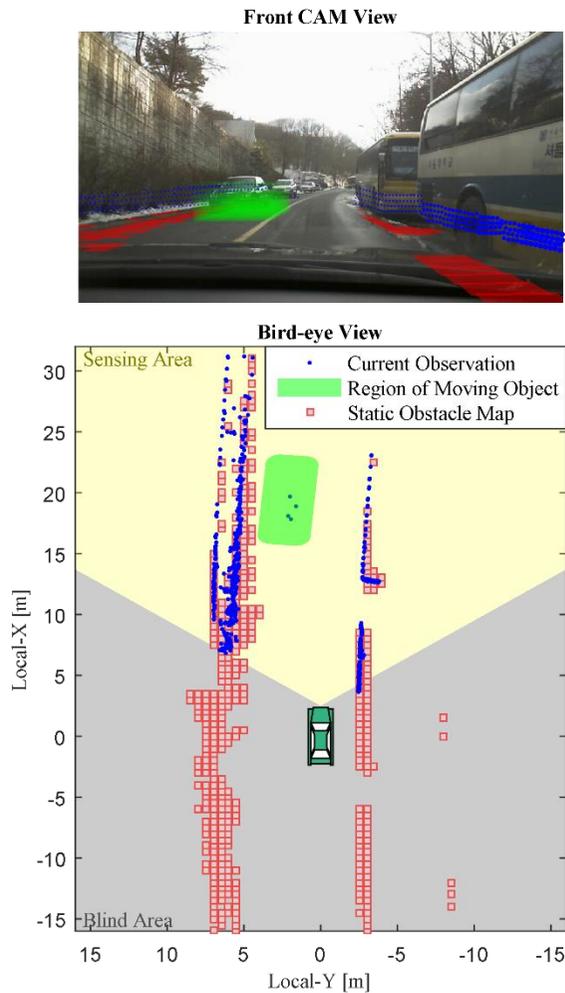


Figure 3.3. An example of static obstacle map construction.

Chapter 4 Moving Object Tracking and Estimation

For the generic assessment and the total management of collision risks with multi-traffics in complex driving situations, it is essential to estimate and represent the target vehicles' overall behaviors such as heading, yaw rate, absolute velocity and acceleration as well as relative position and relative velocity which are the state of the target vehicle with respect to the host vehicle.

To achieve this, this chapter describes an interacting multiple model (IMM) approach using extended Kalman filters (EKF) to improve multi-target states estimation performance with utilization of automotive radars. Automotive radar is the best fitted vehicular surround sensing technology with respect to functionality, robustness, reliability, dependence on weather conditions etc.

In an application of the automotive radar, the most important issue is to handle an uncertain measurement model problem that is wandering on the target's physical boundary. To cope with this problem, multi-models are formulated and a new multi-target tracking algorithm is developed based on IMM approach, global nearest neighbor (GNN) based data association and EKF method with elaborated modeling of automotive radar.

The performance of the proposed multi-target tracking algorithm is verified via vehicle tests in real roads. It is shown that the target vehicle's overall behavior can be estimated by proposed elaborated models and the estimation performance can be significantly enhanced.

4.1. Problem Formulation

In this section, a stochastic hybrid system is formulated in the form of an IMM/EKF algorithm for tracking multiple target vehicles. A standard process model to describe behaviors of target vehicle is derived and discretized. And a standard measurement model is defined to describe automotive radar characteristics.

4.1.1. Stochastic hybrid system

Assume that there are total N_{target} targets and the system of each target can be modeled as one of the N_{mode} hypothesized models where the target set is denoted as $\mathbf{T} \equiv \{1, 2, \dots, N_{target}\}$ and the model set is denoted as $\mathbf{M} \equiv \{1, 2, \dots, N_{mode}\}$. From the work of Li and Bar-shalom [Li'93], a general description for a stochastic hybrid system of target $n (\in \mathbf{T})$ during the sampling period $(t_{k-1}, t_k]$ can be given as:

$$\mathbf{x}_n^j[k] = \mathbf{f}_n^j(\mathbf{x}_n^j[k-1], \mathbf{u}[k-1]) + \mathbf{w}_n^j[k-1] \quad (3.8)$$

$$\mathbf{z}_n^j[k] = \mathbf{h}_n^j(\mathbf{x}_n^j[k], \mathbf{u}[k]) + \mathbf{v}_n^j[k] \quad (3.9)$$

where $\mathbf{x}_n^j[\cdot]$ is the state vector of target n when model $j (\in \mathbf{M})$ is in effect, $\mathbf{u}[\cdot]$ is the input vector (same for all targets), $\mathbf{f}_n^j(\cdot)$ and $\mathbf{h}_n^j(\cdot)$ are the

nonlinear and time invariant system structure. The process noise $\mathbf{w}_n^j[k]$ and the measurement noise $\mathbf{v}_n^j[k]$ are mutually uncorrelated zero-mean white Gaussian with covariance $\mathbf{W}_n^j[k]$ and $\mathbf{V}_n^j[k]$ respectively. And a (possibly state dependent) Markovian transition probability of the system mode index is given as follows.

$$\begin{aligned} &\forall i, j \in \mathbf{M} \\ &\Pr\{m_n[k+1] = j | m_n[k] = i\} = \Phi_{i,j}(\mathbf{x}_n[k]) \end{aligned} \quad (3.10)$$

where $m_n[k]$ is the system mode index of target n which is in effect at time step k .

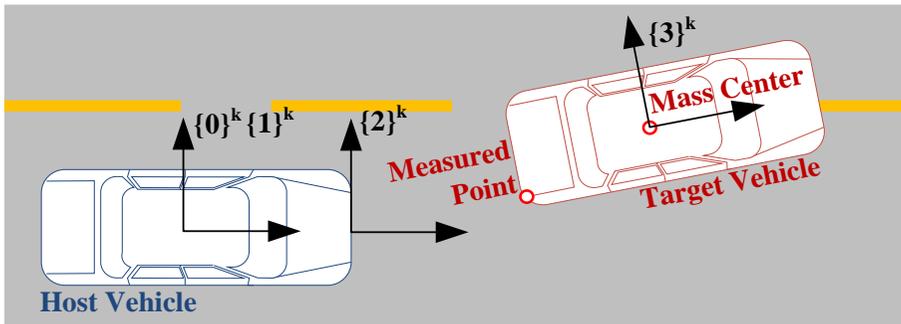
The following notations and definitions are used regarding the measurement. In general, at any time k , some measurements may be due to clutter and some due to the target, i.e. there can be more than a single measurement [Chen'01]. The measurement set (not yet validated or associated) is denoted as $\mathbf{Y}_k \equiv \{\mathbf{y}_1[k], \mathbf{y}_2[k], \dots, \mathbf{y}_{N_{measure}}[k]\}$ where $N_{measure}$ is the number of measurements. The set of validated measurement of target n at time k is denoted by

$$\mathbf{Z}_k \equiv \{\mathbf{z}_1[k], \mathbf{z}_2[k], \dots, \mathbf{z}_{N_{target}}[k]\} \quad (3.11)$$

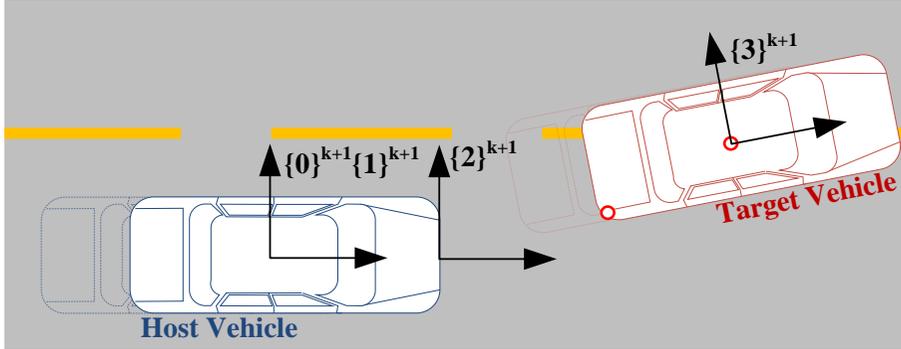
Therefore, the goals of the algorithm can be summarized as follows: 1) to find the set of validated measurement for each target and 2) to find the minimum variance estimated state and the associated error covariance matrix for each target.

4.1.2. Coordinate Systems

In this subsection, coordinate systems used to derive standard process and measurement models are presented. These coordinate systems are described in Figure 4.1. Frame $\{0\}$ is the ground based fixed coordinate system of which the position is same with the host vehicle, frame $\{1\}$ is the host vehicle's body-fixed moving coordinate system, frame $\{2\}$ is the sensor's body-fixed moving coordinate system, and frame $\{3\}$ is the target vehicle's body-fixed moving coordinate system. And superscript means a corresponding time step. The standard process model describes the predicted state of the target vehicle's mass center on the frame of $\{0\}^{k+1}$ based on the current state of the target vehicle's mass center and system input which are defined on the frame of $\{0\}^k$. And the standard measurement model describes the state of measured point such as position and longitudinal velocity on the sensor based moving frame $\{2\}$.



(a) at time k



(b) at time $k+1$

Figure 4.1. Four coordinate systems used to derive standard process model and standard measurement models

4.1.3. Standard Process Model

Various driving patterns of a vehicle including straight line, curve, cut-in/out, U-turn and interchange can be represented by a combination of a constant velocity rectilinear motion, a constant acceleration rectilinear motion, a constant angular velocity curvilinear motion and a constant angular acceleration curvilinear motion [Laugier'11].

To describe all these motions, the state vector and input vector of target n are defined as follows:

$$\mathbf{x}_n = [p_{n,x} \quad p_{n,y} \quad \theta_n \quad v_{n,x} \quad \gamma_n \quad a_{n,x} \quad \dot{\gamma}_n]^T \quad (3.12)$$

$$\mathbf{u} = [v_x \quad \gamma]^T \quad (3.13)$$

where subscript x and y denote x-axis and y-axis of each frame, subscript n denotes “of target n “, p denotes the relative position, θ denotes the

relative yaw angle, v denotes the velocity, γ denotes the yaw rate, a denotes the acceleration, and $\dot{\gamma}$ denotes the yaw acceleration. And variable without subscript n means “variable of the host vehicle”. Therefore v_x denotes the longitudinal velocity of host vehicle and γ denotes the yaw rate of host vehicle. Note that $p_{n,x}$, $p_{n,y}$ and θ_n are defined on the host vehicle’s body-fixed moving frame, $\{1\}$, and other elements are defined on the ground-based fixed frame, $\{0\}$. With no slip assumption ($v_{n,y} \approx 0$), the continuous process model can be formed as:

$$\begin{aligned}
\dot{\mathbf{x}}_n &= \mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \mathbf{q} \\
&= [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_3 \quad \mathbf{a}_4 \quad \mathbf{a}_5 \quad \mathbf{a}_6 \quad \mathbf{a}_7]^T + \mathbf{q} \\
\mathbf{a}_1 &= v_{n,x} \cos \theta_i - v_x + p_{n,y} \cdot \gamma & \mathbf{a}_2 &= v_{n,x} \sin \theta_i - p_{n,x} \cdot \gamma & (3.14) \\
\mathbf{a}_3 &= \gamma_n - \gamma & \mathbf{a}_4 &= a_{n,x} & \mathbf{a}_5 &= \dot{\gamma}_n & \mathbf{a}_6 &= k_a & \mathbf{a}_7 &= k_{\dot{\gamma}} \\
\mathbf{q} &\sim (\mathbf{0}, \mathbf{Q})
\end{aligned}$$

where \mathbf{a} is the nonlinear process vector equation, k_a is the decay-rate of the longitudinal acceleration, $k_{\dot{\gamma}}$ is the decay-rate of the yaw-acceleration, and \mathbf{q} is the continuous white process noise with covariance \mathbf{Q} . Above continuous process model can be discretized via Taylor methods as follows [Kazantzis'99]:

$$\begin{aligned}
\mathbf{x}_n[k+1] &= \mathbf{x}_n(t + \Delta T) \\
&= \mathbf{x}_n(t) + \frac{d}{dt} \mathbf{x}_n \cdot \Delta T + \frac{1}{2} \cdot \frac{d^2}{dt^2} \mathbf{x}_n \cdot \Delta T^2 + \mathbf{h.o.t.} \\
&\cong \mathbf{x}_n(t) + \{\mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \mathbf{q}\} \cdot \Delta T + \frac{1}{2} \cdot \frac{d}{dt} \{\mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \mathbf{q}\} \cdot \Delta T^2 \\
&= \left[\begin{array}{l} \mathbf{x}_n(t) + \{\mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \mathbf{q}\} \cdot \Delta T + \\ \frac{1}{2} \cdot \left\{ \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \dot{\mathbf{x}}_n + \frac{\partial}{\partial \mathbf{u}} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \dot{\mathbf{u}} + \dot{\mathbf{q}} \right\} \cdot \Delta T^2 \end{array} \right] \\
&= \left[\begin{array}{l} \mathbf{x}_n(t) + \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \Delta T + \\ \frac{1}{2} \cdot \left\{ \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \frac{\partial}{\partial \mathbf{u}} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \dot{\mathbf{u}} \right\} \cdot \Delta T^2 + \\ \left\{ \Delta T \cdot \mathbf{I} + \frac{\Delta T^2}{2} \cdot \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \right\} \mathbf{q} \end{array} \right] \\
&= \mathbf{f}(\mathbf{x}_n[k], \mathbf{u}[k]) + \mathbf{w}[k] \\
&= [\mathbf{f}_1 \quad \mathbf{f}_2 \quad \mathbf{f}_3 \quad \mathbf{f}_4 \quad \mathbf{f}_5 \quad \mathbf{f}_6 \quad \mathbf{f}_7]^T + \mathbf{w}[k]
\end{aligned} \tag{3.15}$$

where

$$\begin{aligned}
\mathbf{f}_1 &= - \left(\begin{array}{l} \frac{\gamma_{host} (p_{n,x} \gamma_{host} - v_n \sin(\theta_n))}{2} \\ - \frac{a_n \cos(\theta_n)}{2} + \frac{\gamma_n v_n \sin(\theta_n)}{2} \\ + \frac{a_{host}}{2} - \frac{p_{n,y} \dot{\gamma}_{host}}{2} \end{array} \right) \Delta T^2 + \left(\begin{array}{l} p_{n,y} \gamma_{host} - v_{host} \\ + v_n \cos(\theta_n) \end{array} \right) \Delta T + p_{n,x} \\
\mathbf{f}_2 &= - \left(\begin{array}{l} \frac{a_n \sin(\theta_n)}{2} + \frac{\gamma_n v_n \cos(\theta_n)}{2} + \frac{p_{n,x} \dot{\gamma}_{host}}{2} \\ - \frac{\gamma_{host} (p_{n,y} \gamma_{host} - v_{host} + v_n \cos(\theta_n))}{2} \end{array} \right) \Delta T^2 - \left(\begin{array}{l} p_{n,x} \gamma_{host} \\ - v_n \sin(\theta_n) \end{array} \right) \Delta T + p_{n,y} \\
\mathbf{f}_3 &= \frac{\dot{\gamma}_n}{2} \Delta T^2 + \gamma_n \Delta T + \theta_n & \mathbf{f}_4 &= - \frac{a_n k_a}{2} \Delta T^2 + a_n \Delta T + v_n \\
\mathbf{f}_5 &= - \frac{k_{\dot{\gamma}} \dot{\gamma}_n}{2} \Delta T^2 + \dot{\gamma}_n \Delta T + \gamma_n & \mathbf{f}_6 &= \frac{a_n}{2} (k_a^2 \Delta T^2 - 2k_a \Delta T + 2)
\end{aligned}$$

$$\mathbf{f}_\gamma = \frac{\dot{\gamma}_n}{2} (k_\gamma^2 \Delta T^2 - 2k_\gamma \Delta T + 2)$$

$$\mathbf{w}[k] \sim (0, \mathbf{W}[k])$$

$$\mathbf{W}[k] = E[\mathbf{w} \cdot \mathbf{w}^T]$$

$$\begin{aligned} &= E \left[\left[\left\{ \Delta T \cdot \mathbf{I} + \frac{\Delta T^2}{2} \cdot \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \right\} \mathbf{q} \right] [\dots]^T \right] \\ &= \left\{ \Delta T \cdot \mathbf{I} + \frac{\Delta T^2}{2} \cdot \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \right\} E[\mathbf{q} \cdot \mathbf{q}^T] \{\dots\}^T \\ &= \left\{ \Delta T \cdot \mathbf{I} + \frac{\Delta T^2}{2} \cdot \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \right\} \mathbf{Q} \{\dots\}^T \Bigg|_{\substack{\mathbf{x}_n = \mathbf{x}_n[k] \\ \mathbf{u} = \mathbf{u}[k]}} \end{aligned}$$

$$\dot{\mathbf{u}} = \frac{d}{dt} \begin{bmatrix} v_x & \gamma \end{bmatrix}^T = \begin{bmatrix} a_x & \dot{\gamma} \end{bmatrix}^T$$

where $\dot{\mathbf{u}}$ is time-derivative of input vector where the components' physical meanings are host vehicle's longitudinal acceleration and yaw-acceleration. As the result, $\mathbf{x}_n[k+1]$ describes the predicted state of the target vehicle's mass center on the frame of $\{0\}^{k+1}$. Note that various driving patterns of a vehicle can be represented by the above standard discretized process model by adjusting \mathbf{a} and \mathbf{Q} of continuous process model. The details about specific modes of process model will be discussed in following subsection, 4.2.1.

4.1.4. Standard Measurement Model

In an automotive target tracking, changes in the target aspect with respect to the radar can cause the apparent point of radar reflections (relative position seen by the antenna) to wander significantly [Skolnik'01]. To represent these characteristics, the measurement model can be elaborated by introducing new parameters to specify the sensor position and the measured point. These are

depicted in Figure 4.2. As the result, the measurement vector of target n is defined as follows.

$$\begin{aligned}\mathbf{z}_n[k] &= \mathbf{h}(\mathbf{x}_n[k], \mathbf{u}[k]) + \mathbf{v}_n[k] \\ &= [\mathbf{h}_{n1} \quad \mathbf{h}_{n2} \quad \mathbf{h}_{n3}]^T + \mathbf{v}_n[k] \\ \mathbf{v}_n[k] &\sim (0, \mathbf{V}_n[k])\end{aligned}\tag{3.16}$$

where

$$\begin{aligned}\mathbf{h}_{n1} &= p_{n,x} - s_x + b_{n,x} \cos \theta_n - b_{n,y} \sin \theta_n \\ \mathbf{h}_{n2} &= p_{n,y} - s_y + b_{n,x} \sin \theta_n + b_{n,y} \cos \theta_n \\ \mathbf{h}_{n3} &= v_{n,x} \cos \theta_n - v_x + p_{n,y} \cdot \gamma + (b_{n,x} \sin \theta_n + b_{n,y} \cos \theta_n)(\gamma - \gamma_n)\end{aligned}$$

where \mathbf{h} is nonlinear measurement equation which describe the state of measured point on the sensor based moving frame {2}. \mathbf{s} is sensor position vector defined on the host vehicle's body-fixed moving frame {1} and \mathbf{b}_n is measured point vector of target n defined on the target vehicle's body-fixed moving frame {3}. The first order approximation of the measurement error covariance has been presented in previous works as follows [Lerro'93].

$$\mathbf{V}_n = \begin{bmatrix} \frac{\sigma_r^2 - r_n^2 \sigma_\theta^2}{2} \begin{bmatrix} b_n + \cos 2\theta_n & \sin 2\theta_n \\ \sin 2\theta_n & b_n - \cos 2\theta_n \end{bmatrix} & \mathbf{0}_{2 \times 1} \\ \mathbf{0}_{1 \times 2} & \sigma_v^2 \end{bmatrix}\tag{3.17}$$

where

$$b_n = \frac{\sigma_r^2 + r_n^2 \sigma_\theta^2}{\sigma_r^2 - r_n^2 \sigma_\theta^2} \quad r_n = \sqrt{\mathbf{h}_{n1}^2 + \mathbf{h}_{n2}^2} \quad \theta_n = \tan^{-1} \left(\frac{\mathbf{h}_{n2}}{\mathbf{h}_{n1}} \right)$$

where r_n and θ_n are the range and azimuth measurements, respectively. σ_r , σ_θ and σ_v are the standard deviations of the range, the azimuth and the

relative velocity, respectively. The approximation is validated to be useful for practical parameters [Li'93]. The various measurement patterns due to the target aspect with respect to radar can be represented by the above standard measurement model by adjusting the measured point vector \mathbf{b}_n . The details about specific modes of measurement model will be discussed more fully in subsection, 4.2.2

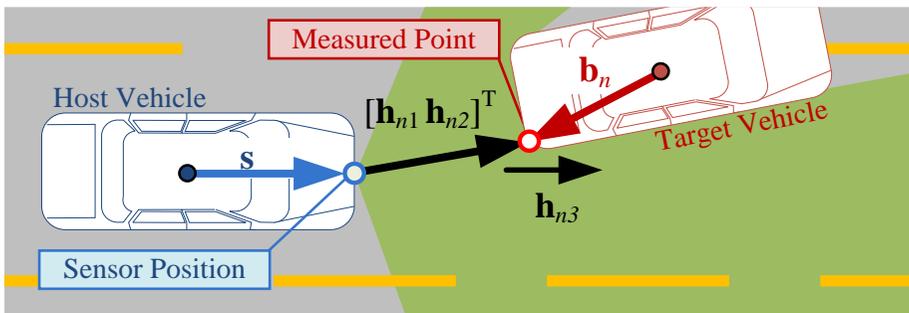


Figure 4.2. Parameters to derive the standard form of the refined measurement model

4.2. Selection of Multiple Model Set and Parameter

Design

To obtain the best estimate, the IMM algorithm has to be properly designed and tuned to meet the special requirements of the automotive application. And these requirements are fulfilled by means of 1) selection of vehicle motion models for all modes of driving, 2) selection of measurement models for all cases of measurements, and 3) determination of the parameters of the underlying Markov chain, that is, the transition probabilities [Li'93]. Therefore, in this section, after analyzing the driving patterns and automotive radar characteristics, four specific modes of process model, twelve specific modes of measurement model, and event dependent transition probability matrix are derived and introduced.

4.2.1. Set of Multiple Process Model

Various driving patterns considered in this section are depicted in Figure 4.3. To describe these various driving patterns of a vehicle, four specific modes of process model will be discussed in this section. As aforementioned, four motions include the constant velocity rectilinear motion, the constant acceleration rectilinear motion, the constant angular velocity curvilinear motion and the constant angular acceleration curvilinear motion are considered. Assuming that accelerations in the steady state are quite small, linear accelerations or decelerations can be reasonably well covered by process noises

with the constant velocity model. With these assumptions, the Constant Velocity Rectilinear (CVR) motion can be modeled by:

$$\begin{aligned}\dot{\mathbf{x}}_n &= \mathbf{a}(x_n, u) + \mathbf{q} \\ &= [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_{3,CVR} \quad 0 \quad 0 \quad 0 \quad 0]^T + \mathbf{q} \\ \mathbf{q} &\sim (\mathbf{0}, \mathbf{Q}_{CVR})\end{aligned}\tag{3.18}$$

where

$$\begin{aligned}\mathbf{a}_{3,CVR} &= -\gamma \\ \mathbf{Q}_{CVR} &= \text{diag}([0 \quad 0 \quad 0 \quad \mathbf{Q}_{44} \quad 0 \quad 0 \quad 0])\end{aligned}$$

where \mathbf{Q} is the appropriate covariance which can represent the zero-mean Gaussian white noise assumed accelerations. In the same way, the Constant Acceleration Rectilinear (CAR) motion can be formulated as follows:

$$\begin{aligned}\dot{\mathbf{x}}_n &= [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_{3,CVR} \quad \mathbf{a}_{4,CAR} \quad 0 \quad 0 \quad 0]^T + \mathbf{q} \\ \mathbf{q} &\sim (\mathbf{0}, \mathbf{Q}_{CAR})\end{aligned}\tag{3.19}$$

where

$$\begin{aligned}\mathbf{a}_{4,CAR} &= a_{n,x} \\ \mathbf{Q}_{CAR} &= \text{diag}([0 \quad 0 \quad 0 \quad 0 \quad 0 \quad \mathbf{Q}_{66} \quad 0])\end{aligned}$$

The constant angular velocity curvilinear motion is a turning motion with a constant yaw rate along a road of constant radius of curvature. With a zero-mean Gaussian white noise to the derivative of yaw rate, the Constant Velocity Curvilinear (CVC) motion can be modeled by:

$$\begin{aligned}\dot{\mathbf{x}}_n &= [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_{3,CVC} \quad 0 \quad 0 \quad 0 \quad 0]^T + \mathbf{q} \\ \mathbf{q} &\sim (\mathbf{0}, \mathbf{Q}_{CVC})\end{aligned}\tag{3.20}$$

where

$$\begin{aligned}\mathbf{a}_{3,CVC} &= \gamma_n - \gamma \\ \mathbf{Q}_{CVC} &= \text{diag}([0 \ 0 \ 0 \ 0 \ \mathbf{Q}_{55} \ 0 \ 0])\end{aligned}$$

With some modifications to (13), the Constant Acceleration Curvilinear (CAC) motion can be derived by:

$$\begin{aligned}\dot{\mathbf{x}}_n &= [\mathbf{a}_1 \ \mathbf{a}_2 \ \mathbf{a}_{3,CVC} \ 0 \ \mathbf{a}_{5,CAC} \ 0 \ 0]^T + \mathbf{q} \\ \mathbf{q} &\sim (\mathbf{0}, \mathbf{Q}_{CAC})\end{aligned}\tag{3.21}$$

where

$$\begin{aligned}\mathbf{a}_{5,CAC} &= \dot{\gamma}_n \\ \mathbf{Q}_{CAC} &= \text{diag}([0 \ 0 \ 0 \ 0 \ 0 \ 0 \ \mathbf{Q}_{77}])\end{aligned}$$

Evidently, the discretized process model derived from (10)-(13) are special forms of (7) and can be summarized by (1). In addition, it is reasonable to assume that the transition between the driving modes of a target vehicle has the Markovian probability governed by (3). Consequently, the process model can be suitable described in the framework of the stochastic hybrid systems.

4.2.2. Set of Multiple Measurement Model

As shown in Figure 4.4, by means of relative position and orientation between host and target vehicle, the configuration of measurement can be altered. Ideally, the detecting point should be located on the four sides or four edges of the target vehicle. As an example, for the case of the cutting in vehicle, a rear right edge is the most likely measurement model while lane changing. However, after cutting-in, the most likely measured point is transited to a center

of rear side of the target vehicle.

As aforementioned, to describe these various measurement patterns, several specific modes can be derived by adjusting \mathbf{b}_n in standard measurement model. From the viewpoint of detecting position defined on the target-body-fixed frame, a specific mode set (depicted in Figure 4.4) can be utilized to describe the variations of the measurement patterns. We can see that the proposed mode set include twelve points around the boundary of the target vehicle and all case examples in Figure 4.5 can be represented well by the proposed mode set. Consequently, derived set of measurement models can be suitable described by the stochastic hybrid systems.

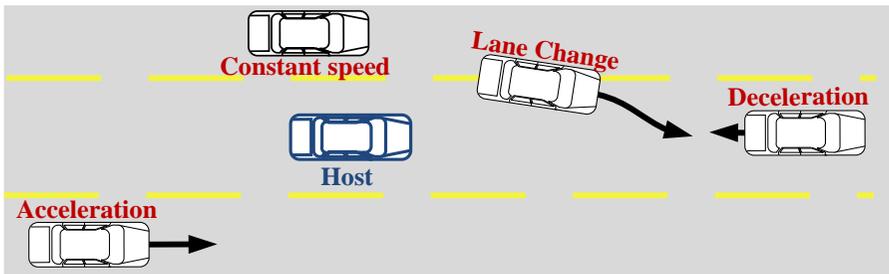


Figure 4.3. Various driving patterns of multi target vehicles.

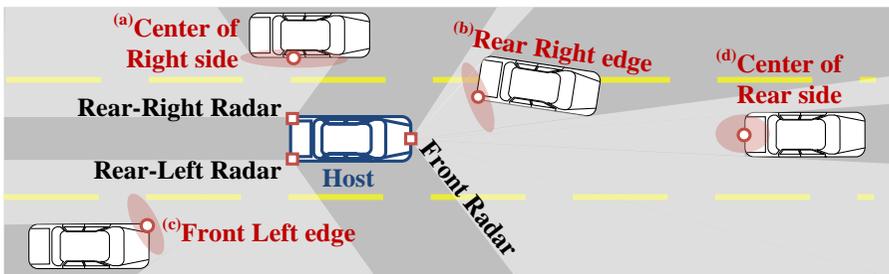


Figure 4.4. Various measurement patterns due to the target aspect with respect

to the host vehicle's radar sensor.

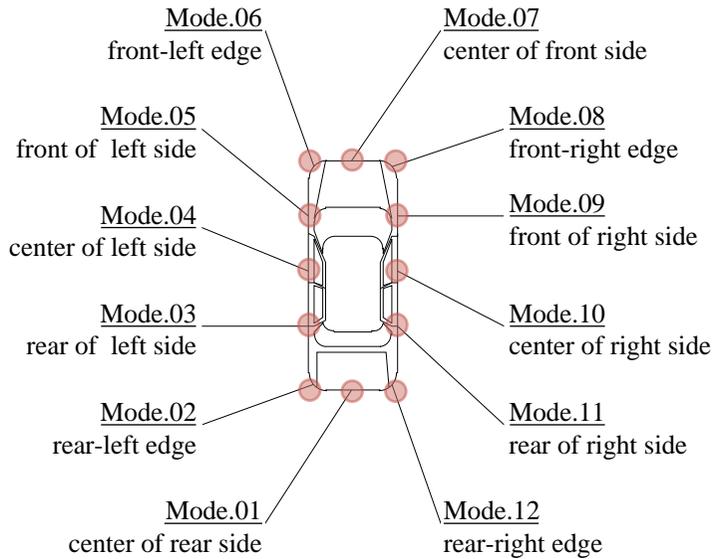


Figure 4.5. Twelve modes of detected point of measurement model to describe various measurement patterns.

4.2.3. Event Dependent Transition Probability Matrix

In a conventional IMM estimator the Markov chain of the mode states is described by a single mode transition matrix. However, unlike the conventional system, mode transition process of an automotive radar system cannot be modeled by a single mode transition probability matrix. As can be seen in Figure 4.6, transitions to invalid modes are infeasible since the radar cannot penetrate the vehicle's body. And a subset of invalid modes can be different for different situations. As an example, mode transition from the front-right edge to rear-left edge is impossible in event.1 while it is possible in event.8. Therefore, we can conclude that the mode transition probabilities of the

automotive radar system are target aspect dependent and multi transition matrices should be derived to describe all cases of the mode jump process.

A transition probability matrix of the proposed algorithm can be decomposed into two sub-matrices; one is the sub-transition probability matrix related to multiple process models and the other matrix is about multiple measurement models. Relation of three matrices is given as:

$$\Phi = \begin{bmatrix} \Phi_{1 \times 1}^{process} \cdot \Phi^{measure} & \Phi_{1 \times 2}^{process} \cdot \Phi^{measure} & \dots & \Phi_{1 \times 1}^{process} \cdot \Phi^{measure} \\ \Phi_{2 \times 1}^{process} \cdot \Phi^{measure} & \Phi_{2 \times 2}^{process} \cdot \Phi^{measure} & \dots & \Phi_{2 \times N_{mode}^{process}} \cdot \Phi^{measure} \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{N_{mode}^{process} \times 1}^{process} \cdot \Phi^{measure} & \Phi_{N_{mode}^{process} \times 2}^{process} \cdot \Phi^{measure} & \dots & \Phi_{N_{mode}^{process} \times N_{mode}^{process}} \cdot \Phi^{measure} \end{bmatrix} \quad (3.22)$$

where $\Phi^{process}$ is the matrix defining a transition probability between process mode indices, $N_{mode}^{process}$ is the number of process model set, $\Phi^{measure}$ defines a transition probability between measurement mode indices. $\Phi^{process}$ can be expressed in a single matrix, but this is impossible in the case of $\Phi^{measure}$ as aforementioned. To solve this problem, different measurement mode transition matrices are used depending on the target aspect event. All eight possible events are defined in Figure 4.7. From the work of Zhang and Bar-shalom [Zhang'11], an approach using state-dependent mode transition probabilities has been presented to track move-stop-move targets. And in our research, this approach is extended to eight stages and applied with some modifications. Target index is omitted unless otherwise required for clarity. First, measurement mode transition probability matrix with no event (physically not allowed) is designed in Gaussian distribution manner as follows:

$\forall i, j \in \mathbf{M}$

$$\Phi_{i,j}^{measure,0} = \frac{1}{\sqrt{2\pi}\sigma_d} \exp\left[-\frac{1}{2}\left(\frac{d_{i,j}}{\sigma_d}\right)^2\right] \quad (3.23)$$

where $d_{i,j}$ is the distance travelled from i to j along the vehicle's body line. And with this raw transition probability, a transition probability conditioned on a specific event E is given by:

$$\Phi^{measure,E} = \Phi^{measure,0} \cdot \Theta^E = \left[\Phi_{i,j}^{measure,0} \right] \cdot \Theta^E \quad (3.24)$$

where Θ^E is an event dependent diagonal matrix of which elements are defines as follows:

$$\Theta_{i,j}^E = \begin{cases} 1 & i = j \in \mathbf{M}_{valid}^{measure,E} \\ 0 & i = j \in \mathbf{M}_{invalid}^{measure,E} \\ 0 & i \neq j \end{cases} \quad (3.25)$$

where $\mathbf{M}_{valid}^{measure,E}$ is the subset of valid measurement modes and $\mathbf{M}_{invalid}^{measure,E}$ is the subset of invalid measurement modes on an event E . Consequently, the conditional transition probability matrix has zero values as entries of columns of invalid modes, which means that under that given event, transitions to invalid modes are not allowed.

Finally, the actual transition probability matrix of the system at time step k is calculated as a combined form which can be written as follows:

$\forall i \in \mathbf{E}$

$$\Phi^{measure} = \sum_i \Pr(E[k]=i | \mathbf{z}[0], \dots, \mathbf{z}[k]) \cdot \Phi^{measure,i} \quad (3.26)$$

Where $\mathbf{E} = \{i | i = 1, \dots, 8\}$ is a set of eight event indexes and $\Pr(E[k] = i | \mathbf{z}[0], \dots, \mathbf{z}[k])$ is an event probability which means the probability of the target being in a specific event i at time step k conditioned on the cumulative set of measurements up to time k . The event probability can be calculated directly as:

$$\forall i \in \mathbf{E} \quad \Pr(E[k] = i | \mathbf{z}[0], \dots, \mathbf{z}[k]) = \int \int_{A_i} f_{\tilde{\mathbf{s}} | \mathbf{z}[0], \dots, \mathbf{z}[k]}(\tilde{\mathbf{s}}[k]) d\tilde{\mathbf{s}} \quad (3.27)$$

where $\tilde{\mathbf{s}}$ denotes a sensor position vector transformed in target-body-fixed coordinate, $f_{\tilde{\mathbf{s}} | \mathbf{z}[0], \dots, \mathbf{z}[k]}(\cdot)$ is a conditional probability density function associated with $\tilde{\mathbf{s}}$ conditioned on the cumulative set of measurements, and A_i is an associated area with event i which is depicted in Figure 4.7.

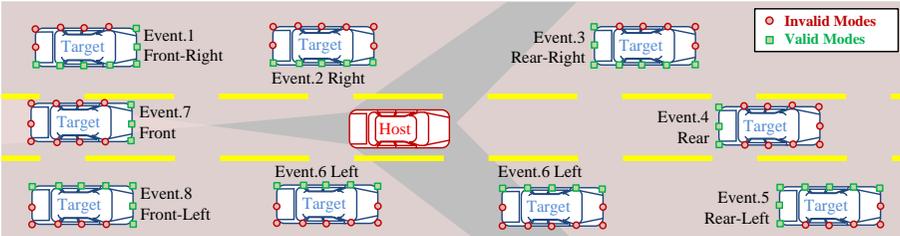


Figure 4.6. Set of measurement model, their two subsets classified by validity (of each target vehicle in multi traffics), and eight events of target aspect.

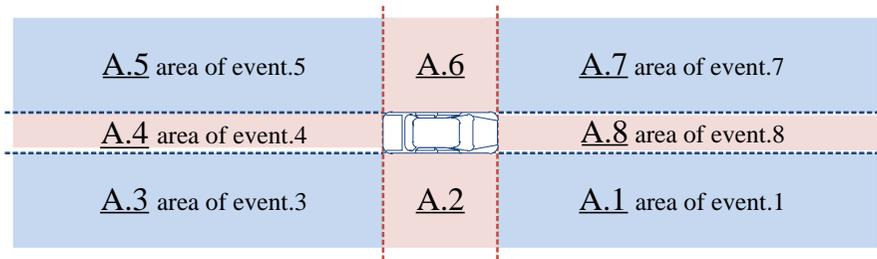


Figure 4.7. Associated area of possible target aspect event on target-centered coordinate

4.3. IMM/EKF Multi Target State estimation

In this section, an IMM/EKF based filtering algorithm for tracking multiple target vehicles is proposed. The overall structure of the algorithm is shown in

Figure 4.8. The host vehicle filter estimates the host vehicle's longitudinal speed and yaw rate which take roll of inputs. And based on IMM approach, the organized measurement set is filtered and the best estimates of multi target vehicles are obtained. Each component of the algorithm is presented in following sections. In the part of track management, the raw measurement cloud is validated and associated for each target by using most likely validation law.

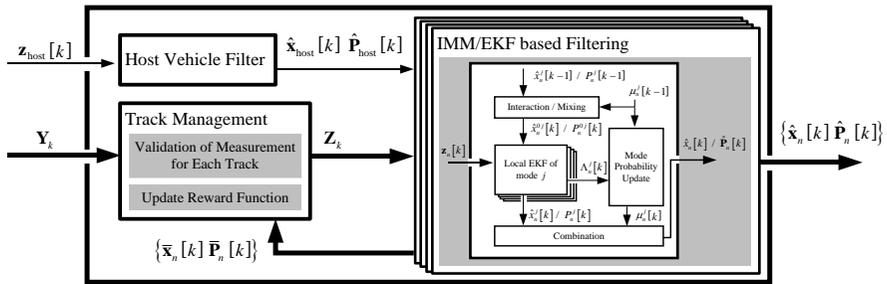


Figure 4.8. Architecture of IMM/EKF multi target state estimation algorithm

4.3.1. Host Vehicle Filter

The Kalman filter is used to estimate host vehicle states such as longitudinal velocity and yaw rate. These values can be estimated from the vehicle sensor signals such as wheel speed, acceleration, and yaw rate of which each time derivative is considered as the Gaussian white noise. The state of the host vehicle filter and its time invariant linear system structures can be defined as follows.

$$\begin{aligned}\dot{\mathbf{x}}_{host} &= \mathbf{A}_{host} \cdot \mathbf{x}_{host} + \mathbf{q}_{host} \\ &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_x & a_x & \gamma \end{bmatrix}^T + \mathbf{q}_{host} \quad (3.29)\end{aligned}$$

$$\mathbf{q}_{host} \sim (\mathbf{0}, \mathbf{Q}_{host})$$

$$\begin{aligned}\mathbf{z}_{host} [k] &= \mathbf{H}_{host} \mathbf{x}_{host} [k] + \mathbf{v}_{host} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x [k] & a_x [k] & \gamma [k] \end{bmatrix}^T + \mathbf{v}_{host} \quad (3.30)\end{aligned} \quad (3.31)$$

$$\mathbf{v}_{host} \sim (\mathbf{0}, \mathbf{V}_{host})$$

The given process model can be discretized as follows:

$$\begin{aligned}\mathbf{x}_{host} [k] &\cong \mathbf{x}_{host} [k-1] + \Delta t \cdot \mathbf{q}_{host} \\ &= \mathbf{F}_{host} \cdot \mathbf{x}_{host} [k-1] + \mathbf{w}_{host} \\ \mathbf{w}_{host} &\sim (\mathbf{0}, \Delta t^2 \cdot \mathbf{Q}_{host}) \\ &\sim (\mathbf{0}, \mathbf{W}_{host})\end{aligned} \quad (3.32)$$

where Δt is the sampling time (0.1s in this study). With above process and measurement model, vehicle states are recursively estimated by using the Kalman filter which is a sequence of time and measurement update steps as following specific equations.

Time update

$$\begin{aligned}\bar{\mathbf{x}}_{host}[k] &= \mathbf{F}_{host} \cdot \hat{\mathbf{x}}_{host}[k-1] \\ \bar{\mathbf{P}}_{host}[k] &= \mathbf{F}_{host} \cdot \hat{\mathbf{P}}_{host}[k-1] \cdot \mathbf{F}_{host}^T + \mathbf{W}_{host}\end{aligned}\quad (3.33)$$

Measurement update

$$\begin{aligned}\mathbf{K}_{host}[k] &= \bar{\mathbf{P}}_{host}[k] \cdot \mathbf{H}_{host}^T \cdot (\mathbf{H}_{host} \cdot \bar{\mathbf{P}}_{host}[k] \cdot \mathbf{H}_{host}^T + \mathbf{V}_{host})^{-1} \\ \hat{\mathbf{P}}_{host}[k] &= (\mathbf{I} - \mathbf{K}_{host}[k] \mathbf{H}_{host}) \cdot \bar{\mathbf{P}}_{host}[k] \\ \hat{\mathbf{x}}_{host}[k] &= \bar{\mathbf{x}}_{host}[k] + \mathbf{K}_{host}[k] \cdot (\mathbf{z}_{host}[k] - \mathbf{H}_{host} \cdot \bar{\mathbf{x}}_{host}[k]) \\ \hat{\mathbf{u}}[k] &= \left[(\hat{\mathbf{x}}_{host}[k])_1 \quad (\hat{\mathbf{x}}_{host}[k])_2 \right]^T\end{aligned}\quad (3.34)$$

4.3.2. IMM/EKF based Filtering

For the each track, after the validated measurement is received, the multiple EKFs are used as the each mode-matched local filter and combined through the IMM method to obtain a refined state estimate. A complete cycle of the IMM/EKF based filtering algorithm can be summarized as follows. Time index is omitted unless otherwise required for clarity.

4.3.2.1. Interaction ($\forall i, j \in \mathbf{M} / \forall n \in \mathbf{T}$):

Starting with the N_{mode} weights $\mu_n^i[k-1]$, the N_{mode} means $\hat{\mathbf{x}}_n^i[k-1]$ and the N_{mode} associated covariance $\hat{\mathbf{P}}_n^i[k-1]$, compute the mixed initial condition for filter matched to mode state j . Note that, when the modes used have different dimension state vectors, an augmentation with zeros the lower dimension state estimate may lead to a bias toward zero for the state components of the larger state vector. In the previous work, a simple procedure

to avoid this “biasing” is presented, together with a suitable augmentation of the covariance of the smaller state that yields an unbiased and consistent mixing [Yuan'12].

Predicted mode probability:

$$\begin{aligned}\bar{\mu}_n^j &\triangleq \Pr\{m_n[k] = j | \mathbf{z}_n[k-1]\} \\ &= \sum_i \{\Phi_{ij}[k-1] \mu_n^i[k-1]\}\end{aligned}\quad (3.35)$$

Mixing probability:

$$\begin{aligned}\mu_n^{ij} &\triangleq \Pr\{m_n[k-1] = i | m_n[k] = j, \mathbf{z}_n[k-1]\} \\ &= \frac{\Phi_{ij}[k-1] \mu_n^i[k-1]}{\bar{\mu}_n^j}\end{aligned}\quad (3.36)$$

Mixed condition:

$$\hat{\mathbf{x}}_n^{0j}[k-1] = \sum_i \hat{\mathbf{x}}_n^i[k-1] \mu_n^{ij} \quad (3.37)$$

$$\hat{\mathbf{P}}_n^{0j}[k-1] = \sum_i \left[\hat{\mathbf{P}}_n^i[k-1] + \begin{Bmatrix} (\hat{\mathbf{x}}_n^i[k-1] - \hat{\mathbf{x}}_n^{0j}[k-1]) \\ \times (\hat{\mathbf{x}}_n^i[k-1] - \hat{\mathbf{x}}_n^{0j}[k-1])^T \end{Bmatrix} \right] \mu_n^{ij} \quad (3.38)$$

4.3.2.2. Extended Kalman Filtering:

Each of the N_{mode} pairs weights $\hat{\mathbf{x}}_n^{0j}[k-1]$, $\hat{\mathbf{P}}_n^{0j}[k-1]$ is used as input to an extended Kalman filter matched to mode state j .

Time update:

$$\mathbf{F}_n^j[k-1] = \left. \frac{\partial \mathbf{f}_n^j}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_n^{0j}[k-1], \hat{\mathbf{u}}[k-1]} \quad (3.39)$$

$$\bar{\mathbf{x}}_n^j[k] = \mathbf{f}_n^j(\hat{\mathbf{x}}_n^{0j}[k-1], \hat{\mathbf{u}}[k-1]) \quad (3.40)$$

$$\bar{\mathbf{P}}_n^j[k] = \mathbf{F}_n^j[k-1] \cdot \hat{\mathbf{P}}_n^{0j} \cdot [k-1] \mathbf{F}_n^j[k-1]^T + \mathbf{W}_n^j[k-1] \quad (3.41)$$

Filter gain:

$$\mathbf{H}_n^j[k] = \left. \frac{\partial \mathbf{h}_n^j}{\partial \mathbf{x}} \right|_{\bar{\mathbf{x}}_n^j[k], \hat{\mathbf{u}}[k]} \quad (3.42)$$

$$\mathbf{S}_n^j[k] = \mathbf{H}_n^j[k] \cdot \bar{\mathbf{P}}_n^j[k] \cdot \mathbf{H}_n^j[k]^T + \mathbf{V}_n^j[k] \mathbf{K}_n^j[k] = \bar{\mathbf{P}}_n^j[k] \cdot \mathbf{H}_n^j[k]^T \cdot \mathbf{S}_n^j[k]^{-1} \quad (3.43)$$

Innovation:

$$\mathbf{r}_n^j[k] = \mathbf{z}_n[k] - \mathbf{h}_n^j(\bar{\mathbf{x}}_n^j[k], \hat{\mathbf{u}}[k]) \quad (3.44)$$

Measurement update:

$$\hat{\mathbf{x}}_n^j[k] = \bar{\mathbf{x}}_n^j[k] + \mathbf{K}_n^j[k] \{\mathbf{r}_n^j[k]\} \quad (3.45)$$

$$\hat{\mathbf{P}}_n^j[k] = (\mathbf{I} - \mathbf{K}_n^j[k] \cdot \mathbf{H}_n^j[k]) \bar{\mathbf{P}}_n^j[k] \quad (3.46)$$

4.3.2.3. Mode Probability Update:

The N_{mode} weights $\bar{\mu}_n^i$ are updated from the innovations of the N_{mode} extended Kalman filters.

