



รายงานการวิจัย

เรื่อง

การสร้างแบบจำลองเพื่อทำนายเส้นทางโคจรของรถยนต์ส่วนบุคคล
ที่มีการขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรงและโค้งโดยใช้ตัวกรองคาลมาน

TRAJECTORY PREDICTION MODELING OF CAR TRAVELING

THROUGH STRAIGHT AND CURVED LANE BOUNDARIES

USING KALMAN FILTER

ชฎาพร เกตุมณี

งานวิจัยนี้ได้รับทุนอุดหนุนการวิจัยจากมหาวิทยาลัยศรีปทุม

ปีการศึกษา 2555



รายงานการวิจัย

เรื่อง

การสร้างแบบจำลองเพื่อทำนายเส้นทางโคจรของรถยนต์ส่วนบุคคล
ที่มีการขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรงและโค้งโดยใช้ตัวกรองคาลมาน

TRAJECTORY PREDICTION MODELING OF CAR TRAVELING

THROUGH STRAIGHT AND CURVED LANE BOUNDARIES

USING KALMAN FILTER

ชฎาพร เกตุมณี

งานวิจัยนี้ได้รับทุนอุดหนุนการวิจัยจากมหาวิทยาลัยศรีปทุม

ปีการศึกษา 2555

คำนำ

รายงานการวิจัยฉบับนี้จัดทำขึ้นเพื่อเป็นประโยชน์ต่อผู้ที่สนใจจะศึกษาเกี่ยวกับเรื่อง การสร้างแบบจำลองทางคณิตศาสตร์เพื่อทำนายเส้นทางโคจรของรถยนต์ส่วนบุคคล ที่มีการขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรงและโค้งโดยใช้ตัวกรองคาลมาน ซึ่งผู้วิจัยได้จัดทำขึ้นเพื่อให้ นักวิจัยที่สนใจเกี่ยวกับตัวกรองสัญญาณดิจิทัลได้ใช้เป็นเอกสารอ่านเพิ่มเติมและอ้างอิงต่อไป

ผู้วิจัยหวังเป็นอย่างยิ่งว่า รายงานการวิจัยฉบับนี้จะเป็นประโยชน์ต่อผู้ที่สนใจเกี่ยวกับการสร้างแบบจำลองทางคณิตศาสตร์ด้วยตัวกรองคาลมาน ซึ่งทำให้ทราบถึงหลักการในการลดสัญญาณรบกวนที่เกิดขึ้นในระบบอันเนื่องมาจากสิ่งแวดล้อม หรือความบกพร่องของเซ็นเซอร์ที่ใช้เป็นต้น รวมทั้งหลักการทางเบื้องต้นของระบบตรวจจับเส้นทางที่ผู้วิจัยได้วิเคราะห์และแสดงตัวอย่างการประมาณค่าโดยใช้ตัวกรองคาลมานให้ผู้อ่านได้นำมาใช้ประโยชน์ได้ต่อไปหากมีข้อบกพร่องประการใด ผู้จัดทำขออภัยไว้ ณ โอกาสนี้ด้วย

ชฎาพร เกตุมณี

พฤศจิกายน พ.ศ. 2556

กิตติกรรมประกาศ

รายงานการวิจัยฉบับนี้สำเร็จและสมบูรณ์เป็นรูปเล่มด้วยความกรุณาและเอาใจใส่เป็นอย่างดีจาก Assoc. Prof. Matthew N. Dailey ผู้ทรงคุณวุฒิที่ได้กรุณาให้คำปรึกษาและแนะแนวทางในการดำเนินการทำรายงานในครั้งนี้โดยไม่มีข้อบกพร่องรวมทั้งข้อเสนอแนะและข้อคิดเห็นต่าง ๆ ตลอดจนการตรวจแก้ไขรายงานฉบับนี้ให้สำเร็จสมบูรณ์ยิ่งขึ้น ทางผู้วิจัยจึงขอขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอขอบพระคุณคุณครูทุกท่านที่ได้ประสิทธิ์ประสาทวิชา ความรู้ และประสบการณ์ ตลอดจนอำนวยความสะดวกสำเร็จให้บังเกิดขึ้น

สุดท้ายนี้ขอขอบคุณคุณคณาจารย์และเจ้าหน้าที่ ทีมมหาวิทยาลัยศรีปทุมและสถาบันเอเชียที่เป็นกำลังใจและให้ความช่วยเหลือในการเก็บรวบรวมข้อมูลรวมทั้งให้คำแนะนำในการทำรายงานการวิจัยครั้งนี้ให้สำเร็จลุล่วงด้วยดีตลอดมา

ชฎาพร เกตุมณี

พฤศจิกายน พ.ศ. 2556

หัวข้อวิจัย : การสร้างแบบจำลองเพื่อทำนายเส้นทางโคจรของรถยนต์ส่วนบุคคลที่มีการ
ขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรงและโค้งโดยใช้ตัวกรองคาลมาน

ผู้วิจัย : นางสาวชญาพร เกตุมณี

หน่วยงาน : มหาวิทยาลัยศรีปทุม วิทยาเขตบางเขน

ปีที่พิมพ์ : พ.ศ. 2556

บทคัดย่อ

เนื่องจากมีผู้เสียชีวิตและผู้บาดเจ็บเพิ่มขึ้นทุกๆปีจากอุบัติเหตุทางรถยนต์ จึงทำให้มีผู้วิจัยเป็นจำนวนมากให้ความสนใจในการพัฒนาระบบการขนส่งและจราจรอัจฉริยะ (ITS) การวิจัยชิ้นนี้เป็นส่วนหนึ่งในการพัฒนาระบบตรวจจับเลนที่ใช้ระบบ Machine vision และเซ็นเซอร์แบบต่างๆ ในการตรวจจับเส้นแบ่งเลนในทุกขณะการขับเคลื่อนของรถยนต์ ซึ่งมักจะได้รับผลกระทบจากสัญญาณรบกวนที่เกิดจากสภาพแวดล้อม เช่น ฝนตก หมอก หรือแม้กระทั่งเงาของเสาไฟฟ้า เพื่อให้ข้อมูลที่ได้จากระบบตรวจจับเส้นเลนมีความถูกต้องแม่นยำและสามารถนำไปใช้ในระบบขับอัตโนมัติ หรือระบบผู้ช่วยคนขับต่อไปได้ บทความการวิจัยชิ้นนี้ได้นำเสนอแนวคิดในการออกแบบจำลองที่มีความเที่ยงตรงเพื่อใช้ในการทำนายเส้นทางโคจรของรถยนต์ส่วนบุคคลที่มีการขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรงและโค้งโดยใช้ตัวกรองคาลมาน ซึ่งแตกต่างจากการวิจัยก่อนหน้านี้ที่นิยมสร้างแบบจำลองสำหรับการขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรงเพียงอย่างเดียว ซึ่งแบบจำลองทางคณิตศาสตร์นี้ถูกออกแบบมาเพื่อใช้ในการระบบตรวจจับเลนที่มีสัญญาณรบกวนหรือการขาดหายไปของข้อมูลโดยใช้ตัวกรองคาลมานเป็นอัลกอริทึมในการลดสัญญาณรบกวนและเพิ่มความถูกต้องแม่นยำของข้อมูล

คำสำคัญ: แบบจำลองทางคณิตศาสตร์ ตัวกรองคาลมาน การทำนายเส้นทางโคจรของรถยนต์ ระบบผู้ช่วยคนขับ ระบบขนส่งและจราจรอัจฉริยะ

Research Title : Trajectory Prediction Modeling of Car Traveling Through Straight and
Curved Lane Boundaries Using Kalman Filter

Name of Researcher : Miss. Chadaporn Keatmanee

Name of Institution : Sripatum University, Bangkhen Campus

Year of Publication : B.E. 2556

ABSTRACT

Due to rise in fatalities and serious injuries because of car accidents many people have paid attention to doing research on Intelligent Transport System (ITS). These researches especially deal with lane detection system using combination of machine vision with the fusion of sensors for obtaining data of road lane boundary detection and vehicle movement in every state to improve data from noisy environment. And hence these data can be used for future work whether for lane departure warning system or automatic car system. This paper proposes to develop a precised lane boundary estimation model which can predict the lateral distance while a vehicle is moving along the road lane boundary, whether curve or straight lane boundary, with that of a different vehicle's movement including straight and circular lanes. This model is different from previous models which were only able to estimate if the lane is straight and flat only. The model is developed under assumption of noisy data due to environment and the Extended Kalman Filter (EKF) is applied to improve the accuracy of the data.

Keywords: Trajectory Model, Kalman Filter, Intelligent Transport System, Driver Awareness,
Lane Detection System

สารบัญ

บทที่	หน้า
1	บทนำ..... 1 ความเป็นมาและความสำคัญของปัญหา..... 1 วัตถุประสงค์ของการวิจัย..... 4 คำถามการวิจัย..... 4 สมมุติฐานการวิจัย..... 4 ขอบเขตของการวิจัย..... 5 นิยามศัพท์ 5
2	วรรณกรรมที่เกี่ยวข้อง..... 7 ความรู้พื้นฐาน..... 7 กรอบความคิดทางทฤษฎี 11 ผลการวิจัยที่เกี่ยวข้อง..... 14 สรุป..... 18
3	ระเบียบวิธีวิจัย..... 20 ระเบียบวิธีวิจัย..... 20 ขั้นตอนการดำเนินงานวิจัย..... 20 การรวบรวมข้อมูล..... 25 การวิเคราะห์ข้อมูล..... 26

สารบัญ(ต่อ)

บทที่		หน้า
4	ผลการวิเคราะห์ข้อมูล.....	27
	ผลการจำลองการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงและโค้งผ่านเส้น แบ่งเลนที่มีลักษณะเป็นเส้นตรง.....	29
	ผลการจำลองการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงและโค้งผ่านเส้น แบ่งเลนที่มีลักษณะเป็นเส้น โค้ง.....	30
	ผลการจำลองการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงและโค้งผ่านเส้น แบ่งเลนที่มีลักษณะเป็นเส้นตรงขณะมีการสูญหายของสัญญาณ.....	31
	ผลการจำลองการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงและโค้งผ่านเส้น แบ่งเลนที่มีลักษณะเป็นเส้น โค้งขณะมีการสูญหายของสัญญาณ.....	32
	ค่า RMSE ที่ได้จำลองการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงและโค้งผ่านเส้น แบ่งเลนที่มีลักษณะเป็นเส้นตรงและ โค้ง.....	33
	ค่า RMSE ที่ได้จำลองการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงและโค้งผ่านเส้น แบ่งเลนที่มีลักษณะเป็นเส้นตรงและ โค้งขณะมีการสูญหายของสัญญาณ.....	33
5	สรุป อภิปราย และข้อเสนอแนะ.....	35
	สรุปผลการวิจัย.....	35
	อภิปรายผล.....	35
	ข้อเสนอแนะ.....	35

สารบัญ(ต่อ)

บทที่	หน้า
บรรณานุกรม	36
ภาคผนวก	37
ภาคผนวก ก โค้ด Matlab	37
ประวัติย่อผู้วิจัย	72

สารบัญภาพประกอบ

ภาพประกอบ	หน้า
1 ตัวอย่างการวัดระยะห่างระหว่างขอบเส้นแบ่งเลนกับรถยนต์ทั้งทางด้านซ้าย (d_l) และด้านขวา (d_r) ของรถยนต์ขณะขับเคลื่อนในลักษณะเส้นตรงผ่านเส้นแบ่งเลนที่เป็นเส้นตรง	3
2 ตัวอย่างการวัดระยะห่างระหว่างขอบเส้นแบ่งเลนกับรถยนต์ทั้งทางด้านซ้าย (d_l) และด้านขวา (d_r) ของรถยนต์ขณะขับเคลื่อนในลักษณะเส้นโค้งผ่านเส้นแบ่งเลนที่เป็นเส้นโค้ง	3
3 กระบวนการทำงานของตัวกรองคาลมาน.....	8
4 กระบวนการทำงานของ Extended Kalman Filter	9
5 ลักษณะการกระจายตัวของข้อมูลแบบ Gaussian	10
6 หลักการพื้นฐานของระบบตรวจจับเลน.....	11
7 ผลการทดลองระบบ Lane detection and tracking for driving aid system	15
8 ตำแหน่งและทิศทางของรถยนต์ที่ใช้ในการทดลอง.....	16
9 แบบจำลองเรขาคณิตวิเคราะห์ของวงโคจรการขับเคลื่อนของรถยนต์ส่วนบุคคล ตำแหน่งและทิศทางของรถยนต์ที่ใช้ในการทดลอง.....	22
10 ค่าตัวแปรที่ใช้ในการคำนวณหาความยาวส่วนโค้ง.....	22
11 ลักษณะการขับเคลื่อนของรถยนต์ผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง ณ ช่วงเวลาต่างๆ	27
12 ลักษณะการขับเคลื่อนของรถยนต์ผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นโค้ง ณ ช่วงเวลาต่างๆ	28

สารบัญภาพประกอบ

ภาพประกอบ	หน้า
13 ผลการประมาณค่าของแบบจำลองเมื่อรถยนต์ขับเคลื่อนแบบเส้นตรงและโค้ง ผ่านเส้นแบ่งเลนที่เป็นเส้นตรง.....	29
14 ผลการประมาณค่าของแบบจำลองเมื่อรถยนต์ขับเคลื่อนแบบเส้นตรงและโค้ง ผ่านเส้นแบ่งเลนที่เป็นเส้นโค้ง.....	30
15 ผลการประมาณค่าของแบบจำลองเมื่อรถยนต์ขับเคลื่อนแบบเส้นตรงและโค้ง ผ่านเส้นแบ่งเลนที่เป็นเส้นตรง ที่มีการจำลองการขาดหายของสัญญาณ.....	31
16 ผลการประมาณค่าของแบบจำลองเมื่อรถยนต์ขับเคลื่อนแบบเส้นตรงและโค้ง ผ่านเส้นแบ่งเลนที่เป็นเส้นโค้ง ที่มีการจำลองการขาดหายของสัญญาณ.....	32
17 กราฟแสดงค่า RMSE	33

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ณ ปัจจุบันนี้มีผู้เสียชีวิตทั่วโลกประมาณ 1.3 ล้านคนและได้รับบาดเจ็บทุกภาพประมาณ 20-50 ล้านคนเนื่องจากอุบัติเหตุทางรถยนต์ทุกปี เนื่องจากความวุ่นวาย ความเหนื่อยล้า และความผิดพลาดต่างๆ (World Health Organization: WHO, 2013) ซึ่งผลกระทบที่เกิดขึ้นจากผลการสูญเสียนี้ส่งผลกระทบโดยตรงต่อการเจริญเติบโตของเศรษฐกิจของประเทศและของโลก เพราะผู้เสียชีวิตคือหนึ่งในแรงงานหลักของประเทศ ผู้ที่ได้รับบาดเจ็บจำเป็นต้องหยุดงานเพื่อเข้ารับการรักษาดูแล ผู้ป่วยต้องลาหยุดงานเพื่อมาดูแลผู้ป่วย รวมถึงค่ารักษาพยาบาล ในที่นี้ยังไม่รวมถึงความสูญเสียอันเนื่องมาจากทุกภาพ ซึ่งบางคนอาจไม่สามารถช่วยเหลือตัวเองได้ สืบเนื่องจากผลกระทบดังกล่าวไปแล้ว จึงทำให้มีการสนับสนุนให้มีการศึกษาและทำวิจัยเรื่อง ระบบขนส่งและจราจรอัจฉริยะ Intelligent Transport Systems (ITS) ซึ่ง Bishop (2000:78-81) ได้อธิบายไว้ว่าเป็นระบบนี้สามารถแบ่งออกได้เป็น 3 ลักษณะคือ

- ระบบที่มีความสามารถในการเตือนผู้ขับขี่ในกรณีที่เกิดการผิดพลาดในการขับขี่ เช่น ระบบเตือนเมื่อมีการเปลี่ยนเลนของรถในลักษณะที่ผิดปกติ (Lane Departure Warning System) ซึ่งอาจจะใช้การเตือนด้วยเสียงหรือระบบสั่น เป็นต้น
- ระบบที่สามารถช่วยเหลือผู้ขับขี่ได้ในกรณีที่มีการขับขี่ที่ผิดปกติเกิดขึ้นเช่น ระบบที่สามารถช่วยคนขับให้ขับอยู่ในเลนปกติได้ในกรณีที่ผู้ขับขี่สูญเสียการควบคุมด้วยเหตุผลใดๆก็ตาม (Lane Keeping Assistance System)
- ระบบที่สามารถขับเคลื่อนด้วยตนเองอัตโนมัติโดยไม่มีคนขับ (Autonomous Driving System)

ในการพัฒนาระบบขนส่งและจราจรอัจฉริยะทั้ง 3 ประเภทจะเกิดขึ้นได้นั้นจำเป็นจะต้องมีระบบที่ใช้ในการตรวจจับเลน(Lane Detection System) ที่มีประสิทธิภาพเพื่อที่ระบบขนส่งและจราจรอัจฉริยะในแต่ละแบบจะสามารถแยกแยะได้ว่าสถานะการขับขี่ที่เกิดขึ้นในปัจจุบันนี้เป็นลักษณะการขับขี่ที่ปลอดภัยหรือไม่

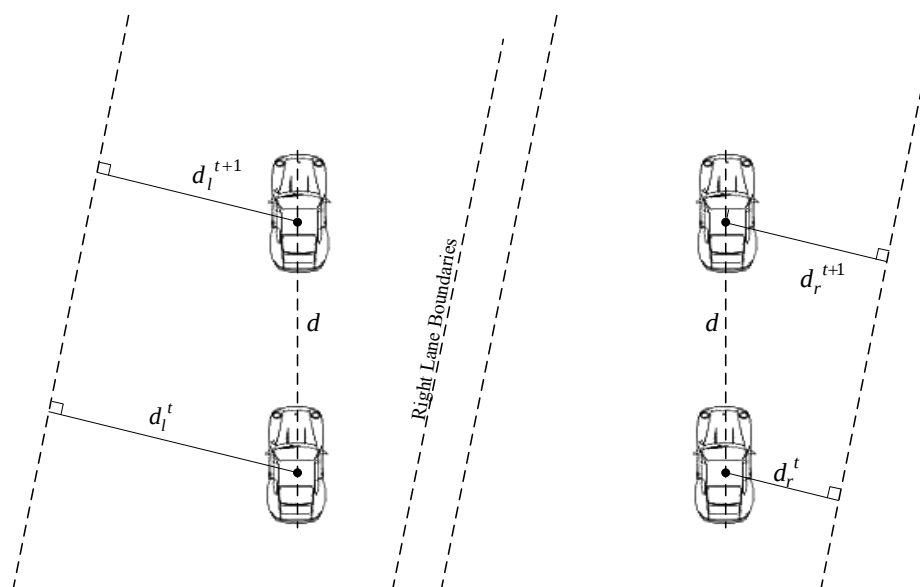
เทคนิคในการสร้างระบบตรวจจับเลนได้แก่ เรดาร์ เซ็นเซอร์แบบต่างๆ GPS และ computer vision โดยระบบที่ได้รับความนิยมในปัจจุบันนี้คือ computer vision เพราะเป็นระบบที่มีราคาไม่สูงมากนัก สามารถใช้กล้องที่มีขายในตลาดได้ โดยติดตั้งเข้ากับรถยนต์เพื่อตรวจจับเลนในขณะขับขี่ และนอกจากนี้ยังไม่จำเป็นต้องมีการอัพเดทข้อมูลให้มีความทันสมัยอยู่เสมออย่าง

ระบบ GPS แต่เพื่อควมมีประสิทธิภาพของระบบตรวจจับเลนผู้พัฒนาส่วนมากนิยมนำเทคนิคหลายๆอย่างมาผสมผสานกันเพื่อเพิ่มประสิทธิภาพในการตรวจจับเลนให้สูงขึ้น

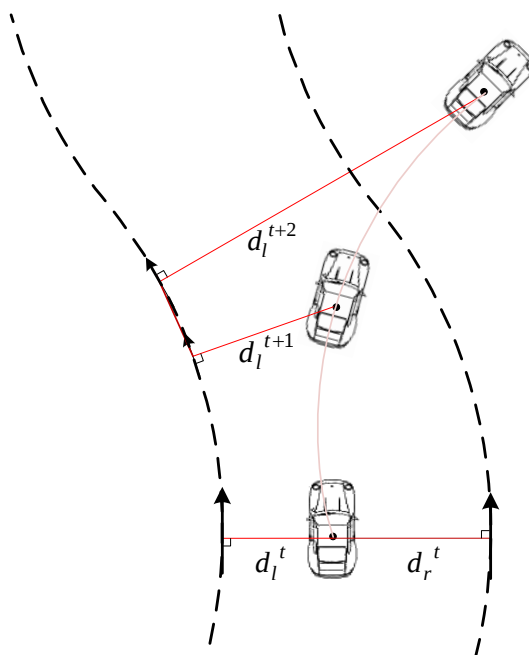
อุปสรรคสำคัญของระบบตรวจจับเลนไม่ว่าจะใช้เทคนิคใดก็ตามคือความไม่สมบูรณ์ของข้อมูลที่เข้าสู่ระบบ โดยมีสาเหตุเนื่องมาจากความไม่สมบูรณ์แบบของตัวอุปกรณ์ เช่น ความผิดพลาดที่เกิดขึ้นจากเซ็นเซอร์ชนิดต่างๆ หรือจากสภาพแวดล้อมที่เกิดขึ้นระหว่างการขับขี่ เช่น เงาของเสาไฟฟ้า หมอก ฝนหรือแม้กระทั่งแสงสะท้อนที่เกิดจากไฟตามถนนในตอนกลางคืน เป็นต้น จึงจำเป็นต้องมีการใช้ตัวกรองดิจิทัลเข้ามาช่วยในการประมาณค่าของข้อมูลในกรณีที่มีสัญญาณรบกวนหรือเกิดการสูญหายของข้อมูลเพื่อเป็นอินพุตให้กับระบบขนส่งและจราจรอัจฉริยะต่อไป ซึ่งตัวกรองดิจิทัลที่ได้รับความนิยมจากอดีตจนถึงปัจจุบันก็คือ ตัวกรองคาลมาน (Kalman Filter) ซึ่งมีประสิทธิภาพในการประมาณค่าให้ใกล้เคียงกับค่าจริงที่ควรที่จะเกิดขึ้น ตัวอย่างของระบบที่ใช้ตัวกรองคาลมานได้แก่ การคำนวณวิถีวงโคจรของดาวเทียม การพยากรณ์ทิศทางของหุ่นภายในตลาดหุ้น การคำนวณเส้นทางการเดินทางของจรวดหรือขีปนาวุธและการทำนายทิศทางในการบังคับรถยนต์ขับเคลื่อนอัตโนมัติ เป็นต้น

ในส่วนของการนำเสนองานวิจัยเพื่อพัฒนาระบบตรวจจับเลนนี้ มีวัตถุประสงค์เพื่อสร้างแบบจำลองสำหรับระบบตรวจจับเลนที่ใช้เทคนิคของระบบ Computer Vision โดยสร้างสูตรคำนวณการพยากรณ์ค่าในสถานะอนาคต (Future state) จากค่าที่ได้จากการรวบรวมข้อมูลในการขับเคลื่อนของรถยนต์ในสถานะปัจจุบัน (Current state) เพื่อให้ได้ค่าประมาณ (Estimated state) ของระยะห่างระหว่างเลนกับรถยนต์ทั้งทางด้านขวามือ (d) และซ้ายมือ (d) แสดงดังภาพประกอบ 1 และ 2 ซึ่งการสร้างแบบจำลองที่ผ่านมาจะเป็นการสร้างแบบจำลองที่อยู่ภายใต้ข้อสันนิฐานที่ว่ารถเคลื่อนที่ในลักษณะเส้นตรงผ่านถนนหรือเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรงเช่นเดียวกัน แสดงดังภาพประกอบ 1 และ 2