Likelihood function:

$$\Lambda_n^j = |2\pi \mathbf{S}_n^j|^{-1/2} \cdot \exp \left[-\frac{1}{2} \cdot (\mathbf{r}_n^j[k])^T \cdot (\mathbf{S}_n^j)^{-1} \cdot (\mathbf{r}_n^j[k]) \right] \quad (3.47)$$

Mode probability:

$$\mu_n^j = \frac{\bar{\mu}_n^j \cdot \Lambda_n^j}{\sum_i \bar{\mu}_n^i \cdot \Lambda_n^i} \quad (3.48)$$

Combination:

for output purpose only, $\hat{\mathbf{x}}_n[k]$ and $\hat{\mathbf{P}}_n[k]$ are computed according to

Combined condition:

$$\hat{\mathbf{x}}_n[k] = \sum_j \hat{\mathbf{x}}_n^j[k] \mu_n^j \quad (3.49)$$

$$\hat{\mathbf{P}}_n[k] = \sum_j \left\{ \hat{\mathbf{P}}_n^j[k] + (\hat{\mathbf{x}}_n^j[k] - \hat{\mathbf{x}}_n[k]) (\hat{\mathbf{x}}_n^j[k] - \hat{\mathbf{x}}_n[k])^T \right\} \mu_n^j \quad (3.50)$$

4.3.3. Track Management

In track management, there are three processes required during each time step: track update, track deletion and track creation. In this study, each track has a rewarding counter that is rewarded when the track is updated with validated measurement and decreased when the measurement is missed. In the part of track update, the measurements are associated with the existing tracks. Following (28), the most likely candidate measurement for target n at time k can be defined as follows:

$$\mathbf{z}_n^{candi}[k] = \min_{\mathbf{y} \in \mathbf{Y}_k} \left[(\mathbf{y} - \hat{\mathbf{z}}_n[k])^T \cdot (\mathbf{S}_n[k])^{-1} \cdot (\mathbf{y} - \hat{\mathbf{z}}_n[k]) \right] \quad (3.51)$$

where

$$\hat{\mathbf{z}}_n^j[k] = \mathbf{h}_n^j(\bar{\mathbf{x}}_n^j[k], \hat{\mathbf{u}}[k])$$

$$\hat{\mathbf{z}}_n[k] = \sum_j \hat{\mathbf{z}}_n^j[k] \bar{\mu}_n^j$$

$$\mathbf{S}_n[k] = \sum_j \left\{ \mathbf{S}_n^j[k] + (\bar{z}_n[k] - \bar{z}_n^j[k])(\bar{z}_n[k] - \bar{z}_n^j[k])^T \right\} \bar{\mu}_n^j$$

The candidate measurement $\mathbf{z}_n^{candi}[k]$ is validated as $\mathbf{z}_n[k]$ if and only if

$$(\mathbf{z}_n^{candi}[k] - \hat{\mathbf{z}}_n[k])^T \cdot (\mathbf{S}_n[k])^{-1} \cdot (\mathbf{z}_n^{candi}[k] - \hat{\mathbf{z}}_n[k]) \leq r_{th} \quad (3.52)$$

where r_{th} is the gate threshold corresponding to the gate probability P_G . Note that all targets share a common threshold. If the track has no validated measurement, the rewarding counter is decreased and the time-prediction is performed as the current estimate (which means a zero-gain EKF). If a rewarding counter drops below a certain threshold, the track is deleted. And newly arriving measurements generate new tracks. The initial state of new track is decided by a model selection strategy. And the new track is assumed to have the same speed with the host vehicle, zero yaw-rate, zero acceleration, and zero yaw acceleration. Additionally, some large value of error covariance is given for the newly generated target.

4.4. Vehicle Tests based Performance Evaluation

The developed IMM/EKF based multi-target tracking algorithm is implemented and verified via vehicle tests in real road. The test scenario is an overtaken situation by a target vehicle to validate all-around target states estimation performance. To investigate the performance enhancement, comparison with model-switching algorithm which is a simple approach to handle the multi-model problem has been conducted. Additionally, the experimental results dealing with multi-target scenarios on the real road are presented to explore the multi-target tracking performance of the proposed algorithm.

As aforementioned, in target tracking research area, considerable IMM approaches are limited to various modes of target motion. And the main novelties of this paper are that process model and measurement model have been elaborated, and multiple measurement models have been derived. Therefore, twelve measurement model set has been implemented and investigated for the proposed IMM/EKF algorithm (with single standard process model).

4.4.1. Configuration of Vehicle Tests

Figure 4.9 shows a host vehicle and a target vehicle which are used in this study. The host vehicle is equipped with a front-radar, two rear-side-radars, wheel speed sensors, and a 6DOF sensor, etc. And a RT-Range is used for an accurate measurement of the relative motion between the host and target vehicle.

The RT-Range is a precision tool for evaluating and verifying the accuracy of radar and tracking algorithm. Since this paper focuses on the target tracking algorithm, additional details about the test vehicles' system configurations have been omitted.

The test scenario is an overtaken situation by a target vehicle as shown in Figure 4.10. The host vehicle (denoted by green vehicle) drives in a straight lane with an initial speed of 50 km/h and a slight deceleration and the target vehicle (denoted by red vehicle) started its overtaking at 7s and finished at 16s while driving with an initial speed of 55kph with a slight acceleration. Typical characteristics of the automotive radar in the given situation are shown in Figure 4.11. We can see that, from 0-12s, the target vehicle is detected by rear radar and left side of the target is in effect during that period. A period from 12-14s is blind area where there is no measurement for the target vehicle by any radar. From 14-20s, the target vehicle is detected by front radar and rear side of the target is in effect during that period.

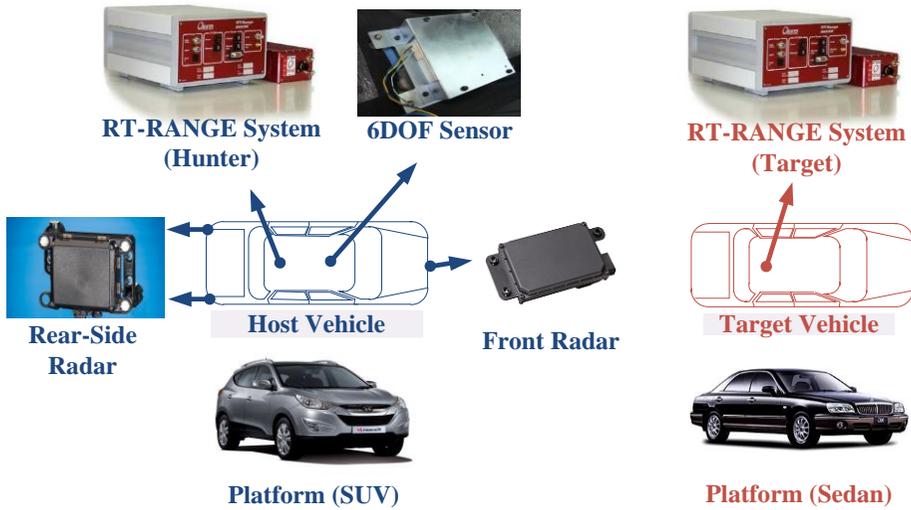


Figure 4.9. Configurations of the host vehicle and the target vehicle

4.4.2. Implementation and Evaluation

As aforementioned, in target tracking research area, considerable IMM approaches are limited to various modes of target motion. And the main novelties of this paper are that process model and measurement model have been elaborated, and multiple measurement models have been derived. Therefore, twelve measurement model set has been implemented and investigated for the proposed IMM/EKF algorithm (with single standard process model). Target states estimation results in the given situation are summarized in Figure 4.12, Figure 4.13, and Figure 4.14.

Figure 4.12 gives three case examples of mode probability update and target state estimation. The host vehicle is depicted by green vehicle, the actual target vehicle by red vehicle, the estimated target vehicle by gray vehicle and the associated measurement by red square. As can be seen in the figure, in all cases,

mode estimation results are quite reasonable and estimated target positions maintain good performance compared to the actual values even in no measurement situation (case.2).

Figure 4.13 shows the results of mode probability adaptation along the time axis. The update results represent the actual mode transition which can be inferred from Figure 3.12. From 0-10 sec, which is a period that the target vehicle is detected by rear radar and front-left edge of the target is mainly in effect, we can see that the probability of mode.6 ranked highest. In a period from 10-12 sec which is blind area, although there are no measurements, the most likely mode has been changed from mode.6 to mode.2 along the target vehicle's left side by the interaction/mixing step of the algorithm. This transition in the blind area is quite reasonable in the view of changes of the target aspect. And finally, from 12-20 sec, which is the period that the target vehicle is detected by front radar, mode.1 (center of rear side), mode.12 (rear-right edge), and mode.2 (rear-left edge) are evaluated as the highly likely modes during the period. Moreover, as the target vehicle becomes the in-lane preceding vehicle, we can see that mode.1 gets stronger mode probability which denoted by red-solid line.

Figure 4.14 presents the estimation results for the target states and their error distribution. From the figure, we can see that the overall trends of each estimated state are similar to actual values and their error distributions can be well approximated by normal distribution. The statistics of estimation error of IMM/EKF based filtering algorithm are given in Table 1.

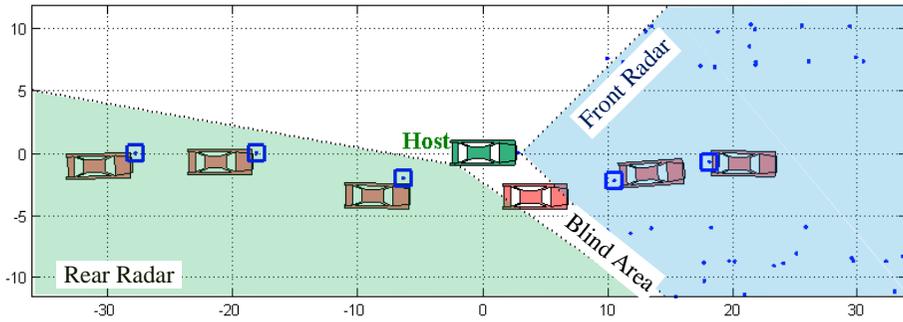


Figure 4.10. Test scenario: overtaking target vehicle

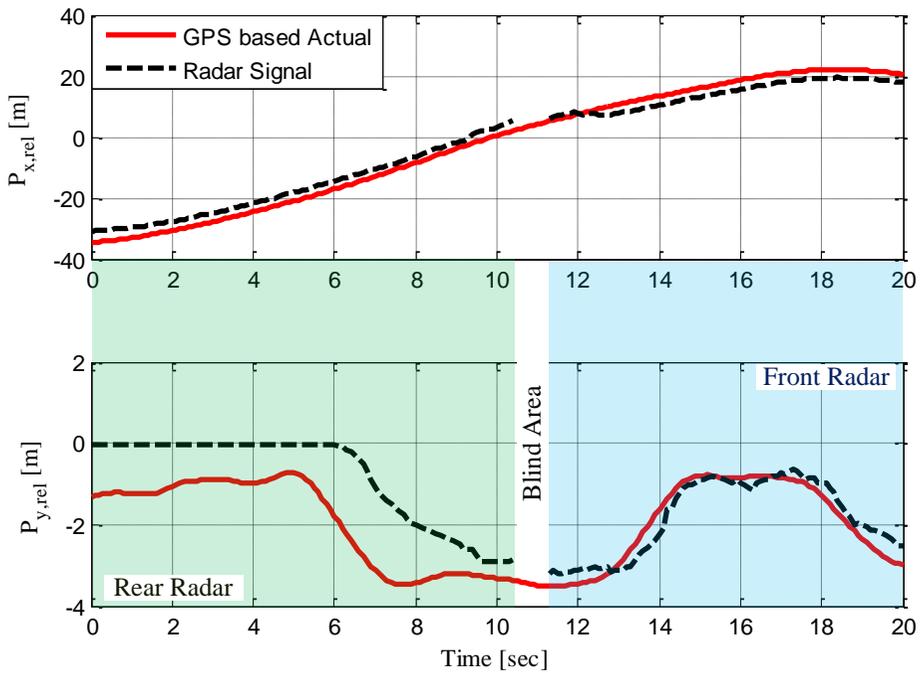


Figure 4.11. Test data: actual position (from RT-Range) and the associated radar signals of target vehicle

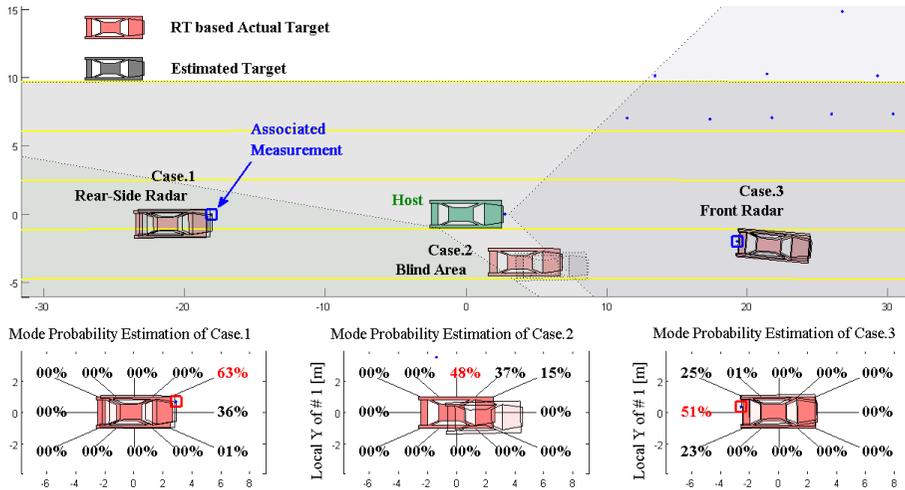


Figure 4.12. Three case examples of mode probability update

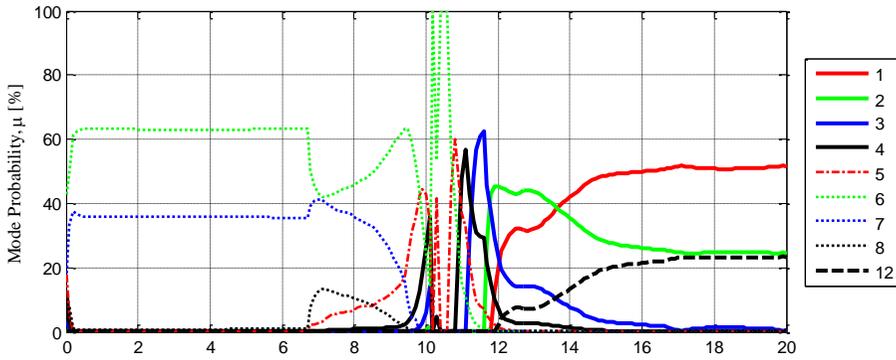


Figure 4.13. Mode probability update results

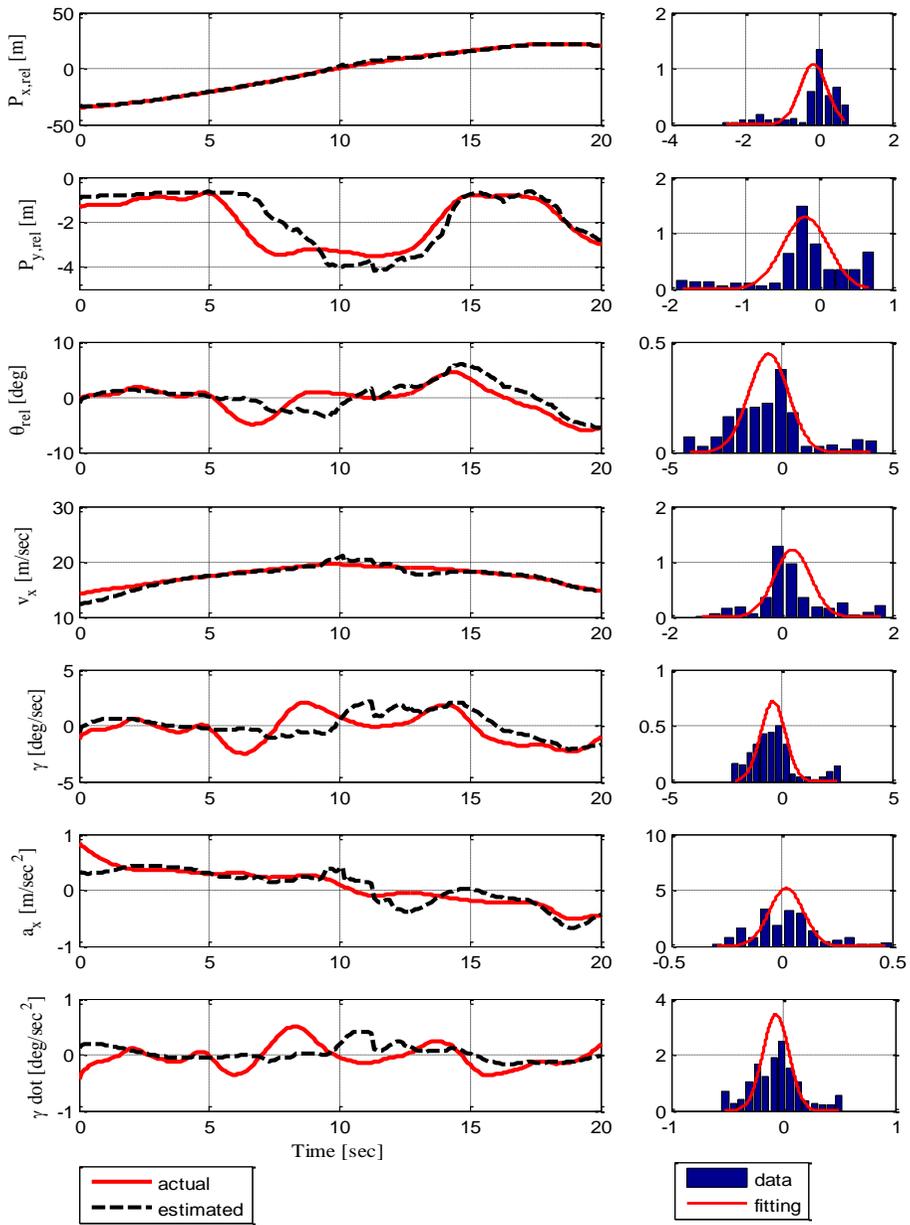


Figure 4.14. Estimated target states and the distributions of the estimation errors

4.4.3. Comparison with Model-switching/EKF

To examine the performance enhancement of the proposed IMM/EKF algorithm, performance comparison with a model-switching/EKF algorithm has been conducted. Model-switching/EKF is the simple approach to handle the multi-model problem by selecting one model from the multi-model set according to some predefined model selection criterion. Architecture and model selection strategy of model-switching/EKF algorithm is summarized in Figure 4.15. The strategy of model selection is grounded on the assumption that radar gives the information of the closest point on the boundary of the target vehicle. As an example, if the measurement is detected from the front-right side, the second measurement model is selected as the model in effect and implemented to the algorithm.

Statistical analysis of estimation errors of model-switching/EKF has been conducted following the same procedure under the same situation of 4.4.2. The statistics are given in Table 1. Over all, the proposed algorithm performed considerably better in all state estimations than the model-switching algorithm. As the standard deviation is the accuracy specifications of each algorithm, the proposed algorithm is approximately three times more accurate than the model-switching/EKF.

Finally, computing complexity of these two approaches have been compared. An average consumption time of IMM/EKF algorithm is about 4.9806ms and mode-switching/EKF is about 1.6787ms. When considering that IMM uses twelve filters in parallel, this three times more complexity is relatively good performance. Moreover, although IMM based algorithm has high complexity

compared to mode-switching method, as the sampling rate of the overall algorithm is 100ms, computing complexity of IMM/EKF algorithm is not in question.

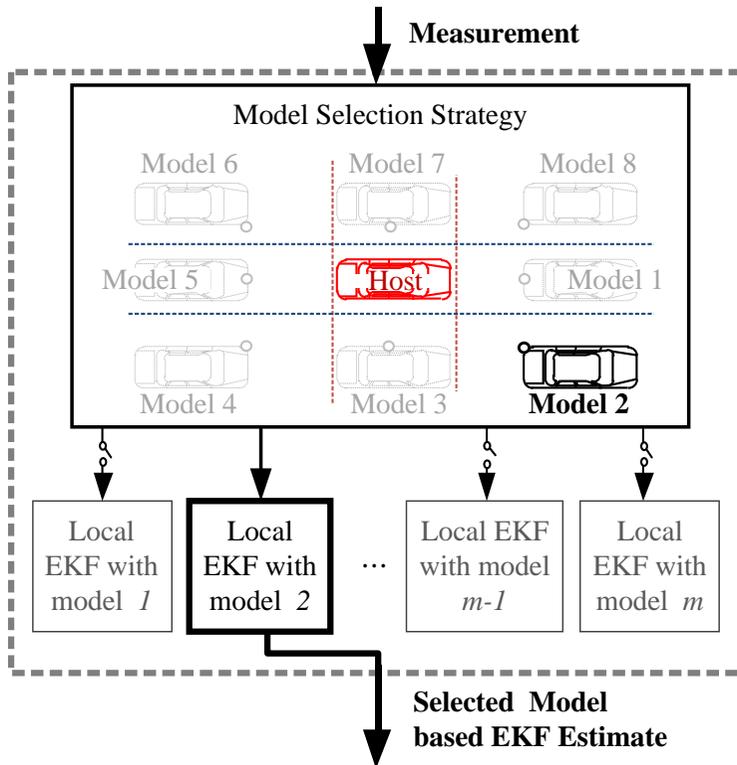


Figure 4.15. Architecture of model-switching/EKF Algorithm

Table 1. Statistics of Estimation Errors (N=400)

	RMS		Model-switching to IMM
	IMM	Model-switching	
$p_{n,x}$ (m)	0.7484	3.8412	5.1326
$p_{n,y}$ (m)	0.6430	1.9908	3.0961
θ_n (deg)	1.8787	5.2473	2.7931
$v_{n,x}$ (m/s)	0.6842	4.0688	5.9465
γ_n (deg/s)	1.1767	5.2730	4.4813
$a_{n,x}$ (m/s ²)	0.1549	0.7476	4.8278
$\dot{\gamma}_n$ (deg/s ²)	0.2403	2.9483	12.2687

4.4.4. Experimental Results with Multi-target Situation

In this section, the experimental results dealing with multi-target scenarios are presented. Real-road driving data were collected from a highway and an urban area. The camera is used to analyze the driving situation and to acquire the lane information. Especially, one test set for validation is extracted from the collected real-driving data to explore the multi-target tracking performance and target state estimation performance. The selected test set is a complex driving scene where the targets may enter and leave the field of view, showing the capability for birth and death of targets.

Figure 4.16 shows a frontal image of the vision camera, and the multi-target state estimation results in complex driving situation. The radar measurements are depicted by blue circles with inner star. And based on the lane information, a region of interest is defined and the outside is represented by red area. As can be seen in the figure, it is evident that the proposed algorithm manages to follow the correct true targets. A true target is most generally defined to be an object that will persist in the tracking volume for at least several scans. False alarms (or false targets) refer to erroneous detection events (such as those caused by random noise or clutter) that do not persist over several scan. In the figure, we can show that those events (caused by road structures such as guardrail in red area) cannot generate new track. At $t=8\text{sec}$, the proposed algorithm tracks 5 targets. And at $t=13\text{sec}$, one target enters the field of view and track#19 is newly generated. And at the same time track#10 starts leaving the sensing range. At $t=15.5\text{sec}$, track#10 leaves the sensing range completely, and zero-gain EKF is

applied. The track#14 starts its lane changing and cuts-in front of the host vehicle. Due to this lane changing, a measurement of hidden vehicle is newly arrived. And at $t=20\text{sec}$, track#12 is newly generated from this measurement while tracks other existing 4 targets. In summary, these results indicate that GNN based track management can be utilized with satisfactory. The GNN based track management approach, which considers the single most likely hypothesis for track update and new track initiation, works well because of widely spaced targets, relatively accurate measurements in the given experiment situation. However, the problem of closely-spaced unresolved targets can occur and this can be solved by MHT which forms multiple hypotheses and manages these hypotheses [Blackman'04].

Figure 4.17 presents the target state estimation results for track#14. Each graph shows each state variables, rewarding function, and mode probability update result as a function of time. As can be inferred from the vision image, the vehicle of track#14 changes the lane and cuts into the host vehicle's lane. From the figure, we can see that the overall trends of each estimated state can represent actual target motion with satisfactorily while the rewarding function of track#14 maintains the maximum value (100 in this paper). In the case of mode probability update results, from 0-18 sec, mode.12 (rear-right edge) is evaluated as the most highly likely mode. This can be explained by the fact that the target vehicle was driving in the left lane during that period. And a transition to mode.1 (center of rear side) from mode.12 has occurred in a period of lane changing. Finally, the probability of mode.1 ranked highest after the target vehicle becomes the in-lane preceding vehicle. Overall, the mode evaluation

results are quite reasonable, in the view of target aspect change.

Figure 4.18 shows the results for track#10. Track#10 drives with lower speed than the host vehicle in the right lane and leaves the field of view at about 13sec. From the figure, each state is estimated reasonably so that can represent actual motion. The rewarding function of track#10 shows a typical track management process. It keeps increasing until 10sec, maintains the maximum value from 10sec to 13sec, and drops after the target disappears from the scope of the radar. And in the case of mode probability update results, from 0-11sec, mode. 2 (rear-left edge) is evaluated as the most highly likely mode. This can be explained by the fact that the target vehicle was driving in the right lane during that period. And a transition to mode.5 (front of the left side) has occurred at 11sec. This drastic mode change can be explained from the change of radar measurement at that moment. In the figure, the radar measurement is depicted as a red circle line and we can show that longitudinal measurement changes drastically at 11sec. This reflects the fact that the mode adaptation of the proposed IMM algorithm is quite reasonable.

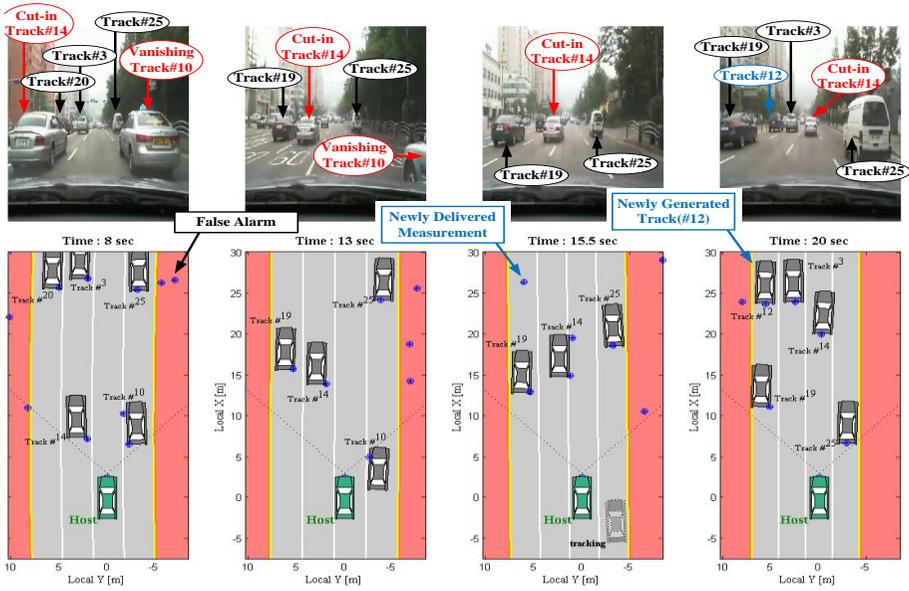


Figure 4.16. Multi-target state estimation in complex driving situation

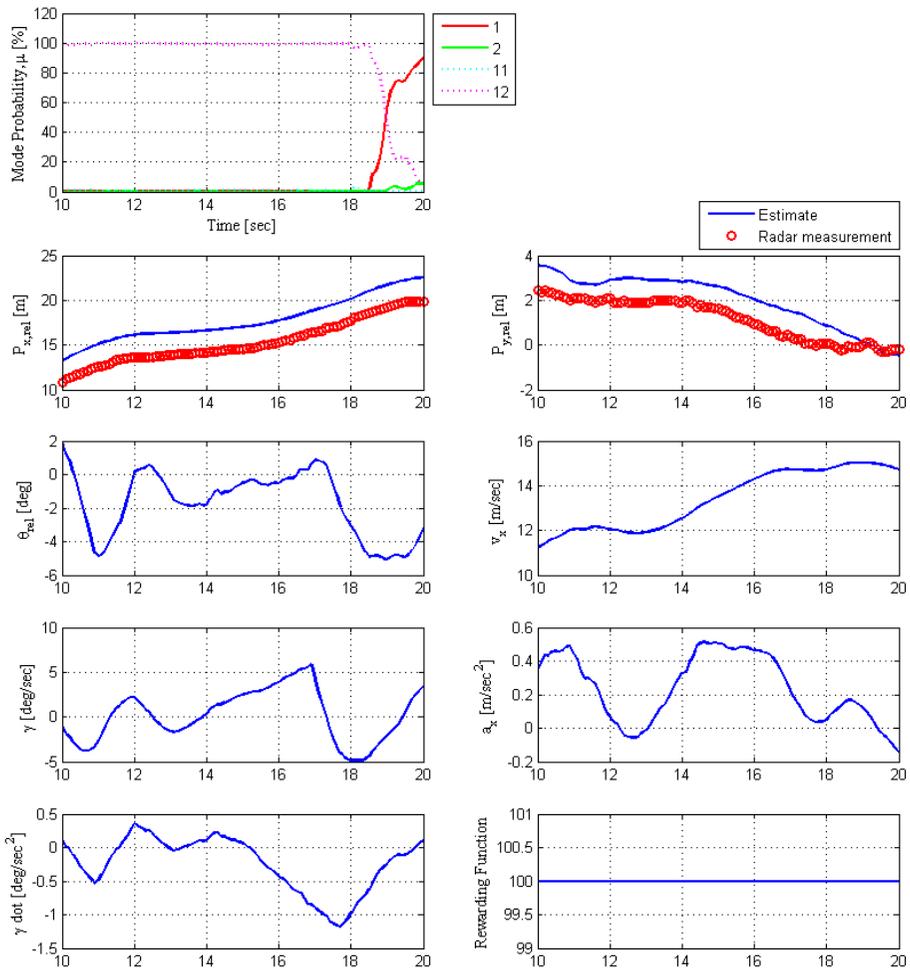


Figure 4.17. Target state estimation results for Track 14 (a target cutting in front of the host vehicle)

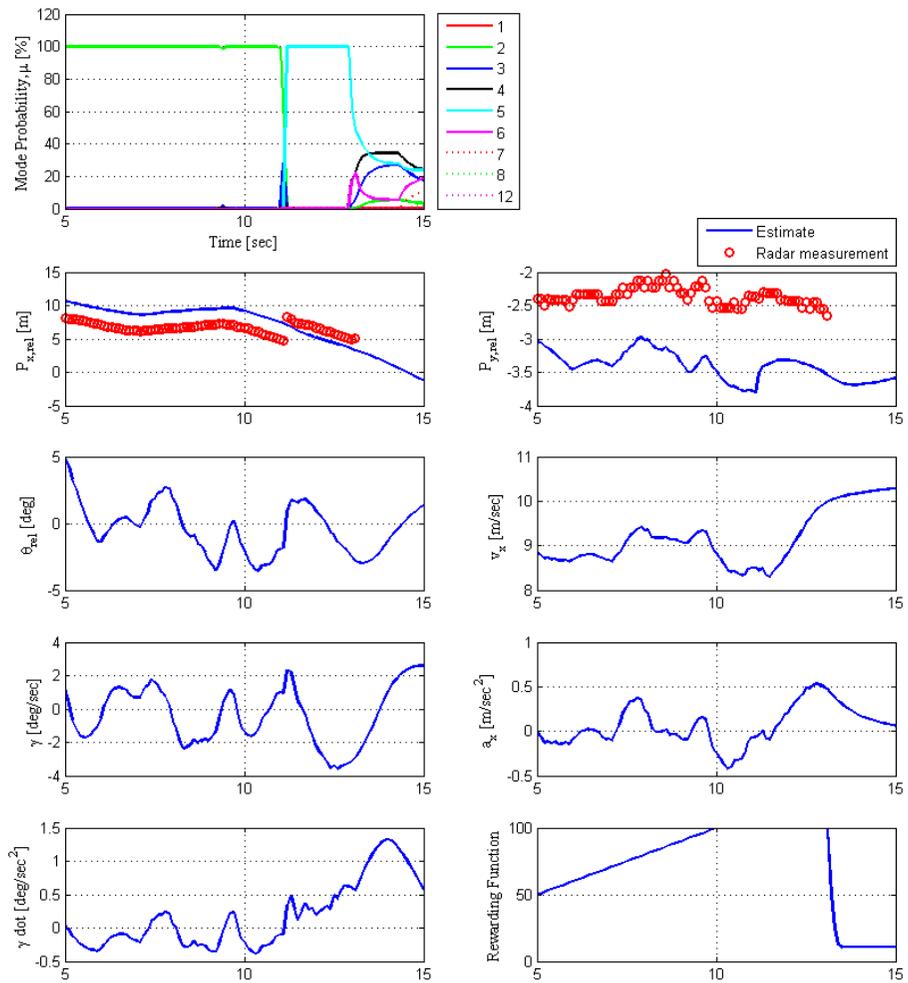


Figure 4.18. Target state estimation results for Track 10 (a target vanishing to rear-right-blind area)

Chapter 5 . Safety Driving Envelope Decision and Motion Optimization

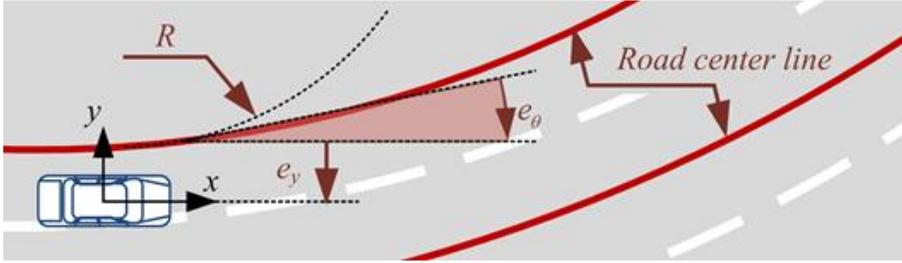
Our approach to motion planning was to separate it into two distinct tasks (cf. Figure 2.1). Firstly, it is responsible for translating perceived moving objects and their prediction, static obstacles from the laser scanner, information from the digital map and vision, etc. into geometric constraints. This top level module is called as safety driving envelope decision. Subsequently, the motion optimization module computes the desired path and desired velocity of the ego-vehicle as a function of time. This trajectory is obtained by solving a geometric problem that has been posed as a convex optimization problem with linear equality/inequality constraints. These two components are addressed in the following subsections.

5.1. Multi-traffic Prediction

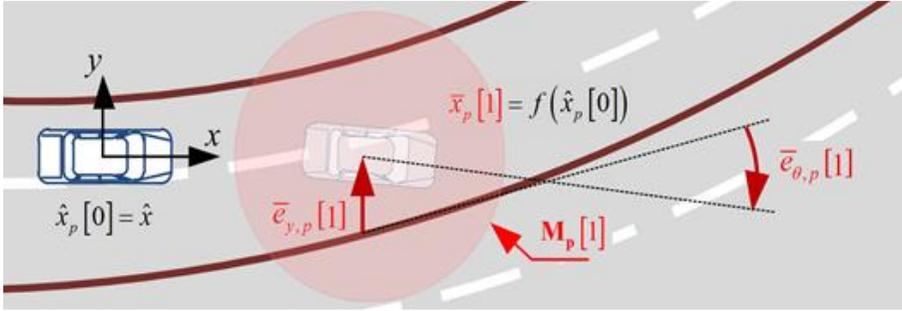
The possible behaviors of surrounding vehicles are predicted considered in determining the safe drivable area decision. In predicting reasonable ranges of the future states of surrounding vehicles, driving data is collected on test track and real roads to analyze the probabilistic movement characteristics of the vehicle.

The future vehicle states are described as a stochastic multi-stage process via Taylor Methods. At every time step of the prediction time horizon, a desired yaw rate is determined by the lane-keeping driver model. Suppose that the measurement noise has a normal distribution with zero mean. Hence, the predictive measurement is linearly related to the time-updated predictive state. Then, the maximum likelihood prediction of the future state is calculated by the Kalman filtering equations.

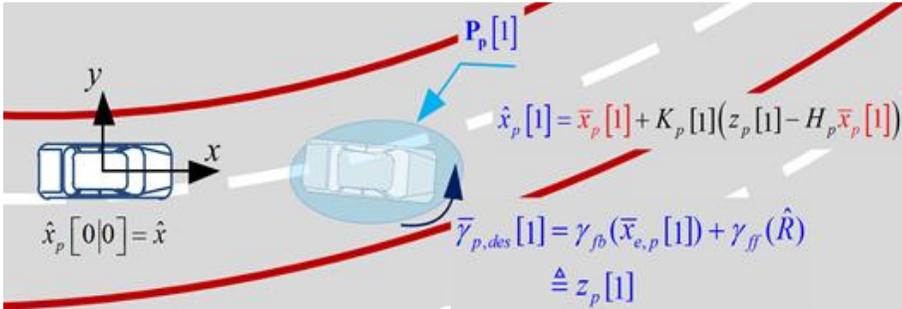
That is, behavior restrictions due to road geometry and driver characteristics such as reasonable acceleration input range are replaced with multi-stage of Kalman filtering problem by the proposed prediction algorithm. As an example, a prediction procedure at the 1st future time step is depicted in Figure 5.1-(b) and Figure 5.1-(c). The overall prediction results for 1 s, 2 s, and 3 s of prediction time at the instant of lane changing are given in Figure 5.1-(d).



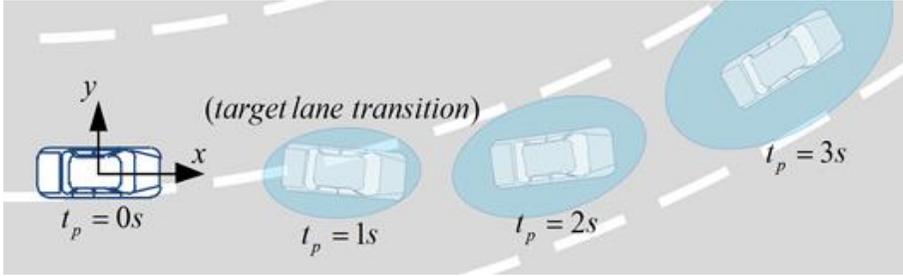
(a) The relationship between the host vehicle and the road center line of each lane



(b) The time-update-predicted host vehicle states and the relative error states with respect to road geometry defined on current body coordinate



(c) The measurement-update-predicted host vehicle states where the predicted desired yaw-rate to keep the lane is defined as virtual measurement



(d) Prediction results for 1s, 2s, and 3s of prediction time at a lane-changing instant

Figure 5.1. Procedure and Concept of Traffic Behavior Prediction

5.1.1. Lane Keeping Behavior Model

The objective of the lane keeping behavior model is to develop a yaw control system for human-like lane keeping. To achieve this goal, it is useful to utilize a dynamic model in which the state variables are in terms of position and orientation error with respect to the road. The error state is defined in terms of fixed coordinates under the assumption of traveling with constant longitudinal velocity on a road of constant radius. The error state is defined in inertial fixed coordinates rather than body-fixed moving coordinates. By using the definition of the road geometry in section A.2, the position error can be defined as:

$$e_y = p_y - \{a_2 \cdot p_x^2 + a_1 \cdot p_x + a_0 + w_{road} \cdot N(\text{current lane})\} \quad (5.1)$$

where p_x is the vehicle's longitudinal position, p_y is the vehicle's lateral position, w_{road} is the width of the road lane, and N is the adjusting integer to represent the current lane. For example, if the vehicle changes lanes to the left, N has the value of minus one. Under small-slip-angle assumptions, the time

derivative of the position error can be defined as:

$$\begin{aligned}\dot{e}_y &= \frac{d}{dt}(p_y) - (2a_2 \cdot p_x + a_1) \cdot \frac{d}{dt}(p_x) \\ &\cong v \sin \theta - (2a_2 \cdot p_x + a_1) \cdot v \cos \theta\end{aligned}\quad (5.2)$$

where v is the longitudinal velocity, and θ is the orientation. The orientation error and its time derivative can be defined as:

$$e_\theta = \theta - \tan^{-1}(2a_2 \cdot p_x + a_1) \quad (5.3)$$

$$\begin{aligned}\dot{e}_\theta &= \dot{\theta} - \frac{d}{dt}\{\tan^{-1}(2a_2 \cdot p_x + a_1)\} \\ &= \gamma - \frac{1}{1 + (2a_2 \cdot p_x + a_1)^2} \cdot \frac{d}{dt}(2a_2 \cdot p_x + a_1) \\ &= \gamma - \frac{1}{1 + (2a_2 \cdot p_x + a_1)^2} (2a_2) \cdot \frac{d}{dt}(p_x) \\ &= \gamma - \frac{2a_2}{1 + (2a_2 \cdot p_x + a_1)^2} \cdot (v \cos \theta)\end{aligned}\quad (5.4)$$

Under assumptions of small road slope and small error, the time derivatives of the error states can be simplified as follows:

$$\begin{aligned}\dot{e}_y &= v \sin \theta - (2a_2 \cdot p_x + a_1) \cdot v \cos \theta \\ &= v \cos \theta \cdot \{\tan \theta - (2a_2 \cdot p_x + a_1)\} \\ &\cong v \cos \theta \cdot e_\theta\end{aligned}\quad (5.5)$$

$$\begin{aligned}\dot{e}_\theta &= \gamma - \frac{2a_2}{1 + (2a_2 \cdot p_x + a_1)^2} \cdot (v \cos \theta) \\ &\cong \gamma - 2a_2 \cdot (v \cos \theta)\end{aligned}\quad (5.6)$$

If the yaw rate dynamics can be approximated as a 1st-order system which has the desired yaw rate as the system input, the state-space model of the tracking error variables is given by following equation:

$$\begin{aligned}
\dot{\mathbf{x}}_e &= \mathbf{F}_e \cdot \mathbf{x}_e + \mathbf{G}_e \cdot \gamma_{des} + \mathbf{G}_w \cdot a_2 \\
&= \begin{bmatrix} 0 & v \cos \theta & 0 \\ 0 & 0 & 1 \\ 0 & 0 & f \end{bmatrix} \cdot \begin{bmatrix} e_y \\ e_\theta \\ \gamma \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -f \end{bmatrix} \gamma_{des} + \begin{bmatrix} 0 \\ -2v \cos \theta \\ 0 \end{bmatrix} a_2
\end{aligned} \tag{5.7}$$

We can see that the first and second rows of the equation describe the road geometry coefficient process model under the assumption of fixed road curvature. Assume that the human drivers determine the desired yaw rate γ_{des} by a state feedback and a feed-forward term that attempts to compensate for the road curvature as follows:

$$\begin{aligned}
\gamma_{des} &= -\mathbf{C} \cdot \mathbf{x}_e + \gamma_{ff} \\
&= -[c_1 \quad c_2 \quad c_3] \mathbf{x}_e + \gamma_{ff}
\end{aligned} \tag{5.8}$$

From a steady-state analysis, we can see that the all error states can be made zero by appropriate choices of feedback gains and the feed-forward term. For example, the feedback gain can be determined by pole placement which is simulating the human driver's behavior characteristics. Then, the feed-forward term can be calculated directly from the above feedback gain and estimated road curvature:

$$\begin{aligned}
\gamma_{des} &= -[c_1 \quad c_2 \quad c_3] \mathbf{x}_e + \gamma_{ff} \\
&= -[c_1 \quad c_2 \quad c_3] \mathbf{x}_e + 2a_2 v \cos \theta (c_3 + 1)
\end{aligned} \tag{5.9}$$

5.1.2. Vehicle Predictor

In the prediction of the vehicle's future states, the available information includes the vehicle's current motion estimates, the road geometry estimates described in the current body coordinates, the lane keeping behavior model, and

the error covariance of each. As mentioned, the lane keeping behavior model-based desired yaw rate is used as the virtual measurement. The future vehicle system can be described as a stochastic multi-stage process via Taylor Methods:

$$\begin{aligned}\mathbf{x}_p[i+1] &= \mathbf{f}_p(\mathbf{x}_p[i]) + \mathbf{w}_p[i], \quad i=0, \dots, N_p-1 \\ &= [f_{1,p} \quad f_{2,p} \quad f_{3,p} \quad f_{4,p} \quad f_{5,p} \quad f_{6,p} \quad f_{7,p}] + \mathbf{w}_p[i]\end{aligned}\quad (5.10)$$

where

$$\begin{aligned}x_p &= [p_{x,p} \quad p_{y,p} \quad \theta_p \quad v_p \quad \gamma_p \quad a_p \quad \dot{\gamma}_p]^T \\ f_{1,p} &= p_{x,p} + (v_p \cos \theta_p) \Delta t + (a_p \cos \theta_p + \gamma_p v_p \sin \theta_p) \frac{\Delta t^2}{2} \\ f_{2,p} &= p_{y,p} + (v_p \sin \theta_p) \Delta t + (a_p \sin \theta_p + \gamma_p v_p \cos \theta_p) \frac{\Delta t^2}{2} \\ f_{3,p} &= \theta_p + (\gamma_p) \Delta t + (\dot{\gamma}_p) \frac{\Delta t^2}{2} \\ f_{4,p} &= v_p + (a_p) \Delta t + (-k_a a_p) \frac{\Delta t^2}{2} \\ f_{5,p} &= \gamma_p + (\dot{\gamma}_p) \Delta t + (-k_{\dot{\gamma}} \dot{\gamma}_p) \frac{\Delta t^2}{2} \\ f_{6,p} &= a_p + (-k_a a_p) \Delta t + (k_a^2 a_p) \frac{\Delta t^2}{2} \\ f_{7,p} &= \dot{\gamma}_p + (-k_{\dot{\gamma}} \dot{\gamma}_p) \Delta t + (k_{\dot{\gamma}}^2 \dot{\gamma}_p) \frac{\Delta t^2}{2}\end{aligned}$$

$$\mathbf{w}_p[i] \sim (0, W_p[i])$$

$$\mathbf{W}_p[i] = \left(\mathbf{B}_p \Delta t + \mathbf{F}_p[i] \mathbf{B}_p \frac{\Delta t^2}{2} \right) \mathbf{Q}_p \left(\mathbf{B}_p \Delta t + \mathbf{F}_p[i] \mathbf{B}_p \frac{\Delta t^2}{2} \right)^T$$

$$\mathbf{F}_p[i] = \left. \frac{\partial \mathbf{f}_p}{\partial \mathbf{x}_p} \right|_{\mathbf{x}_p = \hat{\mathbf{x}}_p[i]} \quad \mathbf{B}_p = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^T$$

where N_p is the length of the pre-defined prediction time horizon, and the subscript ‘p’ is used to denote ‘predictive’. The longitudinal acceleration and yaw acceleration are assumed to decay to zero with corresponding time constants of k_a and $k_{\dot{\gamma}}$. At every time step of the prediction time horizon, a desired yaw rate is determined by the path following model. Suppose that the measurement noise has a normal distribution with zero mean. Hence, the predictive measurement is linearly related to the time-updated predictive states as follows:

$$\begin{aligned}
\mathbf{z}_p[i] &= \mathbf{H}_p \cdot \mathbf{x}_p[i] + \mathbf{v}_p[i], & i = 0, \dots, N_p \\
&= [0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0] \cdot \mathbf{x}_p[i] + \mathbf{v}_p[i] \\
&= \bar{\gamma}_{des,p}[i] \\
&= -\mathbf{C} \cdot \bar{\mathbf{x}}_{e,p}[i] + \gamma_{ff,p}[i]
\end{aligned} \tag{5.11}$$

where

$$\begin{aligned}
\bar{\mathbf{x}}_{e,p}[i] &= \mathbf{f}_e(\bar{\mathbf{x}}_p[i], \hat{\mathbf{x}}_r[0]) \\
&= [\bar{e}_y[i] \ \bar{e}_\theta[i] \ \bar{\gamma}_p[i]]^T \\
&= \begin{bmatrix} \bar{p}_{y,p}[i] - \{\hat{a}_2 \cdot \bar{p}_{x,p}[i]^2 + \hat{a}_1 \cdot \bar{p}_{x,p}[i] + \hat{a}_0 + \hat{w}_{road} \cdot N[i]\} \\ \bar{\theta}_p[i] - \tan^{-1}(2\hat{a}_2 \cdot \bar{p}_{x,p}[i] + \hat{a}_1) \\ \bar{\gamma}_p[i] \end{bmatrix}
\end{aligned}$$

$$\mathbf{v}_p[i] \sim (0, \mathbf{V}_p[i])$$

$$\mathbf{V}_p[i] = \mathbf{V}_{p,const} + [\bar{e}_y[i] \ \bar{e}_\theta[i]]^T \mathbf{w}_{correct} \begin{bmatrix} \bar{e}_y[i] \\ \bar{e}_\theta[i] \end{bmatrix}$$

where $\mathbf{w}_{correct}$ is the matrix defining the additional covariance of the path-following model-based desired yaw rate. Then, the maximum likelihood

prediction of the future state is given by the following extended Kalman filtering. Because the proposed prediction algorithm is based on a stochastic filtering method, the covariance of the prediction error can be evaluated at each time step. Furthermore, the eigenvalue and eigenvectors of the 2nd leading principal minor of \mathbf{P}_p determine the likelihood ellipse around the predictive position. Using the square root of the eigenvalues as semi-axes measured along the eigenvectors, we can sketch the 39% likelihood ellipse with the center at the most likely predictive position. The size of the 87% likelihood ellipse is two times the size of the 39% ellipse in linear dimensions, and that of the 99% ellipse is three times the size. This analysis is very useful to visualize and compare the prediction performance in view of the accuracy and precision.