ในการนำเสนองานวิจัยในครั้งนี้เป็นการสร้างแบบจำลองที่รองรับการขับเคลื่อนของรถยนต์ในลักษณะที่เป็นเส้นตรง (Straight motion) และเส้นโค้ง (Circular motion) ผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง (Straight lane boundaries) และเส้นโค้ง (Curve lane boundaries) ด้วย ซึ่งลักษณะการขับเคลื่อนผ่านเส้นแบ่งเลนในแต่ละแบบแสดงดังภาพประกอบ 1 และ 2 เพื่อเป็นการรองรับการประมาณค่าของรถยนต์ในทุกๆสถานะของการขับเคลื่อน ซึ่งล้วนแต่มีสัญญาณรบกวนเกิดขึ้นทั้งสิ้นรวมทั้งบางครั้งอาจเกิดการสูญหายของข้อมูลเกิดขึ้นด้วย เช่น เมื่อรถขับเคลื่อนในลักษณะ U-Turn เป็นต้น



ภาพประกอบ 1 ตัวอย่างการวัดระยะห่างระหว่างขอบเส้นแบ่งเลนกับรถยนต์ทั้งทางด้านซ้าย (d_l) และด้านขวา (d_r) ของรถยนต์ขณะขับเคลื่อนในลักษณะเส้นตรงผ่านเส้นแบ่งเลนที่เป็นเส้นตรง



ภาพประกอบ 2 ตัวอย่างการวัดระยะห่างระหว่างขอบเส้นแบ่งเลนกับรถยนต์ทั้งทางด้านซ้าย (d_l) และด้านขวา (d_r) ของรถยนต์ขณะขับเคลื่อนในลักษณะเส้นโค้งผ่านเส้นแบ่งเลนที่เป็นเส้นโค้ง

1.2 วัตถุประสงค์ของการวิจัย

ผู้วิจัยต้องการสร้างแบบจำลองที่รองรับการขับเคลื่อนแบบไม่เป็นเชิงเส้น (Non-linear movement) ในลักษณะ 2 มิติโดยใช้เรขาคณิตวิเคราะห์ในการคำนวณหาค่าตัวแปรที่คาดว่าจะเกิดขึ้นในอนาคตโดยมีวัตถุประสงค์หลักดังนี้

1. สร้างแบบจำลองในการประมาณค่าระยะห่างระหว่างขอบของเส้นแบ่งเลนกับรถยนต์ทางด้านซ้ายมือและขวามือ เมื่อรถขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรงและโค้ง โดยรองรับลักษณะการขับเคลื่อนทั้งที่เป็นเส้นตรงและโค้ง
2. แบบจำลองสามารถลดสัญญาณรบกวนที่เกิดขึ้นจากสภาพแวดล้อม หรือจากความบกพร่องของระบบเซ็นเซอร์ได้อย่างมีประสิทธิภาพเพื่อสามารถนำไปใช้ในส่วนของการประมาณค่าและติดตามในระบบตรวจจับเลน

1.3 คำถามการวิจัย

1. แบบจำลองสามารถทำนายการโคจรในสถานะอนาคตของรถยนต์ได้โดยใช้ข้อมูลที่ได้จากสถานะอดีตและปัจจุบันได้อย่างไร
2. แบบจำลองสามารถลดสัญญาณรบกวนที่เกิดขึ้นกับระบบได้เท่าไรหรือคิดเป็นกี่เปอร์เซ็นต์
3. แบบจำลองที่สร้างขึ้นแตกต่างกับแบบจำลองที่เคยมีการเผยแพร่อย่างไร

1.4 สมมุติฐานการวิจัย

1. แบบจำลองสามารถทำนายระยะห่างระหว่างขอบของเส้นแบ่งเลนกับรถยนต์ทางด้านซ้ายมือและขวามือ เมื่อรถยนต์ขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรงและโค้ง โดยรองรับลักษณะการขับเคลื่อนทั้งที่เป็นเส้นตรงและโค้ง
2. แบบจำลองสามารถลดสัญญาณรบกวนที่เกิดขึ้นจากสภาพแวดล้อม หรือจากความบกพร่องของระบบเซ็นเซอร์ได้อย่างมีประสิทธิภาพ โดยค่าระยะห่างระหว่างขอบของเส้นแบ่งเลนกับรถยนต์ทางด้านซ้ายมือและขวามือที่ได้จากการประมาณสามารถลดสัญญาณรบกวนได้มากกว่า 50 เปอร์เซ็นต์ (คำนวณโดยใช้สมการของ Root Mean Square Error: RMSE)

1.5 ขอบเขตของการวิจัย

1. ศึกษาทฤษฎีที่เกี่ยวข้องในการสร้างแบบจำลองโดยใช้ตัวกรองกาลมาน
2. เก็บรวบรวมข้อมูลที่ใช้ในการทดสอบแบบจำลอง
 - 2.1 จำลองข้อมูลที่ได้จากภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง
 - 2.2 จำลองข้อมูลที่ได้จากภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นโค้งผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง
 - 2.3 จำลองข้อมูลที่ได้จากภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นโค้ง
 - 2.4 จำลองข้อมูลที่ได้จากภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นโค้งผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นโค้ง
3. สร้างแบบจำลอง
4. ทดสอบประสิทธิภาพของแบบจำลองที่สร้างขึ้นโดยใช้ RMSE

1.6 นิยามศัพท์

- Kalman Filter คืออัลกอริทึมที่ใช้หลักการของความน่าจะเป็นในการสร้างสมมติฐาน ของสถานะที่กำลังจะเกิดขึ้น (สถานะอนาคต) โดยใช้ข้อมูลที่ได้จากสถานะก่อนหน้านี้ (สถานะอดีต) กับค่าของเซ็นเซอร์ที่ได้จากสถานะปัจจุบัน มาทำการคำนวณด้วยแบบจำลองที่สร้างขึ้นมาโดยแบบจำลองที่จำเป็นต้องสร้างขึ้นสำหรับตัวกรองกาลมานมีอยู่ด้วยกัน 2 แบบจำลอง คือแบบจำลองระบบ (System model or State transition model) และแบบจำลองการวัด (Measurement model)
- Intelligent Transport Systems (ITS) คือการใช้เทคโนโลยีสารสนเทศและการสื่อสารเข้ามาช่วยในการบริหารจัดการในระบบคมนาคม การขนส่ง และการจราจร เพื่อช่วยเพิ่มประสิทธิภาพของระบบบนท้องถนน และระบบขนส่งมวลชน ความปลอดภัย และช่วยลดการติดขัดของการจราจร
- Computer Vision คือระบบการมองเห็นหรือระบบการมองภาพของคอมพิวเตอร์ซึ่งมีลักษณะการทำงานเหมือนกับการมองเห็นของมนุษย์คือ สามารถมองเห็นสิ่งต่างๆรอบๆตัวด้วยดวงตาทั้งสองและทำความเข้าใจ จดจำและแยกแยะสิ่งต่างๆที่มนุษย์มองเห็นได้ด้วยสมอง ซึ่งในระบบ Computer Vision จะใช้กล้องวิดีโอหรือกล้องถ่ายรูปแทนดวงตาของ

มนุษย์ ที่ทำหน้าที่ในการรับภาพเข้ามาในให้สมองซึ่งในที่นี้ก็คือคอมพิวเตอร์ทำการวิเคราะห์และแยกแยะเพื่อนำไปใช้ประโยชน์ต่อไป

- Lane Detection System คือระบบที่ใช้ในการวัดระยะห่างระหว่างรถกับเส้นขอบของเลนซึ่งค่าที่ได้จะเป็นระยะห่างระหว่างจุดกึ่งกลางของรถกับจุดที่ตั้งฉากกับขอบของเลนทั้งทางด้านซ้ายมือและขวามือ

บทที่ 2

วรรณกรรมที่เกี่ยวข้อง

2.1 ความรู้พื้นฐาน

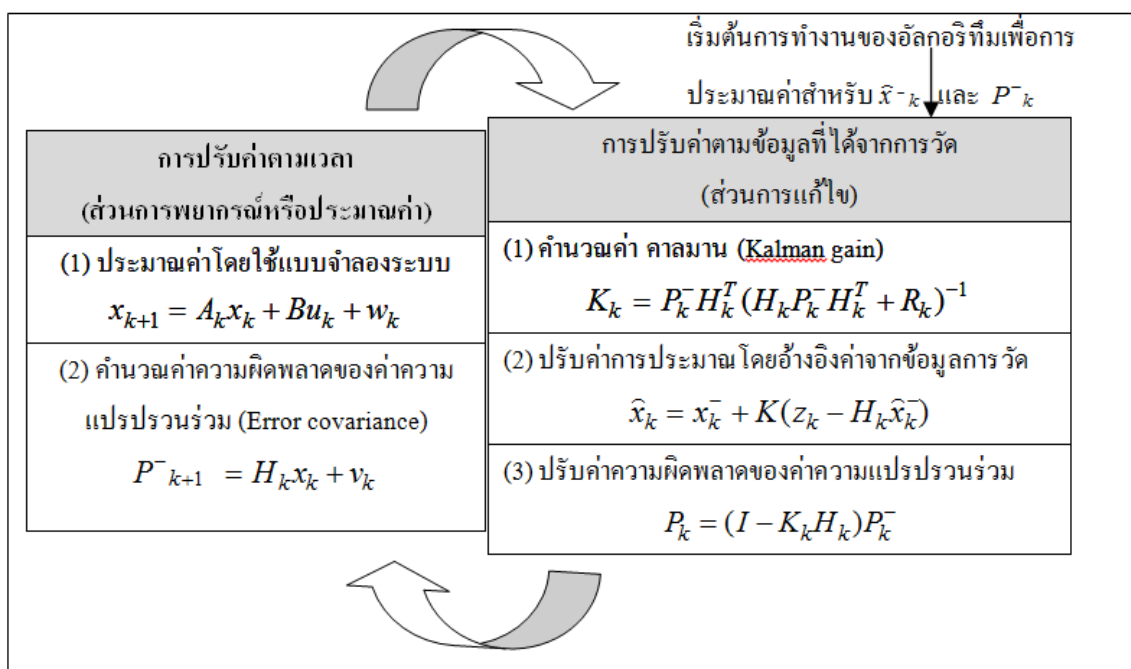
ตัวกรองคาลมานถูกคิดค้นขึ้นโดย Dr. Rudolf Kalman (1960:35-45) เมื่อปีค.ศ.1960 โดยถูกนำไปใช้กับการพัฒนาโครงการทางด้านอวกาศของนาซา (NASA) สหรัฐอเมริกา เพื่อแก้ปัญหา ระบบนำร่องของยาน Apollo ในการลงจอดบนดวงจันทร์ ซึ่งก็เป็นตัวอย่างการใช้งานที่ได้รับความสำเร็จเป็นอย่างมาก