Initialization of prediction at kth time step of the ego-vehicle

$$\hat{\mathbf{x}}_p[0] = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \hat{\mathbf{x}}[k] \end{bmatrix} \quad \mathbf{P}_p[0] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{P}[k] \end{bmatrix} \quad (5.12)$$

Initialization of prediction at kth time step of the nth-target-vehicle

$$\hat{\mathbf{x}}_p[0] = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \hat{\mathbf{x}}[k] \end{bmatrix} \quad \mathbf{P}_p[0] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{P}[k] \end{bmatrix} \quad (5.13)$$

Time update (rough prediction)

$$\begin{aligned}
\bar{\mathbf{x}}_p [i] &= \mathbf{f}_p (\hat{\mathbf{x}}_p [i-1]) \\
\mathbf{F}_p [i-1] &= \left[\frac{\partial \mathbf{f}_p}{\partial \mathbf{x}_p} \right]_{\mathbf{x}_p = \hat{\mathbf{x}}_p [i-1]} \\
\mathbf{M}_p [i] &= \mathbf{F}_p [i-1] \cdot \mathbf{P}_p [i-1] \cdot \mathbf{F}_p [i-1]^T + \mathbf{W}_p
\end{aligned} \tag{5.14}$$

Measurement update (refined prediction)

$$\begin{aligned}
\bar{\mathbf{x}}_{e,p} [i] &= \mathbf{f}_e (\bar{\mathbf{x}}_p [i], \hat{\mathbf{x}}_r [0]) \\
\mathbf{z}_p [i] &= -\mathbf{C} \cdot \bar{\mathbf{x}}_{e,p} [i] + \gamma_{ff,p} [i] \\
\mathbf{V}_p [i] &= \mathbf{V}_{p,\text{const}} + \begin{bmatrix} \bar{e}_y [i] & \bar{e}_\theta [i] \end{bmatrix}^T \mathbf{w}_{\text{correct}} \begin{bmatrix} \bar{e}_y [i] \\ \bar{e}_\theta [i] \end{bmatrix} \\
\mathbf{K}_p [i] &= \mathbf{M}_p [i] \mathbf{H}_p^T \cdot (\mathbf{H}_p \mathbf{M}_p [i] \mathbf{H}_p^T + \mathbf{V}_p [i])^{-1} \\
\mathbf{P}_p [i] &= (\mathbf{I} - \mathbf{K}_p [i] \mathbf{H}_p) \cdot \mathbf{M}_p [i] \\
\hat{\mathbf{x}}_p [i] &= \bar{\mathbf{x}}_p [i] + \mathbf{K}_p [i] \cdot (\mathbf{z}_p [i] - \mathbf{H}_p \cdot \bar{\mathbf{x}}_p [i])
\end{aligned} \tag{5.15}$$

5.1.3. Test Data based Implementation and Performance Evaluation

The vehicle tests were conducted to implement the algorithm (and to determine the parameters of the algorithm) and to investigate the performance improvement by the proposed Probabilistic Fusion Model (PFM) through post-processing analysis. The tests were conducted with 100 randomly selected drivers with at least 5 years of driving experience on both urban and highway roads in various weather conditions. With the acquired data, the parameters of the algorithm were calculated by post-analysis and applied to the algorithm. Hence, the parameter extraction is based on this driving data, and the parameters presented in the paper can be said to represent nominal driver.

In addition to the implementation, comparisons with a Fixed Yaw Rate Model (FYRM) and a Lane Keeping Model (LKM) were conducted. The FYRM assumes that the vehicle may maintain the current yaw behavior, and LKM assumes that the current lane position is maintained without consideration of the current vehicle motion.

The parameters such as the three covariance matrixes of the sensor noise, the three

covariance matrixes of the process noise, and the feedback gain of the path-following model should be calculated or estimated. Accurate evaluations of these parameters are necessary to represent the realistic characteristics of the sensors, processes, and driver yaw behavior.

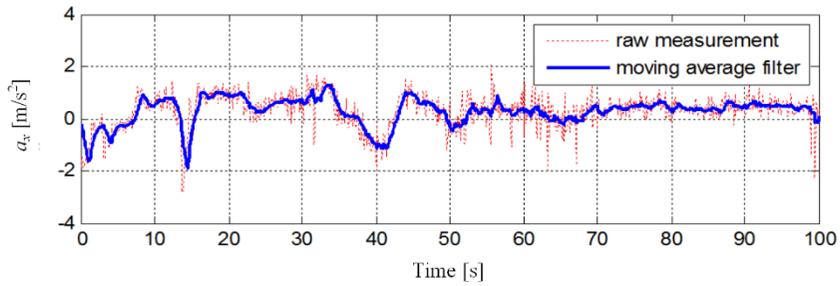
A running average filter is used to generate the quasi-actual reference value. As an example, the procedure to estimate the covariance of sensor noise of longitudinal acceleration is depicted in Figure 5.3-(a) and Figure 5.3-(b). From the figures, we can see that the white Gaussian sensor noise assumption is very realistic. In the case of the process noise, the same procedure is applied, and the covariance calculation is conducted. The process noise is defined as the difference between the predicted value and actual next-step value. As an example, the procedure to estimate the covariance of process noise of the longitudinal acceleration is depicted in Figure 5.3-(d). We can see that the white Gaussian process noise assumption is also quite realistic. The feedback gain of the path-following model is decided by a linear-quadratic-based method that was proposed and validated in previous research [Lee'14, Lee'11, Moon'08]. From the analysis of modeling error of the path-following model (compared to the real yaw rate data), it has been shown that the modeling errors have zero mean and normal distribution, as shown in Figure 5.3-(e).

The FYRM and LKM can be implemented simply by some parameter modifications. For the implementation of the FYRM, the decay rate of yaw acceleration should have the maximum value to represent a fixed yaw rate. In addition to this, the predictive sensor noise covariance V_p should have a large value so that the desired yaw rate has little effect on the prediction (i.e. not to use the road geometry information). In the case of the LKM, very small covariance should be applied for a predictive measurement to make the corrections by the road geometry be dominant throughout the prediction range.

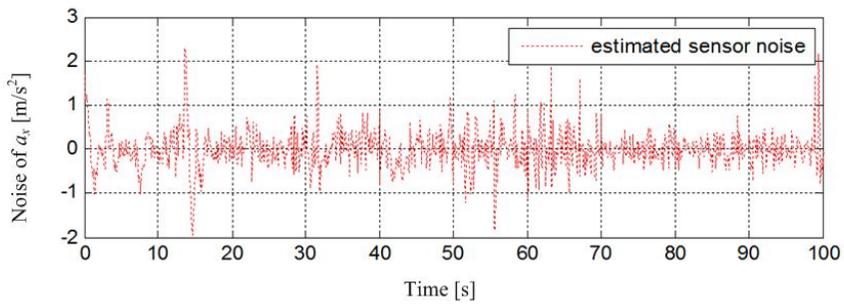
The proposed algorithm is worthy of note because of its flexibility and adaptability. In addition, quantitative measurement and visualization of the prediction performance have been made possible not only for the proposed method but also for existing

conventional methods.

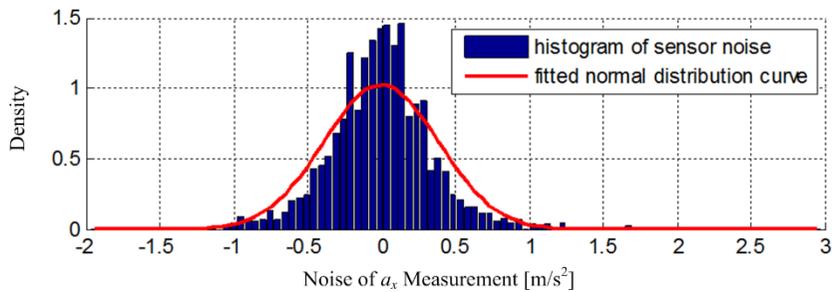
The general parameters of each prediction method are shown in Table 1. Note that the process noise of the exact process model has some small value to represent the discretized error and no-slip assumption error.



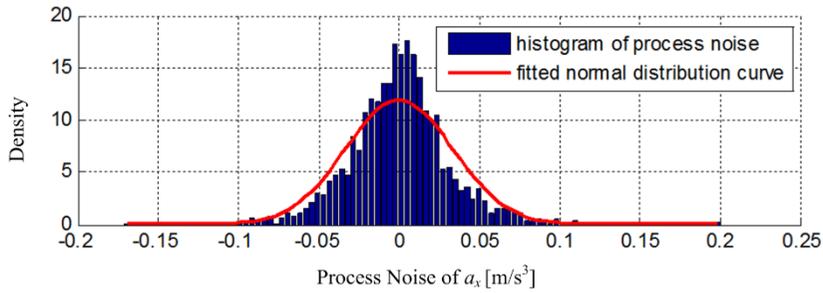
(a) Raw measurement and moving-average-filtered value



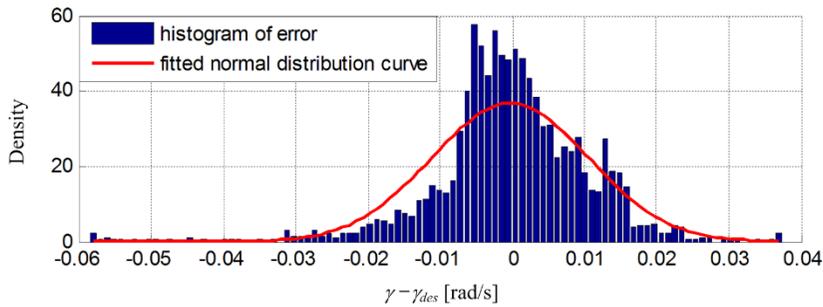
(b) Estimated sensor noise



(c) Histogram and fitted normal distribution of estimated sensor noise



(d) Histogram and fitted normal distribution of estimated process noise based on moving average filter



(e) Histogram and fitted normal distribution of modeling error between actual yaw rate and PFM-based desired yaw rate

Figure 5.2. The procedure to calculate the parameter values: covariance of sensor noise, process noise, and modeling error

Figure 5.3 shows test results of other traffic tracking and its possible motion prediction. The sensing area of long range radar is depicted as green region and mid-range radar is depicted as blue region. Red squares are current observation of radar. Blue-filled-vehicle is the estimated oncoming vehicle and blue-lined-vehicles are prediction results at 0.5s time interval. From the front cam view of Figure 5.3, we can see that all predicted positions are in the center of the lane.

In this way, the prediction performance can be said to be clearly reasonable indirectly.

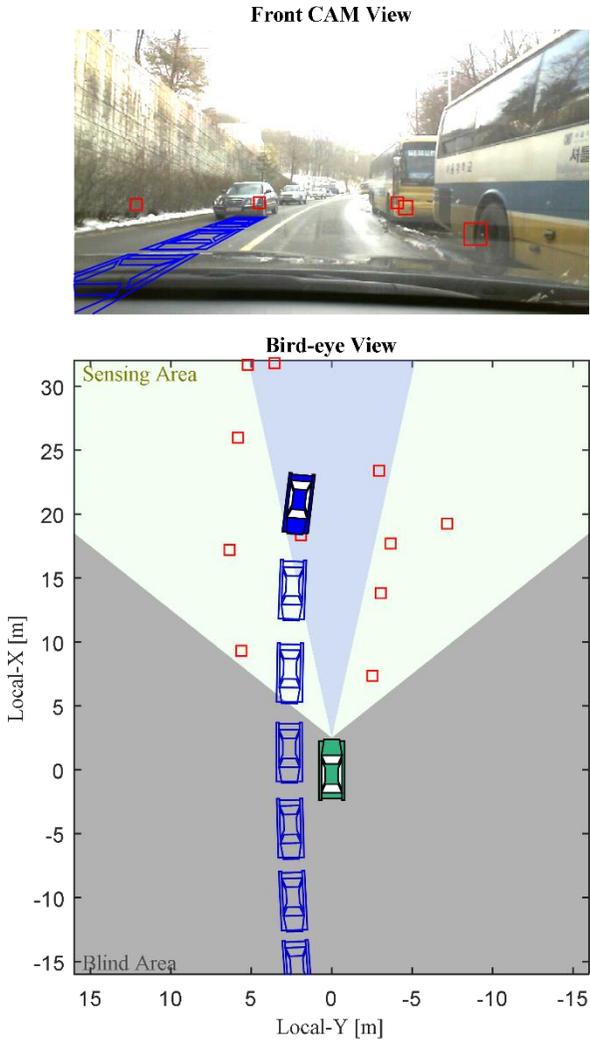


Figure 5.3. An example of moving object tracking and prediction.

The comparison analysis is conducted for general driving situations by a statistical method. These general driving situations include almost every case of driving scenario, such as straight lane keeping, curved lane keeping, curve entry, curve exit, etc. As an example, one test case (which is included in the data set for the comparison analysis) is given in Figure 5.4. In general driving situations, human drivers regulate time headway to maintain a safe distance from the preceding vehicle. On average, the specific range (0-5 s) of time headway accounted for 95 % of the human data [Fairclough'97]. Therefore, from a practical viewpoint, the data within the range of 0-5 s of future time (substituting time headway) have been analyzed.

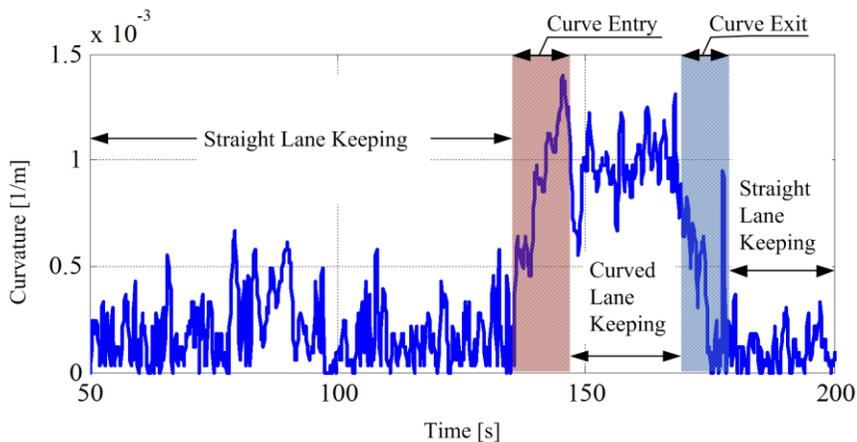


Figure 5.4. Curvature measurements of the Mobileye system including 4 cases of driving scenarios

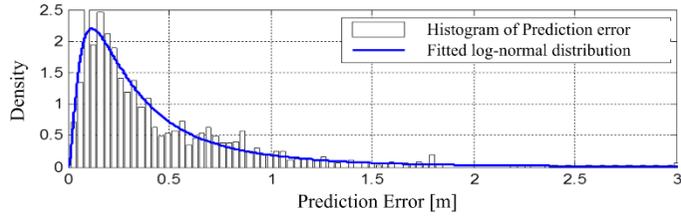
As shown in Figure 5.5 and Figure 5.6, the distribution of prediction errors at some time step can be fitted by the log normal distribution very accurately [Balakrishnan'99, Fairclough'97]. Such tendency is independent of the prediction range or prediction

method. From the figures, we can see that all statistical properties (such as the accuracy and precision) of the proposed algorithm are superior to those of the other two prediction methods, regardless of the prediction time step. Note that this independent superiority from the prediction time step can be defined as the robustness of the algorithm compared to both conventional methods.

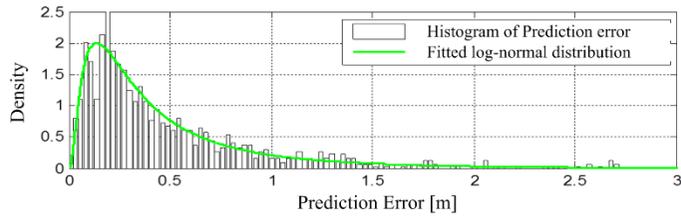
In addition to this cross-sectional analysis, from the fitted density distribution functions of the prediction error, the reliability of prediction can be defined and calculated. If the prediction error exceeds a specific value (the error threshold), the associated predicted position can be said to be unreliable. The value of the error threshold is dependent mainly on the road width. Therefore, in this paper, the value of this error threshold is set to one half of the general road width. In this representation, the reliability of position prediction can be defined as the probability that the prediction error is less than a given error threshold. This definition is depicted in Figure 5.7-(a). As shown in Figure 5.7-(a), the reliability of the position prediction is represented as an area under the density distribution curve within the error threshold. Alternatively, this can be also represented as a specific point on the cumulative distribution curve of the prediction error.

Applying the definition of the prediction reliability to the error distribution functions at each future time step, a comparison of the prediction reliabilities can be conducted as depicted in Figure 5.7-(b) through Figure 5.7-(d). The solid red line represents the reliability of the proposed probabilistic sensor-fusion-based prediction method, the dotted blue line represents the prediction method based on the conventional fixed yaw rate assumption, and the dashed green line represents the prediction method based on the conventional lane keeping assumption. As shown in the figure, the effect of sensor fusion is quite visible and significant. We can see that the fixed yaw rate model, which has a dominant confidence in current vehicle states, has shown superior performance in a short range of prediction time ($t_p=0.0-2.1$ s), while the performance of the lane keeping model is outstanding in a long range of prediction time (after $t_p=2.1$ s). These

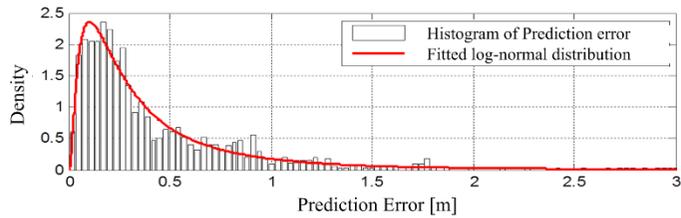
results have reminded us of the necessity of proper fusion between two conventional methods (the current-state-based and road-geometry-based methods). As a result of the proper probabilistic fusion by the proposed algorithm, it is shown that the proposed fusion algorithm can make the most out of the relative merits of each conventional method throughout all the prediction range and predict the future vehicle states more reliably. Moreover, the synergistic effect is predominant in the range of 1.7-4.0 s of prediction time. The proposed algorithm has improved the reliability of prediction by up to 20% compared to prediction based on a fixed yaw rate model, and up to 5% compared to the road-geometry-based prediction method in the range of 1-3 s of future time, which is the main interesting region of various IVSS systems.



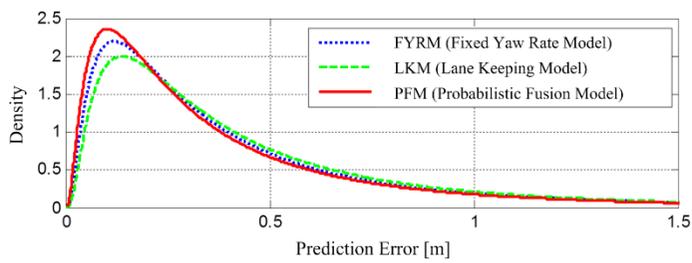
(a) FYRM



(b) LKM

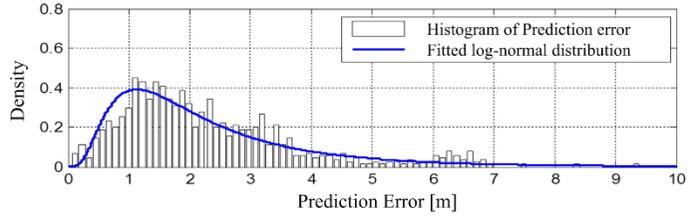


(c) PFM

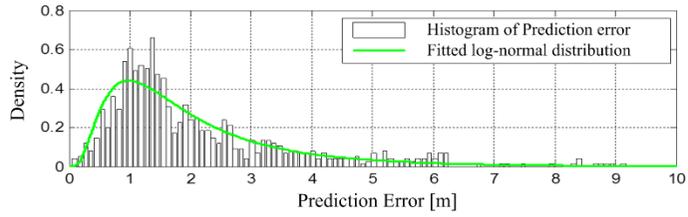


(d) Comparison of fitted error distribution

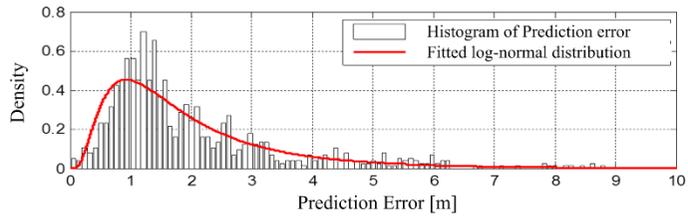
Figure 5.5. Density distribution of the prediction error of each method at $t_p=1.0s$ (i.e. short range prediction)



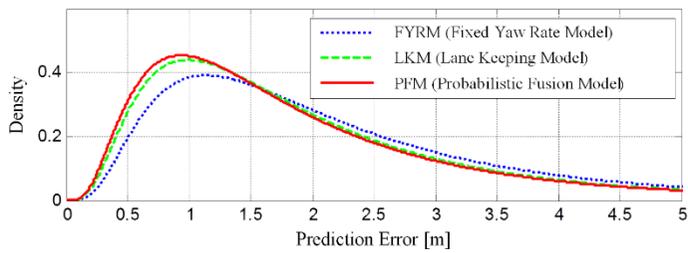
(a) FYRM



(b) LKM

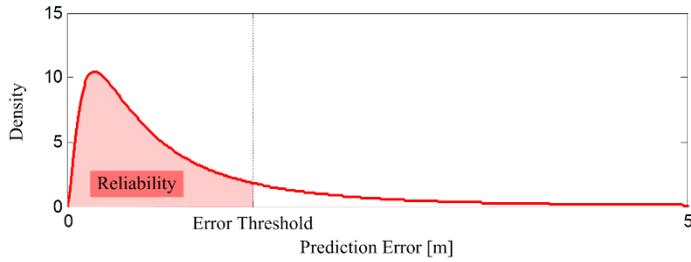


(c) PFM

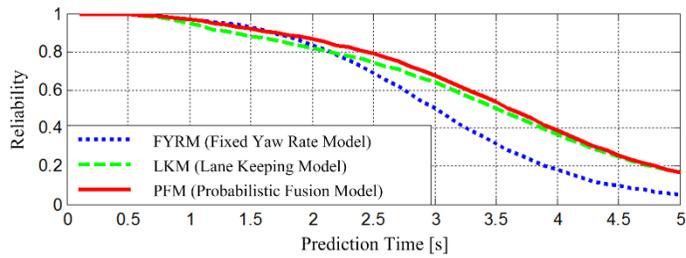


(d) Comparison of fitted error distribution

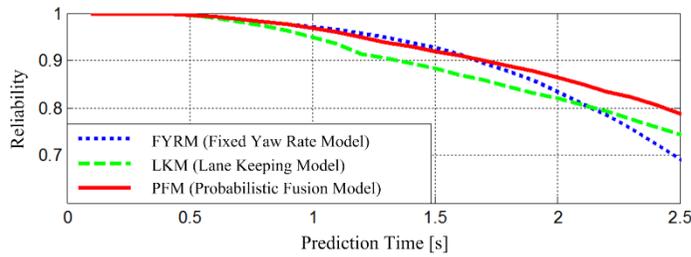
Figure 5.6. Density distribution of the prediction error of each method at $t_p=3.5s$ (i.e. long-range prediction)



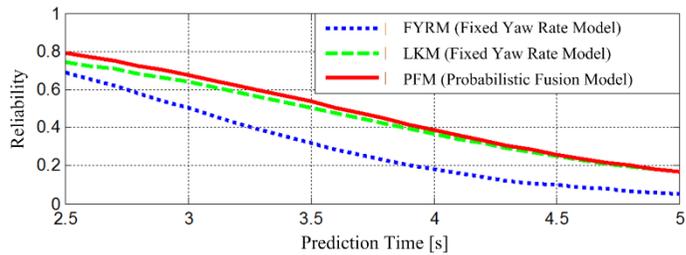
(a) Definition of the Reliability of prediction



(b) Reliability curve for the total prediction range ($[tp]=[0.0, 5.0]$)



(c) Reliability curve for short-range prediction ($[tp]=[0.0, 2.5]$)



(d) Reliability curve for long-range prediction ($[tp]=[2.5, 5.0]$)

Figure 5.7. Prediction reliability evaluation along the prediction time axis

5.2. Safety Driving Envelope Decision

In order to develop a highly automated driving system, a safe driving envelope which indicates the drivable boundaries for safe driving over a finite prediction horizon should be determined with the consideration of not only current states of traffic situation surrounding the subject vehicle but also probable future states of that simultaneously [Ferguson'14].

The safety envelope decision module uses the initial driving corridor as a constraint, and asserts that the vehicle stays in its bounds. Starting from this initial driving corridor, static obstacles, which are represented as grid map, are combined with either the left or right corridor bound. For this coupling, for all obstacles grids, it is decided whether the ego-vehicle is supposed to pass them on the left or right. After having assigned all obstacle grids to either the left or right corridor bound, geometric constraints for motion optimization are newly computed considering driver acceptance, sensor uncertainty, and control uncertainty, etc.

To handle the probable risky situation due to surrounding vehicles within the finite time-horizon, the probable future states of other traffic participants should be considered. To achieve this, every moving object estimated from the sensor system is associated with one lane and a future motion for the object is predicted in probabilistic methodology, assuming that the vehicle follows the lane with acceptable deviation. Similar to the static obstacles, safety envelopes are created for each of the prediction results. However, because the object is in motion, each envelope is active for a specific time step only.

Then the relative complement of predicted area of moving object and unsafe region due to static obstacle in the initial driving corridor is the set of drivable area at each time step, also termed “safety driving envelope”.

In Figure 5.8, these safety envelop decision results are presented for the case of an oncoming vehicle and right side static obstacles (parked buses). All variables are expressed in term of the current local coordinate of ego vehicle. In the figure, the initial driving corridor is depicted as green area, newly computed geometric constraints due to static obstacle as red area, and active constraints due to moving object at each time step as blue area. And the blue vehicles mean the predicted oncoming vehicle and the green vehicle is the predicted ego-vehicle. Note that active constraints due to moving object is propagated as the prediction interval grows longer. Then the safety driving envelope can be defined as the relative complement of blue and red area in green area, at each time step.

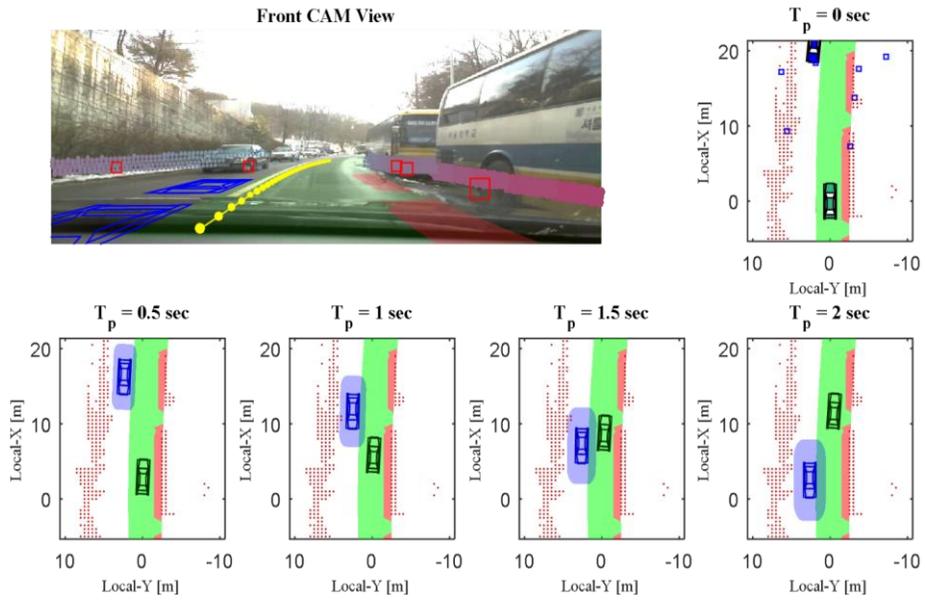


Figure 5.8. Safety envelop decision results at different time steps.

5.3. Model Predictive Control based Motion Planning

In order to determine the desired motion to track, a control architecture based on MPC approach is adopted. The sampling time is chosen as 0.1s and the length of the prediction horizon, N_p , is chosen as 21. These receding horizon optimization problems are solved at each time step and the first terms of the optimal control sequences are applied to the system. Then receding horizon optimization problems for a shifted prediction horizon are solved to obtain new optimal control inputs at next time step. To solve MPC problem in MATLAB, CVXGEN which is designed to be utilizable in MATLAB is used as solver [Mattingley'12]. The MPC problem is defined using CVXGEN syntax, and the CVXGEN returns convex optimization solver for the defined optimization solver for the defined optimization problem.

Particle motion model is used as the system model for MPC approach. State vector, input vector, and dynamic model are defined as follows:

$$\mathbf{x}_{\text{MPC},[j|k]} = \begin{bmatrix} p_{x,[j|k]} & p_{y,[j|k]} & v_{x,[j|k]} & v_{y,[j|k]} \end{bmatrix}^T \quad (5.16)$$

$$\mathbf{u}_{\text{MPC},[j|k]} = \begin{bmatrix} a_{x,[j|k]} & a_{y,[j|k]} \end{bmatrix}^T \quad (5.17)$$

$$\mathbf{x}_{\text{MPC},[j|k+1]} = \mathbf{A}_{\text{MPC},[j|k]} \mathbf{x}_{\text{MPC},[j|k]} + \mathbf{B}_{\text{MPC},[j|k]} \mathbf{u}_{\text{MPC},[j|k]} \quad (5.18)$$

where the system matrices are defined as follows:

$$\mathbf{A}_{\text{MPC},[j|k]} = \begin{bmatrix} 1 & 0 & \Delta t_{\text{MPC}} & 0 \\ 0 & 1 & 0 & \Delta t_{\text{MPC}} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{B}_{\text{MPC},[j|k]} = \begin{bmatrix} \frac{\Delta t_{\text{MPC}}^2}{2} & 0 \\ 0 & \frac{\Delta t_{\text{MPC}}^2}{2} \\ \Delta t_{\text{MPC}} & 0 \\ 0 & \Delta t_{\text{MPC}} \end{bmatrix}$$

MPC approach determines optimal control sequences which minimize the predefined cost function while satisfying constraints. Therefore, before the formulation of MPC problem, we should design the cost function corresponding to the objective of the control algorithm. The predictive expression of the cost function can be represented as follows:

$$\begin{aligned} J = & \sum_{j=1}^{N_p} \left\{ \left(\mathbf{x}_{\text{MPC},[j|k]} - \mathbf{x}_{\text{model},[j|k]} \right)^T \mathbf{Q}_{[k]} \left(\mathbf{x}_{\text{MPC},[j|k]} - \mathbf{x}_{\text{model},[j|k]} \right) \right\} \\ & + \sum_{j=0}^{N_p-1} \left(\mathbf{u}_{\text{MPC},[j|k]}^T \mathbf{W}_{\mathbf{u},[j|k]} \mathbf{u}_{\text{MPC},[j|k]} \right) \\ & + \sum_{j=1}^{N_p-1} \left\{ \left(\mathbf{u}_{\text{MPC},[j|k]} - \mathbf{u}_{\text{MPC},[j-1|k]} \right)^T \mathbf{W}_{\Delta \mathbf{u},[j|k]} \left(\mathbf{u}_{\text{MPC},[j|k]} - \mathbf{u}_{\text{MPC},[j-1|k]} \right) \right\} \end{aligned} \quad (5.19)$$

where, $\mathbf{x}_{\text{model},[j|k]}$ is the initial guessed ego-vehicle states of j-th prediction time step, which predicted by the methodology described in 5.1. \mathbf{Q} is predefined weighting matrix, which penalize the differences between states and rule-based predicted states, $\mathbf{W}_{\mathbf{u}}$ and $\mathbf{W}_{\Delta \mathbf{u}}$ are predefined weighting matrices for the reduction of magnitudes of control input sequences and the rate of change in control sequences respectively. These matrices are positive-definite symmetric.

The driving limit usually becomes dominated by the friction limit of the tires. This limit can be thought of as a circle of forces [Pacejka'05], and these

constraints are given as follows:

$$\|\mathbf{u}_{\text{MPC},[j|k]}\| \leq u_{\text{max}} \quad (5.20)$$

Safety driving envelope decision results at each predated time step should be representable as Quadratic Program to be solved with CVXGEN.

Safety driving envelope decision results at each predated time step should be representable as Quadratic Program to be solved with CVXGEN. For these QP representation, the constraints due to safety driving envelope are approximated as maximal rectangles as shown in Figure 4.4. The rotation of each rectangle is given from initial prediction of ego-vehicle and the maximal rectangle is decided by iterative method. Consequently, the rectangle representations can be expressed as two inequality linear equations as follows:

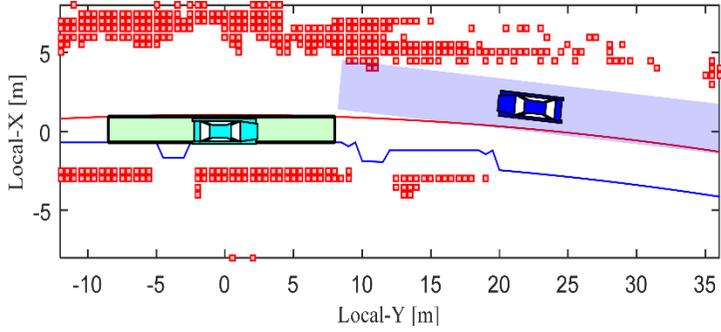
$$\left| \left\{ \mathbf{R}_{\text{model},[j|k]} (\mathbf{x}_{\text{MPC},[j|k]} - \mathbf{x}_{\text{center},[j|k]}) \right\}^1 \right| \leq p_{x,\text{margin}} \quad (5.21)$$

$$\left| \left\{ \mathbf{R}_{\text{model},[j|k]} (\mathbf{x}_{\text{MPC},[j|k]} - \mathbf{x}_{\text{center},[j|k]}) \right\}^2 \right| \leq p_{y,\text{margin}} \quad (5.22)$$

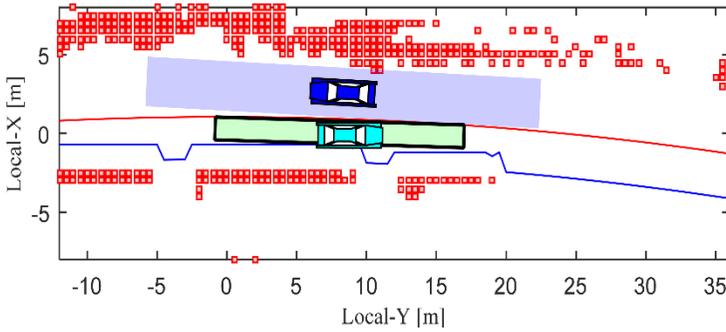
where rotation matrix $\mathbf{R}_{\text{model}}$ is defined by the initial guessed ego-vehicle's states as follows:

$$\mathbf{R}_{\text{model},[j|k]} = \begin{bmatrix} +\cos(\mathbf{x}_{\text{model},[j|k]}^3) & +\sin(\mathbf{x}_{\text{model},[j|k]}^3) \\ -\sin(\mathbf{x}_{\text{model},[j|k]}^3) & +\cos(\mathbf{x}_{\text{model},[j|k]}^3) \end{bmatrix} \quad (5.23)$$

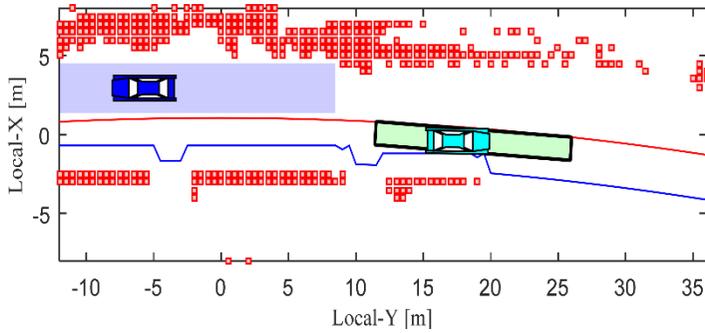
Figure 5.9 shows an example of MPC based motion planning results at the instance of meeting an oncoming vehicle. The maximal rectangle at each predicted time step is depicted as a sky-color-filled square and a cyan vehicle is the MPC solution at that time step. As can be seen in the figure, the ego-vehicle is guided along the safe region with appropriate control input sequences.



(a) at $t_p = 0.0s$



(c) at $t_p = 1.0s$



(e) at $t_p = 2.0s$

Figure 5.9. An example of MPC based motion planning at the instance of meeting an oncoming vehicle.

Chapter 6 Vehicle Tests based Performance Evaluation

The proposed automated driving system is evaluated through test-data based computer simulations and vehicle tests. The test-data based simulation is constructed using the commercial vehicle software, the Carsim and Matlab/Simulink with collected driving data. Data is collected under a various different task conditions while driving on urban roads and expressways. The control output of the proposed algorithm is compared with manual driving of human drivers. And the vehicle tests have been conducted for several times at the internal road of Seoul National University. The given test roads have quite complicated environments to drive automatically. However, from the test results, it is apparent that the test vehicle successfully completed the test route with the proposed automated driving algorithm.

6.1. Test-Data based Simulation

Manual driving of human driver (Human) and controlled behavior of the proposed automated driving system ('ADS') has been compared in the case of driving in busy urban traffic. To verify a driver acceptance and a ride comfort of the proposed algorithm, drivers' relative positions with respect to this safe driving envelope rectangle have been collected and analyzed. As aforementioned, the constraints due to safety driving envelope are approximated as maximal rectangles. Since the vehicle defines the sequences of the safe driving rectangles while it moves, we first transform the previously defined safe driving rectangles to the current local position (including orientation) of the vehicle.

Figure 6.1 shows the data of relative positions with respect to normalized safe driving rectangles which are defined 1 step (0.1s) ahead. The normalized safe driving rectangle is depicted by red-line and the relative normalized position of human driver is depicted by black-square. As can be seen in the figure, drivers control the vehicle to remain in safe driving rectangle.

Figure 6.2 shows the histogram and contour of the data of relative normalized position of human driver. The histogram and contour graph have shown more significant results. As can be seen in Figure 6.2, the driver absolutely stay within the safe driving rectangle. However, it is note that the relative positions do not converge to the center but only stay close enough to the center. It can thus be suggested that a tracking a center-path of drivable area is not a good solution for a driver acceptance and a ride comfort. Figure 6.3 and Figure 6.4

have shown the histogram and contour of the data of relative position of human driver with respect to normalized safe driving rectangles which are defined 3 steps (0.3s) ahead and 5 steps (0.5s) ahead each. As can be seen in the figures, the relative positions do not converge to the center but only stay close enough to it as the case of 1 step ahead.

Figure 6.5 shows the distributions of normalized relative position of each direction. As shown in Figure 6.5, the range of relative lateral position has the larger magnitude. A possible explanation for these results may be the safe driving envelop rectangle (not square). And these results once again stress that driver do not tracking the center of the drivable area but only stay in drivable area with a wide range of choices.

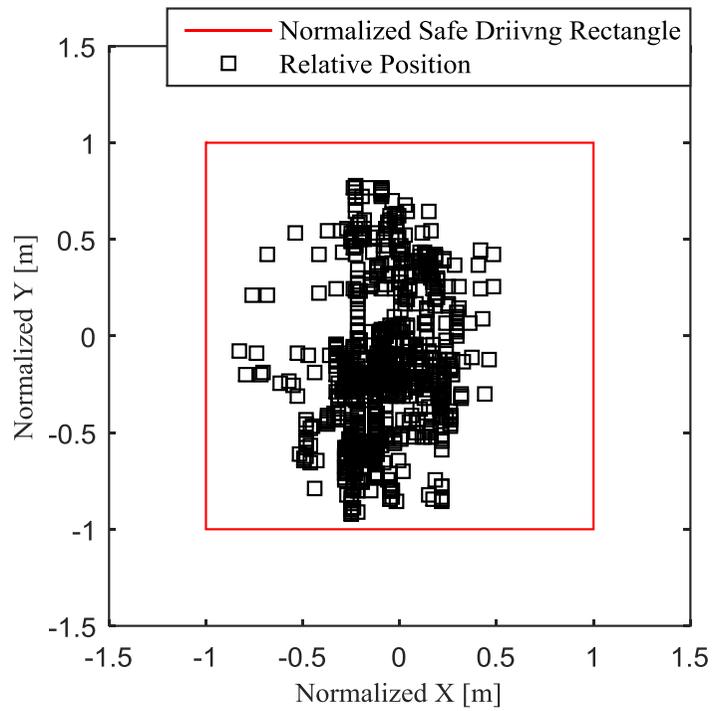
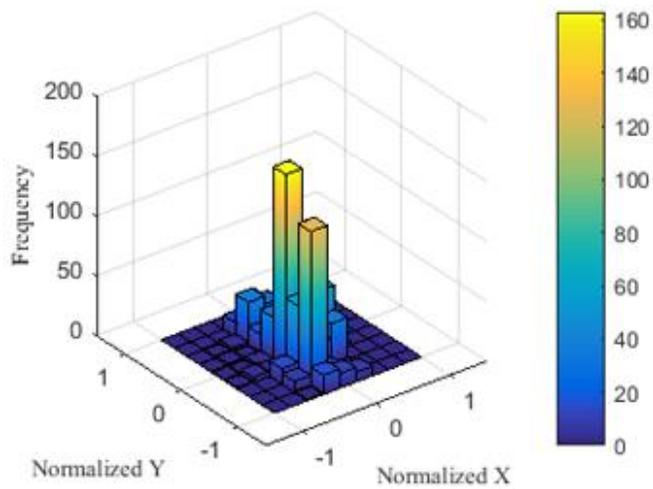
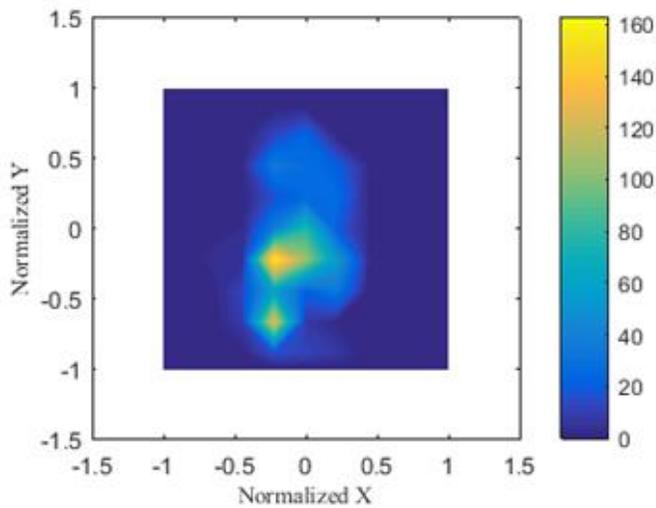


Figure 6.1. Normalized relative positions of human driver with respect to the safe driving rectangle which is defined 1 step (0.1s) ahead.

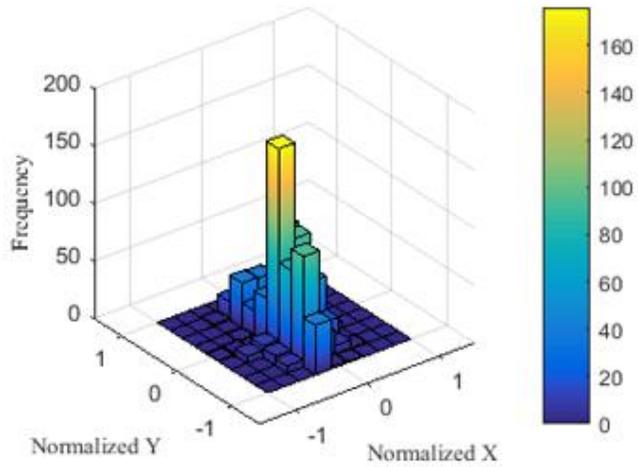


(a) Histogram

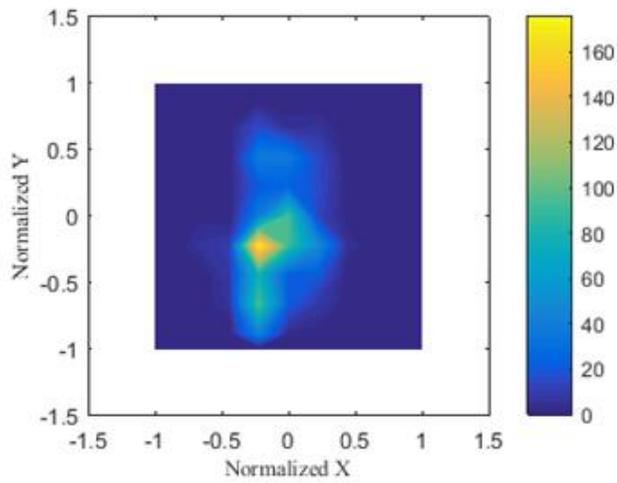


(b) Contour

Figure 6.2. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 1 step (0.1s) ahead.

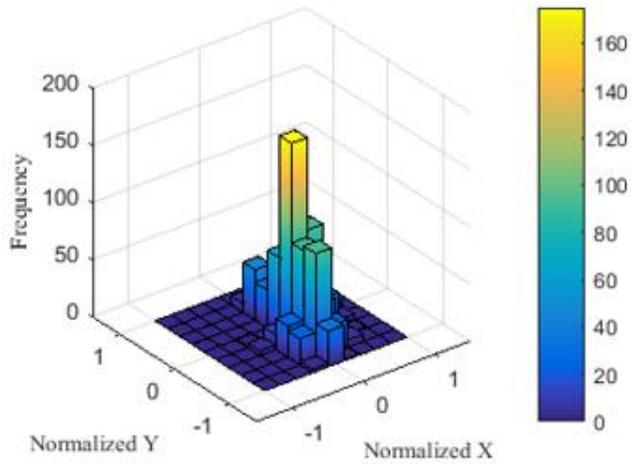


(a) Histogram

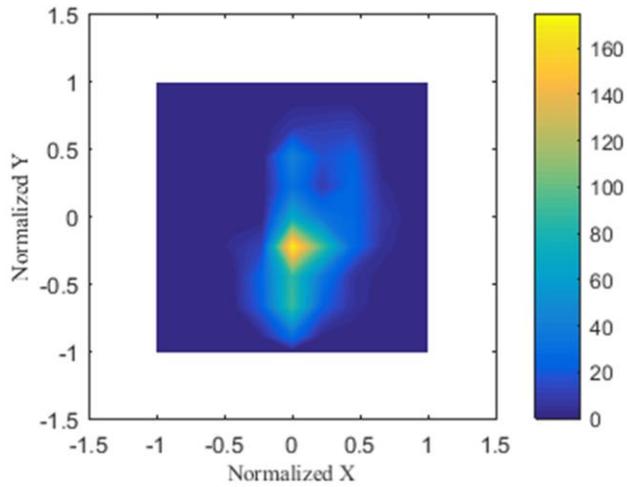


(b) Contour

Figure 6.3. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 3 steps (0.3s) ahead.