ตัวกรองคาลมานเป็นอัลกอริทึมที่ใช้หลักการทำงานแบบวนซ้ำ (recursive) เพื่อใช้คำนวณประมาณค่าหรือพยากรณ์สถานะที่จะเกิดขึ้นกับระบบที่มีการเคลื่อนที่ตลอดเวลา (Dynamic system) โดยใช้ข้อมูลที่ได้จากสถานะก่อนหน้านี้ (สถานะอดีต) ร่วมกับข้อมูลจากเซ็นเซอร์ของสถานะปัจจุบันเท่านั้น ซึ่งไม่จำเป็นที่จะต้องใช้หน่วยความจำเป็นจำนวนมากเพื่อเก็บข้อมูลที่เกิดขึ้นภายในอดีตทั้งหมด ส่งผลให้ความเร็วในการคำนวณเพิ่มมากขึ้น จึงเหมาะกับการนำไปใช้กับระบบที่มีการเปลี่ยนแปลงค่าของข้อมูลตลอดเวลา การใช้งานตัวกรองคาลมานเป็นเรื่องที่ไม่ยุ่งยาก ซับซ้อน แต่ผู้ใช้จำเป็นต้องใช้ความรู้ในหลายสาขาด้วยกันในการพัฒนาแบบจำลอง (model) ขึ้นมาได้แก่ความรู้ทางด้านสถิติ ความน่าจะเป็นและด้านคณิตศาสตร์ เป็นต้น แบบจำลองที่ถูกสร้างขึ้นมาสสำหรับตัวกรองคาลมานโดยใช้ทฤษฎีทางด้านคณิตศาสตร์นั้นจะเป็นแบบจำลองที่ใช้ฟังก์ชันแบบไม่เป็นเชิงเส้น (Nonlinear function) เป็นส่วนใหญ่ เพราะระบบต่างๆ ที่ถูกพัฒนาขึ้นมาในชีวิตประจำวันยากที่จะหาฟังก์ชันที่เป็นแบบเชิงเส้น ฟังก์ชันที่ไม่เป็นเชิงเส้นได้แก่เอ็กซ์โพเนนเชียล ลอการิทึมหรือตรีโกณมิติ เป็นต้น ดังนั้นตัวกรองคาลมานที่ใช้กับฟังก์ชันที่ไม่เป็นเชิงเส้น (Extended Kalman Filter หรือ EKF) จึงถูกนำมาใช้งานมากกว่าตัวกรองคาลมานแบบธรรมดา ซึ่งเป็นเหตุผลที่ผู้วิจัยเริ่มศึกษาวิธีการใช้งาน EKF ตั้งแต่การเริ่มคิดค้นแบบจำลองจนถึงขั้นตอนการจำลองการทำงานของตัวอัลกอริทึมกับเหตุการณ์ที่สมมุติขึ้นมามีด้วย โปรแกรม Matlab

ในการศึกษา EKF นั้น จำเป็นจะต้องศึกษาถึงหลักการทำงานพื้นฐานของตัวกรองคาลมานเสียก่อน เพราะ EKF ถูกพัฒนามาจากตัวกรองคาลมาน (ระบบที่พัฒนาขึ้นมาเพื่อฟังก์ชันที่มีการทำงานในลักษณะเชิงเส้น) สิ่งที่แตกต่างกันคือการใช้อนุพันธ์ (Differential Equation) เข้ามาช่วยในการแปลงฟังก์ชันไม่เป็นเชิงเส้นให้เป็นฟังก์ชันเชิงเส้นได้ (Linearization) ซึ่งวิธีที่ถูกนำมาใช้คือการสร้างเมทริกซ์ จาคอบีเยน (Jacobian Matrix) โดยส่วนมากจะทำการหาอนุพันธ์อันดับหนึ่งเพื่อหาความเปลี่ยนแปลงของค่าเอาต์พุตเมื่อค่าอินพุตมีการเปลี่ยนแปลง

2.1.1 ตัวกรองคาลมานที่ใช้กับฟังก์ชันที่เป็นเชิงเส้น (Kalman Filter)

ตัวกรองคาลมาน(Kalman Filter) เป็นอัลกอริทึมที่ใช้หลักการของความน่าจะเป็นในการสร้างสมมติฐาน ของสถานะที่กำลังจะเกิดขึ้น โดยใช้ข้อมูลที่ได้จากสถานะก่อนหน้านี้ กับค่าของเซ็นเซอร์ที่ได้จากสถานะปัจจุบัน มาทำการคำนวณด้วยแบบจำลองที่สร้างขึ้น (Welch and Bishop, 1997) โดยแบบจำลองที่จำเป็นต้องสร้างขึ้นมาเพื่อใช้ในตัวกรองคาลมานมีด้วยกัน 2 แบบจำลองคือ แบบจำลองระบบ (System model or State transition model) และแบบจำลองการวัด (Measurement model) ซึ่งหลักการทำงานของอัลกอริทึมแสดงดังภาพประกอบ 3



x = state matrix

A = State transition matrix

B = Control matrix

u = Control variable

w = Gaussian white noise

P = State variance matrix

H = Measurement matrix

K = Kalman gain

R = Measurement variance matrix

z = Measurement variables

x_k = current state

$\hat{x}$ = Estimation state

x_{k+1} = future state

I = Identity matrix

P^- = previous state variance

ภาพประกอบ 3 กระบวนการทำงานของตัวกรองคาลมาน

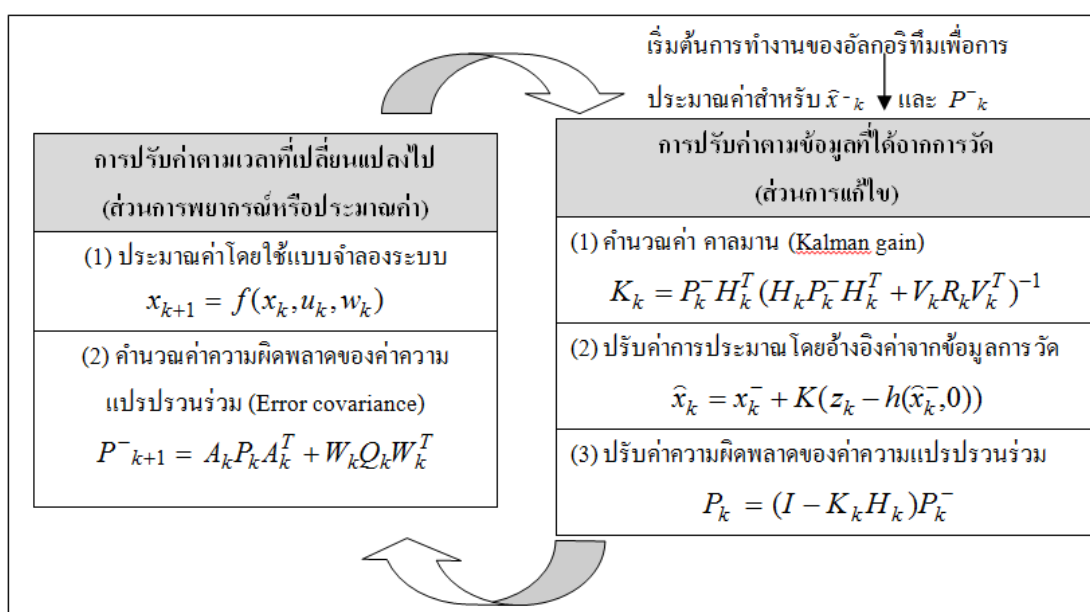
ที่มา: Welch and Bishop, 1997

2.1.2 ตัวกรองคาลมานที่ใช้กับฟังก์ชันที่ไม่เป็นเชิงเส้น (Extended Kalman Filter)

โดยหลักการทำงานของอัลกอริทึมแสดงดังภาพประกอบ 4 สิ่งที่แตกต่างกันจากตัวกรองคาลมานแบบปกติก็คือฟังก์ชันที่ถูกนำไปใช้ในแบบจำลองจะเป็นฟังก์ชันแบบไม่เป็นเชิงเส้นดังนี้

$$\text{แบบจำลองระบบ} \quad x_{k+1} = f(x_k, u_k, w_k)$$

$$\text{แบบจำลองการวัด} \quad z_k = h(x_k + v_k)$$



ภาพประกอบ 4 กระบวนการทำงานของ Extended Kalman Filter

ที่มา: Welch and Bishop, 1997

กระบวนการที่ถูกเพิ่มขึ้นใน Extended Kalman Filter คือการแปลงฟังก์ชันที่ไม่เป็นเชิงเส้นทั้ง f และ h ให้เป็นฟังก์ชันที่เป็นเชิงเส้นในทุกๆสถานะที่กำลังเปลี่ยนแปลงไปของแบบจำลอง โดยใช้เมทริกซ์จาโคเบียน (Jacobian matrix) ซึ่งเป็นรูปแบบการหาอนุพันธ์ โดยจะใช้อนุพันธ์อันดับหนึ่ง (First order derivative) ซึ่งทำให้ได้ค่าโดยประมาณของแบบจำลองทั้งสองแบบเรียงตามลำดับดังนี้

$$\text{แบบจำลองระบบ} \quad x_{k+1} \approx \tilde{x}_{k+1} + A_k(x_k - \hat{x}_k) + Ww_k$$

$$\text{แบบจำลองการวัด} \quad z_k \approx \tilde{z}_k + H(x_k - \hat{x}_k) + Vv_k$$

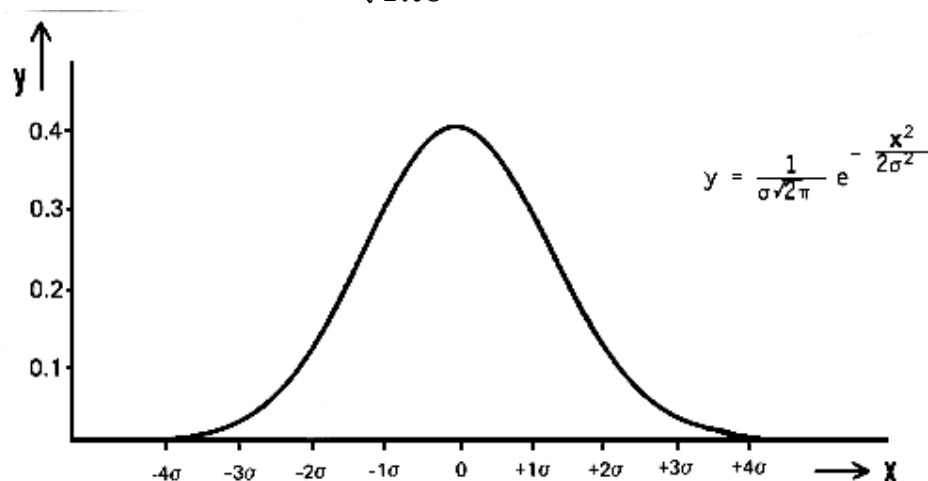
ซึ่งตัวอย่างสมการที่ใช้หาค่าเมทริกซ์จacobian ของ State transition matrix แสดงดังนี้

$$A_{[i,j]} = \frac{\partial f_{[i]}}{\partial x_{[i]}}(\hat{x}_k, u_k, 0)$$

2.1.3 การสุ่มตัวอย่างแบบ Gaussian

ใช้การสุ่มตัวอย่างของสัญญาณรบกวนใช้แบบ Gaussian Noise (Kailath, 1969:350-361) ซึ่งเป็นสัญญาณรบกวนที่ถูกนำไปใช้ในทางปฏิบัติจริงบ่อยครั้ง เนื่องจากในทางคณิตศาสตร์ค่าฟังก์ชันการกระจายของความน่าจะเป็นอยู่ในรูปของการกระจายตัวแบบปกติ (Normal Distribution)

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\mu)^2}{2\sigma^2}}$$



ภาพประกอบ 5 ลักษณะการกระจายตัวของข้อมูลแบบ Gaussian

ที่มา: Kailath, 1969

2.1.4 การคำนวณค่า Root Mean Squared Error (RMSE)