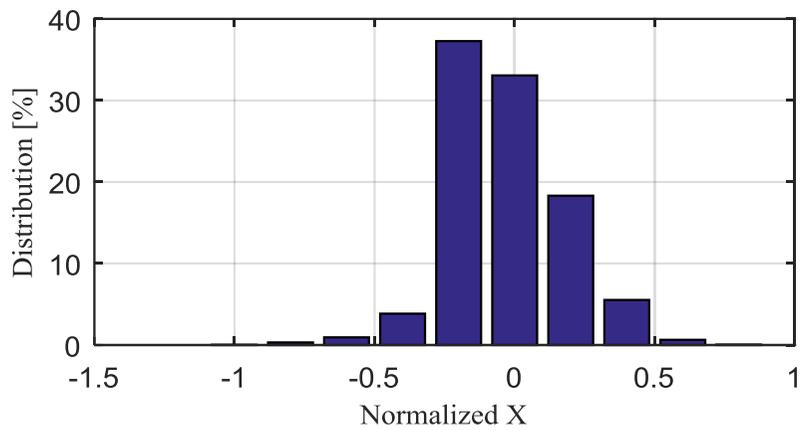


(a) Histogram

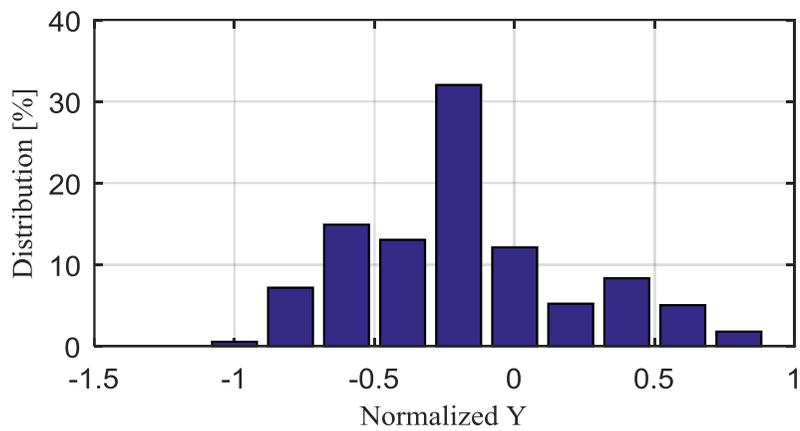


(b) Contour

Figure 6.4. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 5 steps (0.5s) ahead.



(a) Distribution of normalized-x position



(b) Distribution of normalized-y position

Figure 6.5. Distribution of normalized human driver's relative position with respect to the safe driving rectangle defined 1~5 step (0.1s~0.5s) ahead.

6.2. Vehicle Tests: Automated Driving on Urban Roads

Vehicle tests have been conducted for several times at the internal road of Seoul National University. The details of test roads are depicted in Figure 5.6. The given test roads have quite complicated environments to drive automatically. As can be seen in Figure 6.6, the test vehicle drives the given route fully autonomously without a driver manipulation. The lanes were hard to be distinguished because the paint was worn off in places. We need to consider other traffic participants like oncoming and preceding vehicles while there are a lot of buses parked along the road as shown in Figure 6.6. Also, there exist non-vehicle obstacles such as pedestrian and guardrail.

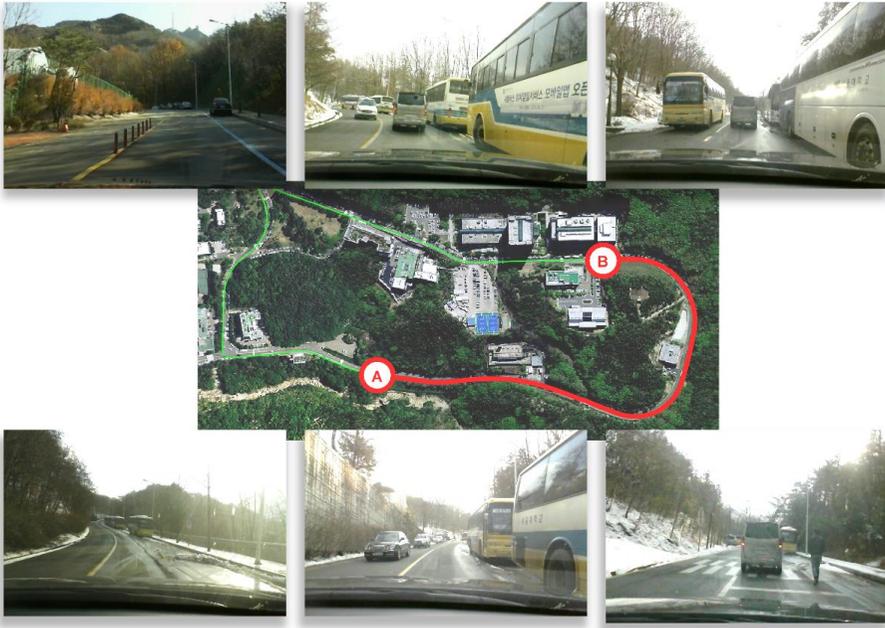


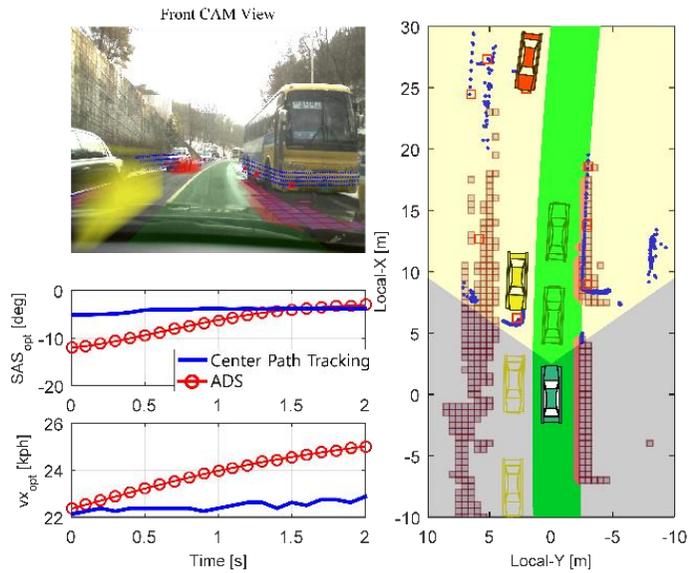
Figure 6.6. Configuration of test route in SNU campus (2km). The route contains a variety of different traffic situations as e.g. narrow passages, curved roads and inclined roads with other traffic participants such as oncoming vehicles, preceding vehicle, and parked busses and non-vehicle obstacles such as guardrails, pedestrian crossings, etc.

The proposed automated driving algorithm has shown the satisfactory control performance and the test results are given in Figure 6.7. As shown in Figure 6.7, the ego-vehicle drives through a narrow urban environment with static infrastructure (buildings, trees, poles, etc.), parking cars on the right, and a preceding vehicle. Furthermore comparisons with center path tracking with the detected lane (without information fusion with GPS/Map) are given. Control input sequences from MPC solver are depicted as red line with circle marker and center-path tracking inputs as blue solid line.

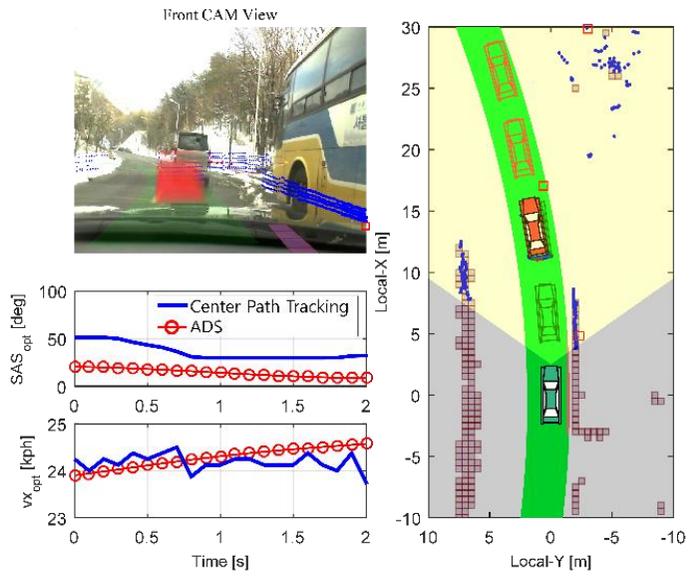
Figure 6.7-(a) shows the result at a driving scene with two oncoming vehicles with right side static obstacles. From the figure, we can see that MPC approach guarantee the safety with oncoming vehicle by giving defense steering command. This results suggest that proposed algorithm may enhance the driver acceptance and driver comfort. Figure 6.7-(b) obtained from the situation with a preceding vehicle at the curved roads. In the case of the normal lane keeping, the proposed algorithm shows the smaller range of control while the center path tracking shows the frequent changes. Figure 6.7-(c) provides the driving scene with a preceding vehicle with both side obstacles (parked buses). Also in this case, the proposed algorithm shows the smaller range of control with higher speed while the center path tracking shows the frequent changes of steering with lower speed. Figure 6.7-(d) presents the stop control behind the decelerating preceding vehicle. In this situation, the lane is invisible due to close preceding vehicle. Consequently, the center path tracking shows the weird steering input while the proposed algorithm gives a stable performance. From the test results, it is apparent that the test vehicle successfully completed the test route with the proposed automated driving algorithm.

To verify a safety performance of the proposed algorithm, normalized relative positions of ADS with respect to this safe driving envelope rectangle have been analyzed. Figure 6.8 shows the histogram and contour of the normalized relative position data of ADS. As can be seen in Figure 6.8, the ADS control the ego-vehicle to absolutely stay within the safe driving rectangle. However, it is note that the relative positions do not converge to the center but only stay close enough to the center. As aforementioned, it is a good solution

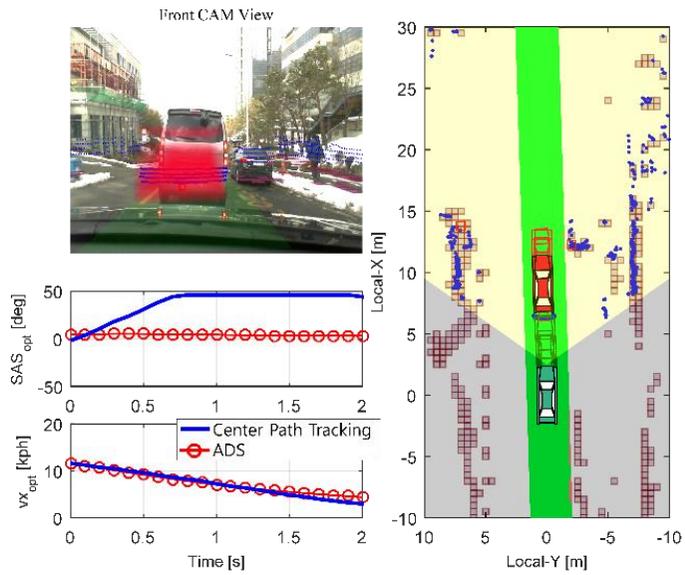
for a driver acceptance and a ride comfort. And Figure 6.9 shows the distributions of normalized relative position of each direction. As shown in Figure 6.9, the range of relative lateral position has the larger magnitude as in the case of human driver. These results once again stress that the proposed algorithm can represent the driver characteristics that do not tracking the center of the drivable area but only stay in drivable area with a wide range of choice.



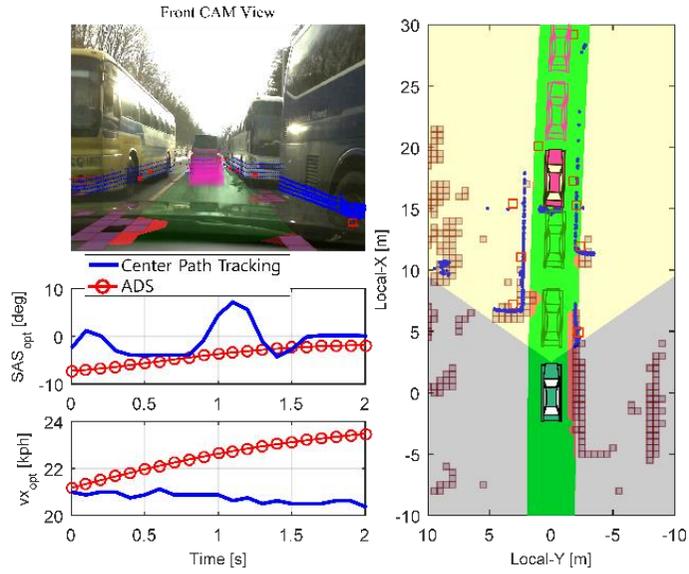
(a) Two oncoming vehicles



(b) Preceding vehicle at curved roads

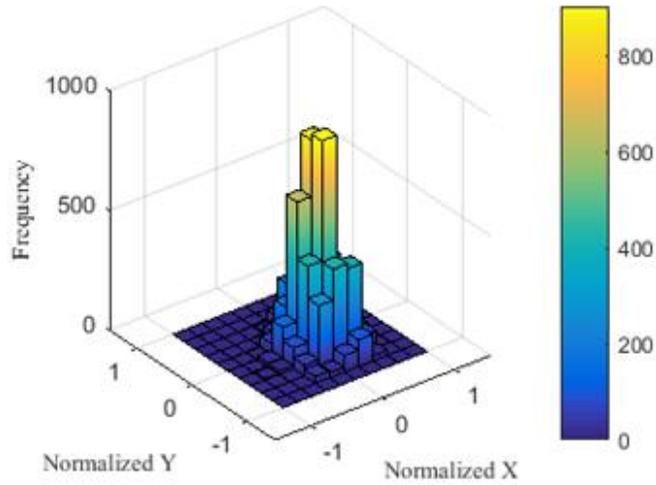


(c) Stop behind the stopped preceding vehicle

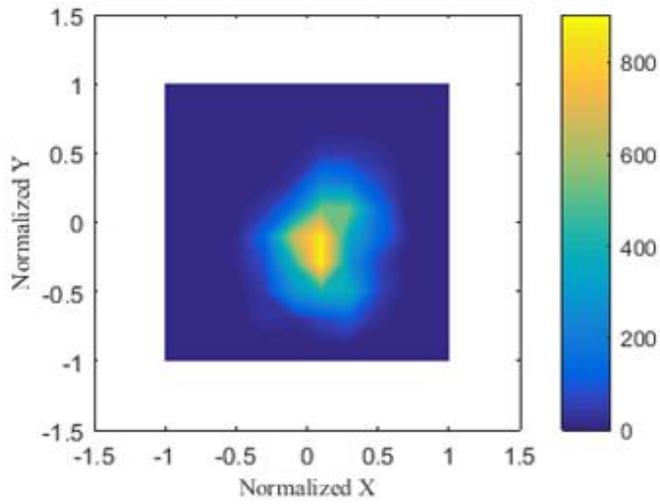


(d) Preceding vehicle with both side obstacles (parked buses)

Figure 6.7. Test results with proposed automated driving algorithm

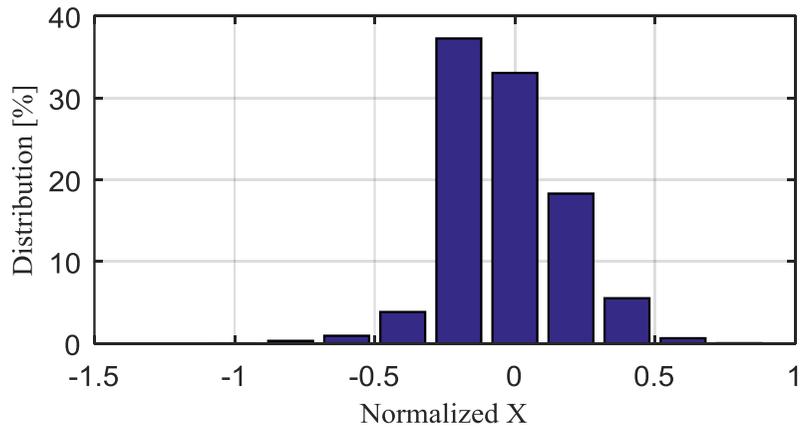


(a) Histogram

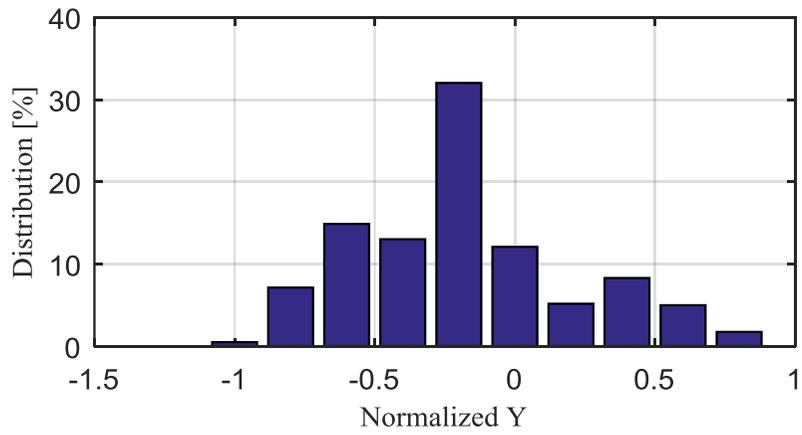


(b) Contour

Figure 6.8. Histogram and contours of normalized relative position of ADS with respect to the safe driving rectangles which are defined 1~5-step ahead.



(a) Normalized-x position



(b) Normalized-y position

Figure 6.9. Distribution of normalized relative position of ADS with respect to the safe driving rectangles which are defined 1~5-step ahead.

Chapter 7 Conclusions and Future Works

This dissertation has proposed a fully automated driving algorithm which is capable of automated driving on urban roads with guaranteed safety. The proposed algorithm consisted of the following three steps: an environment representation, a motion optimization, and a vehicle control. In an environment representation, algorithms for lane-level localization, static/moving obstacle detection, and drivable area representation for safe and comfortable autonomous driving has been developed. And a motion optimization algorithm which is separated into two distinct tasks, safety driving envelope decision and motion optimization, has been developed. The developed motion optimization algorithm solves a geometric constraint problem as a convex optimization problem with linear equality/inequality constraints.

The effectiveness of the proposed automated driving algorithm has been evaluated via test-data based simulations and vehicle tests. In the case of the IMM/EKF based multi-target state estimation algorithm, it is found that the mode probability adaptations are quite reasonable and the proposed algorithm can provide good estimates of the target vehicle's overall behaviors. To investigate the performance enhancement, comparison with model-switching algorithm which is a simple approach to handle the multiple-model problem has been conducted. It is shown that the target state estimation performance can

be significantly enhanced by the proposed algorithm with approximately three times more accurate results while maintain real-time performance with reasonable computation complexity. The IMM/EKF algorithm makes possible the generic assessment and the total management of collision risks with multi-target vehicles in complex driving situations for high-level automated driving. From the results it has been shown that proposed automated driving algorithm can provide the robust performance on an inner-city street scenario.

A further improvement of the autonomous vehicle's ability to interpret a given traffic scenario and to obtain a meaningful behavior prediction of other traffic participants is imperative to achieve incomparable performance and to extend the covering area of autonomous driving.

Bibliography

- ALTHOFF, M., STURSBURG, O. & BUSS, M. 2009. Model-based probabilistic collision detection in autonomous driving. *Intelligent Transportation Systems, IEEE Transactions on*, 10, 299-310.
- ANDERSON, S. J., PETERS, S. C., PILUTTI, T. E. & IAGNEMMA, K. 2010. An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios. *International Journal of Vehicle Autonomous Systems*, 8, 190-216.
- BALAKRISHNAN, N. & CHEN, W. W. 1999. *Handbook of tables for order statistics from lognormal distributions with applications*, Springer Science & Business Media.
- BISHOP, R. A survey of intelligent vehicle applications worldwide. Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE, 2000. IEEE, 25-30.
- BLACKMAN, S. S. 2004. Multiple hypothesis tracking for multiple target tracking. *Ieee Aerospace and Electronic Systems Magazine*, 19, 5-18.
- BORRELLI, F., FALCONE, P., KEVICZKY, T., ASGARI, J. & HROVAT, D. 2005. MPC-based approach to active steering for autonomous vehicle systems. *International Journal of Vehicle Autonomous Systems*, 3, 265-291.
- CARVALHO, A., GAO, Y., GRAY, A., TSENG, H. E. & BORRELLI, F. Predictive control of an autonomous ground vehicle using an iterative linearization approach. Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on, 2013. IEEE, 2335-2340.
- CHEN, B. & TUGNAIT, J. K. 2001. Tracking of multiple maneuvering targets in clutter using IMM/JPDA filtering and fixed-lag smoothing. *Automatica*, 37, 239-249.
- ERLIEN, S., FUJITA, S. & GERDES, J. C. Safe driving envelopes for shared control of ground vehicles. *Advances in Automotive Control*, 2013. 831-836.

- ERLIEN, S. M., FUNKE, J. & GERDES, J. C. Incorporating non-linear tire dynamics into a convex approach to shared steering control. American Control Conference (ACC), 2014, 2014. IEEE, 3468-3473.
- ESKANDARIAN, A. 2012. *Handbook of intelligent vehicles*, Springer London.
- FAIRCLOUGH, S. H., MAY, A. J. & CARTER, C. 1997. The effect of time headway feedback on following behaviour. *Accident Analysis & Prevention*, 29, 387-397.
- FALCONE, P., BORRELLI, F., ASGARI, J., TSENG, H. E. & HROVAT, D. 2007a. Predictive active steering control for autonomous vehicle systems. *Control Systems Technology, IEEE Transactions on*, 15, 566-580.
- FALCONE, P., BORRELLI, F., TSENG, H. E., ASGARI, J. & HROVAT, D. 2008. Linear time-varying model predictive control and its application to active steering systems: Stability analysis and experimental validation. *International journal of robust and nonlinear control*, 18, 862-875.
- FALCONE, P., TUFO, M., BORRELLI, F., ASGARI, J. & TSENG, H. E. A linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems. Decision and Control, 2007 46th IEEE Conference on, 2007b. IEEE, 2980-2985.
- FERGUSON, D. I. F. & DOLGOV, D. A. 2014. Modifying behavior of autonomous vehicle based on predicted behavior of other vehicles. Google Patents.
- FURGALE, P., SCHWESINGER, U., RUFLI, M., DERENDARZ, W., GRIMMETT, H., MUHLFELLNER, P., WONNEBERGER, S., TIMPNER, J., ROTTMANN, S. & LI, B. Toward automated driving in cities using close-to-market sensors: An overview of the V-Charge Project. Intelligent Vehicles Symposium (IV), 2013 IEEE, 2013. IEEE, 809-816.
- GAO, Y., GRAY, A., TSENG, H. E. & BORRELLI, F. 2014. A tube-based robust nonlinear predictive control approach to semiautonomous ground vehicles. *Vehicle System Dynamics*, 52, 802-823.
- GORDON, T., SARDAR, H., BLOWER, D., LJUNG AUST, M., BAREKET, Z., BARNES, M., BLANKESPOOR, A., ISAKSSON-HELLMAN, I.,

- IVARSSON, J. & JUHAS, B. 2010. Advanced crash avoidance technologies (ACAT) program–Final report of the Volvo-Ford-UMTRI project: safety impact methodology for lane departure warning–Method development and estimation of benefits.
- GRAY, A., ALI, M., GAO, Y., HEDRICK, J. K. & BORRELLI, F. 2012. Semi-autonomous vehicle control for road departure and obstacle avoidance. *IFAC Control of Transportation Systems*, 1-6.
- GRAY, A., GAO, Y., HEDRICK, J. K. & BORRELLI, F. Robust predictive control for semi-autonomous vehicles with an uncertain driver model. Intelligent Vehicles Symposium (IV), 2013 IEEE, 2013. IEEE, 208-213.
- HOEDEMAEKER, M. & BROOKHUIS, K. A. 1998. Behavioural adaptation to driving with an adaptive cruise control (ACC). *Transportation Research Part F: Traffic Psychology and Behaviour*, 1, 95-106.
- HUANG, S., REN, W. & CHAN, S. C. 2000. Design and performance evaluation of mixed manual and automated control traffic. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 30, 661-673.
- KASTNER, R., MICHALKE, T., ADAMY, J., FRITSCH, J. & GOERICK, C. 2011. Task-based environment interpretation and system architecture for next generation ADAS. *Intelligent Transportation Systems Magazine, IEEE*, 3, 20-33.
- KATO, S., TSUGAWA, S., TOKUDA, K., MATSUI, T. & FUJII, H. 2002. Vehicle control algorithms for cooperative driving with automated vehicles and intervehicle communications. *Ieee Transactions on Intelligent Transportation Systems*, 3, 155-161.
- KAZANTZIS, N. & KRAVARIS, C. 1999. Time-discretization of nonlinear control systems via Taylor methods. *Computers & chemical engineering*, 23, 763-784.
- KIM, B. & YI, K. 2014. Probabilistic and Holistic Prediction of Vehicle States Using Sensor Fusion for Application to Integrated Vehicle Safety Systems. *Intelligent Transportation Systems, IEEE Transactions on*, 15, 2178-2190.

- KOTHARE, M. V., BALAKRISHNAN, V. & MORARI, M. 1996. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32, 1361-1379.
- LAUGIER, C., PAROMTCHIK, I. E., PERROLLAZ, M., YONG, M., YODER, J.-D., TAY, C., MEKHNACHA, K. & N GRE, A. 2011. Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety. *Intelligent Transportation Systems Magazine, IEEE*, 3, 4-19.
- LEE, J., CHOI, J., YI, K., SHIN, M. & KO, B. 2014. Lane-keeping assistance control algorithm using differential braking to prevent unintended lane departures. *Control Engineering Practice*, 23, 1-13.
- LEE, T., YI, K. & JEONG, C. 2011. Integrated stochastic driver model for evaluation of the vehicle active safety systems. *Fast-zero 2011*.
- LERRO, D. & BARSHALOM, Y. 1993. Tracking with Debiased Consistent Converted Measurements Versus EKF. *Ieee Transactions on Aerospace and Electronic Systems*, 29, 1015-1022.
- LI, X. R. & BAR-SHALOM, Y. 1993. Design of an interacting multiple model algorithm for air traffic control tracking. *Control Systems Technology, IEEE Transactions on*, 1, 186-194.
- MATTINGLEY, J. & BOYD, S. 2012. CVXGEN: a code generator for embedded convex optimization. *Optimization and Engineering*, 13, 1-27.
- MAYNE, D. Q., RAWLINGS, J. B., RAO, C. V. & SCOKAERT, P. O. 2000. Constrained model predictive control: Stability and optimality. *Automatica*, 36, 789-814.
- MOON, S., MOON, I. & YI, K. 2009. Design, tuning, and evaluation of a full-range adaptive cruise control system with collision avoidance. *Control Engineering Practice*, 17, 442-455.
- MOON, S. & YI, K. 2008. Human driving data-based design of a vehicle adaptive cruise control algorithm. *Vehicle System Dynamics*, 46, 661-690.

- NARANJO, J. E., GONZALEZ, C., GARCIA, R. & DE PEDRO, T. 2008. Lane-change fuzzy control in autonomous vehicles for the overtaking maneuver. *Intelligent Transportation Systems, IEEE Transactions on*, 9, 438-450.
- NETTO, M. S., CHAIB, S. & MAMMAR, S. Lateral adaptive control for vehicle lane keeping. American Control Conference, 2004. Proceedings of the 2004, 2004. IEEE, 2693-2698.
- PACEJKA, H. 2005. *Tire and vehicle dynamics*, Elsevier.
- POHL, J., BIRK, W. & WESTERVALL, L. 2007. A driver-distraction-based lane-keeping assistance system. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 221, 541-552.
- POLYCHRONOPOULOS, A., TSOGAS, M., AMDITIS, A. J. & ANDREONE, L. 2007. Sensor fusion for predicting vehicles' path for collision avoidance systems. *Intelligent Transportation Systems, IEEE Transactions on*, 8, 549-562.
- RE, K. 1960. A New Approach to Linear Filtering and Prediction. *Transaction of the ASME-Journal of Basic Engineering*, 35-45.
- ROSSETTER, E. J. & GERDES, J. C. 2006. Lyapunov based performance guarantees for the potential field lane-keeping assistance system. *Journal of dynamic systems, measurement, and control*, 128, 510-522.
- SATTEL, T. & BRANDT, T. 2008. From robotics to automotive: Lane-keeping and collision avoidance based on elastic bands. *Vehicle System Dynamics*, 46, 597-619.
- SKOLNIK, M. I. 2001. *Radar Systems*. McGraw-Hill, NY.
- SWARTZ, D. Clothoid road geometry unsuitable for sensor fusion. IEEE Intelligent Vehicles Symposium, 2003. 484-488.
- TALVALA, K. L., KRITAYAKIRANA, K. & GERDES, J. C. 2011. Pushing the limits: From lanekeeping to autonomous racing. *Annual Reviews in Control*, 35, 137-148.

- TIDEMAN, M., VAN DER VOORT, M., VAN AREM, B. & TILLEMA, F. A review of lateral driver support systems. *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE, 2007. IEEE, 992-999.*
- TINGVALL, C. & HAWORTH, N. Vision Zero: an ethical approach to safety and mobility. *6th ITE International Conference Road Safety & Traffic Enforcement: Beyond, 2000.*
- VANHOLME, B., GRUYER, D., LUSETTI, B., GLASER, S. & MAMMAR, S. 2013. Highly automated driving on highways based on legal safety. *Intelligent Transportation Systems, IEEE Transactions on*, 14, 333-347.
- YUAN, T., BAR-SHALOM, Y., WILLETT, P., MOZESON, E., POLLAK, S. & HARDIMAN, D. 2012. A multiple IMM estimation approach with unbiased mixing for thrusting projectiles. *Aerospace and Electronic Systems, IEEE Transactions on*, 48, 3250-3267.
- ZHANG, Y. & BAR-SHALOM, Y. 2011. Tracking move-stop-move targets with state-dependent mode transition probabilities. *Aerospace and Electronic Systems, IEEE Transactions on*, 47, 2037-2054.
- ZIEGLER, J., BENDER, P., SCHREIBER, M., LATEGAHN, H., STRAUSS, T., STILLER, C., DANG, T., FRANKE, U., APPENRODT, N. & KELLER, C. G. 2014. Making bertha drive—An autonomous journey on a historic route. *Intelligent Transportation Systems Magazine, IEEE*, 6, 8-20.

초 록

주행 안전 및 승차감 향상을 위한 강건 모델 예측 기법 기반 자율 주행 제어 알고리즘 개발

최근 차량용 센싱기술이 급진적으로 발전함에 따라, 차량 연구자들의 관심사는 수동안전시스템(Passive Safety System)에서 능동안전시스템(Active Safety System)으로, 그리고 그 개념이 확장되어 자율주행시스템으로 이동하였다. 예를 들어 적응형 순항 제어 시스템 (ACC: Adaptive Cruise Control), 차선 유지 보조장치 (LKAS: Lane Keeping Assistance System), 차선 변경 보조장치 (LCAS: Lane Change Assistance System), 자동주차 보조시스템 (APA: automated Parking Assist System) 그리고 사각지대 감지경보 (BSI: Blind Spot Intervention) 등의 다양한 능동안전시스템들이 이미 차량제조사들에 의해 출시되었다. 이러한 능동안전시스템들의 출시와 더불어 운전자의 완전한 안전을 보장하기 위한 다양한 프로젝트들이 진행되고 있다. 전세계적으로 다양한 연구팀들이 지속적으로 자율주행분야에 대한 연구를 진행 중이고, 차량제조사들은 개별시스템들을 통합하여 안전기능을 강화하는 연구를 진행하고 있다.

다수의 참고문헌들을 살펴본 결과, 자율주행기술은 잠재적으로 교통혼잡을 완화하고 운전자의 안전을 증진할 것으로 기대된다. 하지만 현재의 자율주행기술은 위성항법보정시스템 (DGPS: Differential Global Positioning System), 다층레이저스캐너 (Velodyne) 등의 고가의 정밀센서가 요구되어 시장에 출시되기에는 비용적인 측면에서 한계점을 가지고 있다. 기술의 발전에 따라 센서들의 가격이 낮아지고 있지만, 차량에 센서들을 장착하고 적용하는 것은 자율주행개발에 있어서 하나의 장벽이 되고 있다.

따라서 본 논문에서는 현재 양산된 차량용 센서들을 조합하여 가격적으로 경제적이고 기술적으로 완성된 센서구성을 활용하여 복잡다차량환경에서의 자율주행이 가능한 시스템을 개발하는 것을 목표로 한다. 전체 알고리즘은 크게 환경인지모듈, 안전영역판단모듈, 그리고 최적요구모션결정모듈로 구성된다.

환경인지모듈에서는 레이더, 라이다, 그리고 차량샤시센서 정보를 활용하여 도로영역, 정지장애물, 이동장애물 등을 인지하고, 안전영역판단모듈에서는 인지된 정보들을 종합하여 자차량이 안전하게 주행 가능한 영역을 유한한 시간 내에서 동적으로 결정한다. 마지막으로 최적요구모션결정 모듈에서는 모델 예측 제어 기법 (MPC: Model Predictive Control)을 활용하여 최적모션을 계산한다.

제안된 자율주행제어 알고리즘의 성능은 실차실험을 통해 검증되었다. 실험결과를 통해 시내주행에서의 인지성능, 판단성능 그리고 안전주행성능을 확인하였고, 일반도로에서 발생 가능한 다양한 시나리오에서 위 성능들이 강건하게 확보됨을 확인하였다.

주요어: 자율 주행 자동차, 지능형 안전 자동차, 모델 예측 제어 기법, 강건 제어, 안전 주행 영역 판단, 확률적 거동 예측, 자율 주행 제어 알고리즘, 통합 안전 제어

학 번: 2011-20689



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

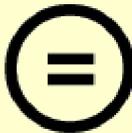
다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사학위논문

**통합환경인지 및 모델예측제어기법
기반 안전자율주행시스템 개발**

**Automated Driving System with Guaranteed
Safety based on Generic Environment
Representation and Model Predictive Control**

2016년 2월

서울대학교 대학원

기계항공공학부

김 범 준

Abstract

Automated Driving System with Guaranteed Safety based on Generic Environment Representation and Model Predictive Control

Beomjun Kim

School of Mechanical and Aerospace Engineering

The Graduate School

Seoul National University

Recently, the interest of automotive researches changes from the passive safety system to the active safety system and, by extension, automated driving system due to advances in sensing technologies. For example, active safety applications, such as vehicle stability control (VSC), adaptive cruise control (ACC), lane keeping assistance (LKA) and lane change assistance (LCA) system), automated parking assist system (APA) and blind spot intervention (BSI), already have been commercialized by major automakers. Furthermore, there are various ongoing projects which are trying to achieve the zero fatality. Several research teams around the world are continuously advancing the field of autonomous driving. And some of major automakers have been researching to integrate individual active safety system for the enhancement of safety. GM is trying to develop and introduce ‘Super Cruise’ system which can drive on the highway without human driver’s intervention. Toyota has undertaken researches to develop ‘Automatic Highway Driving Assist’ technology. The BMW managed to drive 100% automated in real traffic on the freeway from Munich to Ingolstadt, showing a robust, comfortable, and safe driving behavior, even during multiple automated LC maneuvers and the Mercedes Benz

developed ‘Intelligent Drive’ system and followed the route from Mannheim to Pforzheim, Germany, in fully autonomous manner

From a careful review of considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption.

Therefore, this dissertation focused on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier. Mainly three research issues are considered: an environment representation, a motion planning, and a vehicle control.

In the remainder of this paper, we will provide an overview of the overall architecture of the proposed automated driving control algorithm and the experimental results which shown the effectiveness of the proposed automated driving algorithm. The effectiveness of the proposed automated driving algorithm is evaluated via vehicle tests. Test results show the robust performance on an inner-city street scenario.

Keywords: Automated driving vehicle, Model predictive control, Automated driving control algorithm, Environment representation, Safe driving envelope decision

Student Number: 2011-20689

List of Figures

Figure 2.1. System overview of the proposed automated driving system. The proposed algorithm consists of the following three steps: an environment representation, a motion planning, and a vehicle control.....	11
Figure 3.1. The experimental vehicle and its sensor-setup.....	13
Figure 3.2. Sequences of initial driving corridor decision.	18
Figure 3.3. An example of static obstacle map construction.....	20
Figure 4.1. Four coordinate systems used to derive standard process model and standard measurement models	25
Figure 4.2. Parameters to derive the standard form of the refined measurement model	30
Figure 4.3. Various driving patterns of multi target vehicles.	34
Figure 4.4. Various measurement patterns due to the target aspect with respect to the host vehicle’s radar sensor.....	34
Figure 4.5. Twelve modes of detected point of measurement model to describe various measurement patterns.....	35
Figure 4.6. Set of measurement model, their two subsets classified by validity (of each target vehicle in multi traffics), and eight events of target aspect.	38
Figure 4.7. Associated area of possible target aspect event on target-centered coordinate.....	39
Figure 4.8. Architecture of IMM/EKF multi target state estimation algorithm	40
Figure 4.9. Configurations of the host vehicle and the target vehicle	49
Figure 4.10. Test scenario: overtaking target vehicle	51
Figure 4.11. Test data: actual position (from RT-Range) and the associated radar signals of target vehicle	51
Figure 4.12. Three case examples of mode probability update	52

Figure 4.13. Mode probability update results.....	52
Figure 4.14. Estimated target states and the distributions of the estimation errors	53
Figure 4.15. Architecture of model-switching/EKF Algorithm.....	55
Figure 4.16. Multi-target state estimation in complex driving situation	60
Figure 4.17. Target state estimation results for Track 14 (a target cutting in front of the host vehicle)	61
Figure 4.18. Target state estimation results for Track 10 (a target vanishing to rear-right-blind area)	62
Figure 5.1. Procedure and Concept of Traffic Behavior Prediction	66
Figure 5.2. The procedure to calculate the parameter values: covariance of sensor noise, process noise, and modeling error	75
Figure 5.3. An example of moving object tracking and prediction.	76
Figure 5.4. Curvature measurements of the Mobileye system including 4 cases of driving scenarios	77
Figure 5.5. Density distribution of the prediction error of each method at $t_p=1.0s$ (i.e. short range prediction)	80
Figure 5.6. Density distribution of the prediction error of each method at $t_p=3.5s$ (i.e. long-range prediction).....	81
Figure 5.7. Prediction reliability evaluation along the prediction time axis ..	82
Figure 5.8. Safety envelop decision results at different time steps.	85
Figure 5.9. An example of MPC based motion planning at the instance of meeting an oncoming vehicle.	89
Figure 6.1. Normalized relative positions of human driver with respect to the safe driving rectangle which is defined 1 step (0.1s) ahead.....	93
Figure 6.2. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 1 step (0.1s) ahead.	94
Figure 6.3. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 3 steps (0.3s) ahead.	95
Figure 6.4. Histogram and contours of normalized human driver's relative	

positions with respect to the safe driving rectangle which is defined 5 steps (0.5s) ahead.	96
Figure 6.5. Distribution of normalized human driver's relative position with respect to the safe driving rectangle defined 1~5 step (0.1s~0.5s) ahead.	97
Figure 6.6. Configuration of test route in SNU campus (2km). The route contains a variety of different traffic situations as e.g. narrow passages, curved roads and inclined roads with other traffic participants such as oncoming vehicles, preceding vehicle, and parked busses and non-vehicle obstacles such as guardrails, pedestrian crossings, etc.	99
Figure 6.7. Test results with proposed automated driving algorithm	103
Figure 6.8. Histogram and contours of normalized relative position of ADS with respect to the safe driving rectangles which are defined 1~5-step ahead.	104
Figure 6.9. Distribution of normalized relative position of ADS with respect to the safe driving rectangles which are defined 1~5-step ahead.....	105

Contents

Chapter 1 Introduction	1
1.1. Background and Motivation	1
1.2. Previous Researches	4
1.3. Thesis Objectives	7
1.4. Thesis Outline	8
Chapter 2 Overview of an Automated Driving System	9
Chapter 3 Environment Representation	12
3.1. Driving Corridor Decision	14
3.2. Static Obstacle Map Construction	19
Chapter 4 Moving Object Tracking and Estimation	21
4.1. Problem Formulation	22
4.1.1. Stochastic hybrid system	22
4.1.2. Coordinate Systems	24
4.1.3. Standard Process Model	25
4.1.4. Standard Measurement Model.....	28
4.2. Selection of Multiple Model Set and Parameter Design.....	31
4.2.1. Set of Multiple Process Model	31
4.2.2. Set of Multiple Measurement Model.....	33
4.2.3. Event Dependent Transition Probability Matrix	35
4.3. IMM/EKF Multi Target State estimation	40
4.3.1. Host Vehicle Filter	41
4.3.2. IMM/EKF based Filtering	42

4.3.3. Track Management.....	45
4.4. Vehicle Tests based Performance Evaluation.....	47
4.4.1. Configuration of Vehicle Tests	47
4.4.2. Implementation and Evaluation.....	49
4.4.3. Comparison with Model-switching/EKF	54
4.4.4. Experimental Results with Multi-target Situation.....	57
 Chapter 5 . Safety Driving Envelope Decision and Motion Optimization.....	 63
5.1. Multi-traffic Prediction	64
5.1.1. Lane Keeping Behavior Model	66
5.1.2. Vehicle Predictor.....	68
5.1.3. Test Data based Implementation and Performance Evaluation	 72
5.2. Safety Driving Envelope Decision	83
5.3. Model Predictive Control based Motion Planning	86
 Chapter 6 Vehicle Tests based Performance Evaluation	 90
6.1. Test-Data based Simulation	91
6.2. Vehicle Tests: Automated Driving on Urban Roads	98
 Chapter 7 Conclusions	 106
 Bibliography.....	 107
 Abstract in Korean	 114

Chapter 1 Introduction

1.1. Background and Motivation

Since traffic on the road becomes more and more congested over the last decade, a large number of traffic accidents have occurred. According to the previous researches, traffic accidents caused by inattention or fatigue of the driver have been accounted for 90 percent of all traffic accidents [Huang'00]. For instance, according to the previous research [Pohl'07], unintended lane departure accounted for 15 percent of all traffic accidents occurred over the last 10 years in German roadways, and 35 percent of those have been fatal. Furthermore, traffic accidents during a lane change maneuver accounted for about 12 percent of all traffic accidents. Moreover, during many types of collision accidents, most drivers do not attempt to avoid crashes due to unawareness of collision risks [Tideman'07].

Recently, the interest of automotive researches changes from the passive safety system to the active safety system and, by extension, automated driving system due to advances in sensing technologies. For example, active safety applications, such as vehicle stability control (VSC), adaptive cruise control (ACC), lane keeping assistance (LKA) and lane change assistance (LCA system), automated parking assist system (APA) and blind spot intervention (BSI), already have been commercialized by major automakers

[Hoedemaeker'98, Bishop'00, Tingvall'00, Kato'02, Netto'04, Tideman'07, Naranjo'08, Moon'09, Gordon'10, Kastner'11, Zhang'11].

In recent years, the automotive industry is trying to achieve the zero fatality. For instance, there is an ongoing project called 'Vision Zero' in Sweden to reduce fatalities to zero by 2020. In order to achieve the zero fatality, many researches have been undertaken to integrate individual active safety systems for the development of an automated driving system [Bishop'00, Eskandarian'12]. Furthermore, some of major automakers have been researching to integrate individual active safety system for the enhancement of safety. For instance, GM is trying to develop and introduce 'Super Cruise' system which can drive on the highway without human driver's intervention. In this system, adaptive cruise control technology and active lane keeping control technology would be combined. Toyota has undertaken researches to develop 'Automatic Highway Driving Assist' technology. The objective of this system is to maintain safety of the subject vehicle and reduce fatigue on the driver. This system is developed by integrating cooperative adaptive cruise control technology, which communicates with surrounding vehicles wirelessly, and lane trace control technology.

Nowadays many major automakers have undertaken researches to develop an automated driving system. Moreover Google is active in developing technology for automated driving vehicle. The automated driving system is required to detect surrounding environment, recognize traffic situation surrounding the subject vehicle and travel autonomously without human driver's support.

From a considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption [Furgale'13]. And in order to develop a highly automated driving control algorithm, mainly three research issues should be considered: an environment representation, a motion planning, and a vehicle control.

Therefore, this dissertation focuses on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier.

1.2. Previous Researches

A number of studies have been introduced for the development of an automated driving control algorithm. Rossetter et al. designed an active lane-keeping assistance system to apply an additional guidance force by using a potential field approach [Rossetter'06]. Sattel et al. proposed a path planning algorithm to keep the lane and avoid a collision based on the potential field approach [Sattel'08]. Naranjo et al. designed a lane change control algorithm which mimics human behaviors during over taking maneuvers using fuzzy logic for autonomous vehicles [Naranjo'08]. Talvala et al. designed an autonomous driving control algorithm combining look-ahead controller and longitudinal controller to travel the reference path [Talvala'11].

In developing an automated driving system, it is required to operate in a wide operating region and limit the set of permissible states and inputs. Therefore MPC approach has been used widely because of its capability to handle system constraints in a systematic way and adaptability for various driving scenarios [Anderson'10, Falcone'07a, Falcone'08, Gray'12, Gray'13].

Borrelli et al. proposed a MPC based steering control algorithm to track a predefined desired path while satisfying constraints and evaluated this approach via simulation studies [Borrelli'05]. Falcone et al. presented a MPC based active steering controller for tracking the desired trajectory as close as possible while satisfying various constraints. In this research, the performance of the control algorithm using a nonlinear vehicle model as a predictive model is compared

with that using successive linearized vehicle model via simulation and experiment test. Then, Falcone et al. presented a sufficient stability condition for the predictive active steering control algorithm based on linear time-varying model. These approaches mentioned above only focused on the development of a lateral control law. However, there could be a limitation to follow the desired path only with a steering control. Falcone et al. designed a MPC based automated driving control algorithm to control the front steering angle and independent tractive torque at each wheel and evaluated this approach via simulation studies [Borrelli'05, Falcone'07a, Falcone'08, Falcone'07b].

In order to compensate the effect on the control performance by model uncertainties and exogenous disturbances, robust MPC approach which adds a linear feedback control input to the nominal control inputs based on the analysis of robust invariant sets have been introduced and used to design an autonomous control algorithm [Gao'14, Gray'13, Kothare'96, Mayne'00].

In most of these researches mentioned above, it is assumed that the desired trajectory over a finite horizon is known. However, in order to design an automated driving system, we should decide a drivable area or desired path in real time. Erlien et al. use a safe driving envelope which means a safe region of states in which the system should be constrained. In this research, the safe driving envelope consists of a stable handling envelope to ensure vehicle stability and an environmental envelope to constrain the position states for the collision avoidance. The environmental envelope is defined based on the current states of surrounding environment of the subject vehicle. Carvalho et al. presented a control algorithm for an autonomous ground vehicle to follow the

centerline while avoiding collisions with obstacle and evaluate this approach via experiment test. In experiment test, it is assumed that an obstacle is moving at constant speed [Carvalho'13, Erlie'13, Erlie'14].

However, according to the patent of Google at 2013, an autonomous vehicle should be controlled based on the current state of the subject vehicle, the current state of surrounding vehicles and predicted behaviors of surrounding vehicles [Ferguson'14]. Furthermore, according to previous researches, a probabilistic prediction of probable behaviors of surrounding vehicles is required to prevent a potential collision accident in the foreseeable future [Althoff'09, Kim'14, Polychronopoulos'07].

From a considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption [Furgale'13].

Therefore, in this research, we focus on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier.

1.3. Thesis Objectives

This dissertation focuses on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier.

From a considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption [Furgale'13].

Mainly three research issues are considered: an environment representation, a motion planning, and a vehicle control. In the remainder of this paper, we will provide an overview of the overall architecture of the proposed automated driving control algorithm and the experimental results which shown the effectiveness of the proposed automated driving algorithm. The effectiveness of the proposed automated driving algorithm is evaluated via vehicle tests. Test results show the robust performance on an inner-city street scenario.

1.4. Thesis Outline

This dissertation is structured in the following manner. An overall architecture of the proposed automated driving control algorithm is described in Chapter 2. In Chapter 3 and 4, an environment representation algorithm are introduced. The main objectives of the environment representation algorithm are a lane-level localization, a static obstacle detection, and moving object tracking and estimation. In Chapter 5, the concept of the probabilistic prediction of surrounding vehicle's behavior and a determination of the safe driving envelope based on the probabilistic prediction are described. Then an algorithm for a motion optimization and a vehicle control is designed based on MPC approach. Chapter 6 shows the experiment results for the evaluation of the performance of the proposed algorithm. Then the conclusion which describes the summary and contribution of the proposed automated driving control algorithm and future works is presented in Chapter 7.

Chapter 2 Overview of an Automated Driving System

From a considerable amount of literature, automated driving technology has the potential to reduce the environmental impact of driving, reduce traffic jams, and increase the safety of motor vehicle travel. However, the current state-of-the-art in automated vehicle technology requires precise, expensive sensors such as differential global positioning systems, and highly accurate inertial navigation systems and scanning laser rangefinders. While the cost of these sensors is going down, integrating them into cars will increase the price and represent yet another barrier to adoption. Therefore, in this research, we focus on developing a fully automated driving algorithm which is capable of automated driving in complex scenarios while a chosen sensor configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous vehicles presented earlier.

As aforementioned, mainly three research issues are considered: an environment representation, a motion planning, and a vehicle control. The system architecture of the algorithm is outlined in Figure 2.1. The environment representation system consists of three main modules: object classification, vehicle/non-vehicle tracking and map/lane based localization. All system modules make use of information from equipped various sensors. The main sensing components are a vision, radars and vehicle sensors. The chosen sensor

configuration is closer to current automotive serial production in terms of cost and technical maturity than in many autonomous robots presented earlier. The objective of the motion planning modules is to derive an optimal trajectory as a function of time, from the environment representation results. A safety envelope definition module determines the complete driving corridor that leads to the destination while assigning all objects to either the left or right corridor bound. In the case of moving objects such as other traffic participants, their behaviors are anticipated in the near future. An optimal trajectory planner uses the safety envelop as a constraint and computes a trajectory that the vehicle stays in its bounds. The vehicle control module feeds back the pose estimate of the localization module to guide the vehicle along the planned trajectory.

In the remainder of this paper, we will provide an overview of the overall architecture of the proposed automated driving control algorithm and the experimental results which shown the effectiveness of the proposed automated driving algorithm.

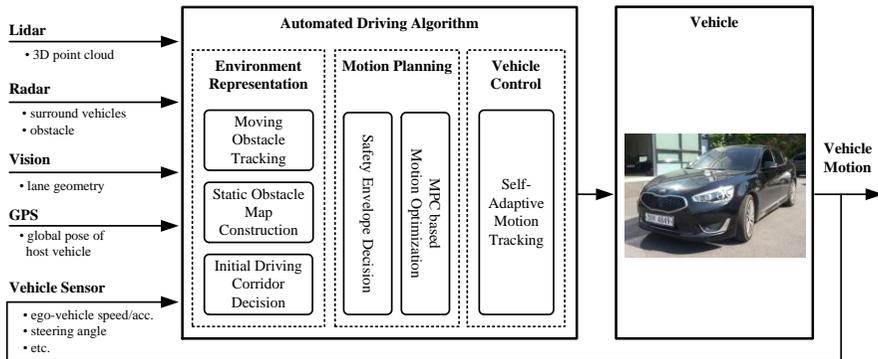


Figure 2.1. System overview of the proposed automated driving system. The proposed algorithm consists of the following three steps: an environment representation, a motion planning, and a vehicle control.

Chapter 3 Environment Representation

Precise and comprehensive environment perception is the basis for safe and comfortable autonomous driving in complex traffic situations such as busy cities [Vanholme'13, Ziegler'14]. We modified the serial-production sensor setup already available in our test vehicles as follows: A multilayer laser scanner was added for monitoring static obstacles with increased precision. For lane detection, an additional monocular vision system was mounted on the windshield. And a low-cost-GPS was equipped for a rough-precision ego-localization which is used for initial corridor decision. The complete sensor setup is shown in Figure 3.1. Depicted in yellow is a monocular vision system for lane detection. Front radar system is depicted in blue and two rear-side radars are depicted in green. A multilayer laser scanner for obstacle monitoring is shown in red and low-cost GPS is depicted in purple signal. The main objectives of these sensors are lane-level localization, static/moving obstacle detection, and drivable area representation for safe and comfortable autonomous driving.

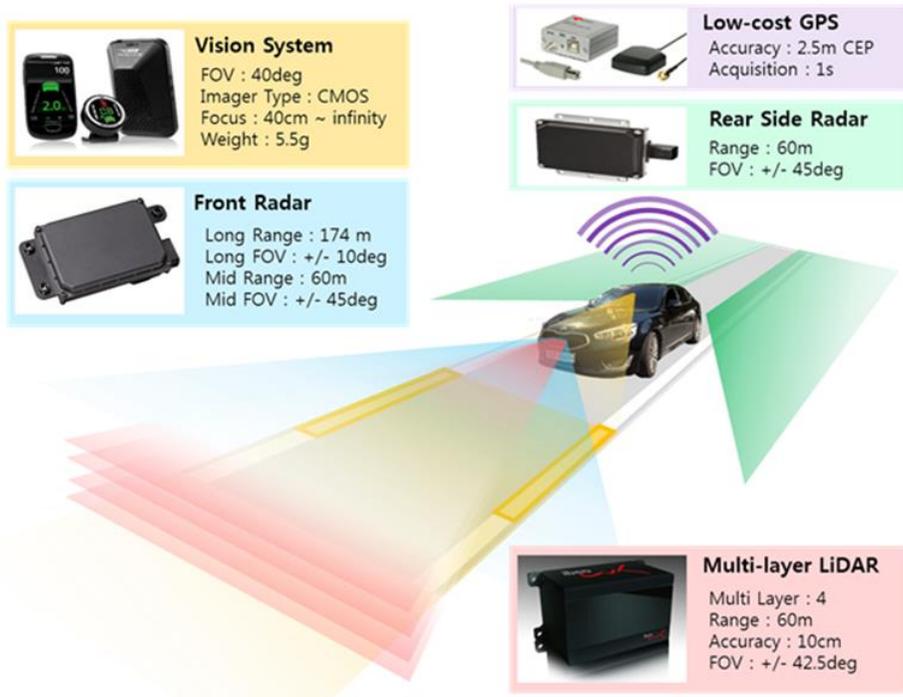


Figure 3.1. The experimental vehicle and its sensor-setup.

3.1. Driving Corridor Decision

An initial driving corridor is determined from detected lanes, rough-precision localization, and digital map which is stored in advance. The map contains properties of the environment which are necessary for driving, but cannot be reliably detected by sensors such as road width, and global waypoint of centerline of total route.

The rough-precision localization is accomplished with ego chassis sensor signals and low-cost GPS. The state vector for localization are defined as follows:

$$\mathbf{x} = [p_x \quad p_y \quad \psi \quad v_x \quad \gamma \quad a_x]^T \quad (3.1)$$

where the subscript x and y denote x-axis and y-axis of each frame, p denotes the relative position, θ denotes the relative yaw angle, v denotes the velocity, γ denotes the yaw rate, and a denotes the acceleration. The process model is discretized based on no-slip assumption as follows:

$$\begin{aligned} \mathbf{x}[k] &= \mathbf{f}(\mathbf{x}[k-1]) + \mathbf{w}[k] \\ &= \mathbf{x}[k-1] + \Delta T \cdot (\mathbf{a}(\mathbf{x}[k-1])) + \mathbf{w}[k] \end{aligned} \quad (3.2)$$

where

$$\begin{aligned} \mathbf{a}_1 &= v_x \cos \psi & \mathbf{a}_2 &= v_x \sin \psi & \mathbf{a}_3 &= \gamma & \mathbf{a}_4 &= a_x & \mathbf{a}_5 &= 0 & \mathbf{a}_6 &= 0 \\ \mathbf{w}[k] &\sim (\mathbf{0}, \mathbf{W}[k]) \end{aligned}$$

Where \mathbf{a} is the nonlinear process vector equation, and \mathbf{w} is the process noise vector which is the white noise with a covariance matrix, \mathbf{W} . The measurement model for the dead-reckoning is derived as follows:

$$\mathbf{z}_{DR}[k] = \mathbf{h}_{DR}(\mathbf{x}[k]) + \mathbf{v}_{DR}[k] \quad (3.3)$$

where

$$\begin{aligned} \mathbf{h}_{DR,1} &= v_x - \frac{w}{2} \cdot \gamma & \mathbf{h}_{DR,2} &= v_x + \frac{w}{2} \cdot \gamma & \mathbf{h}_{DR,5} &= \gamma & \mathbf{h}_{DR,6} &= a_x \\ \mathbf{h}_{DR,3} &= \begin{pmatrix} \cos \delta \left(v_x - \frac{w}{2} \cdot \gamma \right) \\ + \sin \delta (l \cdot \gamma) \end{pmatrix} & \mathbf{h}_{DR,4} &= \begin{pmatrix} \cos \delta \left(v_x + \frac{w}{2} \cdot \gamma \right) \\ + \sin \delta (l \cdot \gamma) \end{pmatrix} \\ \mathbf{v}_{DR}[k] &\sim (0, \mathbf{V}_{DR}[k]) \end{aligned}$$

Where \mathbf{h} is nonlinear measurement vector where the elements are rear-left wheel speed, rear-right wheel speed, front-left wheel speed, front-right wheel speed, yaw rate, and longitudinal acceleration in numeric order. When the GPS is updated, an augmented measurement model is modified as follows:

$$\begin{aligned} \mathbf{z}_{+GPS}[k] &= \mathbf{h}_{+GPS}(\mathbf{x}[k]) + \mathbf{v}_{+GPS}[k] \\ &= \left[\mathbf{h}_{DR}(\mathbf{x}[k]) \quad \mathbf{h}_{+GPS,7} \quad \mathbf{h}_{+GPS,8} \quad \mathbf{h}_{+GPS,9} \right]^T + \mathbf{v}_{+GPS}[k] \end{aligned} \quad (3.4)$$

where

$$\begin{aligned} \mathbf{h}_{+GPS,7} &= p_x & \mathbf{h}_{+GPS,8} &= p_y & \mathbf{h}_{+GPS,9} &= \psi \\ \mathbf{v}_{+GPS}[k] &\sim \left(0, \begin{bmatrix} \mathbf{V}_{+DR}[k] & 0 \\ 0 & \mathbf{V}_{+GPS}[k] \end{bmatrix} \right) \end{aligned}$$

From the global pose estimate of the ego vehicle, a nearest segment is selected from the map and transformed to local coordinate. From the detected lane information, distances and angle deviation between the vehicle's pose and the centerline of the local segment are updated with improved accuracy.

It is common practice to describe the forward road geometry by a 2nd-order polynomial [Swartz'03]. The relation between the ego-vehicle and the road

center line can be described by two factors: the relative lateral position and the relative heading angle. With these two factors, the road geometry, which has a curvature radius R , can be approximated by [Swartz'03]:

$$\begin{aligned} y_r(x) &= \frac{1}{2R}x^2 - \tan e_\theta \cdot x - e_y \\ &= a_2 \cdot x^2 + a_1 \cdot x + a_0 \end{aligned} \quad (3.5)$$

where x is the down-range distance, and y_r is the lateral position of the corresponding road center in the current body coordinates. As the vehicle drives with velocity v and yaw rate γ , the coefficients describing the road geometry change according to the motion of the vehicle. If the state vector are defined as these coefficients, the process model and the measurement model can be derived by:

$$\begin{aligned} \dot{\mathbf{x}}_r &= \mathbf{A}_r \mathbf{x}_r + \mathbf{B}_r \mathbf{u}_r + \mathbf{w}_r \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 2v_x & 0 & 0 \\ 0 & v_x & 0 \end{bmatrix} \begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \gamma + \mathbf{w}_r \end{aligned} \quad (3.6)$$

$$\mathbf{w}_r \sim (\mathbf{0} \quad \mathbf{Q}_r)$$

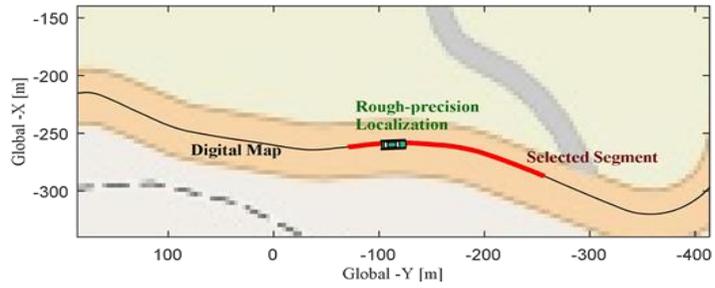
$$\begin{aligned} \mathbf{z}_r[k] &= \mathbf{H}_r \mathbf{x}_r[k] + \mathbf{v}_r[k] \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}_r[k] + \mathbf{v}_r[k] \end{aligned} \quad (3.7)$$

$$\mathbf{v}_r[k] \sim (\mathbf{0} \quad \mathbf{V}_r)$$

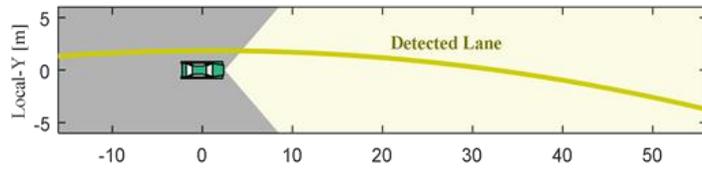
where the subscript 'r' is used to denote relation to the road geometry states. The Kalman filter is used for the estimation of road geometry coefficients. The best estimate result of the localization are used for the longitudinal velocity and the yaw rate. Hence, the covariance of the process noise should be well defined so that it can represent the effect of the estimate error of the vehicle motion

filtering. As a result, the estimate of the initial driving corridor coefficients is recursively estimated using the Kalman filter [RE'60]. Both detected lane and selected local segment are used for measurement update. Note that the covariance of the measurement noise should be well defined so that it can represent each of two errors: the localization error and lane detection error.

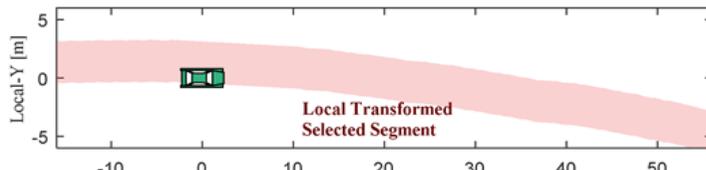
Figure 3.2 shows an example of sequence of initial driving corridor decision. Figure 3.2-(a) shows a nearest segment selection from the map data. Digital map is depicted as black line, rough-precision global pose estimate result is depicted as green vehicle, and the selected nearest segment is depicted as red-thick line. Figure 3.2-(b) shows a lane detection results. From the figure, we can see that yellow centerline is detected. Figure 3.2-(c) shows transformed segment to local coordinate. Due to the error of the global pose estimate, it seems that the ego-vehicle deviated from the lane (actually not). Figure 3.2-(d) shows lane-level segment matching result. Because the locally transformed segment is updated with the detected lane, the updated local segment has more improved accuracy.



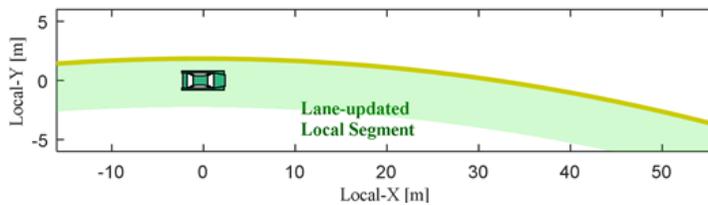
(a) Nearest segment selection



(b) Lane detection



(c) Segment transformation to local coordinate



(d) Lane-level segment matching

Figure 3.2. Sequences of initial driving corridor decision.

3.2. Static Obstacle Map Construction

A grid map representation was used for the static obstacle recognition. To cope with noisy measurement of laser scanner, we accumulate multiple observations to obtain reliable map information. Each grid of the map has the counter which indicates how many times obstacles have been observed in that grid. At each observation, the counter of each grid where obstacles exist is incremented. If the counter value of a grid is higher than a certain threshold, the grid is considered unsafe. The set of unsafe grids constitutes the current static obstacle map.

In the case of the candidates for moving objects, they are detected by comparing the current observation with estimated moving objects from radar. If the point is completely inside the region of estimated moving object, the point is considered as a part of a moving object.

Since the vehicle generates a map while it moves, we first transform the existing map to the current local position (including orientation) of the vehicle. And then current measured data is integrated to the transformed map. The position deviation of the vehicle is estimated from the odometry. To reduce the effect of accumulated error when the vehicle moves by a long distance, only thirteen latest observations are used for making the static obstacle map construction.

Figure 3.3 shows an example of static obstacle map construction. The sensing area of laser scanner is depicted as yellow region. Blue dots are current

observation while the red squares are unsafe-grids of the constructed static obstacle map. We can see that parked-buses on the right side and left-barrier are well tracked though they are hidden in blind area. And the part of a moving object is recognized very well.

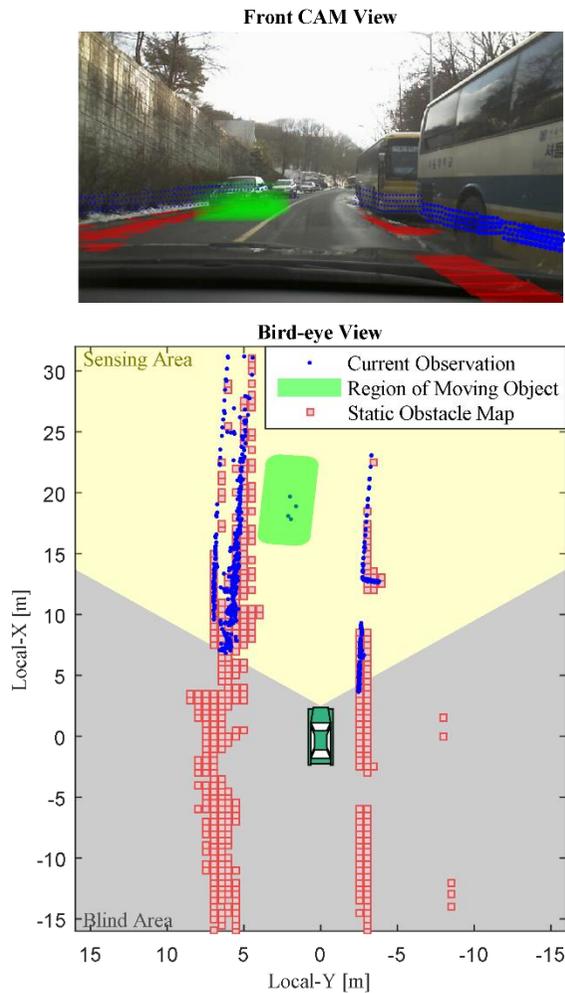


Figure 3.3. An example of static obstacle map construction.

Chapter 4 Moving Object Tracking and Estimation

For the generic assessment and the total management of collision risks with multi-traffics in complex driving situations, it is essential to estimate and represent the target vehicles' overall behaviors such as heading, yaw rate, absolute velocity and acceleration as well as relative position and relative velocity which are the state of the target vehicle with respect to the host vehicle.

To achieve this, this chapter describes an interacting multiple model (IMM) approach using extended Kalman filters (EKF) to improve multi-target states estimation performance with utilization of automotive radars. Automotive radar is the best fitted vehicular surround sensing technology with respect to functionality, robustness, reliability, dependence on weather conditions etc.

In an application of the automotive radar, the most important issue is to handle an uncertain measurement model problem that is wandering on the target's physical boundary. To cope with this problem, multi-models are formulated and a new multi-target tracking algorithm is developed based on IMM approach, global nearest neighbor (GNN) based data association and EKF method with elaborated modeling of automotive radar.

The performance of the proposed multi-target tracking algorithm is verified via vehicle tests in real roads. It is shown that the target vehicle's overall behavior can be estimated by proposed elaborated models and the estimation performance can be significantly enhanced.

4.1. Problem Formulation

In this section, a stochastic hybrid system is formulated in the form of an IMM/EKF algorithm for tracking multiple target vehicles. A standard process model to describe behaviors of target vehicle is derived and discretized. And a standard measurement model is defined to describe automotive radar characteristics.

4.1.1. Stochastic hybrid system

Assume that there are total N_{target} targets and the system of each target can be modeled as one of the N_{mode} hypothesized models where the target set is denoted as $\mathbf{T} \equiv \{1, 2, \dots, N_{target}\}$ and the model set is denoted as $\mathbf{M} \equiv \{1, 2, \dots, N_{mode}\}$. From the work of Li and Bar-shalom [Li'93], a general description for a stochastic hybrid system of target $n(\in \mathbf{T})$ during the sampling period $(t_{k-1}, t_k]$ can be given as:

$$\mathbf{x}_n^j[k] = \mathbf{f}_n^j(\mathbf{x}_n^j[k-1], \mathbf{u}[k-1]) + \mathbf{w}_n^j[k-1] \quad (3.8)$$

$$\mathbf{z}_n^j[k] = \mathbf{h}_n^j(\mathbf{x}_n^j[k], \mathbf{u}[k]) + \mathbf{v}_n^j[k] \quad (3.9)$$

where $\mathbf{x}_n^j[\cdot]$ is the state vector of target n when model $j(\in \mathbf{M})$ is in effect, $\mathbf{u}[\cdot]$ is the input vector(same for all targets), $\mathbf{f}_n^j(\cdot)$ and $\mathbf{h}_n^j(\cdot)$ are the

nonlinear and time invariant system structure. The process noise $\mathbf{w}_n^j[k]$ and the measurement noise $\mathbf{v}_n^j[k]$ are mutually uncorrelated zero-mean white Gaussian with covariance $\mathbf{W}_n^j[k]$ and $\mathbf{V}_n^j[k]$ respectively. And a (possibly state dependent) Markovian transition probability of the system mode index is given as follows.

$$\begin{aligned} \forall i, j \in \mathbf{M} \\ \Pr\{m_n[k+1] = j | m_n[k] = i\} = \Phi_{i,j}(\mathbf{x}_n[k]) \end{aligned} \quad (3.10)$$

where $m_n[k]$ is the system mode index of target n which is in effect at time step k .

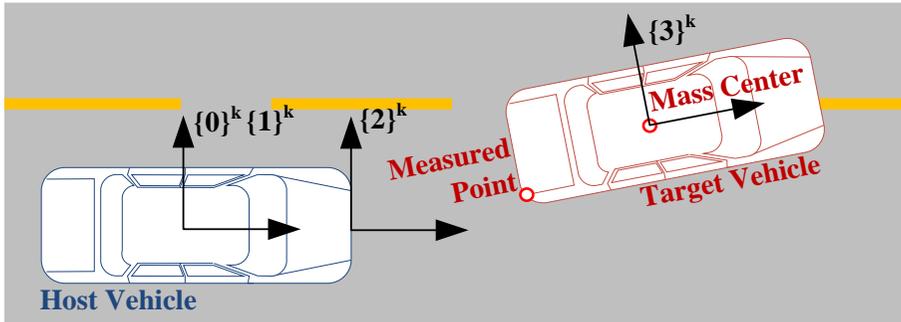
The following notations and definitions are used regarding the measurement. In general, at any time k , some measurements may be due to clutter and some due to the target, i.e. there can be more than a single measurement [Chen'01]. The measurement set (not yet validated or associated) is denoted as $\mathbf{Y}_k \equiv \{\mathbf{y}_1[k], \mathbf{y}_2[k], \dots, \mathbf{y}_{N_{measure}}[k]\}$ where $N_{measure}$ is the number of measurements. The set of validated measurement of target n at time k is denoted by

$$\mathbf{Z}_k \equiv \{\mathbf{z}_1[k], \mathbf{z}_2[k], \dots, \mathbf{z}_{N_{target}}[k]\} \quad (3.11)$$

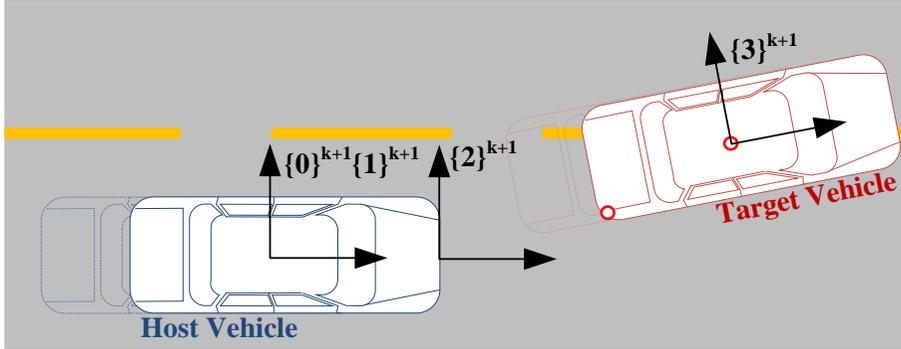
Therefore, the goals of the algorithm can be summarized as follows: 1) to find the set of validated measurement for each target and 2) to find the minimum variance estimated state and the associated error covariance matrix for each target.

4.1.2. Coordinate Systems

In this subsection, coordinate systems used to derive standard process and measurement models are presented. These coordinate systems are described in Figure 4.1. Frame $\{0\}$ is the ground based fixed coordinate system of which the position is same with the host vehicle, frame $\{1\}$ is the host vehicle's body-fixed moving coordinate system, frame $\{2\}$ is the sensor's body-fixed moving coordinate system, and frame $\{3\}$ is the target vehicle's body-fixed moving coordinate system. And superscript means a corresponding time step. The standard process model describes the predicted state of the target vehicle's mass center on the frame of $\{0\}^{k+1}$ based on the current state of the target vehicle's mass center and system input which are defined on the frame of $\{0\}^k$. And the standard measurement model describes the state of measured point such as position and longitudinal velocity on the sensor based moving frame $\{2\}$.



(a) at time k



(b) at time $k+1$

Figure 4.1. Four coordinate systems used to derive standard process model and standard measurement models

4.1.3. Standard Process Model

Various driving patterns of a vehicle including straight line, curve, cut-in/out, U-turn and interchange can be represented by a combination of a constant velocity rectilinear motion, a constant acceleration rectilinear motion, a constant angular velocity curvilinear motion and a constant angular acceleration curvilinear motion [Laugier'11].

To describe all these motions, the state vector and input vector of target n are defined as follows:

$$\mathbf{x}_n = [p_{n,x} \quad p_{n,y} \quad \theta_n \quad v_{n,x} \quad \gamma_n \quad a_{n,x} \quad \dot{\gamma}_n]^T \quad (3.12)$$

$$\mathbf{u} = [v_x \quad \gamma]^T \quad (3.13)$$

where subscript x and y denote x-axis and y-axis of each frame, subscript n denotes “of target n “, p denotes the relative position, θ denotes the

relative yaw angle, v denotes the velocity, γ denotes the yaw rate, a denotes the acceleration, and $\dot{\gamma}$ denotes the yaw acceleration. And variable without subscript n means “variable of the host vehicle”. Therefore v_x denotes the longitudinal velocity of host vehicle and γ denotes the yaw rate of host vehicle. Note that $p_{n,x}$, $p_{n,y}$ and θ_n are defined on the host vehicle’s body-fixed moving frame, $\{1\}$, and other elements are defined on the ground-based fixed frame, $\{0\}$. With no slip assumption ($v_{n,y} \approx 0$), the continuous process model can be formed as:

$$\begin{aligned}
\dot{\mathbf{x}}_n &= \mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \mathbf{q} \\
&= [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_3 \quad \mathbf{a}_4 \quad \mathbf{a}_5 \quad \mathbf{a}_6 \quad \mathbf{a}_7]^T + \mathbf{q} \\
\mathbf{a}_1 &= v_{n,x} \cos \theta_i - v_x + p_{n,y} \cdot \gamma & \mathbf{a}_2 &= v_{n,x} \sin \theta_i - p_{n,x} \cdot \gamma & (3.14) \\
\mathbf{a}_3 &= \gamma_n - \gamma & \mathbf{a}_4 &= a_{n,x} & \mathbf{a}_5 &= \dot{\gamma}_n \quad \mathbf{a}_6 = k_a \quad \mathbf{a}_7 = k_{\dot{\gamma}} \\
\mathbf{q} &\sim (\mathbf{0}, \mathbf{Q})
\end{aligned}$$

where \mathbf{a} is the nonlinear process vector equation, k_a is the decay-rate of the longitudinal acceleration, $k_{\dot{\gamma}}$ is the decay-rate of the yaw-acceleration, and \mathbf{q} is the continuous white process noise with covariance \mathbf{Q} . Above continuous process model can be discretized via Taylor methods as follows [Kazantzis'99]:

$$\begin{aligned}
\mathbf{x}_n[k+1] &= \mathbf{x}_n(t + \Delta T) \\
&= \mathbf{x}_n(t) + \frac{d}{dt} \mathbf{x}_n \cdot \Delta T + \frac{1}{2} \cdot \frac{d^2}{dt^2} \mathbf{x}_n \cdot \Delta T^2 + \mathbf{h.o.t.} \\
&\cong \mathbf{x}_n(t) + \{\mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \mathbf{q}\} \cdot \Delta T + \frac{1}{2} \cdot \frac{d}{dt} \{\mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \mathbf{q}\} \cdot \Delta T^2 \\
&= \left[\begin{array}{l} \mathbf{x}_n(t) + \{\mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \mathbf{q}\} \cdot \Delta T + \\ \frac{1}{2} \cdot \left\{ \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \dot{\mathbf{x}}_n + \frac{\partial}{\partial \mathbf{u}} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \dot{\mathbf{u}} + \dot{\mathbf{q}} \right\} \cdot \Delta T^2 \end{array} \right] \\
&= \left[\begin{array}{l} \mathbf{x}_n(t) + \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \Delta T + \\ \frac{1}{2} \cdot \left\{ \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \mathbf{a}(\mathbf{x}_n, \mathbf{u}) + \frac{\partial}{\partial \mathbf{u}} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \cdot \dot{\mathbf{u}} \right\} \cdot \Delta T^2 + \\ \left\{ \Delta T \cdot \mathbf{I} + \frac{\Delta T^2}{2} \cdot \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \right\} \mathbf{q} \end{array} \right] \\
&= \mathbf{f}(\mathbf{x}_n[k], \mathbf{u}[k]) + \mathbf{w}[k] \\
&= [\mathbf{f}_1 \quad \mathbf{f}_2 \quad \mathbf{f}_3 \quad \mathbf{f}_4 \quad \mathbf{f}_5 \quad \mathbf{f}_6 \quad \mathbf{f}_7]^T + \mathbf{w}[k]
\end{aligned} \tag{3.15}$$

where

$$\begin{aligned}
\mathbf{f}_1 &= - \left(\begin{array}{l} \frac{\gamma_{host} (p_{n,x} \gamma_{host} - v_n \sin(\theta_n))}{2} \\ - \frac{a_n \cos(\theta_n)}{2} + \frac{\gamma_n v_n \sin(\theta_n)}{2} \\ + \frac{a_{host}}{2} - \frac{p_{n,y} \dot{\gamma}_{host}}{2} \end{array} \right) \Delta T^2 + \left(\begin{array}{l} p_{n,y} \gamma_{host} - v_{host} \\ + v_n \cos(\theta_n) \end{array} \right) \Delta T + p_{n,x} \\
\mathbf{f}_2 &= - \left(\begin{array}{l} \frac{a_n \sin(\theta_n)}{2} + \frac{\gamma_n v_n \cos(\theta_n)}{2} + \frac{p_{n,x} \dot{\gamma}_{host}}{2} \\ - \frac{\gamma_{host} (p_{n,y} \gamma_{host} - v_{host} + v_n \cos(\theta_n))}{2} \end{array} \right) \Delta T^2 - \left(\begin{array}{l} p_{n,x} \gamma_{host} \\ - v_n \sin(\theta_n) \end{array} \right) \Delta T + p_{n,y} \\
\mathbf{f}_3 &= \frac{\dot{\gamma}_n}{2} \Delta T^2 + \gamma_n \Delta T + \theta_n & \mathbf{f}_4 &= - \frac{a_n k_a}{2} \Delta T^2 + a_n \Delta T + v_n \\
\mathbf{f}_5 &= - \frac{k_{\dot{\gamma}} \dot{\gamma}_n}{2} \Delta T^2 + \dot{\gamma}_n \Delta T + \gamma_n & \mathbf{f}_6 &= \frac{a_n}{2} (k_a^2 \Delta T^2 - 2k_a \Delta T + 2)
\end{aligned}$$

$$\mathbf{f}_\gamma = \frac{\dot{\gamma}_n}{2} (k_\gamma^2 \Delta T^2 - 2k_\gamma \Delta T + 2)$$

$$\mathbf{w}[k] \sim (0, \mathbf{W}[k])$$

$$\mathbf{W}[k] = E[\mathbf{w} \cdot \mathbf{w}^T]$$

$$\begin{aligned} &= E \left[\left[\left\{ \Delta T \cdot \mathbf{I} + \frac{\Delta T^2}{2} \cdot \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \right\} \mathbf{q} \right] [\dots]^T \right] \\ &= \left\{ \Delta T \cdot \mathbf{I} + \frac{\Delta T^2}{2} \cdot \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \right\} E[\mathbf{q} \cdot \mathbf{q}^T] \{\dots\}^T \\ &= \left\{ \Delta T \cdot \mathbf{I} + \frac{\Delta T^2}{2} \cdot \frac{\partial}{\partial \mathbf{x}_n} \mathbf{a}(\mathbf{x}_n, \mathbf{u}) \right\} \mathbf{Q} \{\dots\}^T \Bigg|_{\substack{\mathbf{x}_n = \mathbf{x}_n[k] \\ \mathbf{u} = \mathbf{u}[k]}} \end{aligned}$$

$$\dot{\mathbf{u}} = \frac{d}{dt} \begin{bmatrix} v_x & \gamma \end{bmatrix}^T = \begin{bmatrix} a_x & \dot{\gamma} \end{bmatrix}^T$$

where $\dot{\mathbf{u}}$ is time-derivative of input vector where the components' physical meanings are host vehicle's longitudinal acceleration and yaw-acceleration. As the result, $\mathbf{x}_n[k+1]$ describes the predicted state of the target vehicle's mass center on the frame of $\{0\}^{k+1}$. Note that various driving patterns of a vehicle can be represented by the above standard discretized process model by adjusting \mathbf{a} and \mathbf{Q} of continuous process model. The details about specific modes of process model will be discussed in following subsection, 4.2.1.

4.1.4. Standard Measurement Model

In an automotive target tracking, changes in the target aspect with respect to the radar can cause the apparent point of radar reflections (relative position seen by the antenna) to wander significantly [Skolnik'01]. To represent these characteristics, the measurement model can be elaborated by introducing new parameters to specify the sensor position and the measured point. These are

depicted in Figure 4.2. As the result, the measurement vector of target n is defined as follows.

$$\begin{aligned}\mathbf{z}_n[k] &= \mathbf{h}(\mathbf{x}_n[k], \mathbf{u}[k]) + \mathbf{v}_n[k] \\ &= [\mathbf{h}_{n1} \quad \mathbf{h}_{n2} \quad \mathbf{h}_{n3}]^T + \mathbf{v}_n[k] \\ \mathbf{v}_n[k] &\sim (0, \mathbf{V}_n[k])\end{aligned}\quad (3.16)$$

where

$$\begin{aligned}\mathbf{h}_{n1} &= p_{n,x} - s_x + b_{n,x} \cos \theta_n - b_{n,y} \sin \theta_n \\ \mathbf{h}_{n2} &= p_{n,y} - s_y + b_{n,x} \sin \theta_n + b_{n,y} \cos \theta_n \\ \mathbf{h}_{n3} &= v_{n,x} \cos \theta_n - v_x + p_{n,y} \cdot \gamma + (b_{n,x} \sin \theta_n + b_{n,y} \cos \theta_n)(\gamma - \gamma_n)\end{aligned}$$

where \mathbf{h} is nonlinear measurement equation which describe the state of measured point on the sensor based moving frame {2}. \mathbf{s} is sensor position vector defined on the host vehicle's body-fixed moving frame {1} and \mathbf{b}_n is measured point vector of target n defined on the target vehicle's body-fixed moving frame {3}. The first order approximation of the measurement error covariance has been presented in previous works as follows [Lerro'93].

$$\mathbf{V}_n = \begin{bmatrix} \frac{\sigma_r^2 - r_n^2 \sigma_\theta^2}{2} \begin{bmatrix} b_n + \cos 2\theta_n & \sin 2\theta_n \\ \sin 2\theta_n & b_n - \cos 2\theta_n \end{bmatrix} & \mathbf{0}_{2 \times 1} \\ \mathbf{0}_{1 \times 2} & \sigma_v^2 \end{bmatrix} \quad (3.17)$$

where

$$b_n = \frac{\sigma_r^2 + r_n^2 \sigma_\theta^2}{\sigma_r^2 - r_n^2 \sigma_\theta^2} \quad r_n = \sqrt{\mathbf{h}_{n1}^2 + \mathbf{h}_{n2}^2} \quad \theta_n = \tan^{-1} \left(\frac{\mathbf{h}_{n2}}{\mathbf{h}_{n1}} \right)$$

where r_n and θ_n are the range and azimuth measurements, respectively. σ_r , σ_θ and σ_v are the standard deviations of the range, the azimuth and the

relative velocity, respectively. The approximation is validated to be useful for practical parameters [Li'93]. The various measurement patterns due to the target aspect with respect to radar can be represented by the above standard measurement model by adjusting the measured point vector \mathbf{b}_n . The details about specific modes of measurement model will be discussed more fully in subsection, 4.2.2

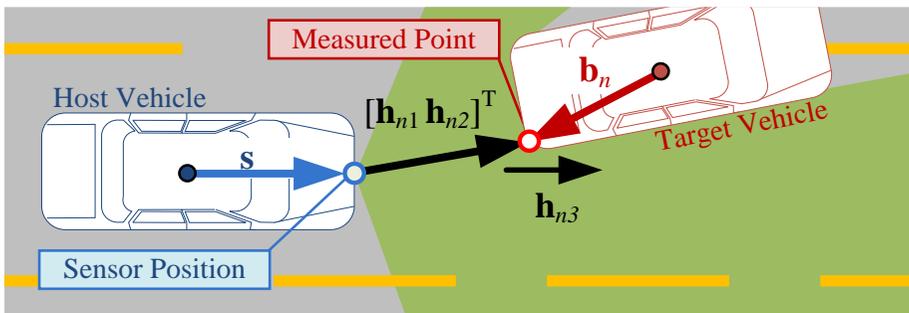


Figure 4.2. Parameters to derive the standard form of the refined measurement model

4.2. Selection of Multiple Model Set and Parameter

Design

To obtain the best estimate, the IMM algorithm has to be properly designed and tuned to meet the special requirements of the automotive application. And these requirements are fulfilled by means of 1) selection of vehicle motion models for all modes of driving, 2) selection of measurement models for all cases of measurements, and 3) determination of the parameters of the underlying Markov chain, that is, the transition probabilities [Li'93]. Therefore, in this section, after analyzing the driving patterns and automotive radar characteristics, four specific modes of process model, twelve specific modes of measurement model, and event dependent transition probability matrix are derived and introduced.

4.2.1. Set of Multiple Process Model

Various driving patterns considered in this section are depicted in Figure 4.3. To describe these various driving patterns of a vehicle, four specific modes of process model will be discussed in this section. As aforementioned, four motions include the constant velocity rectilinear motion, the constant acceleration rectilinear motion, the constant angular velocity curvilinear motion and the constant angular acceleration curvilinear motion are considered. Assuming that accelerations in the steady state are quite small, linear accelerations or decelerations can be reasonably well covered by process noises

with the constant velocity model. With these assumptions, the Constant Velocity Rectilinear (CVR) motion can be modeled by:

$$\begin{aligned}\dot{\mathbf{x}}_n &= \mathbf{a}(x_n, u) + \mathbf{q} \\ &= [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_{3,CVR} \quad 0 \quad 0 \quad 0 \quad 0]^T + \mathbf{q} \\ \mathbf{q} &\sim (\mathbf{0}, \mathbf{Q}_{CVR})\end{aligned}\tag{3.18}$$

where

$$\begin{aligned}\mathbf{a}_{3,CVR} &= -\gamma \\ \mathbf{Q}_{CVR} &= \text{diag}([0 \quad 0 \quad 0 \quad \mathbf{Q}_{44} \quad 0 \quad 0 \quad 0])\end{aligned}$$

where \mathbf{Q} is the appropriate covariance which can represent the zero-mean Gaussian white noise assumed accelerations. In the same way, the Constant Acceleration Rectilinear (CAR) motion can be formulated as follows:

$$\begin{aligned}\dot{\mathbf{x}}_n &= [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_{3,CVR} \quad \mathbf{a}_{4,CAR} \quad 0 \quad 0 \quad 0]^T + \mathbf{q} \\ \mathbf{q} &\sim (\mathbf{0}, \mathbf{Q}_{CAR})\end{aligned}\tag{3.19}$$

where

$$\begin{aligned}\mathbf{a}_{4,CAR} &= a_{n,x} \\ \mathbf{Q}_{CAR} &= \text{diag}([0 \quad 0 \quad 0 \quad 0 \quad 0 \quad \mathbf{Q}_{66} \quad 0])\end{aligned}$$

The constant angular velocity curvilinear motion is a turning motion with a constant yaw rate along a road of constant radius of curvature. With a zero-mean Gaussian white noise to the derivative of yaw rate, the Constant Velocity Curvilinear (CVC) motion can be modeled by:

$$\begin{aligned}\dot{\mathbf{x}}_n &= [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_{3,CVC} \quad 0 \quad 0 \quad 0 \quad 0]^T + \mathbf{q} \\ \mathbf{q} &\sim (\mathbf{0}, \mathbf{Q}_{CVC})\end{aligned}\tag{3.20}$$

where

$$\begin{aligned}\mathbf{a}_{3,CVC} &= \gamma_n - \gamma \\ \mathbf{Q}_{CVC} &= \text{diag}([0 \ 0 \ 0 \ 0 \ \mathbf{Q}_{55} \ 0 \ 0])\end{aligned}$$

With some modifications to (13), the Constant Acceleration Curvilinear (CAC) motion can be derived by:

$$\begin{aligned}\dot{\mathbf{x}}_n &= [\mathbf{a}_1 \ \mathbf{a}_2 \ \mathbf{a}_{3,CVC} \ 0 \ \mathbf{a}_{5,CAC} \ 0 \ 0]^T + \mathbf{q} \\ \mathbf{q} &\sim (\mathbf{0}, \mathbf{Q}_{CAC})\end{aligned}\tag{3.21}$$

where

$$\begin{aligned}\mathbf{a}_{5,CAC} &= \dot{\gamma}_n \\ \mathbf{Q}_{CAC} &= \text{diag}([0 \ 0 \ 0 \ 0 \ 0 \ 0 \ \mathbf{Q}_{77}])\end{aligned}$$

Evidently, the discretized process model derived from (10)-(13) are special forms of (7) and can be summarized by (1). In addition, it is reasonable to assume that the transition between the driving modes of a target vehicle has the Markovian probability governed by (3). Consequently, the process model can be suitable described in the framework of the stochastic hybrid systems.

4.2.2. Set of Multiple Measurement Model

As shown in Figure 4.4, by means of relative position and orientation between host and target vehicle, the configuration of measurement can be altered. Ideally, the detecting point should be located on the four sides or four edges of the target vehicle. As an example, for the case of the cutting in vehicle, a rear right edge is the most likely measurement model while lane changing. However, after cutting-in, the most likely measured point is transited to a center

of rear side of the target vehicle.

As aforementioned, to describe these various measurement patterns, several specific modes can be derived by adjusting \mathbf{b}_n in standard measurement model. From the viewpoint of detecting position defined on the target-body-fixed frame, a specific mode set (depicted in Figure 4.4) can be utilized to describe the variations of the measurement patterns. We can see that the proposed mode set include twelve points around the boundary of the target vehicle and all case examples in Figure 4.5 can be represented well by the proposed mode set. Consequently, derived set of measurement models can be suitable described by the stochastic hybrid systems.

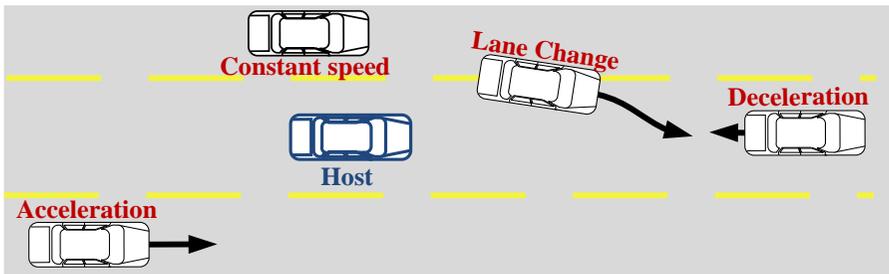


Figure 4.3. Various driving patterns of multi target vehicles.

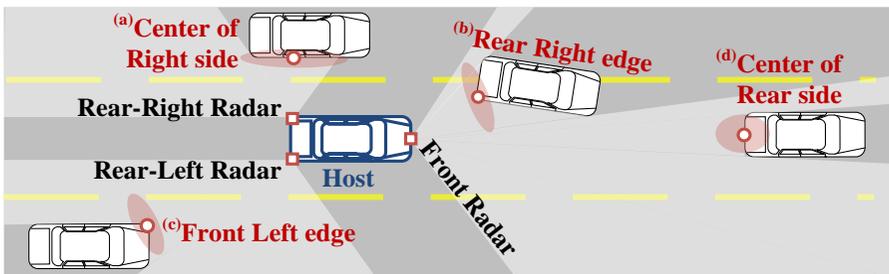


Figure 4.4. Various measurement patterns due to the target aspect with respect

to the host vehicle's radar sensor.

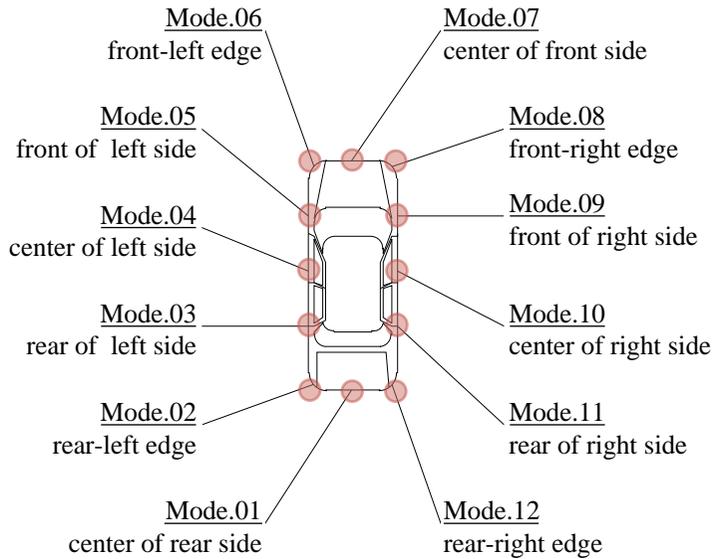


Figure 4.5. Twelve modes of detected point of measurement model to describe various measurement patterns.

4.2.3. Event Dependent Transition Probability Matrix

In a conventional IMM estimator the Markov chain of the mode states is described by a single mode transition matrix. However, unlike the conventional system, mode transition process of an automotive radar system cannot be modeled by a single mode transition probability matrix. As can be seen in Figure 4.6, transitions to invalid modes are infeasible since the radar cannot penetrate the vehicle's body. And a subset of invalid modes can be different for different situations. As an example, mode transition from the front-right edge to rear-left edge is impossible in event.1 while it is possible in event.8. Therefore, we can conclude that the mode transition probabilities of the

automotive radar system are target aspect dependent and multi transition matrices should be derived to describe all cases of the mode jump process.

A transition probability matrix of the proposed algorithm can be decomposed into two sub-matrices; one is the sub-transition probability matrix related to multiple process models and the other matrix is about multiple measurement models. Relation of three matrices is given as:

$$\Phi = \begin{bmatrix} \Phi_{1 \times 1}^{process} \cdot \Phi^{measure} & \Phi_{1 \times 2}^{process} \cdot \Phi^{measure} & \dots & \Phi_{1 \times 1}^{process} \cdot \Phi^{measure} \\ \Phi_{2 \times 1}^{process} \cdot \Phi^{measure} & \Phi_{2 \times 2}^{process} \cdot \Phi^{measure} & \dots & \Phi_{2 \times N_{mode}^{process}} \cdot \Phi^{measure} \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{N_{mode}^{process} \times 1}^{process} \cdot \Phi^{measure} & \Phi_{N_{mode}^{process} \times 2}^{process} \cdot \Phi^{measure} & \dots & \Phi_{N_{mode}^{process} \times N_{mode}^{process}} \cdot \Phi^{measure} \end{bmatrix} \quad (3.22)$$

where $\Phi^{process}$ is the matrix defining a transition probability between process mode indices, $N_{mode}^{process}$ is the number of process model set, $\Phi^{measure}$ defines a transition probability between measurement mode indices. $\Phi^{process}$ can be expressed in a single matrix, but this is impossible in the case of $\Phi^{measure}$ as aforementioned. To solve this problem, different measurement mode transition matrices are used depending on the target aspect event. All eight possible events are defined in Figure 4.7. From the work of Zhang and Bar-shalom [Zhang'11], an approach using state-dependent mode transition probabilities has been presented to track move-stop-move targets. And in our research, this approach is extended to eight stages and applied with some modifications. Target index is omitted unless otherwise required for clarity. First, measurement mode transition probability matrix with no event (physically not allowed) is designed in Gaussian distribution manner as follows:

$\forall i, j \in \mathbf{M}$

$$\Phi_{i,j}^{measure,0} = \frac{1}{\sqrt{2\pi\sigma_d}} \exp\left[-\frac{1}{2}\left(\frac{d_{i,j}}{\sigma_d}\right)^2\right] \quad (3.23)$$

where $d_{i,j}$ is the distance travelled from i to j along the vehicle's body line. And with this raw transition probability, a transition probability conditioned on a specific event E is given by:

$$\Phi^{measure,E} = \Phi^{measure,0} \cdot \Theta^E = \left[\Phi_{i,j}^{measure,0} \right] \cdot \Theta^E \quad (3.24)$$

where Θ^E is an event dependent diagonal matrix of which elements are defines as follows:

$$\Theta_{i,j}^E = \begin{cases} 1 & i = j \in \mathbf{M}_{valid}^{measure,E} \\ 0 & i = j \in \mathbf{M}_{invalid}^{measure,E} \\ 0 & i \neq j \end{cases} \quad (3.25)$$

where $\mathbf{M}_{valid}^{measure,E}$ is the subset of valid measurement modes and $\mathbf{M}_{invalid}^{measure,E}$ is the subset of invalid measurement modes on an event E . Consequently, the conditional transition probability matrix has zero values as entries of columns of invalid modes, which means that under that given event, transitions to invalid modes are not allowed.