ค่า RMSE (Pindyck and Robert, 1998:47:98) คือ การวัดค่าความแตกต่างระหว่างค่าจริง และค่าที่ได้จากการประมาณจากแบบจำลอง หากค่า RMSE มีค่าน้อยแสดงว่าแบบจำลอง สามารถประมาณค่าได้ใกล้เคียงกับค่าจริง ดังนั้นหากค่านี้มีค่าเท่ากับศูนย์ หมายความว่า ไม่เกิดความคลาดเคลื่อนในแบบจำลองนี้เลย โดยสามารถคำนวณได้ดังนี้

$$RMSE = \sqrt{\frac{1}{T} \sum_{t=1}^T (Y_t^s - Y_t^o)^2}$$

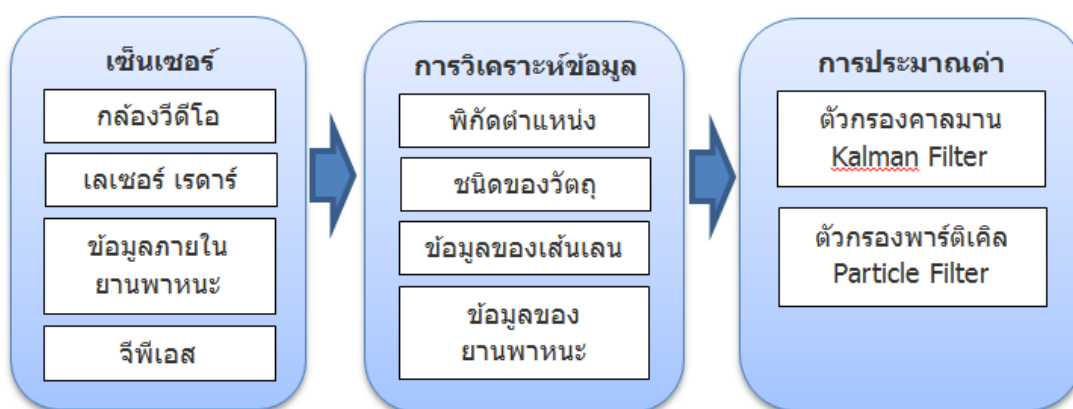
เมื่อ Y_t^s คือค่าประมาณจากแบบจำลอง

Y_t^o คือค่าที่แท้จริง

T คือจำนวนคาบเวลาที่ใช้ในการประมาณแบบจำลอง

2.2 กรอบความคิดทางทฤษฎี

ระบบตรวจจับเลนสามารถแบ่งออกได้เป็น 3 ส่วน คือ การตรวจจับเส้นแบ่งเลน (Lane feature detection) การประมาณระยะห่างระหว่างขอบเลนกับยานพาหนะ (Lane boundary estimation) และการติดตามเส้นเลน (Lane tracking) โดยบทความวิจัยชิ้นนี้จะศึกษาเฉพาะส่วนของการสร้างแบบจำลองเพื่อใช้ในการประมาณค่าระยะห่างระหว่างขอบเลนกับรถยนต์แบบส่วนบุคคล ถึงแม้ว่าจะมีการพัฒนาระบบตรวจจับเส้นเลนขึ้นมาเป็นจำนวนมากแต่หลักการพื้นฐานของระบบสามารถแสดงได้ดังภาพประกอบ 6



ภาพประกอบ 6 หลักการพื้นฐานของระบบตรวจจับเลน

2.2.1 เซ็นเซอร์ ในการออกแบบระบบบางครั้งจำเป็นต้องใช้งานเซ็นเซอร์มากกว่า 1 ตัว เพื่อเพิ่มประสิทธิภาพในการทำงานของระบบ (Handmann, Leefken, Tzomakas, Seele, and Biologie, 1999) โดยเซ็นเซอร์ที่นิยมใช้มีดังนี้

- กล้องวิดีโอและอุปกรณ์ตรวจจับภาพอื่นๆ (Leng and Chen, 2010) เป็นอุปกรณ์ที่ทำให้ง่ายต่อการตรวจจับและเก็บข้อมูลสภาพแวดล้อมโดยรอบของยานพาหนะได้ นอกจากนี้ยังมีราคาไม่สูงมากในปัจจุบันและยังไม่ต้องมีการปรับปรุงข้อมูลให้ทันสมัยอยู่เสมอเหมือนกับระบบจีพีเอส ซึ่งงานวิจัยชิ้นนี้ใช้ข้อมูลที่ได้จากกล้องวิดีโอในการสร้างแบบจำลอง
- เลเซอร์และเรดาร์ ยังเป็นอุปกรณ์ที่นิยมใช้ในการตรวจจับวัตถุที่อยู่ห่างออกไปจากตัวยานพาหนะ เช่น ต้นไม้ ภูเขา ซึ่งเหมาะกับถนนในชนบท (Ma, Lakshmanan, and Hero, 2000)
- เซ็นเซอร์ภายในระบบการขับเคลื่อนของยานพาหนะ เช่น ความเร็วรอบในการขับเคลื่อน อัตราความเร่ง หรือ ความเร็วของล้อรถ เป็นต้น
- จีพีเอส (Goldbeck, Huertge, Ernst, and Kelch, 2000) ใช้เพื่อระบุตำแหน่งของยานพาหนะ ซึ่งความถูกต้องแม่นยำขึ้นอยู่กับความทันสมัยของข้อมูลของอุปกรณ์เป็นหลัก

2.2.2 การวิเคราะห์ข้อมูล คือขั้นตอนในการวิเคราะห์ข้อมูลที่ได้จากเซ็นเซอร์ต่างๆ เพื่อนำข้อมูลที่ได้ไปใช้ในระบบตรวจจับเลน เช่น ค่าตำแหน่งของยานพาหนะ วัตถุที่ขีดขวางหรือชนิดของวัตถุต่างๆ เช่น เสา ยานพาหนะอื่นๆ เสาไฟฟ้า ค่าความโค้งของเส้นแบ่งเลน ค่ามุมของรถที่กระทำกับเส้นแบ่งเลน ค่ามุมของล้อรถ เป็นต้น

2.2.3 การประมาณค่าข้อมูล คือขั้นตอนในการเพิ่มความถูกต้องแม่นยำของข้อมูลที่ได้จากขั้นตอนการวิเคราะห์ข้อมูลซึ่งจะถูกนำไปใช้ในระบบติดตามต่อไป สาเหตุที่ต้องมีการประมาณค่าข้อมูลนั้นอาจเกิดจากความบกพร่องของเซ็นเซอร์ หรือสัญญาณรบกวนที่เกิดขึ้นจากสภาพแวดล้อมโดยรอบ ตัวอย่างเช่น ถ้าใช้กล้องวิดีโอเป็นตัวตรวจจับข้อมูลในบางครั้งขณะที่รถเลี้ยวข้อมูลของเส้นแบ่งเลนก็จะขาดหายไปหรือในบางครั้งมีเงาของเสาไฟฟ้าพาดผ่านเส้นแบ่งเลนก็จะทำให้ข้อมูลหายไปบางส่วนเช่นกันหรือในกรณีที่ฝนตกหรือมีหมอกลงจัดก็จะทำให้ภาพของเส้นแบ่งเลนขาดความคมชัดทำให้ค่าตัวแปรต่างๆที่ได้จากการตรวจจับขาดความเที่ยงตรง เป็นต้น ดังนั้นจึงต้องมีการประมาณค่าให้กับข้อมูลที่ขาดหายไปหรือข้อมูลที่มีความผิดพลาดสูงอันเนื่องมาจากสัญญาณรบกวนที่เกิดขึ้นกับระบบ

ในการเพิ่มความเสถียรและความน่าเชื่อถือของข้อมูลก่อนที่จะถูกนำไปใช้ในส่วนของการติดตามซึ่งเป็นขั้นตอนต่อไปของระบบตรวจจับเลนจึงมีความสำคัญเป็นอย่างมาก (McCall, and Trivedi, 2006:365-347) อัลกอริทึมในการประมาณค่าถึงถูกนำมาใช้งานโดยตัวกรองที่นิยมนำมาใช้ในระบบตรวจจับเส้นเลนได้แก่ ตัวกรองคาลมาน (Borkar, Hayes, and Smith, 2012) (Lim, Seng, and Ang, 2012) และ ตัวกรองพาร์ติเคิล (Chiku, and Miura, 2012:39-44)

ตัวกรองคาลมานถูกนำมาใช้และเป็นที่รู้จักอย่างแพร่หลาย เพราะการใช้งานที่ไม่ยุ่งยากและมีประสิทธิภาพในการประมาณค่าสูงในระบบประมาณค่าและติดตาม (Ristic, Arulampalam, and Gordon, 2004:5-123) โดยเฉพาะอย่างยิ่งในระบบที่ใช้ในการพยากรณ์การเคลื่อนที่แบบต่างๆ ในระบบตรวจจับเส้นเลนตัวกรองคาลมานสามารถนำไปใช้ได้ในกรณีที่ต้องการประมาณค่าเนื่องจากค่าที่ได้รับจากเซ็นเซอร์มีความผิดพลาดสูงหรือในกรณีที่ข้อมูลเกิดการสูญหายไป เช่น เส้นเลน องศา ความเร็ว ความเร่ง หรือค่าความโค้งของเส้นเลน เป็นต้น จึงทำให้ข้อมูลที่ได้มีความเสถียรและเที่ยงตรงมากยิ่งขึ้นก็จะส่งผลให้ประสิทธิภาพในการทำงานของระบบตรวจจับเลนสูงขึ้นเช่นกัน (Jurgen, and S of Automotive Engineers, 2007)

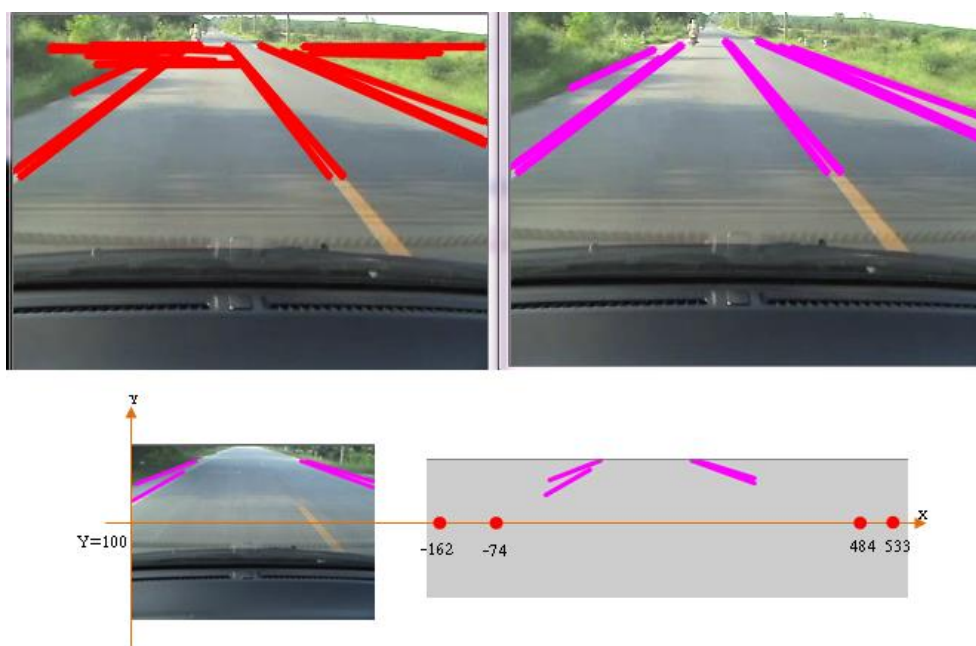
หลักในการทำงานพื้นฐานของตัวกรองคาลมานคือค่าที่ได้ในสถานะพยากรณ์เกิดจากการคำนวณค่าความเป็นไปโดยใช้ข้อมูลจากสถานะก่อนหน้า ข้อมูลที่ได้จากเซ็นเซอร์และข้อมูลจากสถานะปัจจุบันมาใช้เพื่อคำนวณหาค่าความเป็นไปได้เพื่อพยากรณ์ค่าข้อมูลสำหรับสถานะอนาคต

ตัวอย่างการใช้ตัวกรองคาลมานในระบบการขนส่งและจราจรอัจฉริยะที่มีชื่อเสียงทั่วโลกได้แก่ รถขับเคลื่อนอัตโนมัติของบริษัท Google: Google Lobbies Nevada self-driving car (Wright, 2011:16-18) และรถยนต์อัจฉริยะที่ชนะการแข่งขัน DARPA Grand Challenge ที่มีชื่อว่า Standley ซึ่งเป็นทีมจากมหาวิทยาลัยสแตนฟอร์ด (Thrun, Montemerlo, Dahlkamp, Stavens, Aron, Diebel, Fong, Gale, Halpenny, Hoffmann, Lau, Oakley, Palatucci, Stang, Strohsand, Dupont, Jendrosseck, Koelen, Markey, Rummel, Niekerk, Jensen, Alessandrini, Brandski, Davies, Ettinger, Kaehler, Nefinan, and Mahoney, 2006) โดยรถอัจฉริยะดังกล่าวใช้ค่าที่ได้จากเซ็นเซอร์ต่างๆ เช่น กล้อง จีพีเอส เลเซอร์ เรดาร์ และอื่นๆ ในการตรวจจับข้อมูลภายในและภายนอกของรถ นอกจากนี้ยังใช้ตัวกรองคาลมาน (Unscented Kalman Filter: UKF) ในการประมาณค่า ระยะพิกัดของรถ ตำแหน่ง และความเร็วในการขับเคลื่อนของรถอัจฉริยะซึ่งเป็นรถที่สามารถขับเคลื่อนได้เองโดยปราศจากคนขับ

2.3 ผลการวิจัยที่เกี่ยวข้อง

การศึกษาของนายประมวล ตียะพานิชย์ (2553) เรื่อง **Kalman Filter** ผลของการศึกษาตัวกรองคาลมานถูกใช้ในการรวมเซ็นเซอร์และการรวมข้อมูล โดยปกติระบบเรียลไทม์จะสร้างการวัดซึ่งต่อเนื่องกันหลายชุดมากกว่าที่จะทำการวัดเพียงครั้งเดียว เพื่อให้ได้สถานะของระบบ การวัดจำนวนมากเหล่านี้ในที่สุดแล้วจะถูกนำมารวมกันทางการคำนวณทางคณิตศาสตร์เพื่อสร้างระบบของสถานะที่เกิดขึ้นในขณะนั้น ตัวอย่างการประยุกต์ใช้โดยพิจารณาจากปัญหาของการกำหนดตำแหน่งที่แม่นยำของรถบรรทุก รถบรรทุกสามารถถูกติดตั้งด้วยระบบจีพีเอสที่ให้ค่าประมาณของตำแหน่งภายในไม่กี่เมตรค่าประมาณของจีพีเอสมีแนวโน้มที่จะมีความผิดพลาดการอ่านค่าคลาดเคลื่อนอย่างรวดเร็ว ถึงแม้ว่าตลอดเวลาจะอยู่ภายในไม่กี่เมตรจากตำแหน่งจริงก็ตามตำแหน่งของรถบรรทุกยังสามารถประมาณโดยการผสมผสานความเร็วและทิศทางในช่วงเวลาหนึ่งซึ่งพิจารณาจากการนับจำนวนคันเร่งที่ถูกเหยียบและพวงมาลัยมีการหมุนเท่าใด เทคนิคนี้มีชื่อเรียกว่า dead reckoning โดยปกติแล้วการคำนวณนี้จะให้ค่าประมาณตำแหน่งของรถบรรทุกที่ราบรื่นมาก แต่มันจะผิดพลาดบางครั้งเมื่อเวลาผ่านไป โดยเกิดขึ้นเป็นความผิดพลาดเล็กๆที่สะสมกัน นอกจากนี้รถบรรทุกจะถูกคาดหวังว่าจะต้องปฏิบัติตามกฎของฟิสิกส์ ดังนั้นตำแหน่งของมันจึงควรที่จะเปลี่ยนไปตามสัดส่วนของความเร็วของรถในตัวอย่างนี้ตัวกรองคาลมานสามารถเป็นตัวช่วยในการดำเนินการในสองขั้นตอนที่แตกต่างกันการประมาณการณ์และการปรับปรุง ในขั้นตอนการประมาณการณ์ ตำแหน่งเก่าของรถบรรทุกจะมีการปรับเปลี่ยนตามกฎการเคลื่อนที่ทางกายภาพรวมถึงกับการเปลี่ยนแปลงใดๆ ที่เกิดขึ้นโดยการเหยียบคันเร่งและการหมุนพวงมาลัย ไม่เพียงแต่การประมาณตำแหน่งใหม่จะถูกคำนวณขึ้น แต่ความแปรปรวนร่วมใหม่จะถูกนำมาคำนวณด้วยเช่นกัน บางทีความแปรปรวนร่วมอาจเป็นส่วนเดียวกันกับความเร็วของรถบรรทุก เพราะเรามีความไม่แน่นอนเกี่ยวกับความถูกต้องของค่าประมาณ dead reckoning ที่ความเร็วสูง แต่มีความชัดเจนเกี่ยวกับตำแหน่งเมื่อรถเคลื่อนไหวยาวๆ ถัดมาในช่วงการปรับปรุง การวัดตำแหน่งของรถบรรทุกจะมาจากระบบจีพีเอส การวัดนี้นำมาซึ่งค่าความไม่แน่นอนจำนวนหนึ่ง และค่าความแปรปรวนร่วมเทียบกับค่าการคาดการณ์จากขั้นตอนก่อนหน้านี้ จะเป็นตัวกำหนดว่า จำนวนการวัดใหม่เท่าไรจึงจะส่งผลกระทบต่อค่าการคาดการณ์ที่ถูกปรับปรุง โดยอุดมคติแล้ว หากค่าประมาณ dead reckoning มีแนวโน้มที่จะผิดพลาดไปจากตำแหน่งจริงการวัดจีพีเอสจึงควรที่จะดึงค่าประมาณตำแหน่งกลับสู่ตำแหน่งจริง แต่ไม่ไปรบกวนจนถึงจุดที่จะทำให้เกิดการเปลี่ยนแปลงอย่างรวดเร็วและมีความผิดพลาดเกิดขึ้น ตัวกรองคาลมานจึงควรนำมาใช้เพื่อลดความผิดพลาดที่เกิดขึ้นนั่นเอง

การศึกษาของปัทม์ วัลยะเพ็ชร (2553) เรื่อง **Lane detection and tracking for driving aid system** ผลของการศึกษาการทำงานของระบบพบว่าเป็นระบบที่สามารถตรวจจับและติดตามเส้นแบ่งเลนของถนน โดยใช้หลักการทางด้านการประมวลผลภาพ คือ Canny Edge Detection, Hough Transform, Histogram Vertical Projection, Tracking และอัลกอริทึมอื่นๆ เข้ามาช่วย แล้วจึงวิเคราะห์ว่ารถยนต์อยู่ตำแหน่งใดบนท้องถนนจากนั้นระบบก็จะทำการเตือนด้วยเสียงหากขับรถออกนอกเลนโดยไม่ได้เปิดไฟเลี้ยวและในงานวิจัยชิ้นนี้จะเน้นการจับเส้นเลนของถนนเป็นหลัก การทดลองจะเป็นการตรวจจับเส้นเลนที่มีลักษณะเป็นเส้นตรงกับการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรง โดยผลลัพธ์ที่ได้นั้นระบบจะสามารถตรวจจับได้ดีหรือไม่ขึ้นขึ้นอยู่กับสภาพแสงและสภาพถนน (มีสัญญาณรบกวนเกิดขึ้นกับข้อมูล) หากเส้นเลนมีชัดเจนและมีแสงสว่างที่พอเหมาะ ก็สามารถตรวจจับเส้นได้อย่างแม่นยำ ตัวอย่างดังภาพประกอบ 7



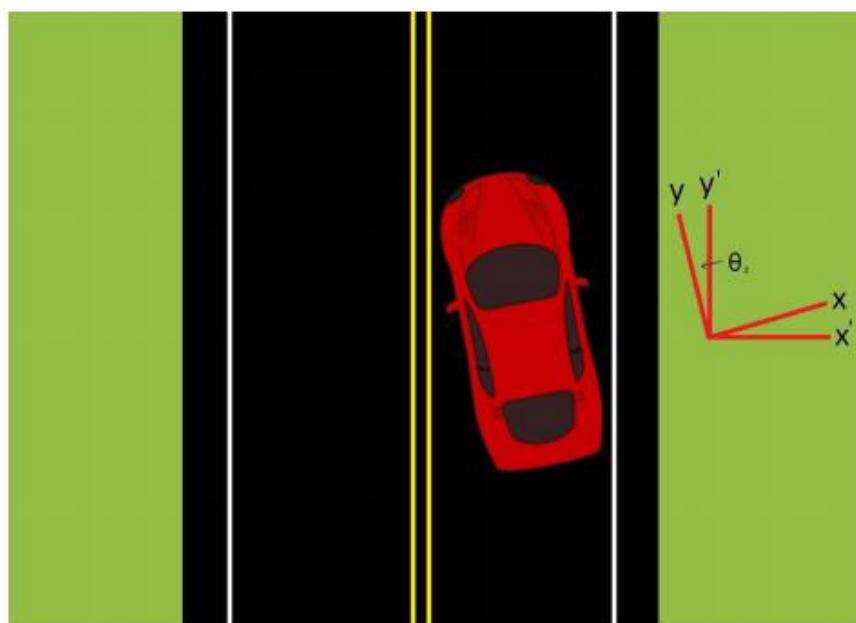
ภาพประกอบ 7 ผลการทดลองระบบ Lane detection and tracking for driving aid system

ที่มา: ปัทม์, 2553

การศึกษาของ Chung-Hsien Yu (2012) เรื่อง **An Improved Lane Detection Algorithm and the Definition of the Error Rate Standard** ผลของการศึกษาพบว่างานวิจัยนำเสนอวิธีการในการพัฒนาระบบตรวจจับเลน ที่ได้รับผลกระทบจากสัญญาณรบกวนแบบพัลส์ (Pulse) ซึ่งเป็นวิธีการที่นำเสนอถูกคิดค้นมาเพื่อช่วยให้วิธีตรวจจับเลนแบบ Error Rate Objectively สามารถทำงานได้อย่างมีประสิทธิภาพมากขึ้น โดยใช้อัลกอริทึม Sobel edge detection และ Canny edge

detection โดยผลการทดลองที่ได้มีประสิทธิภาพที่น่าพอใจ แต่เป็นการทดลองที่ใช้การขับเคลื่อนแบบเส้นตรงผ่านเส้นเลนที่มีลักษณะเป็นเส้นตรงเช่นเดียวกัน

การศึกษาของ P. K. Wang, P. A. Torrion, L. M. Collins, and Kenneth D. Morton (2012) เรื่อง **Rapid Position Estimation and Tracking for Autonomous Driving** ผลของการศึกษาพบว่างานวิจัยนำเสนอวิธีการระบบช่วยเหลือในการขับรถยนต์ที่มีการขับเคลื่อนด้วยความเร็วสูงบนถนนในชนบท โดยใช้เทคนิคในการตรวจจับเส้นเลน โดยการหาระยะห่างระหว่างจุดกึ่งกลางของรถยนต์กับเส้นเลนส์โดยใช้ตรีโกณมิติ และ คำนวณหาองศาโดยใช้หลักการของ Slope ซึ่งการทดลองนี้ทำอยู่ภายในข้อสันนิฐานที่ว่า รถยนต์ขับเคลื่อนไปในทิศทางที่เป็นเส้นตรงผ่านถนนที่เป็นเส้นตรงเช่นเดียวกันดังแสดงในภาพประกอบ 8



ภาพประกอบ 8 ตำแหน่งและทิศทางของรถยนต์ที่ใช้ในการทดลอง
ที่มา Wang et al, 2012

การศึกษาเอกสารประกอบการสอนวิชา M163 มหาวิทยาลัย University of North Carolina (2008) เรื่อง **Kalman Filter Application** ผลของการศึกษาพบว่าตัวกรองคาลมานสามารถลดปัญหาที่เกิดขึ้นจากสัญญาณรบกวนได้โดยใช้แบบจำลองที่สร้างขึ้น ดังนั้นแบบจำลองที่สร้าง