Finally, the actual transition probability matrix of the system at time step k is calculated as a combined form which can be written as follows:

$\forall i \in \mathbf{E}$

$$\Phi^{measure} = \sum_i \Pr(E[k]=i | \mathbf{z}[0], \dots, \mathbf{z}[k]) \cdot \Phi^{measure,i} \quad (3.26)$$

Where $\mathbf{E} = \{i | i = 1, \dots, 8\}$ is a set of eight event indexes and $\Pr(E[k] = i | \mathbf{z}[0], \dots, \mathbf{z}[k])$ is an event probability which means the probability of the target being in a specific event i at time step k conditioned on the cumulative set of measurements up to time k . The event probability can be calculated directly as:

$$\forall i \in \mathbf{E} \quad \Pr(E[k] = i | \mathbf{z}[0], \dots, \mathbf{z}[k]) = \int \int_{A_i} f_{\tilde{\mathbf{s}} | \mathbf{z}[0], \dots, \mathbf{z}[k]}(\tilde{\mathbf{s}}[k]) d\tilde{\mathbf{s}} \quad (3.27)$$

where $\tilde{\mathbf{s}}$ denotes a sensor position vector transformed in target-body-fixed coordinate, $f_{\tilde{\mathbf{s}} | \mathbf{z}[0], \dots, \mathbf{z}[k]}(\cdot)$ is a conditional probability density function associated with $\tilde{\mathbf{s}}$ conditioned on the cumulative set of measurements, and A_i is an associated area with event i which is depicted in Figure 4.7.

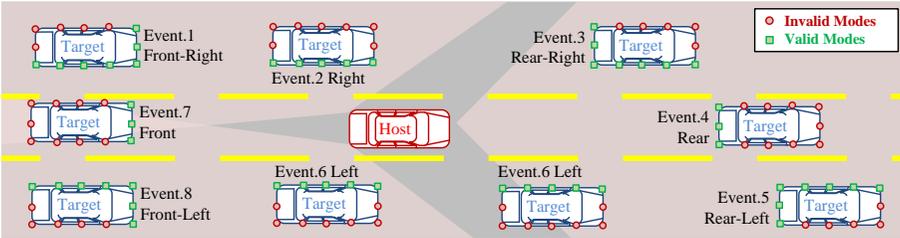


Figure 4.6. Set of measurement model, their two subsets classified by validity (of each target vehicle in multi traffics), and eight events of target aspect.

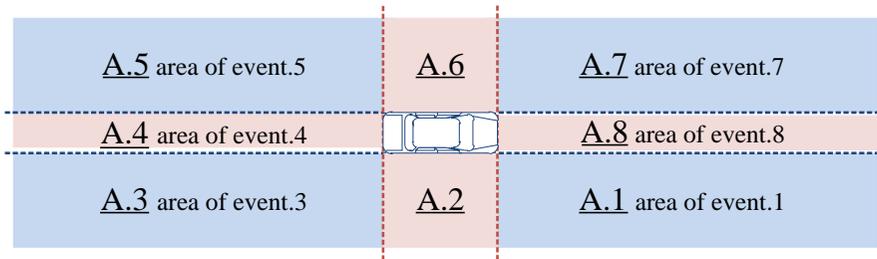


Figure 4.7. Associated area of possible target aspect event on target-centered coordinate

4.3. IMM/EKF Multi Target State estimation

In this section, an IMM/EKF based filtering algorithm for tracking multiple target vehicles is proposed. The overall structure of the algorithm is shown in

Figure 4.8. The host vehicle filter estimates the host vehicle's longitudinal speed and yaw rate which take roll of inputs. And based on IMM approach, the organized measurement set is filtered and the best estimates of multi target vehicles are obtained. Each component of the algorithm is presented in following sections. In the part of track management, the raw measurement cloud is validated and associated for each target by using most likely validation law.

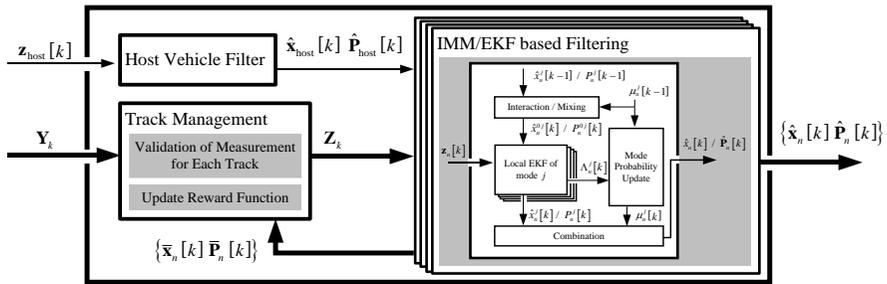


Figure 4.8. Architecture of IMM/EKF multi target state estimation algorithm

4.3.1. Host Vehicle Filter

The Kalman filter is used to estimate host vehicle states such as longitudinal velocity and yaw rate. These values can be estimated from the vehicle sensor signals such as wheel speed, acceleration, and yaw rate of which each time derivative is considered as the Gaussian white noise. The state of the host vehicle filter and its time invariant linear system structures can be defined as follows.

$$\begin{aligned}\dot{\mathbf{x}}_{host} &= \mathbf{A}_{host} \cdot \mathbf{x}_{host} + \mathbf{q}_{host} \\ &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_x & a_x & \gamma \end{bmatrix}^T + \mathbf{q}_{host} \quad (3.29)\end{aligned}$$

$$\mathbf{q}_{host} \sim (\mathbf{0}, \mathbf{Q}_{host})$$

$$\begin{aligned}\mathbf{z}_{host} [k] &= \mathbf{H}_{host} \mathbf{x}_{host} [k] + \mathbf{v}_{host} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x [k] & a_x [k] & \gamma [k] \end{bmatrix}^T + \mathbf{v}_{host} \quad (3.30)\end{aligned} \quad (3.31)$$

$$\mathbf{v}_{host} \sim (\mathbf{0}, \mathbf{V}_{host})$$

The given process model can be discretized as follows:

$$\begin{aligned}\mathbf{x}_{host} [k] &\cong \mathbf{x}_{host} [k-1] + \Delta t \cdot \mathbf{q}_{host} \\ &= \mathbf{F}_{host} \cdot \mathbf{x}_{host} [k-1] + \mathbf{w}_{host} \\ \mathbf{w}_{host} &\sim (\mathbf{0}, \Delta t^2 \cdot \mathbf{Q}_{host}) \\ &\sim (\mathbf{0}, \mathbf{W}_{host})\end{aligned} \quad (3.32)$$

where Δt is the sampling time (0.1s in this study). With above process and measurement model, vehicle states are recursively estimated by using the Kalman filter which is a sequence of time and measurement update steps as following specific equations.

Time update

$$\begin{aligned}\bar{\mathbf{x}}_{host}[k] &= \mathbf{F}_{host} \cdot \hat{\mathbf{x}}_{host}[k-1] \\ \bar{\mathbf{P}}_{host}[k] &= \mathbf{F}_{host} \cdot \hat{\mathbf{P}}_{host}[k-1] \cdot \mathbf{F}_{host}^T + \mathbf{W}_{host}\end{aligned}\quad (3.33)$$

Measurement update

$$\begin{aligned}\mathbf{K}_{host}[k] &= \bar{\mathbf{P}}_{host}[k] \cdot \mathbf{H}_{host}^T \cdot (\mathbf{H}_{host} \cdot \bar{\mathbf{P}}_{host}[k] \cdot \mathbf{H}_{host}^T + \mathbf{V}_{host})^{-1} \\ \hat{\mathbf{P}}_{host}[k] &= (\mathbf{I} - \mathbf{K}_{host}[k] \mathbf{H}_{host}) \cdot \bar{\mathbf{P}}_{host}[k] \\ \hat{\mathbf{x}}_{host}[k] &= \bar{\mathbf{x}}_{host}[k] + \mathbf{K}_{host}[k] \cdot (\mathbf{z}_{host}[k] - \mathbf{H}_{host} \cdot \bar{\mathbf{x}}_{host}[k]) \\ \hat{\mathbf{u}}[k] &= \left[(\hat{\mathbf{x}}_{host}[k])_1 \quad (\hat{\mathbf{x}}_{host}[k])_2 \right]^T\end{aligned}\quad (3.34)$$

4.3.2. IMM/EKF based Filtering

For the each track, after the validated measurement is received, the multiple EKFs are used as the each mode-matched local filter and combined through the IMM method to obtain a refined state estimate. A complete cycle of the IMM/EKF based filtering algorithm can be summarized as follows. Time index is omitted unless otherwise required for clarity.

4.3.2.1. Interaction ($\forall i, j \in \mathbf{M} / \forall n \in \mathbf{T}$):

Starting with the N_{mode} weights $\mu_n^i[k-1]$, the N_{mode} means $\hat{\mathbf{x}}_n^i[k-1]$ and the N_{mode} associated covariance $\hat{\mathbf{P}}_n^i[k-1]$, compute the mixed initial condition for filter matched to mode state j . Note that, when the modes used have different dimension state vectors, an augmentation with zeros the lower dimension state estimate may lead to a bias toward zero for the state components of the larger state vector. In the previous work, a simple procedure

to avoid this “biasing” is presented, together with a suitable augmentation of the covariance of the smaller state that yields an unbiased and consistent mixing [Yuan'12].

Predicted mode probability:

$$\begin{aligned}\bar{\mu}_n^j &\triangleq \Pr\{m_n[k] = j | \mathbf{z}_n[k-1]\} \\ &= \sum_i \{\Phi_{ij}[k-1] \mu_n^i[k-1]\}\end{aligned}\quad (3.35)$$

Mixing probability:

$$\begin{aligned}\mu_n^{ij} &\triangleq \Pr\{m_n[k-1] = i | m_n[k] = j, \mathbf{z}_n[k-1]\} \\ &= \frac{\Phi_{ij}[k-1] \mu_n^i[k-1]}{\bar{\mu}_n^j}\end{aligned}\quad (3.36)$$

Mixed condition:

$$\hat{\mathbf{x}}_n^{0j}[k-1] = \sum_i \hat{\mathbf{x}}_n^i[k-1] \mu_n^{ij} \quad (3.37)$$

$$\hat{\mathbf{P}}_n^{0j}[k-1] = \sum_i \left[\hat{\mathbf{P}}_n^i[k-1] + \begin{Bmatrix} (\hat{\mathbf{x}}_n^i[k-1] - \hat{\mathbf{x}}_n^{0j}[k-1]) \\ \times (\hat{\mathbf{x}}_n^i[k-1] - \hat{\mathbf{x}}_n^{0j}[k-1])^T \end{Bmatrix} \right] \mu_n^{ij} \quad (3.38)$$

4.3.2.2. Extended Kalman Filtering:

Each of the N_{mode} pairs weights $\hat{\mathbf{x}}_n^{0j}[k-1]$, $\hat{\mathbf{P}}_n^{0j}[k-1]$ is used as input to an extended Kalman filter matched to mode state j .

Time update:

$$\mathbf{F}_n^j[k-1] = \left. \frac{\partial \mathbf{f}_n^j}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_n^{0j}[k-1], \hat{\mathbf{u}}[k-1]} \quad (3.39)$$

$$\bar{\mathbf{x}}_n^j[k] = \mathbf{f}_n^j(\hat{\mathbf{x}}_n^{0j}[k-1], \hat{\mathbf{u}}[k-1]) \quad (3.40)$$

$$\bar{\mathbf{P}}_n^j[k] = \mathbf{F}_n^j[k-1] \cdot \hat{\mathbf{P}}_n^{0j} \cdot [k-1] \mathbf{F}_n^j[k-1]^T + \mathbf{W}_n^j[k-1] \quad (3.41)$$

Filter gain:

$$\mathbf{H}_n^j[k] = \left. \frac{\partial \mathbf{h}_n^j}{\partial \mathbf{x}} \right|_{\bar{\mathbf{x}}_n^j[k], \hat{\mathbf{u}}[k]} \quad (3.42)$$

$$\mathbf{S}_n^j[k] = \mathbf{H}_n^j[k] \cdot \bar{\mathbf{P}}_n^j[k] \cdot \mathbf{H}_n^j[k]^T + \mathbf{V}_n^j[k] \mathbf{K}_n^j[k] = \bar{\mathbf{P}}_n^j[k] \cdot \mathbf{H}_n^j[k]^T \cdot \mathbf{S}_n^j[k]^{-1} \quad (3.43)$$

Innovation:

$$\mathbf{r}_n^j[k] = \mathbf{z}_n[k] - \mathbf{h}_n^j(\bar{\mathbf{x}}_n^j[k], \hat{\mathbf{u}}[k]) \quad (3.44)$$

Measurement update:

$$\hat{\mathbf{x}}_n^j[k] = \bar{\mathbf{x}}_n^j[k] + \mathbf{K}_n^j[k] \{\mathbf{r}_n^j[k]\} \quad (3.45)$$

$$\hat{\mathbf{P}}_n^j[k] = (\mathbf{I} - \mathbf{K}_n^j[k] \cdot \mathbf{H}_n^j[k]) \bar{\mathbf{P}}_n^j[k] \quad (3.46)$$

4.3.2.3. Mode Probability Update:

The N_{mode} weights $\bar{\mu}_n^i$ are updated from the innovations of the N_{mode} extended Kalman filters.

Likelihood function:

$$\Lambda_n^j = |2\pi \mathbf{S}_n^j|^{-1/2} \cdot \exp \left[-\frac{1}{2} \cdot (\mathbf{r}_n^j[k])^T \cdot (\mathbf{S}_n^j)^{-1} \cdot (\mathbf{r}_n^j[k]) \right] \quad (3.47)$$

Mode probability:

$$\mu_n^j = \frac{\bar{\mu}_n^j \cdot \Lambda_n^j}{\sum_i \bar{\mu}_n^i \cdot \Lambda_n^i} \quad (3.48)$$

Combination:

for output purpose only, $\hat{\mathbf{x}}_n[k]$ and $\hat{\mathbf{P}}_n[k]$ are computed according to

Combined condition:

$$\hat{\mathbf{x}}_n[k] = \sum_j \hat{\mathbf{x}}_n^j[k] \mu_n^j \quad (3.49)$$

$$\hat{\mathbf{P}}_n[k] = \sum_j \left\{ \hat{\mathbf{P}}_n^j[k] + (\hat{\mathbf{x}}_n^j[k] - \hat{\mathbf{x}}_n[k]) (\hat{\mathbf{x}}_n^j[k] - \hat{\mathbf{x}}_n[k])^T \right\} \mu_n^j \quad (3.50)$$

4.3.3. Track Management

In track management, there are three processes required during each time step: track update, track deletion and track creation. In this study, each track has a rewarding counter that is rewarded when the track is updated with validated measurement and decreased when the measurement is missed. In the part of track update, the measurements are associated with the existing tracks. Following (28), the most likely candidate measurement for target n at time k can be defined as follows:

$$\mathbf{z}_n^{candi}[k] = \min_{\mathbf{y} \in \mathbf{Y}_k} \left[(\mathbf{y} - \hat{\mathbf{z}}_n[k])^T \cdot (\mathbf{S}_n[k])^{-1} \cdot (\mathbf{y} - \hat{\mathbf{z}}_n[k]) \right] \quad (3.51)$$

where

$$\hat{\mathbf{z}}_n^j[k] = \mathbf{h}_n^j(\bar{\mathbf{x}}_n^j[k], \hat{\mathbf{u}}[k])$$

$$\hat{\mathbf{z}}_n[k] = \sum_j \hat{\mathbf{z}}_n^j[k] \bar{\mu}_n^j$$

$$\mathbf{S}_n[k] = \sum_j \left\{ \mathbf{S}_n^j[k] + (\bar{z}_n[k] - \bar{z}_n^j[k])(\bar{z}_n[k] - \bar{z}_n^j[k])^T \right\} \bar{\mu}_n^j$$

The candidate measurement $\mathbf{z}_n^{candi}[k]$ is validated as $\mathbf{z}_n[k]$ if and only if

$$(\mathbf{z}_n^{candi}[k] - \hat{\mathbf{z}}_n[k])^T \cdot (\mathbf{S}_n[k])^{-1} \cdot (\mathbf{z}_n^{candi}[k] - \hat{\mathbf{z}}_n[k]) \leq r_{th} \quad (3.52)$$

where r_{th} is the gate threshold corresponding to the gate probability P_G . Note that all targets share a common threshold. If the track has no validated measurement, the rewarding counter is decreased and the time-prediction is performed as the current estimate (which means a zero-gain EKF). If a rewarding counter drops below a certain threshold, the track is deleted. And newly arriving measurements generate new tracks. The initial state of new track is decided by a model selection strategy. And the new track is assumed to have the same speed with the host vehicle, zero yaw-rate, zero acceleration, and zero yaw acceleration. Additionally, some large value of error covariance is given for the newly generated target.

4.4. Vehicle Tests based Performance Evaluation

The developed IMM/EKF based multi-target tracking algorithm is implemented and verified via vehicle tests in real road. The test scenario is an overtaken situation by a target vehicle to validate all-around target states estimation performance. To investigate the performance enhancement, comparison with model-switching algorithm which is a simple approach to handle the multi-model problem has been conducted. Additionally, the experimental results dealing with multi-target scenarios on the real road are presented to explore the multi-target tracking performance of the proposed algorithm.

As aforementioned, in target tracking research area, considerable IMM approaches are limited to various modes of target motion. And the main novelties of this paper are that process model and measurement model have been elaborated, and multiple measurement models have been derived. Therefore, twelve measurement model set has been implemented and investigated for the proposed IMM/EKF algorithm (with single standard process model).

4.4.1. Configuration of Vehicle Tests

Figure 4.9 shows a host vehicle and a target vehicle which are used in this study. The host vehicle is equipped with a front-radar, two rear-side-radars, wheel speed sensors, and a 6DOF sensor, etc. And a RT-Range is used for an accurate measurement of the relative motion between the host and target vehicle.

The RT-Range is a precision tool for evaluating and verifying the accuracy of radar and tracking algorithm. Since this paper focuses on the target tracking algorithm, additional details about the test vehicles' system configurations have been omitted.

The test scenario is an overtaken situation by a target vehicle as shown in Figure 4.10. The host vehicle (denoted by green vehicle) drives in a straight lane with an initial speed of 50 km/h and a slight deceleration and the target vehicle (denoted by red vehicle) started its overtaking at 7s and finished at 16s while driving with an initial speed of 55kph with a slight acceleration. Typical characteristics of the automotive radar in the given situation are shown in Figure 4.11. We can see that, from 0-12s, the target vehicle is detected by rear radar and left side of the target is in effect during that period. A period from 12-14s is blind area where there is no measurement for the target vehicle by any radar. From 14-20s, the target vehicle is detected by front radar and rear side of the target is in effect during that period.

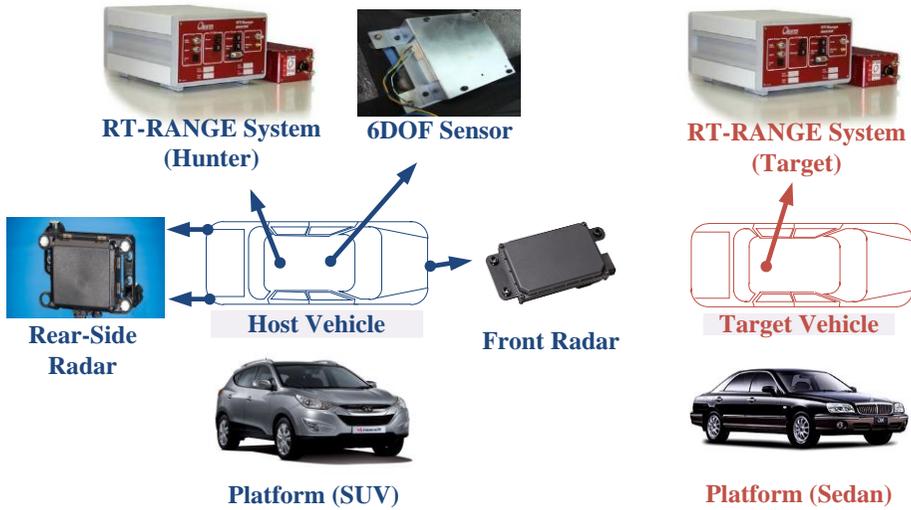


Figure 4.9. Configurations of the host vehicle and the target vehicle

4.4.2. Implementation and Evaluation

As aforementioned, in target tracking research area, considerable IMM approaches are limited to various modes of target motion. And the main novelties of this paper are that process model and measurement model have been elaborated, and multiple measurement models have been derived. Therefore, twelve measurement model set has been implemented and investigated for the proposed IMM/EKF algorithm (with single standard process model). Target states estimation results in the given situation are summarized in Figure 4.12, Figure 4.13, and Figure 4.14.

Figure 4.12 gives three case examples of mode probability update and target state estimation. The host vehicle is depicted by green vehicle, the actual target vehicle by red vehicle, the estimated target vehicle by gray vehicle and the associated measurement by red square. As can be seen in the figure, in all cases,

mode estimation results are quite reasonable and estimated target positions maintain good performance compared to the actual values even in no measurement situation (case.2).

Figure 4.13 shows the results of mode probability adaptation along the time axis. The update results represent the actual mode transition which can be inferred from Figure 3.12. From 0-10 sec, which is a period that the target vehicle is detected by rear radar and front-left edge of the target is mainly in effect, we can see that the probability of mode.6 ranked highest. In a period from 10-12 sec which is blind area, although there are no measurements, the most likely mode has been changed from mode.6 to mode.2 along the target vehicle's left side by the interaction/mixing step of the algorithm. This transition in the blind area is quite reasonable in the view of changes of the target aspect. And finally, from 12-20 sec, which is the period that the target vehicle is detected by front radar, mode.1 (center of rear side), mode.12 (rear-right edge), and mode.2 (rear-left edge) are evaluated as the highly likely modes during the period. Moreover, as the target vehicle becomes the in-lane preceding vehicle, we can see that mode.1 gets stronger mode probability which denoted by red-solid line.

Figure 4.14 presents the estimation results for the target states and their error distribution. From the figure, we can see that the overall trends of each estimated state are similar to actual values and their error distributions can be well approximated by normal distribution. The statistics of estimation error of IMM/EKF based filtering algorithm are given in Table 1.

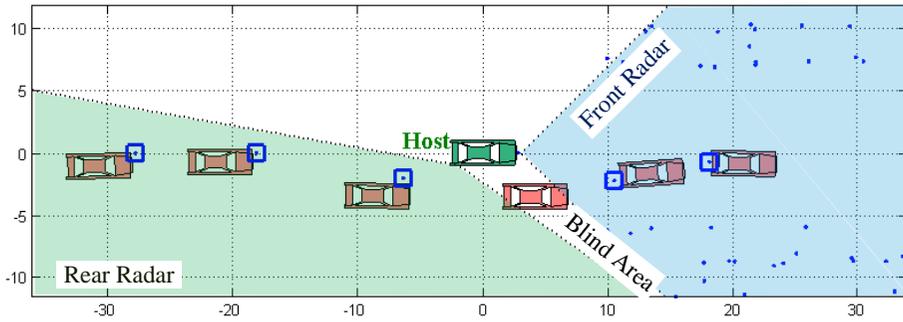


Figure 4.10. Test scenario: overtaking target vehicle

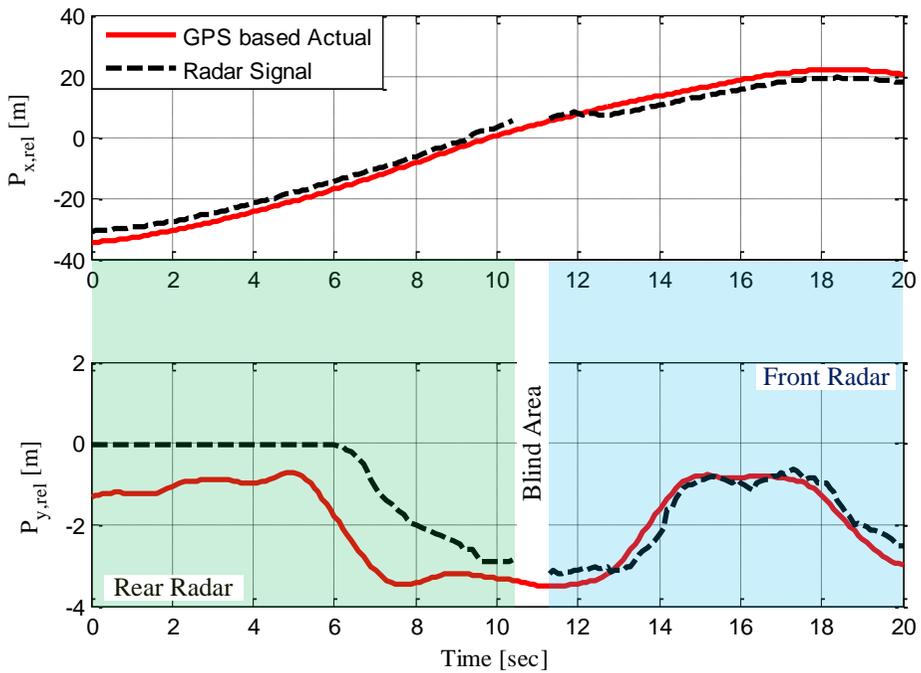


Figure 4.11. Test data: actual position (from RT-Range) and the associated radar signals of target vehicle

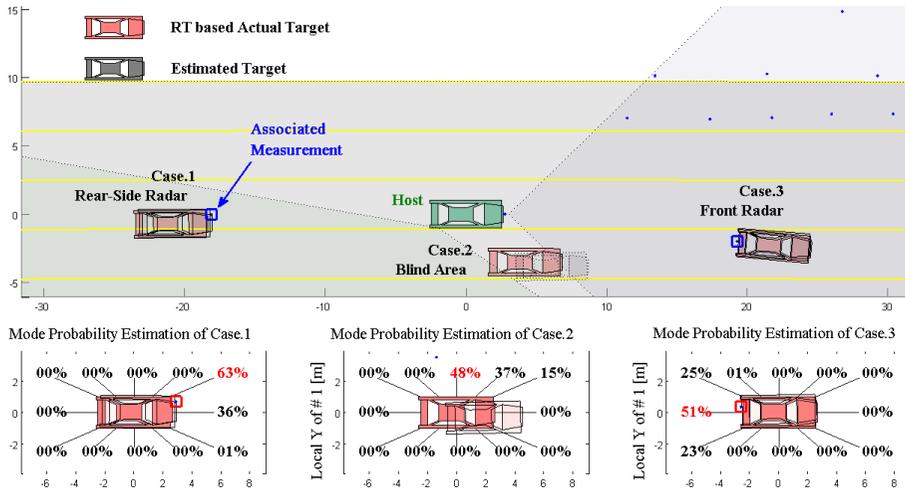


Figure 4.12. Three case examples of mode probability update

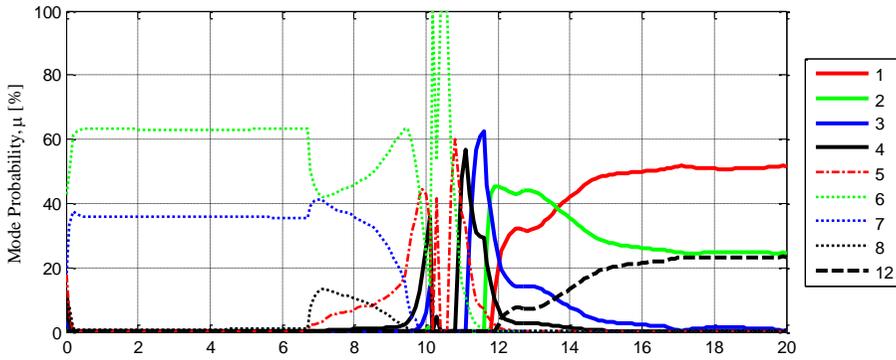


Figure 4.13. Mode probability update results

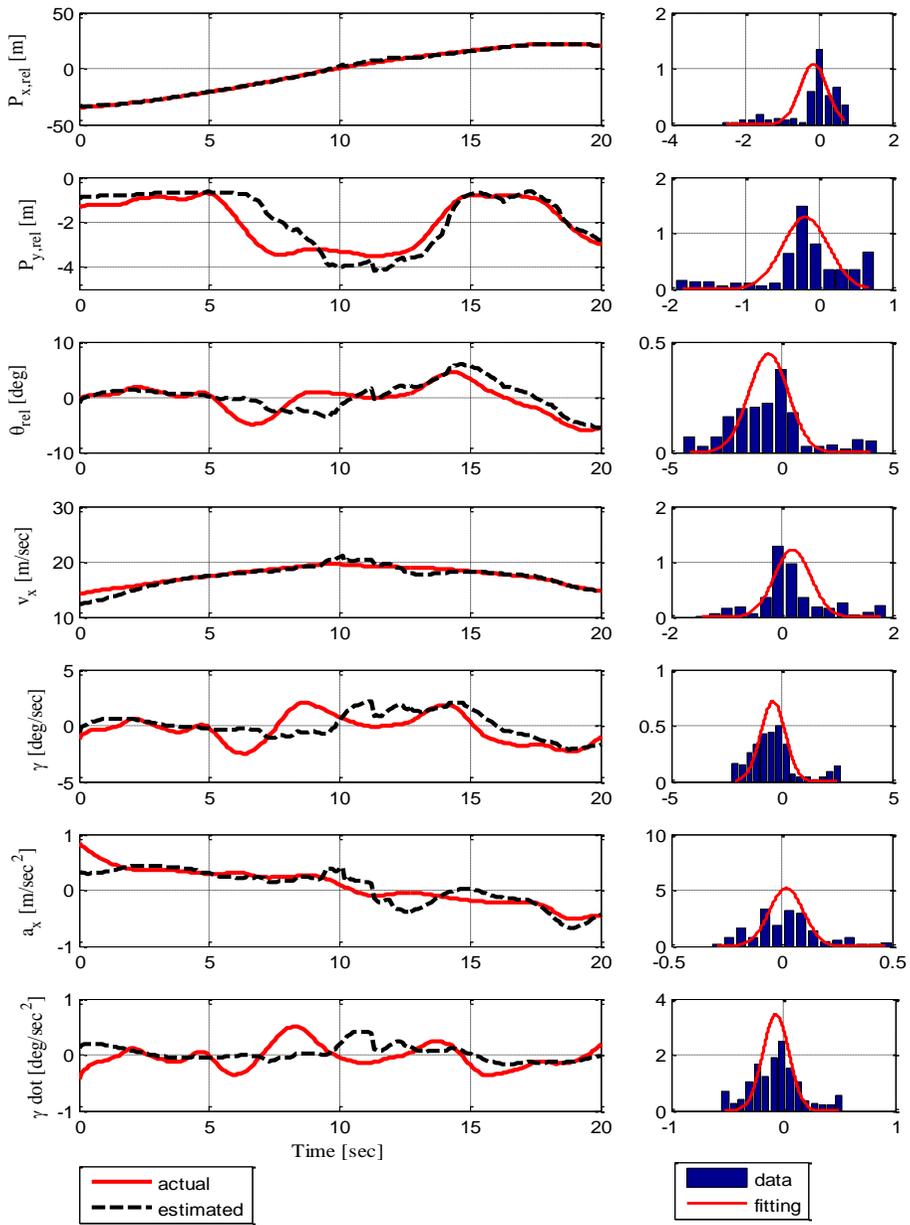


Figure 4.14. Estimated target states and the distributions of the estimation errors

4.4.3. Comparison with Model-switching/EKF

To examine the performance enhancement of the proposed IMM/EKF algorithm, performance comparison with a model-switching/EKF algorithm has been conducted. Model-switching/EKF is the simple approach to handle the multi-model problem by selecting one model from the multi-model set according to some predefined model selection criterion. Architecture and model selection strategy of model-switching/EKF algorithm is summarized in Figure 4.15. The strategy of model selection is grounded on the assumption that radar gives the information of the closest point on the boundary of the target vehicle. As an example, if the measurement is detected from the front-right side, the second measurement model is selected as the model in effect and implemented to the algorithm.

Statistical analysis of estimation errors of model-switching/EKF has been conducted following the same procedure under the same situation of 4.4.2. The statistics are given in Table 1. Over all, the proposed algorithm performed considerably better in all state estimations than the model-switching algorithm. As the standard deviation is the accuracy specifications of each algorithm, the proposed algorithm is approximately three times more accurate than the model-switching/EKF.

Finally, computing complexity of these two approaches have been compared. An average consumption time of IMM/EKF algorithm is about 4.9806ms and mode-switching/EKF is about 1.6787ms. When considering that IMM uses twelve filters in parallel, this three times more complexity is relatively good performance. Moreover, although IMM based algorithm has high complexity

compared to mode-switching method, as the sampling rate of the overall algorithm is 100ms, computing complexity of IMM/EKF algorithm is not in question.

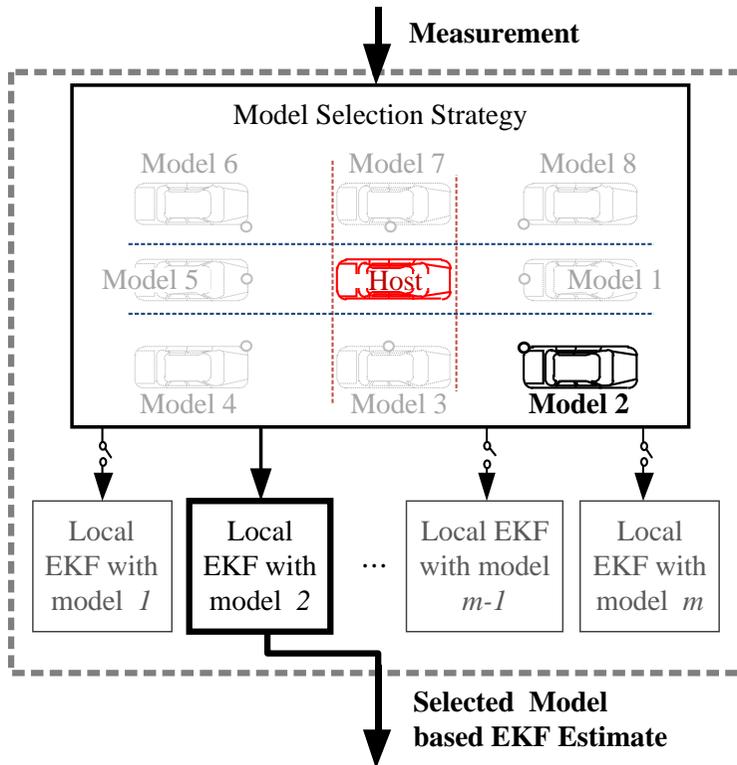


Figure 4.15. Architecture of model-switching/EKF Algorithm

Table 1. Statistics of Estimation Errors (N=400)

	RMS		Model-switching to IMM
	IMM	Model-switching	
$p_{n,x}$ (m)	0.7484	3.8412	5.1326
$p_{n,y}$ (m)	0.6430	1.9908	3.0961
θ_n (deg)	1.8787	5.2473	2.7931
$v_{n,x}$ (m/s)	0.6842	4.0688	5.9465
γ_n (deg/s)	1.1767	5.2730	4.4813
$a_{n,x}$ (m/s ²)	0.1549	0.7476	4.8278
$\dot{\gamma}_n$ (deg/s ²)	0.2403	2.9483	12.2687

4.4.4. Experimental Results with Multi-target Situation

In this section, the experimental results dealing with multi-target scenarios are presented. Real-road driving data were collected from a highway and an urban area. The camera is used to analyze the driving situation and to acquire the lane information. Especially, one test set for validation is extracted from the collected real-driving data to explore the multi-target tracking performance and target state estimation performance. The selected test set is a complex driving scene where the targets may enter and leave the field of view, showing the capability for birth and death of targets.

Figure 4.16 shows a frontal image of the vision camera, and the multi-target state estimation results in complex driving situation. The radar measurements are depicted by blue circles with inner star. And based on the lane information, a region of interest is defined and the outside is represented by red area. As can be seen in the figure, it is evident that the proposed algorithm manages to follow the correct true targets. A true target is most generally defined to be an object that will persist in the tracking volume for at least several scans. False alarms (or false targets) refer to erroneous detection events (such as those caused by random noise or clutter) that do not persist over several scan. In the figure, we can show that those events (caused by road structures such as guardrail in red area) cannot generate new track. At $t=8\text{sec}$, the proposed algorithm tracks 5 targets. And at $t=13\text{sec}$, one target enters the field of view and track#19 is newly generated. And at the same time track#10 starts leaving the sensing range. At $t=15.5\text{sec}$, track#10 leaves the sensing range completely, and zero-gain EKF is

applied. The track#14 starts its lane changing and cuts-in front of the host vehicle. Due to this lane changing, a measurement of hidden vehicle is newly arrived. And at $t=20\text{sec}$, track#12 is newly generated from this measurement while tracks other existing 4 targets. In summary, these results indicate that GNN based track management can be utilized with satisfactory. The GNN based track management approach, which considers the single most likely hypothesis for track update and new track initiation, works well because of widely spaced targets, relatively accurate measurements in the given experiment situation. However, the problem of closely-spaced unresolved targets can occur and this can be solved by MHT which forms multiple hypotheses and manages these hypotheses [Blackman'04].

Figure 4.17 presents the target state estimation results for track#14. Each graph shows each state variables, rewarding function, and mode probability update result as a function of time. As can be inferred from the vision image, the vehicle of track#14 changes the lane and cuts into the host vehicle's lane. From the figure, we can see that the overall trends of each estimated state can represent actual target motion with satisfactorily while the rewarding function of track#14 maintains the maximum value (100 in this paper). In the case of mode probability update results, from 0-18 sec, mode.12 (rear-right edge) is evaluated as the most highly likely mode. This can be explained by the fact that the target vehicle was driving in the left lane during that period. And a transition to mode.1 (center of rear side) from mode.12 has occurred in a period of lane changing. Finally, the probability of mode.1 ranked highest after the target vehicle becomes the in-lane preceding vehicle. Overall, the mode evaluation

results are quite reasonable, in the view of target aspect change.

Figure 4.18 shows the results for track#10. Track#10 drives with lower speed than the host vehicle in the right lane and leaves the field of view at about 13sec. From the figure, each state is estimated reasonably so that can represent actual motion. The rewarding function of track#10 shows a typical track management process. It keeps increasing until 10sec, maintains the maximum value from 10sec to 13sec, and drops after the target disappears from the scope of the radar. And in the case of mode probability update results, from 0-11sec, mode. 2 (rear-left edge) is evaluated as the most highly likely mode. This can be explained by the fact that the target vehicle was driving in the right lane during that period. And a transition to mode.5 (front of the left side) has occurred at 11sec. This drastic mode change can be explained from the change of radar measurement at that moment. In the figure, the radar measurement is depicted as a red circle line and we can show that longitudinal measurement changes drastically at 11sec. This reflects the fact that the mode adaptation of the proposed IMM algorithm is quite reasonable.

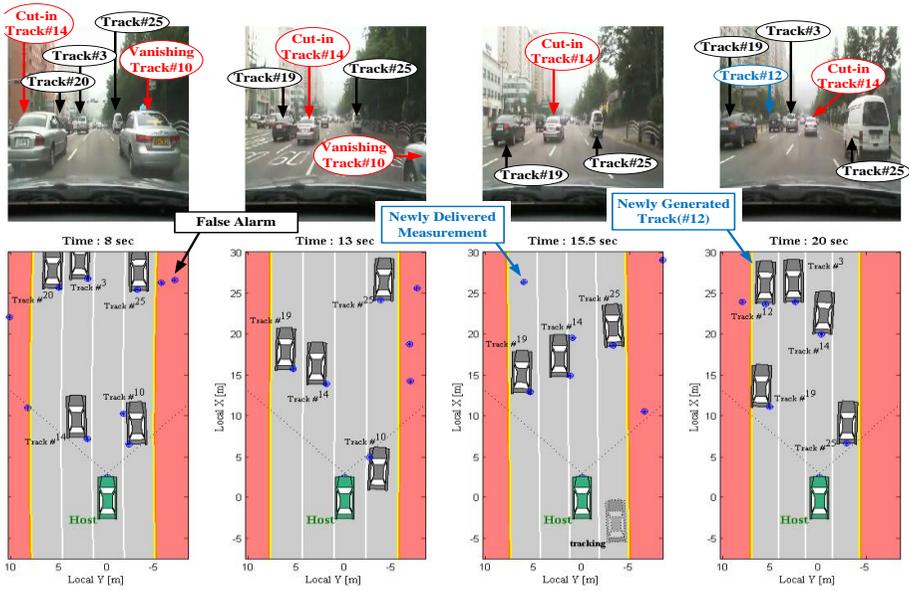


Figure 4.16. Multi-target state estimation in complex driving situation

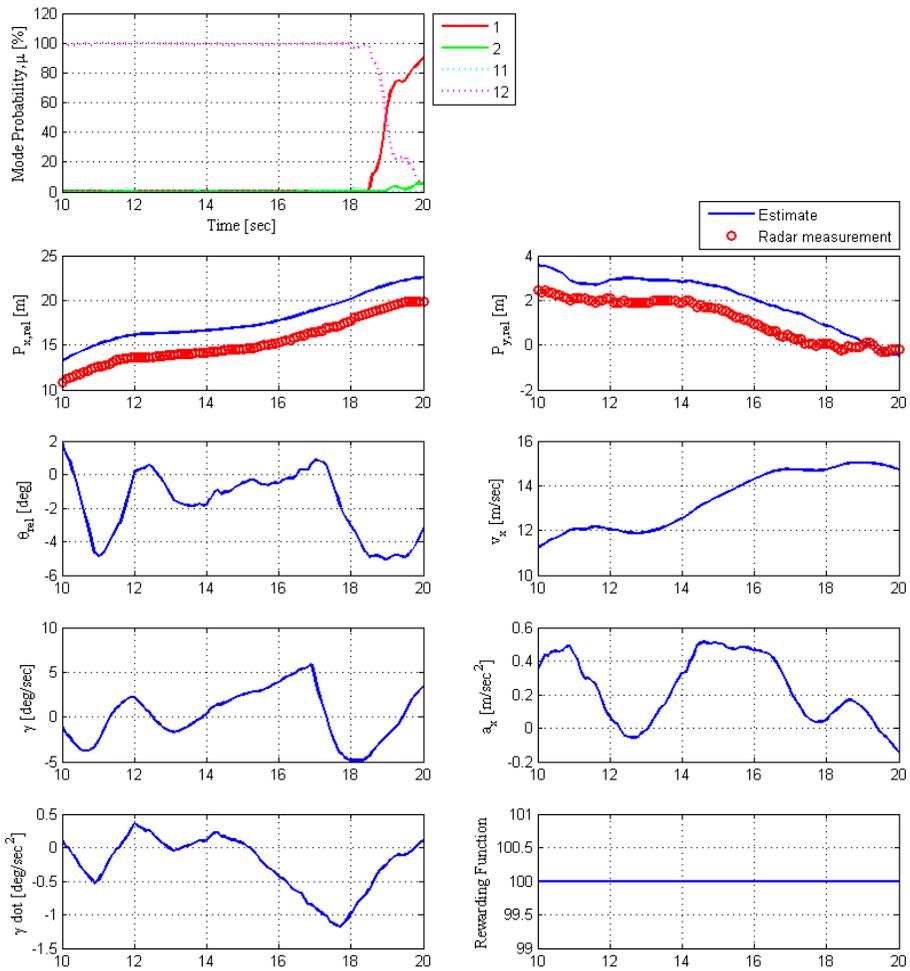


Figure 4.17. Target state estimation results for Track 14 (a target cutting in front of the host vehicle)

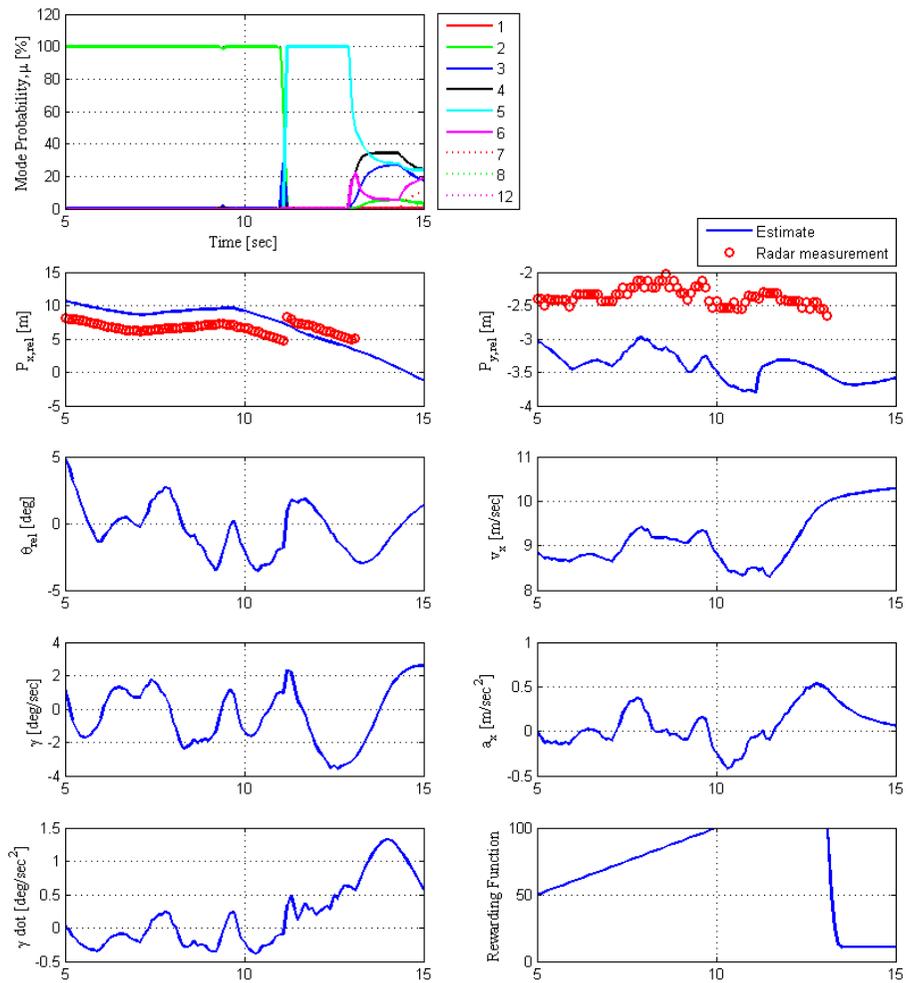


Figure 4.18. Target state estimation results for Track 10 (a target vanishing to rear-right-blind area)

Chapter 5 . Safety Driving Envelope Decision and Motion Optimization

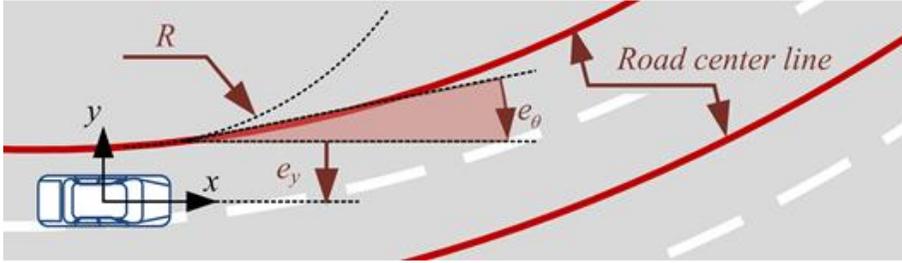
Our approach to motion planning was to separate it into two distinct tasks (cf. Figure 2.1). Firstly, it is responsible for translating perceived moving objects and their prediction, static obstacles from the laser scanner, information from the digital map and vision, etc. into geometric constraints. This top level module is called as safety driving envelope decision. Subsequently, the motion optimization module computes the desired path and desired velocity of the ego-vehicle as a function of time. This trajectory is obtained by solving a geometric problem that has been posed as a convex optimization problem with linear equality/inequality constraints. These two components are addressed in the following subsections.