ขึ้นทั้งแบบจำลองระบบและแบบจำลองการวัดจำเป็นต้องมีความถูกต้อง โดยวิธีการสร้างแบบจำลองมีขั้นตอนที่สำคัญดังนี้

- ทำความเข้าใจปัญหาที่เกิดขึ้นกับระบบ แล้วสร้างแบบจำลอง โดยใช้สมการทางคณิตศาสตร์สร้างแบบจำลองขึ้นมาซึ่งขั้นตอนนี้สำคัญที่สุด
- สร้างแบบจำลองระบบ ซึ่งอาจจะไม่ใช่แบบจำลองที่มีประสิทธิภาพมาก แต่สามารถปรับเปลี่ยนแก้ไขได้ภายหลัง
- สร้างแบบจำลองการวัด ซึ่งข้อมูลในการสร้างแบบจำลองการวัดได้มาจากเซ็นเซอร์ที่ถูกนำมาใช้ร่วมกับระบบ ซึ่งสามารถมีได้มากกว่าหนึ่งอย่าง
- จำลองสัญญาณรบกวนและกำหนดค่า variance ของสัญญาณรบกวน ที่นิยมนำมาใช้ทดสอบระบบมักจะเป็นสัญญาณรบกวนแบบ Gaussian noise เพราะมีค่าเฉลี่ยใกล้เคียงศูนย์
- ทดสอบ EKF ใช้ matlab ในการจำลองการทำงาน
- ปรับค่าตัวแปร เพื่อเพิ่มความเที่ยงตรงในการประมาณค่าของ EKF การปรับเปลี่ยนค่า R (Measurement variance) ของค่าสเกลลาร์ที่ได้จากเครื่องมือวัด โดยถ้าเรามีความเชื่อว่าแบบจำลองการวัดของเรามีความน่าเชื่อถือ ค่า R จะมีค่าน้อย แต่ในทางกลับกัน ถ้าค่า R มีค่าสูงแสดงว่าเราไม่ค่อยมั่นใจกับแบบจำลองการวัดที่สร้างขึ้น ถ้าค่า $R = 0.001$ หมายความว่าเราเชื่อถือค่าที่ได้จากเครื่องมือวัด จึงทำให้ค่าประมาณที่ได้มีค่าใกล้เคียงกับค่าที่ได้จากเครื่องมือวัด แต่ถ้าเราให้ค่า $R = 0.1$ นั้นหมายความว่าเราไม่ค่อยเชื่อมั่นค่าที่ได้จากเครื่องมือวัด จึงทำให้อัลกอริทึมให้ความสำคัญกับแบบจำลองระบบมากกว่าแบบจำลองการวัด ทำให้ค่าที่ได้จากการประมาณมาจากค่าที่คำนวณได้จากเมทริกซ์จาคอบีเยนและค่าที่ได้จากเครื่องมือวัด ซึ่งหลักการนี้ถูกนำไปใช้กับค่า Q (State variance) ได้เช่นเดียวกัน

การศึกษาของอรรถนันท์ เล็กอุทัยม (2535) เรื่อง การประยุกต์ใช้เทคนิคคาลแมนฟิลเตอร์ในการพยากรณ์ปริมาณน้ำหลากเข้าเขื่อนอุบลรัตน์ ผลการศึกษาพบว่าการพยากรณ์ปริมาณน้ำทำในฉบับพลัน (Real Time Forecasting) เป็นการคำนวณน้ำทำโดยใช้ข้อมูลที่บันทึกได้ใหม่ล่าสุดเช่น 1 ชั่วโมง หรือ 1 วัน ก่อนเวลาที่ต้องพยากรณ์มาใช้คำนวณ การพยากรณ์ในลักษณะนี้สามารถให้ผลการคำนวณที่ทันต่อเหตุการณ์ เป็นประโยชน์อย่างยิ่งต่อการดำเนินการอ่างเก็บน้ำ (Reservoir Operation) และเตือนอุทกภัย (Flood Warning) การศึกษาครั้งนี้มุ่งที่จะประยุกต์ใช้ Kalman Filter

Technique ผนวกกับ Lump Model ชนิด Non-Linear Storage Function Model (Nlsfm) เพื่อเปรียบเทียบแบบจำลอง (Calibration) และเปรียบเทียบผลการคำนวณน้ำท่ากับผลจากการประยุกต์ใช้ NLSFM เพียงอย่างเดียว Kalman Filter Technique เป็นเทคนิคที่สามารถใช้ค่าความผิดพลาดจากผลการคำนวณของแบบจำลองคณิตศาสตร์ (NLSFM) ในปัจจุบันไปปรับพารามิเตอร์ของสมการหลักในแบบจำลองเพื่อการคำนวณปริมาณน้ำท่าในครั้งต่อไปได้ถูกต้องยิ่งขึ้น จากผลการเปรียบเทียบแบบจำลองโดยใช้ข้อมูลจำนวน 10 ปี สรุปได้ว่าค่าที่เหมาะสมในการกำหนดค่าพารามิเตอร์เริ่มต้นของแบบจำลองคือค่า Manning Coefficient เท่ากับ 0.030, Initial Error Covariance Matrix เท่ากับ 0.01% ของ State Variable Matrix และกำหนด Initial Model Error Covariance Matrix เท่ากับ 0.01% ของพารามิเตอร์ของสมการหลักในแบบจำลองคณิตศาสตร์ ผลการประยุกต์ใช้ Nlsfm และ Kalman Filter Mode (KFM) กับข้อมูลปริมาณฝนจากการวัดจริงจำนวน 8 ปี สรุปได้ว่าการใช้ KFM ให้ผลการคำนวณปริมาณน้ำท่าที่ดีกว่าผลจาก NLSFM คือให้ค่าเฉลี่ย รากของค่าเฉลี่ยความผิดพลาดกำลังสอง (RMSE) เท่ากับ 1.4644 mm และ 2.2563 mm ค่าเฉลี่ยความผิดพลาดการคำนวณปริมาณน้ำท่าสูงสุดเท่ากับ 15.70% และ 29.05% ค่าความผิดพลาดเวลาที่เกิดปริมาณน้ำท่าสูงสุดเท่ากับ 0-1 วัน และ 0-8 วัน ตามลำดับ ผลการประยุกต์ใช้ NLSFM และ KFM กับข้อมูลปริมาณฝนจากการคำนวณล่วงหน้าแบบ 1- 2- 3- และ 4-Day Average สรุปได้ว่า Kalman Filter Technique สามารถช่วยให้ผลการคำนวณถูกต้องกว่าการใช้ NLSFM อย่างเดียวซึ่งดีกว่าในลักษณะเดียวกับการใช้ข้อมูลฝนจากการวัดจริงทั้งทางด้านค่าเฉลี่ย RMSE ค่าเฉลี่ยความผิดพลาดการคำนวณปริมาณน้ำท่าสูงสุดและค่าความผิดพลาดการคำนวณเวลาที่เกิดปริมาณน้ำท่าสูงสุด จากการวิเคราะห์ทางสถิติด้วยวิธี Student t-test กับผลการประยุกต์ใช้ KFM และ NLSFM กับข้อมูลทั้งแบบปริมาณฝนจากการวัดจริงและปริมาณฝนจากการคำนวณล่วงหน้า ปรากฏว่าผลจากการใช้ KFM ไม่แตกต่างแบบมีนัยสำคัญ (Not Significantly Difference) เมื่อเทียบกับผลจากการใช้ NLSFM แต่มีแนวโน้มว่าให้ผลการคำนวณที่ถูกต้องกว่า

สรุป

งานวิจัยในการเพิ่มความปลอดภัยของผู้ขับขี่และประสิทธิภาพในการใช้งานของพาหนะ คือลักษณะเด่นของระบบการขนส่งและจราจรอัจฉริยะ โดยการใช้เซ็นเซอร์และเทคโนโลยีอัจฉริยะแบบต่างๆ ผสานเข้าด้วยกันในการทำควมเข้าใจสภาพแวดล้อมโดยรวมของยานพาหนะเพื่อใช้ในการช่วยเหลือ เตือนผู้ขับขี่ หรือแม้กระทั่งสามารถขับตัวเองโดยอัตโนมัติได้ ยกตัวอย่าง เช่น ระบบเตือนการเปลี่ยนเลน ระบบป้องกันการชน หรือระบบช่วยขับ เป็นต้น โดยระบบเหล่านี้จำเป็นต้องสังเกตการณ์ผู้ขับขี่และสภาพแวดล้อมโดยรวมของยานพาหนะ ณ เวลานั้นๆ เพื่อที่จะสามารถ

พยากรณ์เหตุการณ์ที่จะเกิดขึ้นได้ เพื่อเตือนผู้ขับขี่หรืออาจมีการช่วยเหลือในการบังคับพวงมาลัยหรือเบรก โดยใช้ตัวกรองที่ประสิทธิภาพและได้รับความนิยมอย่างยาวนานและแพร่หลาย เช่น ตัวกรองคาร์บอน หรือ ตัวกรองพาร์ติเคิล เป็นต้น

ดังนั้นระบบตรวจจับเส้นเลนจึงถือว่าเป็นส่วนประกอบหนึ่งที่สำคัญในการใช้พัฒนาระบบการขนส่งและจราจรอัจฉริยะ โดยระบบตรวจจับเลนจะทำหน้าที่ในการระบุตำแหน่งและบอกลักษณะของเส้นเลน ทั้งบนถนนไฮเวย์หรือถนนในชนบท เพื่อใช้ข้อมูลเหล่านี้เป็นอินพุตเพื่อป้อนให้กับระบบอื่นๆในระบบการขนส่งและจราจรอัจฉริยะต่อไป

บทที่ 3

ระเบียบวิธีวิจัย

3.1 ระเบียบวิธีวิจัย

งานวิจัยชิ้นนี้ได้สร้างแบบจำลองที่รองรับการขับเคลื่อนของรถยนต์ส่วนบุคคลที่มีลักษณะแบบไม่เป็นเชิงเส้นและถูกจำลองแบบ 2 มิติ ในการคำนวณหาตัวแปรที่จำเป็นสำหรับสถานะปัจจุบันที่จะถูกนำไปใช้ร่วมกับค่าที่ได้จากเซ็นเซอร์ในการประมาณค่าความเป็นไปได้ของสถานะอนาคต โดยใช้ความรู้ทางเรขาคณิตวิเคราะห์ในการสร้างแบบจำลองการทำงานวงโคจรของรถยนต์ส่วนบุคคลขึ้นมาเพื่อช่วยให้การประมาณค่าของตัวกรองกาลมาน ซึ่งเป็นส่วนที่ถือว่าเป็นหัวใจสำคัญในการประยุกต์ใช้การตัวกรองกาลมานให้มีความถูกต้องแม่นยำและเหมาะกับการใช้งานในสภาพแวดล้อมที่มีการรบกวนสูงอย่างระบบตรวจจับเลน โดยสร้างแบบจำลองในที่นี้ได้จำลองการทำงานภายใต้สมมุติฐาน ดังนี้

ให้มีการใช้กล้องวิดีโอในการเก็บรวบรวมข้อมูลของถนนซึ่งถูกติดตั้งที่กระจกทางด้านข้าง มีความเร็วในการขับเคลื่อนประมาณ 60 กิโลเมตรต่อชั่วโมง ภาพจะถูกนำมาวิเคราะห์ในอัตราส่วน 30 เฟรมต่อวินาที โดยข้อมูลที่ได้นำมาจากและจำลองความกว้างของถนนอยู่ที่ 3 ถึง 3.7 เมตร และข้อมูลที่ใช้ในการทดสอบระบบได้มาจากการจำลองข้อมูลที่ได้จากภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง การจำลองข้อมูลที่ได้จากภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นโค้งผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง การจำลองข้อมูลที่ได้จากภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นโค้งผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง และการจำลองข้อมูลที่ได้จากภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นโค้งผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นโค้ง