5.1. Multi-traffic Prediction

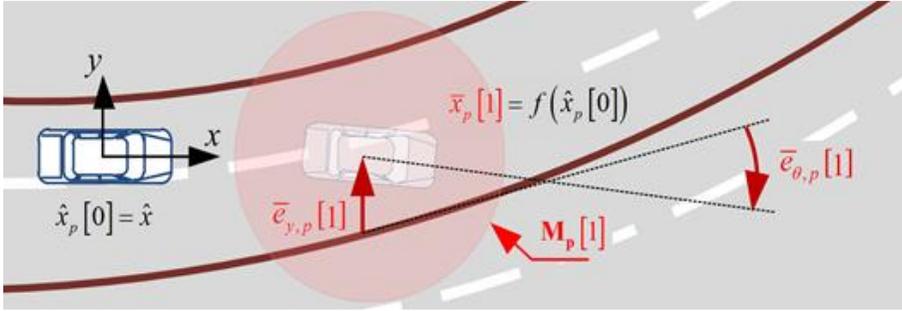
The possible behaviors of surrounding vehicles are predicted considered in determining the safe drivable area decision. In predicting reasonable ranges of the future states of surrounding vehicles, driving data is collected on test track and real roads to analyze the probabilistic movement characteristics of the vehicle.

The future vehicle states are described as a stochastic multi-stage process via Taylor Methods. At every time step of the prediction time horizon, a desired yaw rate is determined by the lane-keeping driver model. Suppose that the measurement noise has a normal distribution with zero mean. Hence, the predictive measurement is linearly related to the time-updated predictive state. Then, the maximum likelihood prediction of the future state is calculated by the Kalman filtering equations.

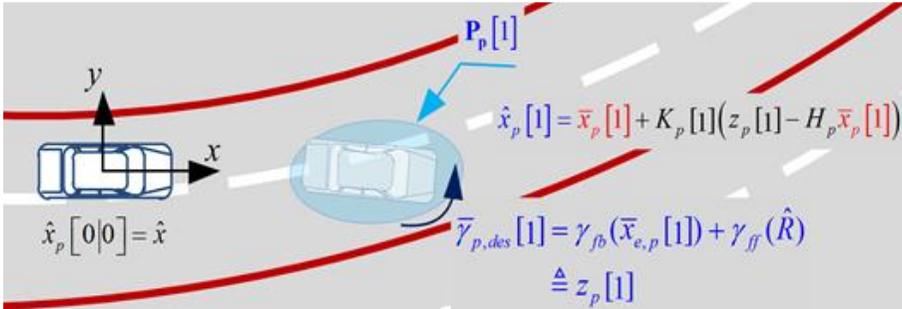
That is, behavior restrictions due to road geometry and driver characteristics such as reasonable acceleration input range are replaced with multi-stage of Kalman filtering problem by the proposed prediction algorithm. As an example, a prediction procedure at the 1st future time step is depicted in Figure 5.1-(b) and Figure 5.1-(c). The overall prediction results for 1 s, 2 s, and 3 s of prediction time at the instant of lane changing are given in Figure 5.1-(d).



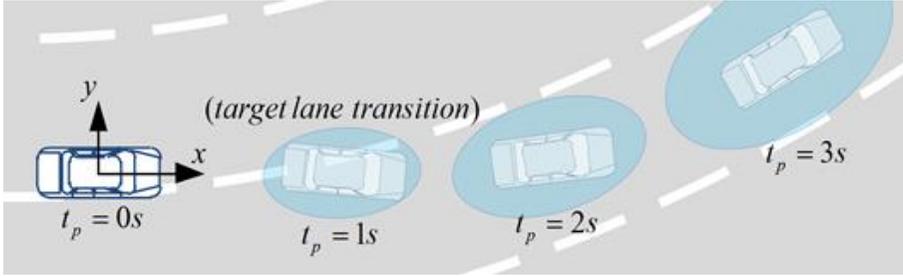
(a) The relationship between the host vehicle and the road center line of each lane



(b) The time-update-predicted host vehicle states and the relative error states with respect to road geometry defined on current body coordinate



(c) The measurement-update-predicted host vehicle states where the predicted desired yaw-rate to keep the lane is defined as virtual measurement



(d) Prediction results for 1s, 2s, and 3s of prediction time at a lane-changing instant

Figure 5.1. Procedure and Concept of Traffic Behavior Prediction

5.1.1. Lane Keeping Behavior Model

The objective of the lane keeping behavior model is to develop a yaw control system for human-like lane keeping. To achieve this goal, it is useful to utilize a dynamic model in which the state variables are in terms of position and orientation error with respect to the road. The error state is defined in terms of fixed coordinates under the assumption of traveling with constant longitudinal velocity on a road of constant radius. The error state is defined in inertial fixed coordinates rather than body-fixed moving coordinates. By using the definition of the road geometry in section A.2, the position error can be defined as:

$$e_y = p_y - \{a_2 \cdot p_x^2 + a_1 \cdot p_x + a_0 + w_{road} \cdot N(\text{current lane})\} \quad (5.1)$$

where p_x is the vehicle's longitudinal position, p_y is the vehicle's lateral position, w_{road} is the width of the road lane, and N is the adjusting integer to represent the current lane. For example, if the vehicle changes lanes to the left, N has the value of minus one. Under small-slip-angle assumptions, the time

derivative of the position error can be defined as:

$$\begin{aligned}\dot{e}_y &= \frac{d}{dt}(p_y) - (2a_2 \cdot p_x + a_1) \cdot \frac{d}{dt}(p_x) \\ &\cong v \sin \theta - (2a_2 \cdot p_x + a_1) \cdot v \cos \theta\end{aligned}\quad (5.2)$$

where v is the longitudinal velocity, and θ is the orientation. The orientation error and its time derivative can be defined as:

$$e_\theta = \theta - \tan^{-1}(2a_2 \cdot p_x + a_1) \quad (5.3)$$

$$\begin{aligned}\dot{e}_\theta &= \dot{\theta} - \frac{d}{dt}\{\tan^{-1}(2a_2 \cdot p_x + a_1)\} \\ &= \gamma - \frac{1}{1 + (2a_2 \cdot p_x + a_1)^2} \cdot \frac{d}{dt}(2a_2 \cdot p_x + a_1) \\ &= \gamma - \frac{1}{1 + (2a_2 \cdot p_x + a_1)^2} (2a_2) \cdot \frac{d}{dt}(p_x) \\ &= \gamma - \frac{2a_2}{1 + (2a_2 \cdot p_x + a_1)^2} \cdot (v \cos \theta)\end{aligned}\quad (5.4)$$

Under assumptions of small road slope and small error, the time derivatives of the error states can be simplified as follows:

$$\begin{aligned}\dot{e}_y &= v \sin \theta - (2a_2 \cdot p_x + a_1) \cdot v \cos \theta \\ &= v \cos \theta \cdot \{\tan \theta - (2a_2 \cdot p_x + a_1)\} \\ &\cong v \cos \theta \cdot e_\theta\end{aligned}\quad (5.5)$$

$$\begin{aligned}\dot{e}_\theta &= \gamma - \frac{2a_2}{1 + (2a_2 \cdot p_x + a_1)^2} \cdot (v \cos \theta) \\ &\cong \gamma - 2a_2 \cdot (v \cos \theta)\end{aligned}\quad (5.6)$$

If the yaw rate dynamics can be approximated as a 1st-order system which has the desired yaw rate as the system input, the state-space model of the tracking error variables is given by following equation:

$$\begin{aligned}
\dot{\mathbf{x}}_e &= \mathbf{F}_e \cdot \mathbf{x}_e + \mathbf{G}_e \cdot \gamma_{des} + \mathbf{G}_w \cdot a_2 \\
&= \begin{bmatrix} 0 & v \cos \theta & 0 \\ 0 & 0 & 1 \\ 0 & 0 & f \end{bmatrix} \cdot \begin{bmatrix} e_y \\ e_\theta \\ \gamma \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -f \end{bmatrix} \gamma_{des} + \begin{bmatrix} 0 \\ -2v \cos \theta \\ 0 \end{bmatrix} a_2
\end{aligned} \tag{5.7}$$

We can see that the first and second rows of the equation describe the road geometry coefficient process model under the assumption of fixed road curvature. Assume that the human drivers determine the desired yaw rate γ_{des} by a state feedback and a feed-forward term that attempts to compensate for the road curvature as follows:

$$\begin{aligned}
\gamma_{des} &= -\mathbf{C} \cdot \mathbf{x}_e + \gamma_{ff} \\
&= -[c_1 \quad c_2 \quad c_3] \mathbf{x}_e + \gamma_{ff}
\end{aligned} \tag{5.8}$$

From a steady-state analysis, we can see that the all error states can be made zero by appropriate choices of feedback gains and the feed-forward term. For example, the feedback gain can be determined by pole placement which is simulating the human driver's behavior characteristics. Then, the feed-forward term can be calculated directly from the above feedback gain and estimated road curvature:

$$\begin{aligned}
\gamma_{des} &= -[c_1 \quad c_2 \quad c_3] \mathbf{x}_e + \gamma_{ff} \\
&= -[c_1 \quad c_2 \quad c_3] \mathbf{x}_e + 2a_2 v \cos \theta (c_3 + 1)
\end{aligned} \tag{5.9}$$

5.1.2. Vehicle Predictor

In the prediction of the vehicle's future states, the available information includes the vehicle's current motion estimates, the road geometry estimates described in the current body coordinates, the lane keeping behavior model, and

the error covariance of each. As mentioned, the lane keeping behavior model-based desired yaw rate is used as the virtual measurement. The future vehicle system can be described as a stochastic multi-stage process via Taylor Methods:

$$\begin{aligned}\mathbf{x}_p[i+1] &= \mathbf{f}_p(\mathbf{x}_p[i]) + \mathbf{w}_p[i], \quad i=0, \dots, N_p-1 \\ &= [f_{1,p} \quad f_{2,p} \quad f_{3,p} \quad f_{4,p} \quad f_{5,p} \quad f_{6,p} \quad f_{7,p}] + \mathbf{w}_p[i]\end{aligned}\quad (5.10)$$

where

$$\begin{aligned}x_p &= [p_{x,p} \quad p_{y,p} \quad \theta_p \quad v_p \quad \gamma_p \quad a_p \quad \dot{\gamma}_p]^T \\ f_{1,p} &= p_{x,p} + (v_p \cos \theta_p) \Delta t + (a_p \cos \theta_p + \gamma_p v_p \sin \theta_p) \frac{\Delta t^2}{2} \\ f_{2,p} &= p_{y,p} + (v_p \sin \theta_p) \Delta t + (a_p \sin \theta_p + \gamma_p v_p \cos \theta_p) \frac{\Delta t^2}{2} \\ f_{3,p} &= \theta_p + (\gamma_p) \Delta t + (\dot{\gamma}_p) \frac{\Delta t^2}{2} \\ f_{4,p} &= v_p + (a_p) \Delta t + (-k_a a_p) \frac{\Delta t^2}{2} \\ f_{5,p} &= \gamma_p + (\dot{\gamma}_p) \Delta t + (-k_{\dot{\gamma}} \dot{\gamma}_p) \frac{\Delta t^2}{2} \\ f_{6,p} &= a_p + (-k_a a_p) \Delta t + (k_a^2 a_p) \frac{\Delta t^2}{2} \\ f_{7,p} &= \dot{\gamma}_p + (-k_{\dot{\gamma}} \dot{\gamma}_p) \Delta t + (k_{\dot{\gamma}}^2 \dot{\gamma}_p) \frac{\Delta t^2}{2}\end{aligned}$$

$$\mathbf{w}_p[i] \sim (0, W_p[i])$$

$$\mathbf{W}_p[i] = \left(\mathbf{B}_p \Delta t + \mathbf{F}_p[i] \mathbf{B}_p \frac{\Delta t^2}{2} \right) \mathbf{Q}_p \left(\mathbf{B}_p \Delta t + \mathbf{F}_p[i] \mathbf{B}_p \frac{\Delta t^2}{2} \right)^T$$

$$\mathbf{F}_p[i] = \left. \frac{\partial \mathbf{f}_p}{\partial \mathbf{x}_p} \right|_{\mathbf{x}_p = \hat{\mathbf{x}}_p[i]} \quad \mathbf{B}_p = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^T$$

where N_p is the length of the pre-defined prediction time horizon, and the subscript ‘p’ is used to denote ‘predictive’. The longitudinal acceleration and yaw acceleration are assumed to decay to zero with corresponding time constants of k_a and $k_{\dot{\gamma}}$. At every time step of the prediction time horizon, a desired yaw rate is determined by the path following model. Suppose that the measurement noise has a normal distribution with zero mean. Hence, the predictive measurement is linearly related to the time-updated predictive states as follows:

$$\begin{aligned}
\mathbf{z}_p[i] &= \mathbf{H}_p \cdot \mathbf{x}_p[i] + \mathbf{v}_p[i], & i = 0, \dots, N_p \\
&= [0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0] \cdot \mathbf{x}_p[i] + \mathbf{v}_p[i] \\
&= \bar{\gamma}_{des,p}[i] \\
&= -\mathbf{C} \cdot \bar{\mathbf{x}}_{e,p}[i] + \gamma_{ff,p}[i]
\end{aligned} \tag{5.11}$$

where

$$\begin{aligned}
\bar{\mathbf{x}}_{e,p}[i] &= \mathbf{f}_e(\bar{\mathbf{x}}_p[i], \hat{\mathbf{x}}_r[0]) \\
&= [\bar{e}_y[i] \ \bar{e}_\theta[i] \ \bar{\gamma}_p[i]]^T \\
&= \begin{bmatrix} \bar{p}_{y,p}[i] - \{\hat{a}_2 \cdot \bar{p}_{x,p}[i]^2 + \hat{a}_1 \cdot \bar{p}_{x,p}[i] + \hat{a}_0 + \hat{w}_{road} \cdot N[i]\} \\ \bar{\theta}_p[i] - \tan^{-1}(2\hat{a}_2 \cdot \bar{p}_{x,p}[i] + \hat{a}_1) \\ \bar{\gamma}_p[i] \end{bmatrix}
\end{aligned}$$

$$\mathbf{v}_p[i] \sim (0, \mathbf{V}_p[i])$$

$$\mathbf{V}_p[i] = \mathbf{V}_{p,const} + [\bar{e}_y[i] \ \bar{e}_\theta[i]]^T \mathbf{w}_{correct} \begin{bmatrix} \bar{e}_y[i] \\ \bar{e}_\theta[i] \end{bmatrix}$$

where $\mathbf{w}_{correct}$ is the matrix defining the additional covariance of the path-following model-based desired yaw rate. Then, the maximum likelihood

prediction of the future state is given by the following extended Kalman filtering. Because the proposed prediction algorithm is based on a stochastic filtering method, the covariance of the prediction error can be evaluated at each time step. Furthermore, the eigenvalue and eigenvectors of the 2nd leading principal minor of \mathbf{P}_p determine the likelihood ellipse around the predictive position. Using the square root of the eigenvalues as semi-axes measured along the eigenvectors, we can sketch the 39% likelihood ellipse with the center at the most likely predictive position. The size of the 87% likelihood ellipse is two times the size of the 39% ellipse in linear dimensions, and that of the 99% ellipse is three times the size. This analysis is very useful to visualize and compare the prediction performance in view of the accuracy and precision.

Initialization of prediction at kth time step of the ego-vehicle

$$\hat{\mathbf{x}}_p[0] = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \hat{\mathbf{x}}[k] \end{bmatrix} \quad \mathbf{P}_p[0] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{P}[k] \end{bmatrix} \quad (5.12)$$

Initialization of prediction at kth time step of the nth-target-vehicle

$$\hat{\mathbf{x}}_p[0] = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \hat{\mathbf{x}}[k] \end{bmatrix} \quad \mathbf{P}_p[0] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{P}[k] \end{bmatrix} \quad (5.13)$$

Time update (rough prediction)

$$\begin{aligned}
\bar{\mathbf{x}}_p [i] &= \mathbf{f}_p (\hat{\mathbf{x}}_p [i-1]) \\
\mathbf{F}_p [i-1] &= \left[\frac{\partial \mathbf{f}_p}{\partial \mathbf{x}_p} \right]_{\mathbf{x}_p = \hat{\mathbf{x}}_p [i-1]} \\
\mathbf{M}_p [i] &= \mathbf{F}_p [i-1] \cdot \mathbf{P}_p [i-1] \cdot \mathbf{F}_p [i-1]^T + \mathbf{W}_p
\end{aligned} \tag{5.14}$$

Measurement update (refined prediction)

$$\begin{aligned}
\bar{\mathbf{x}}_{e,p} [i] &= \mathbf{f}_e (\bar{\mathbf{x}}_p [i], \hat{\mathbf{x}}_r [0]) \\
\mathbf{z}_p [i] &= -\mathbf{C} \cdot \bar{\mathbf{x}}_{e,p} [i] + \gamma_{ff,p} [i] \\
\mathbf{V}_p [i] &= \mathbf{V}_{p,\text{const}} + \begin{bmatrix} \bar{e}_y [i] & \bar{e}_\theta [i] \end{bmatrix}^T \mathbf{w}_{\text{correct}} \begin{bmatrix} \bar{e}_y [i] \\ \bar{e}_\theta [i] \end{bmatrix} \\
\mathbf{K}_p [i] &= \mathbf{M}_p [i] \mathbf{H}_p^T \cdot (\mathbf{H}_p \mathbf{M}_p [i] \mathbf{H}_p^T + \mathbf{V}_p [i])^{-1} \\
\mathbf{P}_p [i] &= (\mathbf{I} - \mathbf{K}_p [i] \mathbf{H}_p) \cdot \mathbf{M}_p [i] \\
\hat{\mathbf{x}}_p [i] &= \bar{\mathbf{x}}_p [i] + \mathbf{K}_p [i] \cdot (\mathbf{z}_p [i] - \mathbf{H}_p \cdot \bar{\mathbf{x}}_p [i])
\end{aligned} \tag{5.15}$$

5.1.3. Test Data based Implementation and Performance Evaluation

The vehicle tests were conducted to implement the algorithm (and to determine the parameters of the algorithm) and to investigate the performance improvement by the proposed Probabilistic Fusion Model (PFM) through post-processing analysis. The tests were conducted with 100 randomly selected drivers with at least 5 years of driving experience on both urban and highway roads in various weather conditions. With the acquired data, the parameters of the algorithm were calculated by post-analysis and applied to the algorithm. Hence, the parameter extraction is based on this driving data, and the parameters presented in the paper can be said to represent nominal driver.

In addition to the implementation, comparisons with a Fixed Yaw Rate Model (FYRM) and a Lane Keeping Model (LKM) were conducted. The FYRM assumes that the vehicle may maintain the current yaw behavior, and LKM assumes that the current lane position is maintained without consideration of the current vehicle motion.

The parameters such as the three covariance matrixes of the sensor noise, the three

covariance matrixes of the process noise, and the feedback gain of the path-following model should be calculated or estimated. Accurate evaluations of these parameters are necessary to represent the realistic characteristics of the sensors, processes, and driver yaw behavior.

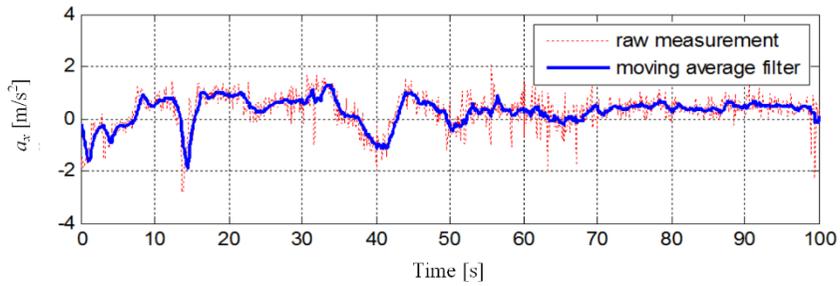
A running average filter is used to generate the quasi-actual reference value. As an example, the procedure to estimate the covariance of sensor noise of longitudinal acceleration is depicted in Figure 5.3-(a) and Figure 5.3-(b). From the figures, we can see that the white Gaussian sensor noise assumption is very realistic. In the case of the process noise, the same procedure is applied, and the covariance calculation is conducted. The process noise is defined as the difference between the predicted value and actual next-step value. As an example, the procedure to estimate the covariance of process noise of the longitudinal acceleration is depicted in Figure 5.3-(d). We can see that the white Gaussian process noise assumption is also quite realistic. The feedback gain of the path-following model is decided by a linear-quadratic-based method that was proposed and validated in previous research [Lee'14, Lee'11, Moon'08]. From the analysis of modeling error of the path-following model (compared to the real yaw rate data), it has been shown that the modeling errors have zero mean and normal distribution, as shown in Figure 5.3-(e).

The FYRM and LKM can be implemented simply by some parameter modifications. For the implementation of the FYRM, the decay rate of yaw acceleration should have the maximum value to represent a fixed yaw rate. In addition to this, the predictive sensor noise covariance V_p should have a large value so that the desired yaw rate has little effect on the prediction (i.e. not to use the road geometry information). In the case of the LKM, very small covariance should be applied for a predictive measurement to make the corrections by the road geometry be dominant throughout the prediction range.

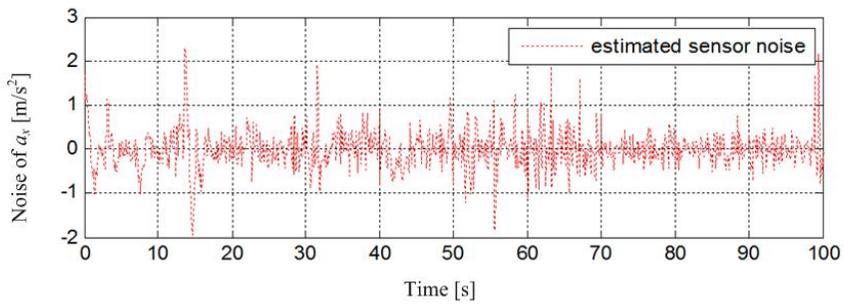
The proposed algorithm is worthy of note because of its flexibility and adaptability. In addition, quantitative measurement and visualization of the prediction performance have been made possible not only for the proposed method but also for existing

conventional methods.

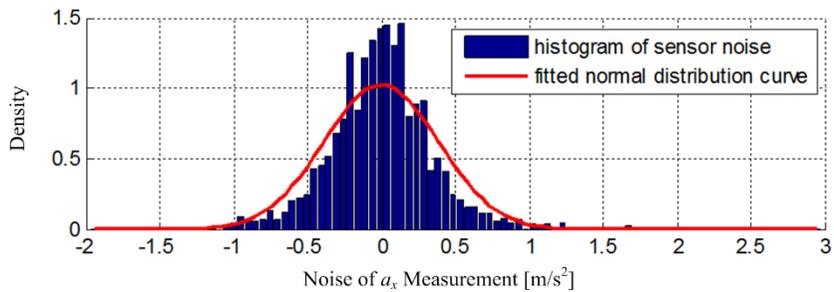
The general parameters of each prediction method are shown in Table 1. Note that the process noise of the exact process model has some small value to represent the discretized error and no-slip assumption error.



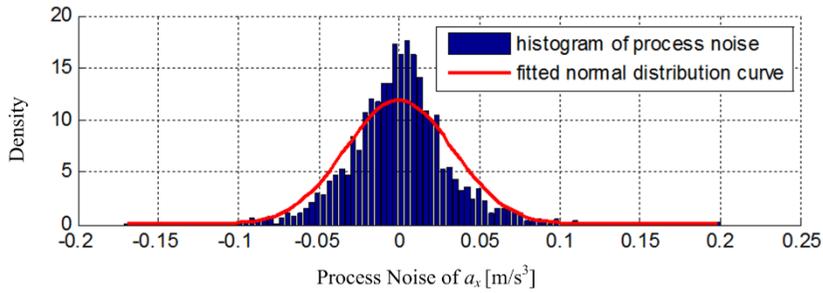
(a) Raw measurement and moving-average-filtered value



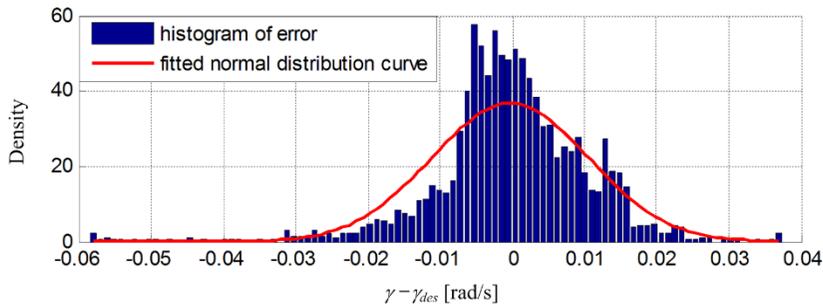
(b) Estimated sensor noise



(c) Histogram and fitted normal distribution of estimated sensor noise



(d) Histogram and fitted normal distribution of estimated process noise based on moving average filter



(e) Histogram and fitted normal distribution of modeling error between actual yaw rate and PFM-based desired yaw rate

Figure 5.2. The procedure to calculate the parameter values: covariance of sensor noise, process noise, and modeling error

Figure 5.3 shows test results of other traffic tracking and its possible motion prediction. The sensing area of long range radar is depicted as green region and mid-range radar is depicted as blue region. Red squares are current observation of radar. Blue-filled-vehicle is the estimated oncoming vehicle and blue-lined-vehicles are prediction results at 0.5s time interval. From the front cam view of Figure 5.3, we can see that all predicted positions are in the center of the lane.

In this way, the prediction performance can be said to be clearly reasonable indirectly.

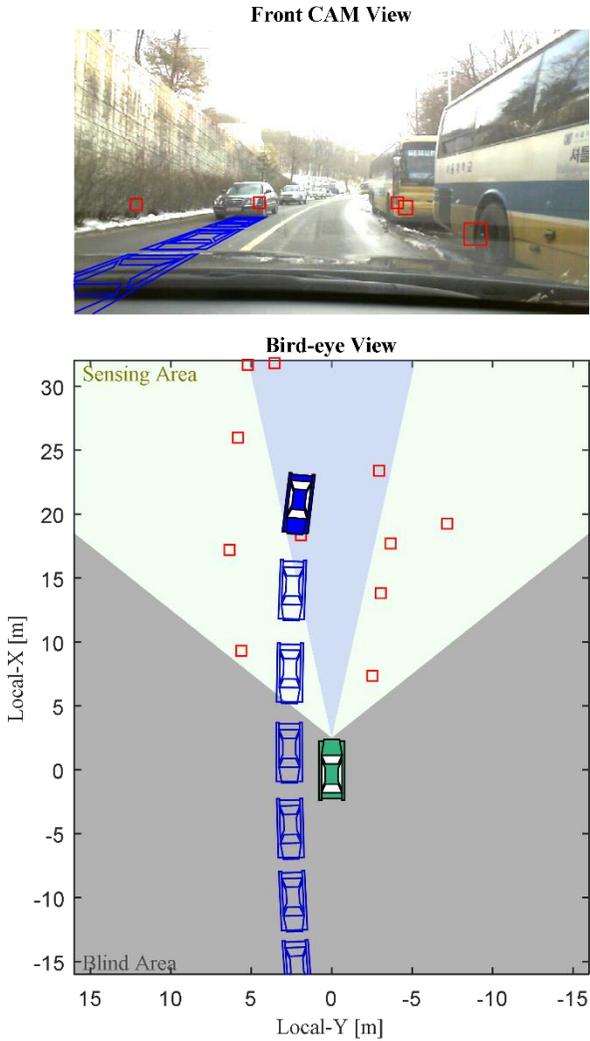


Figure 5.3. An example of moving object tracking and prediction.

The comparison analysis is conducted for general driving situations by a statistical method. These general driving situations include almost every case of driving scenario, such as straight lane keeping, curved lane keeping, curve entry, curve exit, etc. As an example, one test case (which is included in the data set for the comparison analysis) is given in Figure 5.4. In general driving situations, human drivers regulate time headway to maintain a safe distance from the preceding vehicle. On average, the specific range (0-5 s) of time headway accounted for 95 % of the human data [Fairclough'97]. Therefore, from a practical viewpoint, the data within the range of 0-5 s of future time (substituting time headway) have been analyzed.

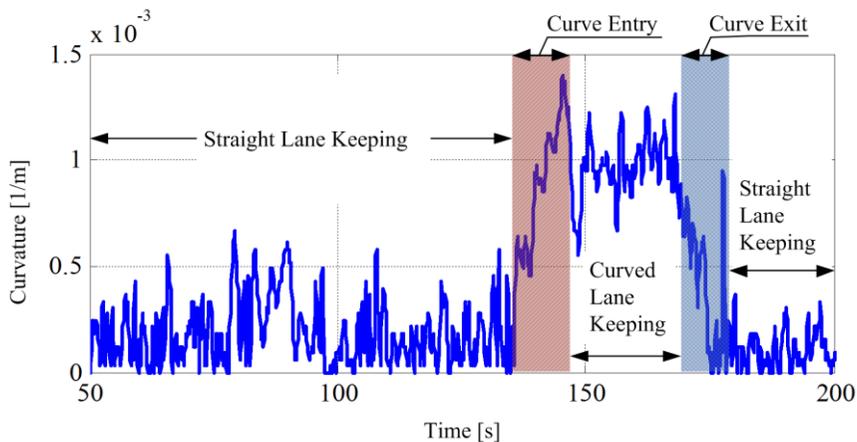


Figure 5.4. Curvature measurements of the Mobileye system including 4 cases of driving scenarios

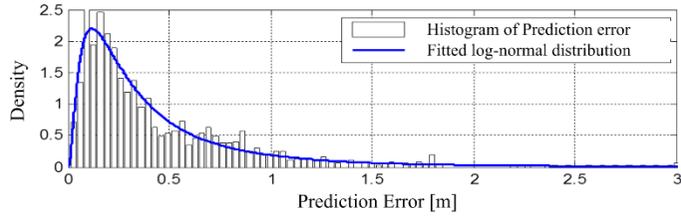
As shown in Figure 5.5 and Figure 5.6, the distribution of prediction errors at some time step can be fitted by the log normal distribution very accurately [Balakrishnan'99, Fairclough'97]. Such tendency is independent of the prediction range or prediction

method. From the figures, we can see that all statistical properties (such as the accuracy and precision) of the proposed algorithm are superior to those of the other two prediction methods, regardless of the prediction time step. Note that this independent superiority from the prediction time step can be defined as the robustness of the algorithm compared to both conventional methods.

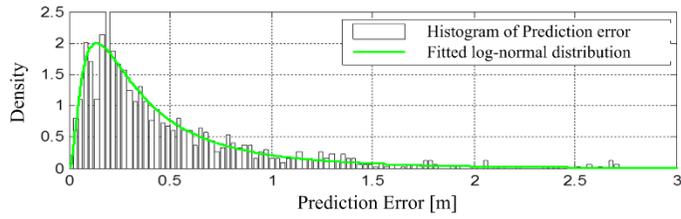
In addition to this cross-sectional analysis, from the fitted density distribution functions of the prediction error, the reliability of prediction can be defined and calculated. If the prediction error exceeds a specific value (the error threshold), the associated predicted position can be said to be unreliable. The value of the error threshold is dependent mainly on the road width. Therefore, in this paper, the value of this error threshold is set to one half of the general road width. In this representation, the reliability of position prediction can be defined as the probability that the prediction error is less than a given error threshold. This definition is depicted in Figure 5.7-(a). As shown in Figure 5.7-(a), the reliability of the position prediction is represented as an area under the density distribution curve within the error threshold. Alternatively, this can be also represented as a specific point on the cumulative distribution curve of the prediction error.

Applying the definition of the prediction reliability to the error distribution functions at each future time step, a comparison of the prediction reliabilities can be conducted as depicted in Figure 5.7-(b) through Figure 5.7-(d). The solid red line represents the reliability of the proposed probabilistic sensor-fusion-based prediction method, the dotted blue line represents the prediction method based on the conventional fixed yaw rate assumption, and the dashed green line represents the prediction method based on the conventional lane keeping assumption. As shown in the figure, the effect of sensor fusion is quite visible and significant. We can see that the fixed yaw rate model, which has a dominant confidence in current vehicle states, has shown superior performance in a short range of prediction time ($t_p=0.0-2.1$ s), while the performance of the lane keeping model is outstanding in a long range of prediction time (after $t_p=2.1$ s). These

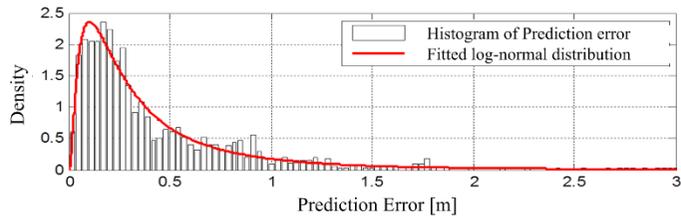
results have reminded us of the necessity of proper fusion between two conventional methods (the current-state-based and road-geometry-based methods). As a result of the proper probabilistic fusion by the proposed algorithm, it is shown that the proposed fusion algorithm can make the most out of the relative merits of each conventional method throughout all the prediction range and predict the future vehicle states more reliably. Moreover, the synergistic effect is predominant in the range of 1.7-4.0 s of prediction time. The proposed algorithm has improved the reliability of prediction by up to 20% compared to prediction based on a fixed yaw rate model, and up to 5% compared to the road-geometry-based prediction method in the range of 1-3 s of future time, which is the main interesting region of various IVSS systems.



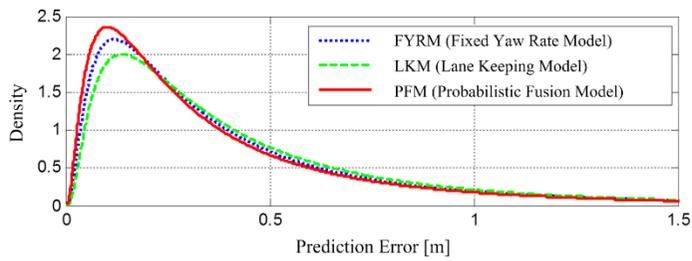
(a) FYRM



(b) LKM

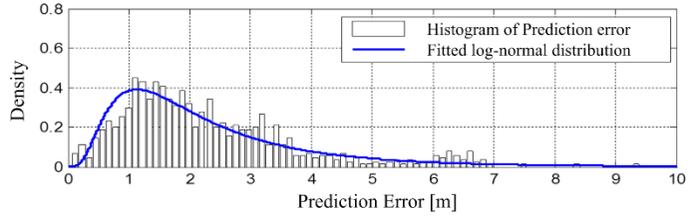


(c) PFM

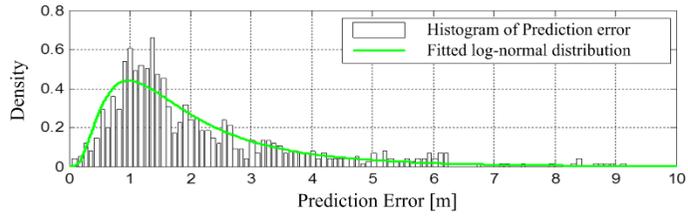


(d) Comparison of fitted error distribution

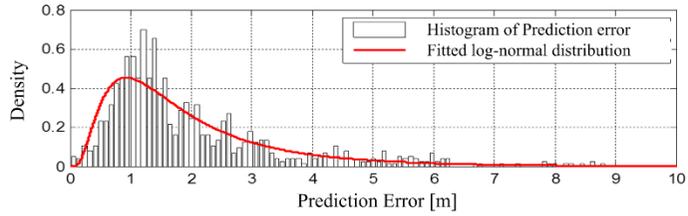
Figure 5.5. Density distribution of the prediction error of each method at $t_p=1.0s$ (i.e. short range prediction)



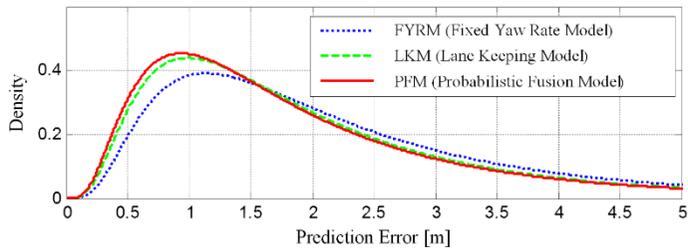
(a) FYRM



(b) LKM

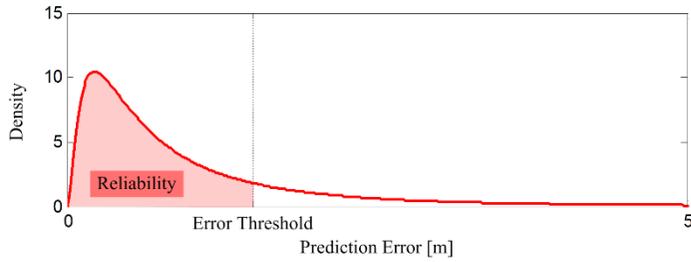


(c) PFM

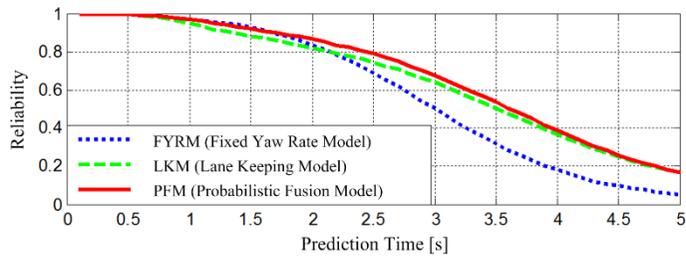


(d) Comparison of fitted error distribution

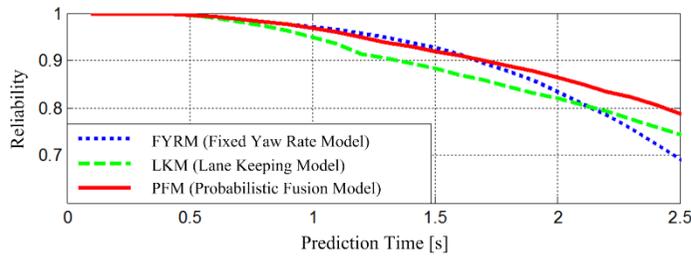
Figure 5.6. Density distribution of the prediction error of each method at $t_p=3.5s$ (i.e. long-range prediction)



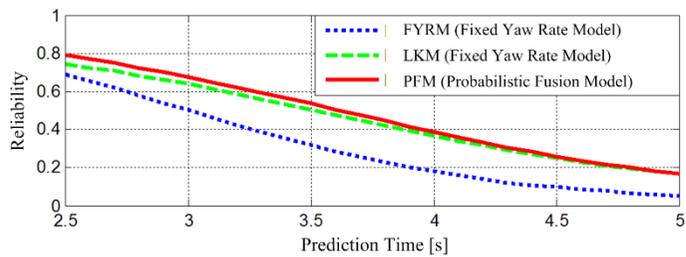
(a) Definition of the Reliability of prediction



(b) Reliability curve for the total prediction range ($[tp]=[0.0, 5.0]$)



(c) Reliability curve for short-range prediction ($[tp]=[0.0, 2.5]$)



(d) Reliability curve for long-range prediction ($[tp]=[2.5, 5.0]$)

Figure 5.7. Prediction reliability evaluation along the prediction time axis

5.2. Safety Driving Envelope Decision

In order to develop a highly automated driving system, a safe driving envelope which indicates the drivable boundaries for safe driving over a finite prediction horizon should be determined with the consideration of not only current states of traffic situation surrounding the subject vehicle but also probable future states of that simultaneously [Ferguson'14].

The safety envelope decision module uses the initial driving corridor as a constraint, and asserts that the vehicle stays in its bounds. Starting from this initial driving corridor, static obstacles, which are represented as grid map, are combined with either the left or right corridor bound. For this coupling, for all obstacles grids, it is decided whether the ego-vehicle is supposed to pass them on the left or right. After having assigned all obstacle grids to either the left or right corridor bound, geometric constraints for motion optimization are newly computed considering driver acceptance, sensor uncertainty, and control uncertainty, etc.

To handle the probable risky situation due to surrounding vehicles within the finite time-horizon, the probable future states of other traffic participants should be considered. To achieve this, every moving object estimated from the sensor system is associated with one lane and a future motion for the object is predicted in probabilistic methodology, assuming that the vehicle follows the lane with acceptable deviation. Similar to the static obstacles, safety envelopes are created for each of the prediction results. However, because the object is in motion, each envelope is active for a specific time step only.

Then the relative complement of predicted area of moving object and unsafe region due to static obstacle in the initial driving corridor is the set of drivable area at each time step, also termed “safety driving envelope”.

In Figure 5.8, these safety envelop decision results are presented for the case of an oncoming vehicle and right side static obstacles (parked buses). All variables are expressed in term of the current local coordinate of ego vehicle. In the figure, the initial driving corridor is depicted as green area, newly computed geometric constraints due to static obstacle as red area, and active constraints due to moving object at each time step as blue area. And the blue vehicles mean the predicted oncoming vehicle and the green vehicle is the predicted ego-vehicle. Note that active constraints due to moving object is propagated as the prediction interval grows longer. Then the safety driving envelope can be defined as the relative complement of blue and red area in green area, at each time step.

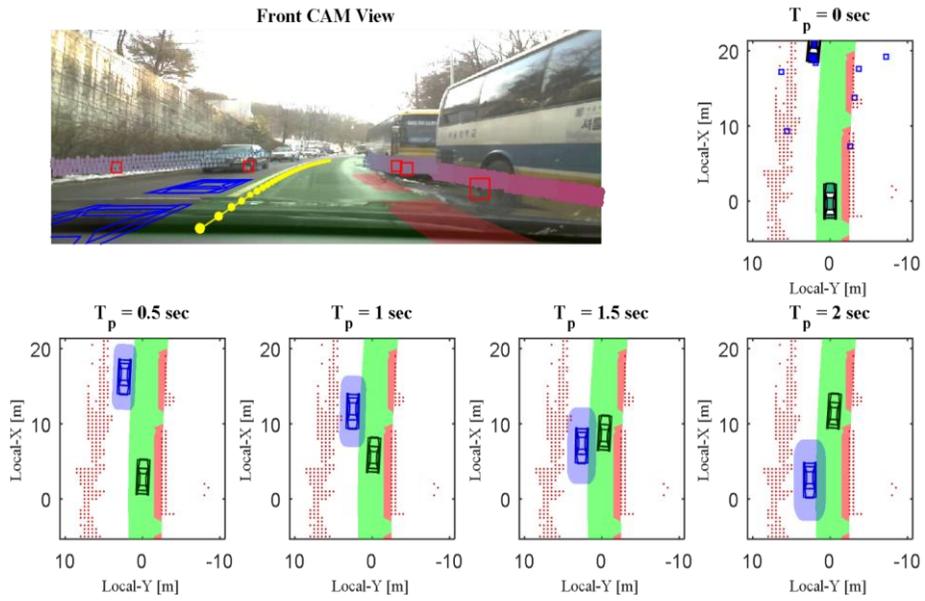


Figure 5.8. Safety envelop decision results at different time steps.

5.3. Model Predictive Control based Motion Planning

In order to determine the desired motion to track, a control architecture based on MPC approach is adopted. The sampling time is chosen as 0.1s and the length of the prediction horizon, N_p , is chosen as 21. These receding horizon optimization problems are solved at each time step and the first terms of the optimal control sequences are applied to the system. Then receding horizon optimization problems for a shifted prediction horizon are solved to obtain new optimal control inputs at next time step. To solve MPC problem in MATLAB, CVXGEN which is designed to be utilizable in MATLAB is used as solver [Mattingley'12]. The MPC problem is defined using CVXGEN syntax, and the CVXGEN returns convex optimization solver for the defined optimization solver for the defined optimization problem.

Particle motion model is used as the system model for MPC approach. State vector, input vector, and dynamic model are defined as follows:

$$\mathbf{x}_{\text{MPC},[j|k]} = \begin{bmatrix} p_{x,[j|k]} & p_{y,[j|k]} & v_{x,[j|k]} & v_{y,[j|k]} \end{bmatrix}^T \quad (5.16)$$

$$\mathbf{u}_{\text{MPC},[j|k]} = \begin{bmatrix} a_{x,[j|k]} & a_{y,[j|k]} \end{bmatrix}^T \quad (5.17)$$

$$\mathbf{x}_{\text{MPC},[j|k+1]} = \mathbf{A}_{\text{MPC},[j|k]} \mathbf{x}_{\text{MPC},[j|k]} + \mathbf{B}_{\text{MPC},[j|k]} \mathbf{u}_{\text{MPC},[j|k]} \quad (5.18)$$

where the system matrices are defined as follows:

$$\mathbf{A}_{\text{MPC},[j|k]} = \begin{bmatrix} 1 & 0 & \Delta t_{\text{MPC}} & 0 \\ 0 & 1 & 0 & \Delta t_{\text{MPC}} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{B}_{\text{MPC},[j|k]} = \begin{bmatrix} \frac{\Delta t_{\text{MPC}}^2}{2} & 0 \\ 0 & \frac{\Delta t_{\text{MPC}}^2}{2} \\ \Delta t_{\text{MPC}} & 0 \\ 0 & \Delta t_{\text{MPC}} \end{bmatrix}$$

MPC approach determines optimal control sequences which minimize the predefined cost function while satisfying constraints. Therefore, before the formulation of MPC problem, we should design the cost function corresponding to the objective of the control algorithm. The predictive expression of the cost function can be represented as follows:

$$\begin{aligned} J = & \sum_{j=1}^{N_p} \left\{ \left(\mathbf{x}_{\text{MPC},[j|k]} - \mathbf{x}_{\text{model},[j|k]} \right)^T \mathbf{Q}_{[k]} \left(\mathbf{x}_{\text{MPC},[j|k]} - \mathbf{x}_{\text{model},[j|k]} \right) \right\} \\ & + \sum_{j=0}^{N_p-1} \left(\mathbf{u}_{\text{MPC},[j|k]}^T \mathbf{W}_{\mathbf{u},[j|k]} \mathbf{u}_{\text{MPC},[j|k]} \right) \\ & + \sum_{j=1}^{N_p-1} \left\{ \left(\mathbf{u}_{\text{MPC},[j|k]} - \mathbf{u}_{\text{MPC},[j-1|k]} \right)^T \mathbf{W}_{\Delta \mathbf{u},[j|k]} \left(\mathbf{u}_{\text{MPC},[j|k]} - \mathbf{u}_{\text{MPC},[j-1|k]} \right) \right\} \end{aligned} \quad (5.19)$$

where, $\mathbf{x}_{\text{model},[j|k]}$ is the initial guessed ego-vehicle states of j-th prediction time step, which predicted by the methodology described in 5.1. \mathbf{Q} is predefined weighting matrix, which penalize the differences between states and rule-based predicted states, $\mathbf{W}_{\mathbf{u}}$ and $\mathbf{W}_{\Delta \mathbf{u}}$ are predefined weighting matrices for the reduction of magnitudes of control input sequences and the rate of change in control sequences respectively. These matrices are positive-definite symmetric.

The driving limit usually becomes dominated by the friction limit of the tires. This limit can be thought of as a circle of forces [Pacejka'05], and these

constraints are given as follows:

$$\|\mathbf{u}_{\text{MPC},[j|k]}\| \leq u_{\text{max}} \quad (5.20)$$

Safety driving envelope decision results at each predated time step should be representable as Quadratic Program to be solved with CVXGEN.

Safety driving envelope decision results at each predated time step should be representable as Quadratic Program to be solved with CVXGEN. For these QP representation, the constraints due to safety driving envelope are approximated as maximal rectangles as shown in Figure 4.4. The rotation of each rectangle is given from initial prediction of ego-vehicle and the maximal rectangle is decided by iterative method. Consequently, the rectangle representations can be expressed as two inequality linear equations as follows:

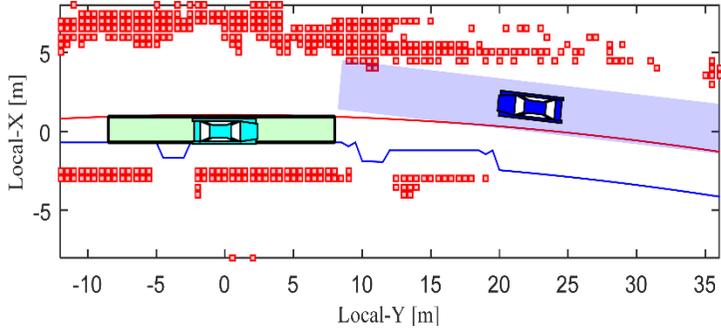
$$\left| \left\{ \mathbf{R}_{\text{model},[j|k]} (\mathbf{x}_{\text{MPC},[j|k]} - \mathbf{x}_{\text{center},[j|k]}) \right\}^1 \right| \leq p_{x,\text{margin}} \quad (5.21)$$

$$\left| \left\{ \mathbf{R}_{\text{model},[j|k]} (\mathbf{x}_{\text{MPC},[j|k]} - \mathbf{x}_{\text{center},[j|k]}) \right\}^2 \right| \leq p_{y,\text{margin}} \quad (5.22)$$

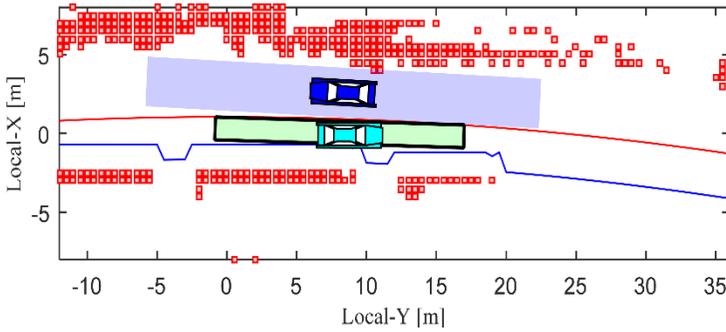
where rotation matrix $\mathbf{R}_{\text{model}}$ is defined by the initial guessed ego-vehicle's states as follows:

$$\mathbf{R}_{\text{model},[j|k]} = \begin{bmatrix} +\cos(\mathbf{x}_{\text{model},[j|k]}^3) & +\sin(\mathbf{x}_{\text{model},[j|k]}^3) \\ -\sin(\mathbf{x}_{\text{model},[j|k]}^3) & +\cos(\mathbf{x}_{\text{model},[j|k]}^3) \end{bmatrix} \quad (5.23)$$

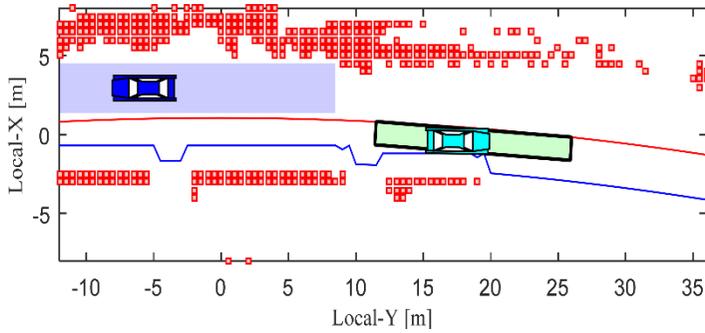
Figure 5.9 shows an example of MPC based motion planning results at the instance of meeting an oncoming vehicle. The maximal rectangle at each predicted time step is depicted as a sky-color-filled square and a cyan vehicle is the MPC solution at that time step. As can be seen in the figure, the ego-vehicle is guided along the safe region with appropriate control input sequences.



(a) at $t_p = 0.0s$



(c) at $t_p = 1.0s$



(e) at $t_p = 2.0s$

Figure 5.9. An example of MPC based motion planning at the instance of meeting an oncoming vehicle.

Chapter 6 Vehicle Tests based Performance Evaluation

The proposed automated driving system is evaluated through test-data based computer simulations and vehicle tests. The test-data based simulation is constructed using the commercial vehicle software, the Carsim and Matlab/Simulink with collected driving data. Data is collected under a various different task conditions while driving on urban roads and expressways. The control output of the proposed algorithm is compared with manual driving of human drivers. And the vehicle tests have been conducted for several times at the internal road of Seoul National University. The given test roads have quite complicated environments to drive automatically. However, from the test results, it is apparent that the test vehicle successfully completed the test route with the proposed automated driving algorithm.

6.1. Test-Data based Simulation

Manual driving of human driver (Human) and controlled behavior of the proposed automated driving system ('ADS') has been compared in the case of driving in busy urban traffic. To verify a driver acceptance and a ride comfort of the proposed algorithm, drivers' relative positions with respect to this safe driving envelope rectangle have been collected and analyzed. As aforementioned, the constraints due to safety driving envelope are approximated as maximal rectangles. Since the vehicle defines the sequences of the safe driving rectangles while it moves, we first transform the previously defined safe driving rectangles to the current local position (including orientation) of the vehicle.

Figure 6.1 shows the data of relative positions with respect to normalized safe driving rectangles which are defined 1 step (0.1s) ahead. The normalized safe driving rectangle is depicted by red-line and the relative normalized position of human driver is depicted by black-square. As can be seen in the figure, drivers control the vehicle to remain in safe driving rectangle.

Figure 6.2 shows the histogram and contour of the data of relative normalized position of human driver. The histogram and contour graph have shown more significant results. As can be seen in Figure 6.2, the driver absolutely stay within the safe driving rectangle. However, it is note that the relative positions do not converge to the center but only stay close enough to the center. It can thus be suggested that a tracking a center-path of drivable area is not a good solution for a driver acceptance and a ride comfort. Figure 6.3 and Figure 6.4

have shown the histogram and contour of the data of relative position of human driver with respect to normalized safe driving rectangles which are defined 3 steps (0.3s) ahead and 5 steps (0.5s) ahead each. As can be seen in the figures, the relative positions do not converge to the center but only stay close enough to it as the case of 1 step ahead.

Figure 6.5 shows the distributions of normalized relative position of each direction. As shown in Figure 6.5, the range of relative lateral position has the larger magnitude. A possible explanation for these results may be the safe driving envelop rectangle (not square). And these results once again stress that driver do not tracking the center of the drivable area but only stay in drivable area with a wide range of choices.

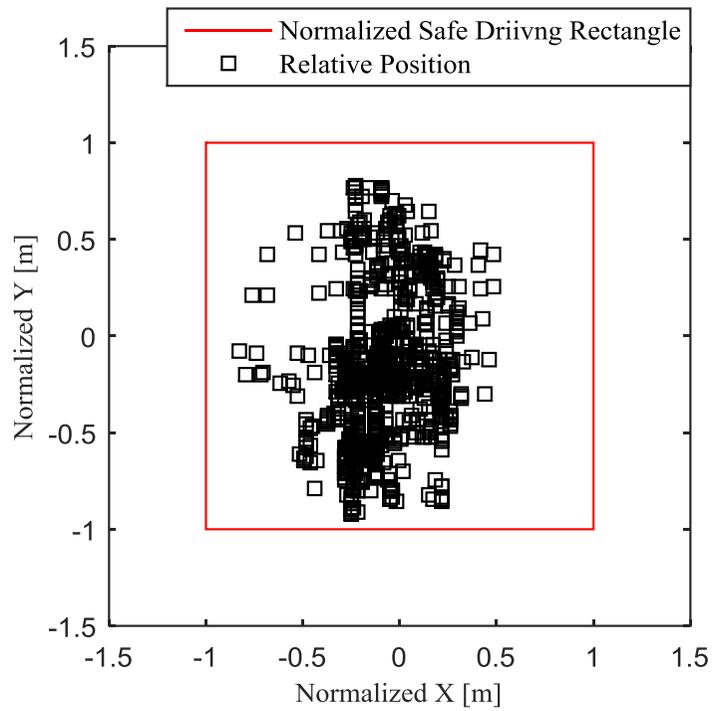
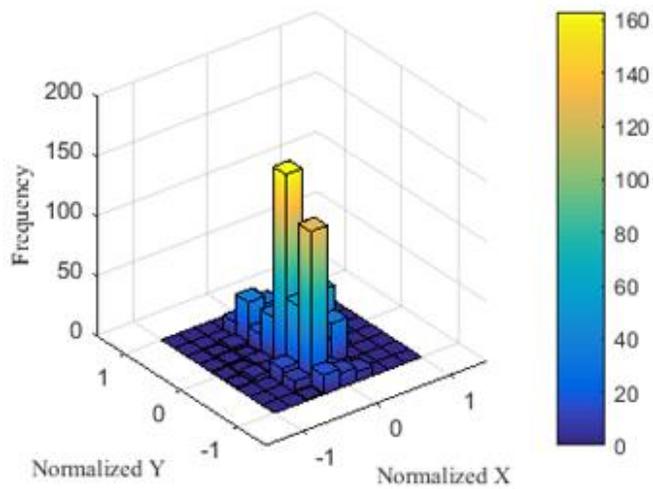
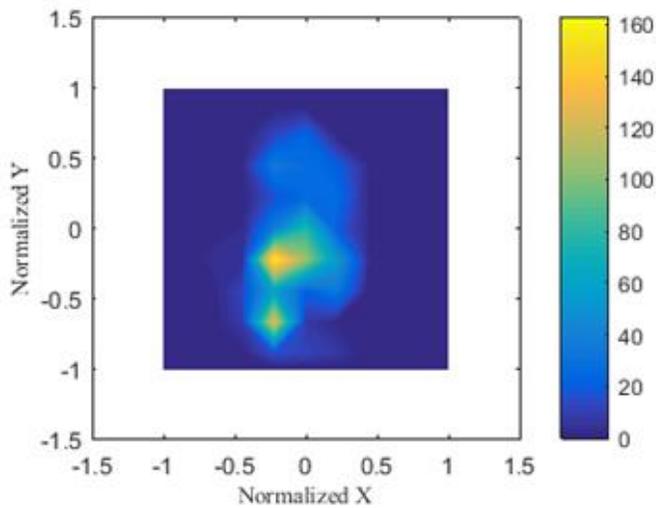


Figure 6.1. Normalized relative positions of human driver with respect to the safe driving rectangle which is defined 1 step (0.1s) ahead.

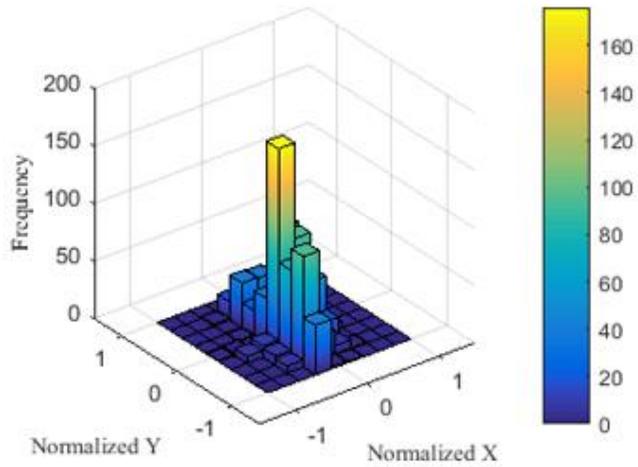


(a) Histogram

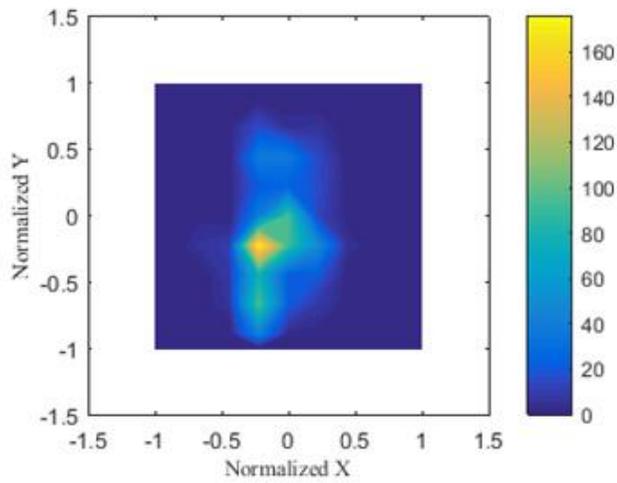


(b) Contour

Figure 6.2. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 1 step (0.1s) ahead.

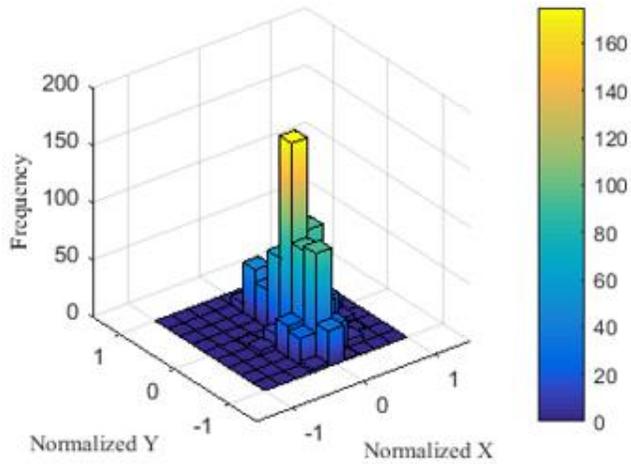


(a) Histogram

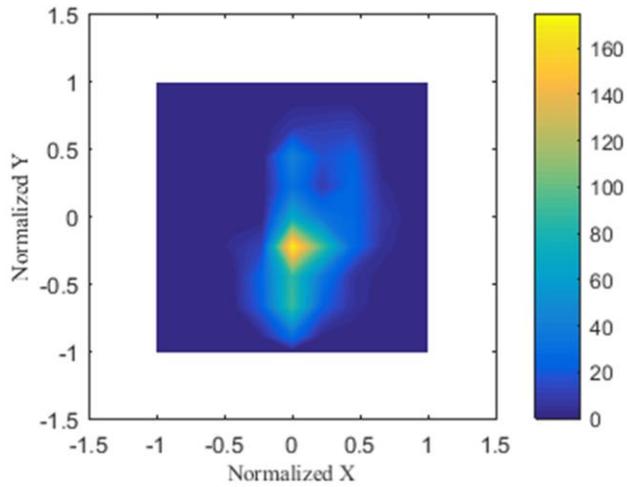


(b) Contour

Figure 6.3. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 3 steps (0.3s) ahead.

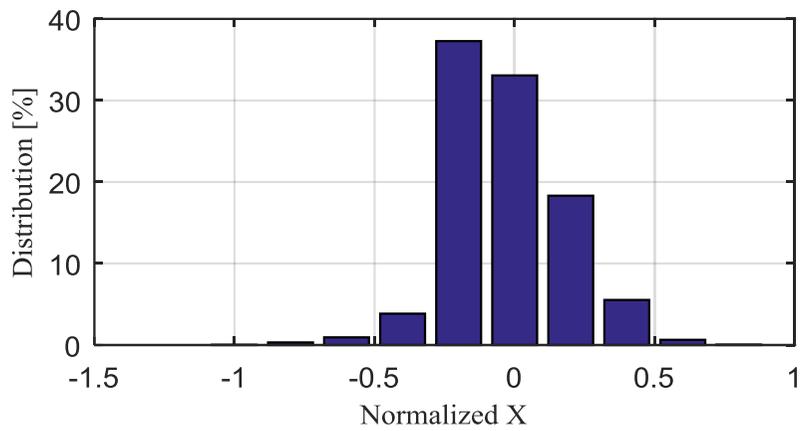


(a) Histogram

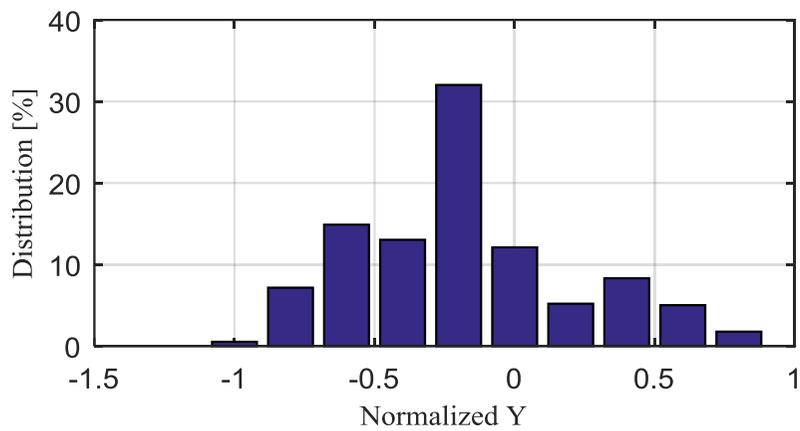


(b) Contour

Figure 6.4. Histogram and contours of normalized human driver's relative positions with respect to the safe driving rectangle which is defined 5 steps (0.5s) ahead.



(a) Distribution of normalized-x position



(b) Distribution of normalized-y position

Figure 6.5. Distribution of normalized human driver's relative position with respect to the safe driving rectangle defined 1~5 step (0.1s~0.5s) ahead.

6.2. Vehicle Tests: Automated Driving on Urban Roads

Vehicle tests have been conducted for several times at the internal road of Seoul National University. The details of test roads are depicted in Figure 5.6. The given test roads have quite complicated environments to drive automatically. As can be seen in Figure 6.6, the test vehicle drives the given route fully autonomously without a driver manipulation. The lanes were hard to be distinguished because the paint was worn off in places. We need to consider other traffic participants like oncoming and preceding vehicles while there are a lot of buses parked along the road as shown in Figure 6.6. Also, there exist non-vehicle obstacles such as pedestrian and guardrail.

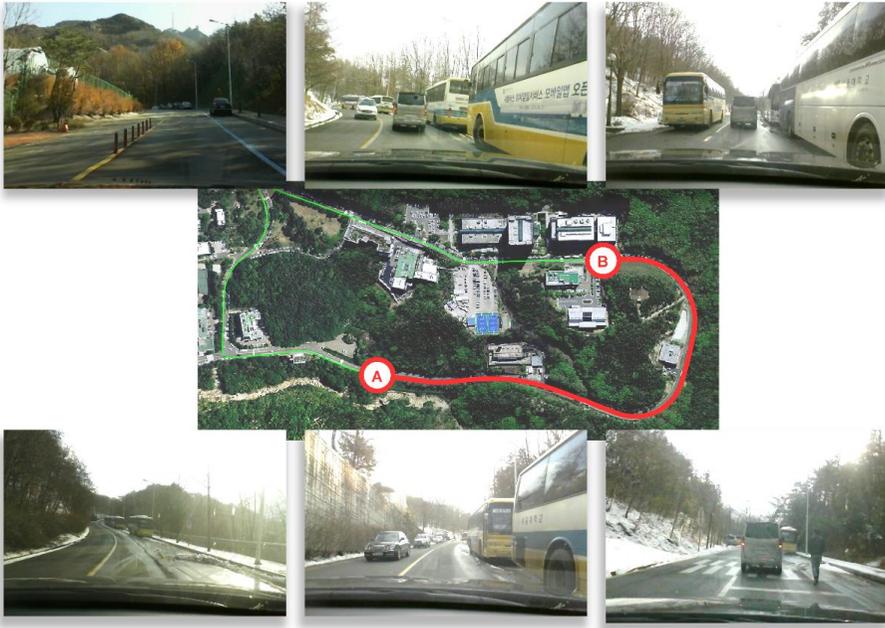


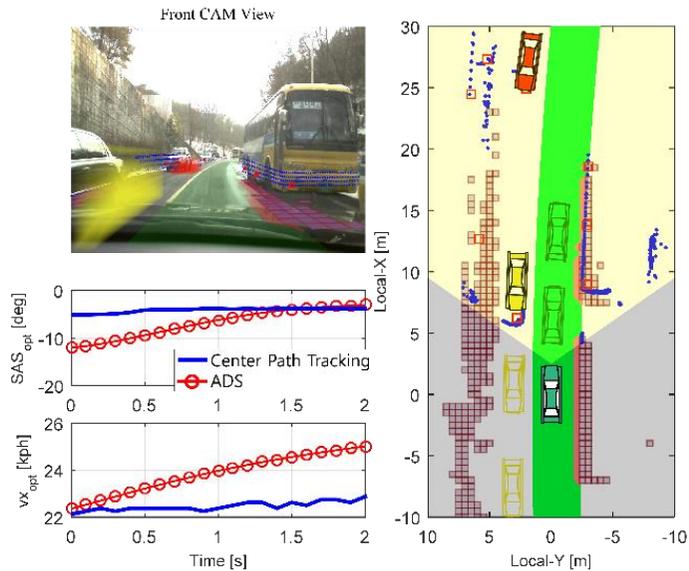
Figure 6.6. Configuration of test route in SNU campus (2km). The route contains a variety of different traffic situations as e.g. narrow passages, curved roads and inclined roads with other traffic participants such as oncoming vehicles, preceding vehicle, and parked busses and non-vehicle obstacles such as guardrails, pedestrian crossings, etc.

The proposed automated driving algorithm has shown the satisfactory control performance and the test results are given in Figure 6.7. As shown in Figure 6.7, the ego-vehicle drives through a narrow urban environment with static infrastructure (buildings, trees, poles, etc.), parking cars on the right, and a preceding vehicle. Furthermore comparisons with center path tracking with the detected lane (without information fusion with GPS/Map) are given. Control input sequences from MPC solver are depicted as red line with circle marker and center-path tracking inputs as blue solid line.

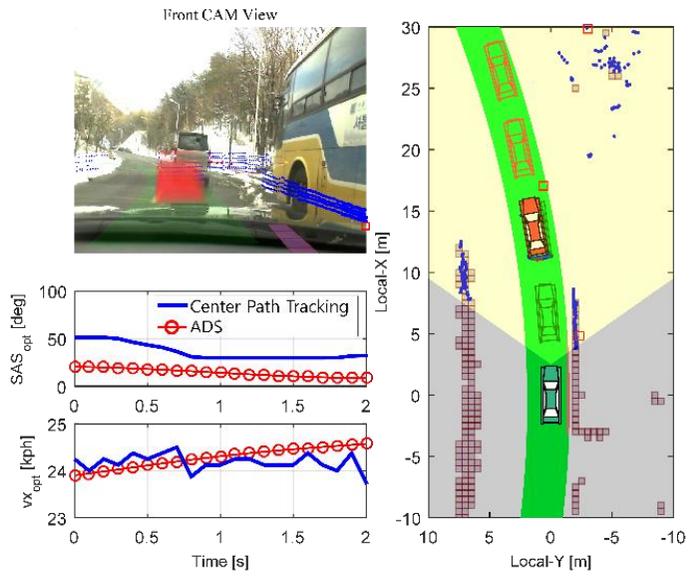
Figure 6.7-(a) shows the result at a driving scene with two oncoming vehicles with right side static obstacles. From the figure, we can see that MPC approach guarantee the safety with oncoming vehicle by giving defense steering command. This results suggest that proposed algorithm may enhance the driver acceptance and driver comfort. Figure 6.7-(b) obtained from the situation with a preceding vehicle at the curved roads. In the case of the normal lane keeping, the proposed algorithm shows the smaller range of control while the center path tracking shows the frequent changes. Figure 6.7-(c) provides the driving scene with a preceding vehicle with both side obstacles (parked buses). Also in this case, the proposed algorithm shows the smaller range of control with higher speed while the center path tracking shows the frequent changes of steering with lower speed. Figure 6.7-(d) presents the stop control behind the decelerating preceding vehicle. In this situation, the lane is invisible due to close preceding vehicle. Consequently, the center path tracking shows the weird steering input while the proposed algorithm gives a stable performance. From the test results, it is apparent that the test vehicle successfully completed the test route with the proposed automated driving algorithm.

To verify a safety performance of the proposed algorithm, normalized relative positions of ADS with respect to this safe driving envelope rectangle have been analyzed. Figure 6.8 shows the histogram and contour of the normalized relative position data of ADS. As can be seen in Figure 6.8, the ADS control the ego-vehicle to absolutely stay within the safe driving rectangle. However, it is note that the relative positions do not converge to the center but only stay close enough to the center. As aforementioned, it is a good solution

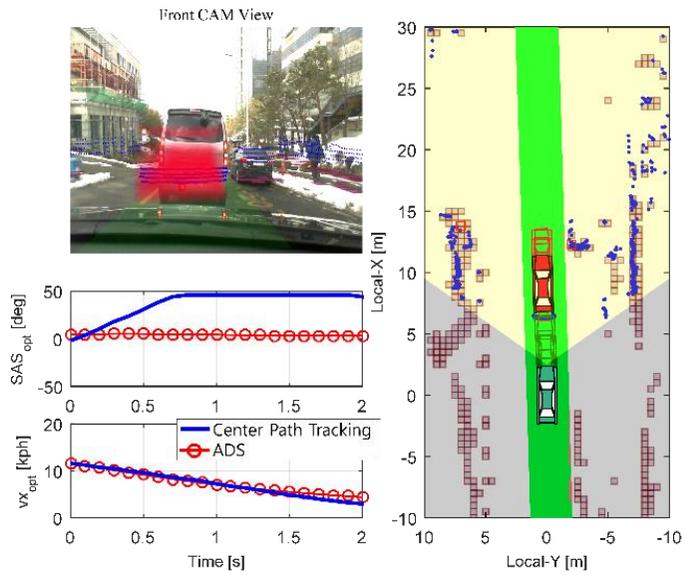
for a driver acceptance and a ride comfort. And Figure 6.9 shows the distributions of normalized relative position of each direction. As shown in Figure 6.9, the range of relative lateral position has the larger magnitude as in the case of human driver. These results once again stress that the proposed algorithm can represent the driver characteristics that do not tracking the center of the drivable area but only stay in drivable area with a wide range of choice.



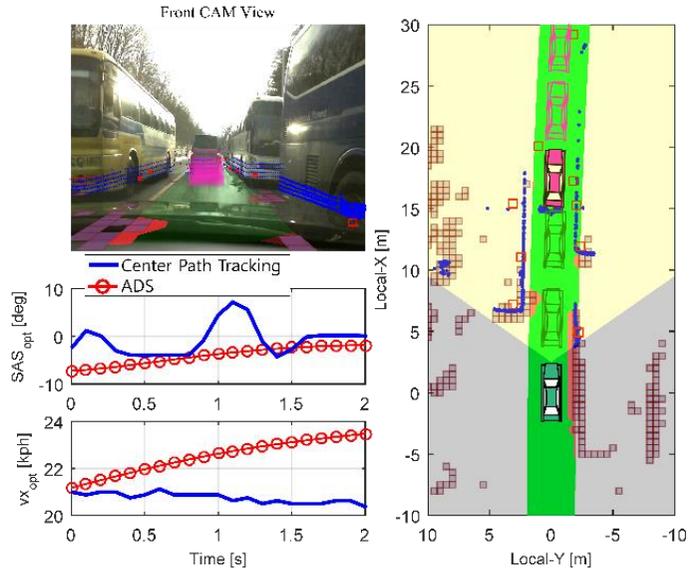
(a) Two oncoming vehicles



(b) Preceding vehicle at curved roads

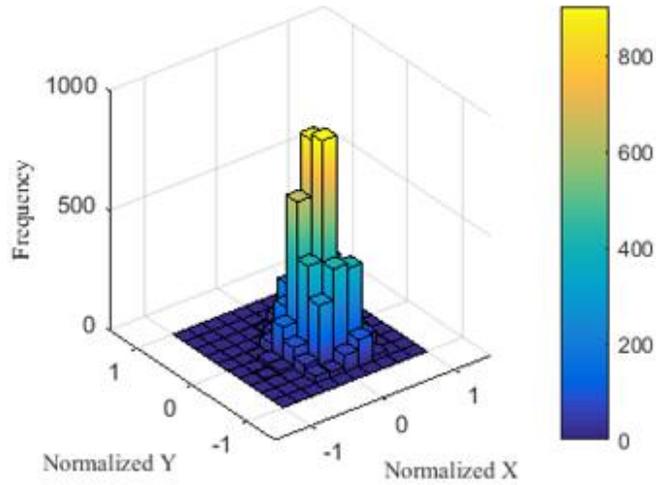


(c) Stop behind the stopped preceding vehicle

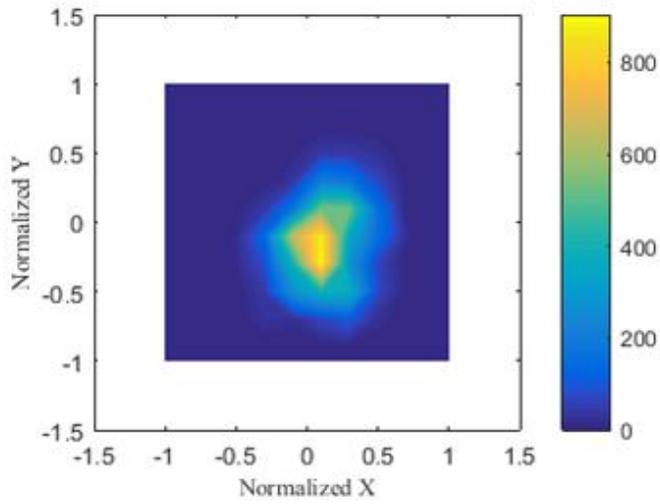


(d) Preceding vehicle with both side obstacles (parked buses)

Figure 6.7. Test results with proposed automated driving algorithm

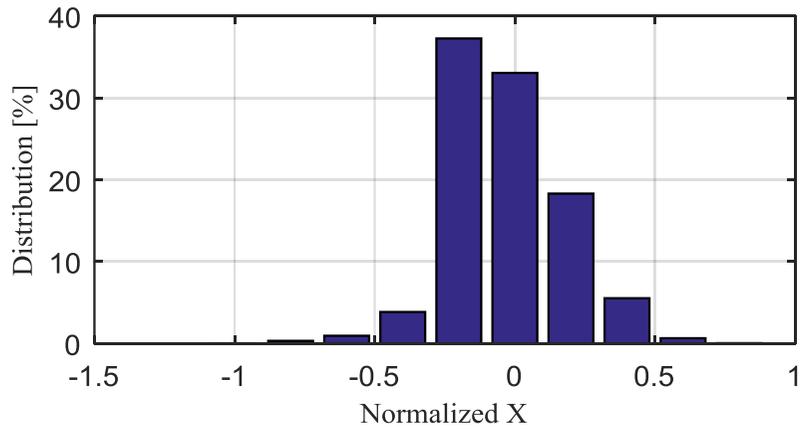


(a) Histogram

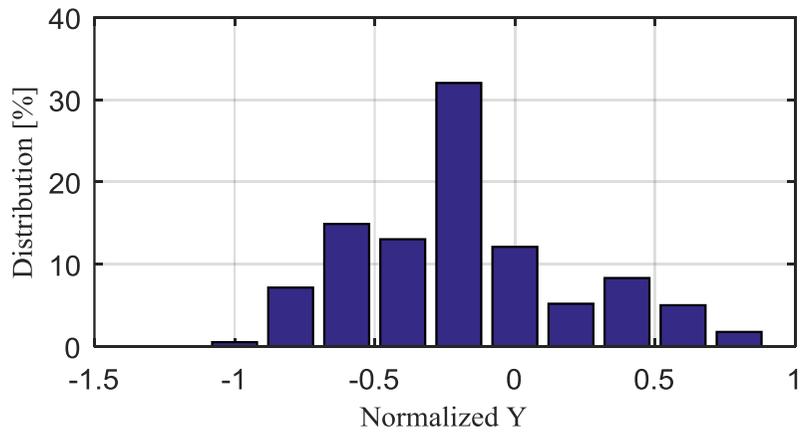


(b) Contour

Figure 6.8. Histogram and contours of normalized relative position of ADS with respect to the safe driving rectangles which are defined 1~5-step ahead.



(a) Normalized-x position



(b) Normalized-y position

Figure 6.9. Distribution of normalized relative position of ADS with respect to the safe driving rectangles which are defined 1~5-step ahead.

Chapter 7 Conclusions and Future Works

This dissertation has proposed a fully automated driving algorithm which is capable of automated driving on urban roads with guaranteed safety. The proposed algorithm consisted of the following three steps: an environment representation, a motion optimization, and a vehicle control. In an environment representation, algorithms for lane-level localization, static/moving obstacle detection, and drivable area representation for safe and comfortable autonomous driving has been developed. And a motion optimization algorithm which is separated into two distinct tasks, safety driving envelope decision and motion optimization, has been developed. The developed motion optimization algorithm solves a geometric constraint problem as a convex optimization problem with linear equality/inequality constraints.

The effectiveness of the proposed automated driving algorithm has been evaluated via test-data based simulations and vehicle tests. In the case of the IMM/EKF based multi-target state estimation algorithm, it is found that the mode probability adaptations are quite reasonable and the proposed algorithm can provide good estimates of the target vehicle's overall behaviors. To investigate the performance enhancement, comparison with model-switching algorithm which is a simple approach to handle the multiple-model problem has been conducted. It is shown that the target state estimation performance can

be significantly enhanced by the proposed algorithm with approximately three times more accurate results while maintain real-time performance with reasonable computation complexity. The IMM/EKF algorithm makes possible the generic assessment and the total management of collision risks with multi-target vehicles in complex driving situations for high-level automated driving. From the results it has been shown that proposed automated driving algorithm can provide the robust performance on an inner-city street scenario.

A further improvement of the autonomous vehicle's ability to interpret a given traffic scenario and to obtain a meaningful behavior prediction of other traffic participants is imperative to achieve incomparable performance and to extend the covering area of autonomous driving.

Bibliography

- ALTHOFF, M., STURSBURG, O. & BUSS, M. 2009. Model-based probabilistic collision detection in autonomous driving. *Intelligent Transportation Systems, IEEE Transactions on*, 10, 299-310.
- ANDERSON, S. J., PETERS, S. C., PILUTTI, T. E. & IAGNEMMA, K. 2010. An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios. *International Journal of Vehicle Autonomous Systems*, 8, 190-216.
- BALAKRISHNAN, N. & CHEN, W. W. 1999. *Handbook of tables for order statistics from lognormal distributions with applications*, Springer Science & Business Media.
- BISHOP, R. A survey of intelligent vehicle applications worldwide. Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE, 2000. IEEE, 25-30.
- BLACKMAN, S. S. 2004. Multiple hypothesis tracking for multiple target tracking. *Ieee Aerospace and Electronic Systems Magazine*, 19, 5-18.
- BORRELLI, F., FALCONE, P., KEVICZKY, T., ASGARI, J. & HROVAT, D. 2005. MPC-based approach to active steering for autonomous vehicle systems. *International Journal of Vehicle Autonomous Systems*, 3, 265-291.
- CARVALHO, A., GAO, Y., GRAY, A., TSENG, H. E. & BORRELLI, F. Predictive control of an autonomous ground vehicle using an iterative linearization approach. Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on, 2013. IEEE, 2335-2340.
- CHEN, B. & TUGNAIT, J. K. 2001. Tracking of multiple maneuvering targets in clutter using IMM/JPDA filtering and fixed-lag smoothing. *Automatica*, 37, 239-249.
- ERLIEN, S., FUJITA, S. & GERDES, J. C. Safe driving envelopes for shared control of ground vehicles. *Advances in Automotive Control*, 2013. 831-836.

- ERLIEN, S. M., FUNKE, J. & GERDES, J. C. Incorporating non-linear tire dynamics into a convex approach to shared steering control. American Control Conference (ACC), 2014, 2014. IEEE, 3468-3473.
- ESKANDARIAN, A. 2012. *Handbook of intelligent vehicles*, Springer London.
- FAIRCLOUGH, S. H., MAY, A. J. & CARTER, C. 1997. The effect of time headway feedback on following behaviour. *Accident Analysis & Prevention*, 29, 387-397.
- FALCONE, P., BORRELLI, F., ASGARI, J., TSENG, H. E. & HROVAT, D. 2007a. Predictive active steering control for autonomous vehicle systems. *Control Systems Technology, IEEE Transactions on*, 15, 566-580.
- FALCONE, P., BORRELLI, F., TSENG, H. E., ASGARI, J. & HROVAT, D. 2008. Linear time-varying model predictive control and its application to active steering systems: Stability analysis and experimental validation. *International journal of robust and nonlinear control*, 18, 862-875.
- FALCONE, P., TUFO, M., BORRELLI, F., ASGARI, J. & TSENG, H. E. A linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems. Decision and Control, 2007 46th IEEE Conference on, 2007b. IEEE, 2980-2985.
- FERGUSON, D. I. F. & DOLGOV, D. A. 2014. Modifying behavior of autonomous vehicle based on predicted behavior of other vehicles. Google Patents.
- FURGALE, P., SCHWESINGER, U., RUFLI, M., DERENDARZ, W., GRIMMETT, H., MUHLFELLNER, P., WONNEBERGER, S., TIMPNER, J., ROTTMANN, S. & LI, B. Toward automated driving in cities using close-to-market sensors: An overview of the V-Charge Project. Intelligent Vehicles Symposium (IV), 2013 IEEE, 2013. IEEE, 809-816.
- GAO, Y., GRAY, A., TSENG, H. E. & BORRELLI, F. 2014. A tube-based robust nonlinear predictive control approach to semiautonomous ground vehicles. *Vehicle System Dynamics*, 52, 802-823.
- GORDON, T., SARDAR, H., BLOWER, D., LJUNG AUST, M., BAREKET, Z., BARNES, M., BLANKESPOOR, A., ISAKSSON-HELLMAN, I.,

- IVARSSON, J. & JUHAS, B. 2010. Advanced crash avoidance technologies (ACAT) program–Final report of the Volvo-Ford-UMTRI project: safety impact methodology for lane departure warning–Method development and estimation of benefits.
- GRAY, A., ALI, M., GAO, Y., HEDRICK, J. K. & BORRELLI, F. 2012. Semi-autonomous vehicle control for road departure and obstacle avoidance. *IFAC Control of Transportation Systems*, 1-6.
- GRAY, A., GAO, Y., HEDRICK, J. K. & BORRELLI, F. Robust predictive control for semi-autonomous vehicles with an uncertain driver model. Intelligent Vehicles Symposium (IV), 2013 IEEE, 2013. IEEE, 208-213.
- HOEDEMAEKER, M. & BROOKHUIS, K. A. 1998. Behavioural adaptation to driving with an adaptive cruise control (ACC). *Transportation Research Part F: Traffic Psychology and Behaviour*, 1, 95-106.
- HUANG, S., REN, W. & CHAN, S. C. 2000. Design and performance evaluation of mixed manual and automated control traffic. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 30, 661-673.
- KASTNER, R., MICHALKE, T., ADAMY, J., FRITSCH, J. & GOERICK, C. 2011. Task-based environment interpretation and system architecture for next generation ADAS. *Intelligent Transportation Systems Magazine, IEEE*, 3, 20-33.
- KATO, S., TSUGAWA, S., TOKUDA, K., MATSUI, T. & FUJII, H. 2002. Vehicle control algorithms for cooperative driving with automated vehicles and intervehicle communications. *Ieee Transactions on Intelligent Transportation Systems*, 3, 155-161.
- KAZANTZIS, N. & KRAVARIS, C. 1999. Time-discretization of nonlinear control systems via Taylor methods. *Computers & chemical engineering*, 23, 763-784.
- KIM, B. & YI, K. 2014. Probabilistic and Holistic Prediction of Vehicle States Using Sensor Fusion for Application to Integrated Vehicle Safety Systems. *Intelligent Transportation Systems, IEEE Transactions on*, 15, 2178-2190.

- KOTHARE, M. V., BALAKRISHNAN, V. & MORARI, M. 1996. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32, 1361-1379.
- LAUGIER, C., PAROMTCHIK, I. E., PERROLLAZ, M., YONG, M., YODER, J.-D., TAY, C., MEKHNACHA, K. & N GRE, A. 2011. Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety. *Intelligent Transportation Systems Magazine, IEEE*, 3, 4-19.
- LEE, J., CHOI, J., YI, K., SHIN, M. & KO, B. 2014. Lane-keeping assistance control algorithm using differential braking to prevent unintended lane departures. *Control Engineering Practice*, 23, 1-13.
- LEE, T., YI, K. & JEONG, C. 2011. Integrated stochastic driver model for evaluation of the vehicle active safety systems. *Fast-zero 2011*.
- LERRO, D. & BARSHALOM, Y. 1993. Tracking with Debiased Consistent Converted Measurements Versus EKF. *Ieee Transactions on Aerospace and Electronic Systems*, 29, 1015-1022.
- LI, X. R. & BAR-SHALOM, Y. 1993. Design of an interacting multiple model algorithm for air traffic control tracking. *Control Systems Technology, IEEE Transactions on*, 1, 186-194.
- MATTINGLEY, J. & BOYD, S. 2012. CVXGEN: a code generator for embedded convex optimization. *Optimization and Engineering*, 13, 1-27.
- MAYNE, D. Q., RAWLINGS, J. B., RAO, C. V. & SCOKAERT, P. O. 2000. Constrained model predictive control: Stability and optimality. *Automatica*, 36, 789-814.
- MOON, S., MOON, I. & YI, K. 2009. Design, tuning, and evaluation of a full-range adaptive cruise control system with collision avoidance. *Control Engineering Practice*, 17, 442-455.
- MOON, S. & YI, K. 2008. Human driving data-based design of a vehicle adaptive cruise control algorithm. *Vehicle System Dynamics*, 46, 661-690.

- NARANJO, J. E., GONZALEZ, C., GARCIA, R. & DE PEDRO, T. 2008. Lane-change fuzzy control in autonomous vehicles for the overtaking maneuver. *Intelligent Transportation Systems, IEEE Transactions on*, 9, 438-450.
- NETTO, M. S., CHAIB, S. & MAMMAR, S. Lateral adaptive control for vehicle lane keeping. American Control Conference, 2004. Proceedings of the 2004, 2004. IEEE, 2693-2698.
- PACEJKA, H. 2005. *Tire and vehicle dynamics*, Elsevier.
- POHL, J., BIRK, W. & WESTERVALL, L. 2007. A driver-distraction-based lane-keeping assistance system. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 221, 541-552.
- POLYCHRONOPOULOS, A., TSOGAS, M., AMDITIS, A. J. & ANDREONE, L. 2007. Sensor fusion for predicting vehicles' path for collision avoidance systems. *Intelligent Transportation Systems, IEEE Transactions on*, 8, 549-562.
- RE, K. 1960. A New Approach to Linear Filtering and Prediction. *Transaction of the ASME-Journal of Basic Engineering*, 35-45.
- ROSSETTER, E. J. & GERDES, J. C. 2006. Lyapunov based performance guarantees for the potential field lane-keeping assistance system. *Journal of dynamic systems, measurement, and control*, 128, 510-522.
- SATTEL, T. & BRANDT, T. 2008. From robotics to automotive: Lane-keeping and collision avoidance based on elastic bands. *Vehicle System Dynamics*, 46, 597-619.
- SKOLNIK, M. I. 2001. *Radar Systems*. McGraw-Hill, NY.
- SWARTZ, D. Clothoid road geometry unsuitable for sensor fusion. IEEE Intelligent Vehicles Symposium, 2003. 484-488.
- TALVALA, K. L., KRITAYAKIRANA, K. & GERDES, J. C. 2011. Pushing the limits: From lanekeeping to autonomous racing. *Annual Reviews in Control*, 35, 137-148.

- TIDEMAN, M., VAN DER VOORT, M., VAN AREM, B. & TILLEMA, F. A review of lateral driver support systems. *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE, 2007. IEEE, 992-999.*
- TINGVALL, C. & HAWORTH, N. Vision Zero: an ethical approach to safety and mobility. *6th ITE International Conference Road Safety & Traffic Enforcement: Beyond, 2000.*
- VANHOLME, B., GRUYER, D., LUSETTI, B., GLASER, S. & MAMMAR, S. 2013. Highly automated driving on highways based on legal safety. *Intelligent Transportation Systems, IEEE Transactions on, 14, 333-347.*
- YUAN, T., BAR-SHALOM, Y., WILLETT, P., MOZESON, E., POLLAK, S. & HARDIMAN, D. 2012. A multiple IMM estimation approach with unbiased mixing for thrusting projectiles. *Aerospace and Electronic Systems, IEEE Transactions on, 48, 3250-3267.*
- ZHANG, Y. & BAR-SHALOM, Y. 2011. Tracking move-stop-move targets with state-dependent mode transition probabilities. *Aerospace and Electronic Systems, IEEE Transactions on, 47, 2037-2054.*
- ZIEGLER, J., BENDER, P., SCHREIBER, M., LATEGAHN, H., STRAUSS, T., STILLER, C., DANG, T., FRANKE, U., APPENRODT, N. & KELLER, C. G. 2014. Making bertha drive—An autonomous journey on a historic route. *Intelligent Transportation Systems Magazine, IEEE, 6, 8-20.*

초 록

주행 안전 및 승차감 향상을 위한 강건 모델 예측 기법 기반 자율 주행 제어 알고리즘 개발

최근 차량용 센싱기술이 급진적으로 발전함에 따라, 차량 연구자들의 관심사는 수동안전시스템(Passive Safety System)에서 능동안전시스템(Active Safety System)으로, 그리고 그 개념이 확장되어 자율주행시스템으로 이동하였다. 예를 들어 적응형 순항 제어 시스템 (ACC: Adaptive Cruise Control), 차선 유지 보조장치 (LKAS: Lane Keeping Assistance System), 차선 변경 보조장치 (LCAS: Lane Change Assistance System), 자동주차 보조시스템 (APA: automated Parking Assist System) 그리고 사각지대 감지경보 (BSI: Blind Spot Intervention) 등의 다양한 능동안전시스템들이 이미 차량제조사들에 의해 출시되었다. 이러한 능동안전시스템들의 출시와 더불어 운전자의 완전한 안전을 보장하기 위한 다양한 프로젝트들이 진행되고 있다. 전세계적으로 다양한 연구팀들이 지속적으로 자율주행분야에 대한 연구를 진행 중이고, 차량제조사들은 개별시스템들을 통합하여 안전기능을 강화하는 연구를 진행하고 있다.

다수의 참고문헌들을 살펴본 결과, 자율주행기술은 잠재적으로 교통혼잡을 완화하고 운전자의 안전을 증진할 것으로 기대된다. 하지만 현재의 자율주행기술은 위성항법보정시스템 (DGPS: Differential Global Positioning System), 다층레이저스캐너 (Velodyne) 등의 고가의 정밀센서가 요구되어 시장에 출시되기에는 비용적인 측면에서 한계점을 가지고 있다. 기술의 발전에 따라 센서들의 가격이 낮아지고 있지만, 차량에 센서들을 장착하고 적용하는 것은 자율주행개발에 있어서 하나의 장벽이 되고 있다.

따라서 본 논문에서는 현재 양산된 차량용 센서들을 조합하여 가격적으로 경제적이고 기술적으로 완성된 센서구성을 활용하여 복잡다차량환경에서의 자율주행이 가능한 시스템을 개발하는 것을 목표로 한다. 전체 알고리즘은 크게 환경인지모듈, 안전영역판단모듈, 그리고 최적요구모션결정모듈로 구성된다.

환경인지모듈에서는 레이더, 라이다, 그리고 차량샤시센서 정보를 활용하여 도로영역, 정지장애물, 이동장애물 등을 인지하고, 안전영역판단모듈에서는 인지된 정보들을 종합하여 자차량이 안전하게 주행 가능한 영역을 유한한 시간 내에서 동적으로 결정한다. 마지막으로 최적요구모션결정 모듈에서는 모델 예측 제어 기법 (MPC: Model Predictive Control)을 활용하여 최적모션을 계산한다.

제안된 자율주행제어 알고리즘의 성능은 실차실험을 통해 검증되었다. 실험결과를 통해 시내주행에서의 인지성능, 판단성능 그리고 안전주행성능을 확인하였고, 일반도로에서 발생 가능한 다양한 시나리오에서 위 성능들이 강건하게 확보됨을 확인하였다.

주요어: 자율 주행 자동차, 지능형 안전 자동차, 모델 예측 제어 기법, 강건 제어, 안전 주행 영역 판단, 확률적 거동 예측, 자율 주행 제어 알고리즘, 통합 안전 제어

학 번: 2011-20689