3.2 ขั้นตอนการดำเนินงานวิจัย

อีกหนึ่งตัวอย่างของการสร้างแบบจำลองวงโคจรของรถยนต์ซึ่งสร้างโดย Wang, Patrick และทีมงาน (Wang and others., 2012) ซึ่งสามารถประมาณค่าตัวแปรที่ใช้ในการเคลื่อนที่ของรถยนต์ได้ แต่ภายใต้เงื่อนไขของถนนที่มีลักษณะเป็นเส้นตรงเท่านั้น ซึ่งทำให้ผู้วิจัยเกิดแนวคิดที่จะสร้างแบบจำลองเพื่อทำนายเส้นทางโคจรของรถยนต์ส่วนบุคคลที่มีการขับเคลื่อนแบบตรงและโค้งผ่านเส้นแบ่งเลนที่เป็นเส้นตรงและโค้งโดยใช้ตัวกรองกาลมาน โดยลำดับการสร้างแบบจำลองแสดงดังต่อไปนี้

3.2.1 การสร้างแบบจำลองระบบ (System State) คือขั้นตอนแรกในการจำลองรูปแบบการเคลื่อนที่ในแบบต่างๆของรถยนต์ผ่านเส้นแบ่งเลนที่มีลักษณะตรงและโค้ง ดังภาพประกอบ 9 เพื่อใช้ในการค้นหาค่าตัวแปรที่ใช้ในการคำนวณหรือประมาณค่าในสถานะอนาคต ($t+1$) จากข้อมูลที่ได้จากสถานะปัจจุบัน (t)

โดยค่าตัวแปรที่ใช้ในการสร้างฟังก์ชันเพื่อประมาณค่าแสดงดังนี้

$$X^t = [d_l^t, \theta_l^t, \kappa_l^t, d_r^t, \theta_r^t, \kappa_r^t, v^t, \dot{\theta}^t]^T$$

โดย

X^t : เมตริกซ์ที่ใช้เก็บค่าตัวแปรของแบบจำลองระบบ

d_l^t และ d_r^t : ระยะห่างระหว่างยานพาหนะกับขอบของเส้นแบ่งเลนทางด้านซ้ายและขวา

θ_l^t และ θ_r^t : ค่าองศา (*Tangent*) ของเส้นแบ่งเลนทางด้านซ้ายและขวา

κ_l^t และ κ_r^t : ค่าความโค้งของเส้นแบ่งเลนทางด้านซ้ายและขวา

v^t : ค่าความเร็วของรถยนต์

$\dot{\theta}^t$: ค่าความเร็วเชิงมุมของรถยนต์

3.2.2 การสร้างสมการระบบ (State transition function) คือขั้นตอนในการคำนวณหาสมการที่จะใช้ในการคำนวณหาตัวแบบของแบบจำลองระบบในสถานะอนาคต โดยใช้ค่าที่ได้จากสถานะอดีตและปัจจุบันแทนค่าลงไปในสมการระบบ ดังนี้

$$d_l^{t+1} = f(d_l^t, \theta_l^t, \kappa_l^t, v^t, \dot{\theta}^t)$$

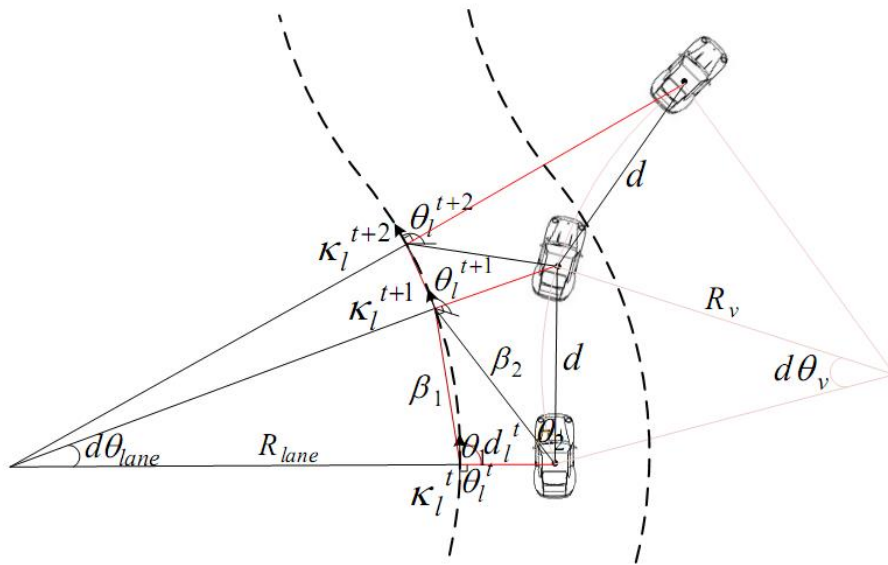
$$\theta_l^{t+1} = f(\theta_l^t + (\dot{\theta}^t \Delta t) + (s\kappa_l^t))$$

$$\kappa_l^{t+1} = \kappa_l^t$$

$$d_r^{t+1} = f(d_r^t, \theta_r^t, \kappa_r^t, v^t, \dot{\theta}^t)$$

$$\theta_r^{t+1} = f(\theta_r^t + (\dot{\theta}^t \Delta t) + (s\kappa_r^t))$$

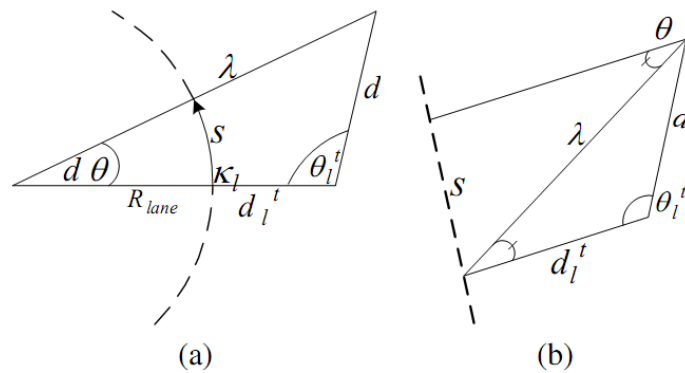
$$\kappa_r^{t+1} = \kappa_r^t$$



ภาพประกอบ 9 แสดงแบบจำลองเรขาคณิตวิเคราะห์
ของวงโคจรการขับเคลื่อนของรถยนต์ส่วนบุคคล

โดยลำดับขั้นตอนในการสร้างสมการเพื่อหาค่าตัวแปรของแบบจำลองระบบแสดงดังต่อไปนี้

1. การคำนวณหาค่าความยาวส่วนโค้ง (Arc-Length: s) แสดงการคำนวณหาค่าตัวแปรดัง
ภาพประกอบ 10



ภาพประกอบ 10 แสดงค่าตัวแปรที่ใช้ในการคำนวณหาค่าความยาวส่วนโค้ง

- (a) ภาพจำลองขณะรถยนต์ขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นโค้ง
- (b) ภาพจำลองขณะรถยนต์ขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรง

สมการในการหาค่าความยาวเส้นโค้งเมื่อรถยนต์ขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรงแสดงดังสมการที่ (1) ส่วนสำหรับเส้นโค้งแสดงดังสมการที่ (2)

$$s = \lambda \sin(\theta) \quad \text{สมการที่ (1)}$$

$$s = \kappa d \theta \quad \text{สมการที่ (2)}$$

2. สมการในการหาค่าระยะห่างระหว่างรถยนต์กับเส้นแบ่งเลนที่เป็นเส้นตรงและเส้นโค้งแสดงดังสมการที่ (3) ถึงสมการที่ (4) โดยมีค่าตัวแปรแสดงดังภาพประกอบ 9

$$\beta_1 = 2 \frac{1}{\kappa} \sin\left(\frac{s\kappa}{2}\right) \quad \text{สมการที่ (3)}$$

$$\beta_2 = \sqrt{\beta_1^2 + d_l^2 - (2\beta_1 d_l \cos(\theta_2))} \quad \text{สมการที่ (4)}$$

$$d = 2 \frac{v}{\dot{\theta}} \sin\left(\frac{\dot{\theta} \Delta t}{2}\right) \quad \text{สมการที่ (5)}$$

$$d_l^{t+1} = \sqrt{\beta_2^2 + d^2 - (2\beta_2 d \cos(\theta_2))} \quad \text{สมการที่ (6)}$$

จากสมการจะเห็นได้ว่าการใช้ค่าตัวแปรสำหรับเส้นแบ่งเลนที่เป็นเส้นโค้งซึ่งสามารถใช้ร่วมกันได้กับการคำนวณสำหรับเส้นแบ่งเลนที่เป็นเส้นตรง เพราะเมื่อค่า κ และ $\dot{\theta}$ มีค่าเท่ากับศูนย์หรือเข้าใกล้ศูนย์จะส่งผลให้สมการที่ (3) และ (5) มีค่าเท่ากับสมการที่ใช้คำนวณการขับเคลื่อนของรถยนต์ผ่านเส้นแบ่งเลนที่เป็นเส้นตรง แสดงดังสมการที่ (7) และ (8)

$$\beta_1^s = \lim_{\kappa \rightarrow 0} 2 \frac{1}{\kappa} \sin\left(\frac{s\kappa}{2}\right) = s \quad \text{สมการที่ (7)}$$

$$d^s = \lim_{\dot{\theta} \rightarrow 0} 2 \frac{v}{\dot{\theta}} \sin\left(\frac{\dot{\theta} \Delta t}{2}\right) = v \Delta t \quad \text{สมการที่ (8)}$$

3.2.3 สมการของระบบ ซึ่งเป็นสมการหลักของแบบจำลองแสดงดังสมการที่ (9)

$$X^{t+1} = f(X^t, u^t) + v^t \quad \text{สมการที่ (9)}$$

3.2.4 สมการของเซ็นเซอร์ ซึ่งเป็นสมการที่ใช้ในการกำหนดค่าตัวแปรสำหรับเซ็นเซอร์ที่จะถูกนำมาใช้ในการสร้างแบบจำลองดังสมการที่ (10) และสมการที่ (11)

$$Z^t = [d_l^t, \theta_l^t, \kappa_l^t, d_r^t, \theta_r^t, \kappa_r^t]^T \quad \text{สมการที่ (10)}$$

$$Z^{t+1} = HX^t + v^t \quad \text{สมการที่ (11)}$$

3.2.5 การแปลงฟังก์ชันแบบไม่เป็นเชิงเส้นให้เป็นแบบเชิงเส้น (Linearization) เนื่องจาก $f(X^t)$ คือฟังก์ชันแบบไม่เป็นเชิงเส้นดังนั้นจึงจำเป็นต้องใช้ *Extended Kalman Filter: EKF* เข้ามาใช้งานแทนตัวกรองคาลมานแบบธรรมดาที่รองรับการใช้งานฟังก์ชันที่เป็นเชิงเส้นเท่านั้น โดยสมการที่ใช้ในการแปลงแสดงดังสมการที่ (12) ซึ่งใช้หลักการของเมทริกซ์จาโคเบียน (Jacobian) แสดงดังสมการที่ (13)

$$f(X^t, u^t) \approx f(\hat{X}^t, u^t) + J^{ft} (X^t - \hat{X}^t) \quad \text{สมการที่ (12)}$$

$$J^{ft} = \frac{\partial f(X^t, u^t)}{\partial X^t} \quad \text{สมการที่ (13)}$$

3.2.6 การกำหนดค่าเริ่มต้นให้กับค่าตัวแปรของสมการระบบ (Initialization) ในที่นี้ผู้วิจัยกำหนดให้ทุกตัวแปรมีค่าดังสมการที่ (14)

$$X_0^t = [d_l^t, \theta_l^t, \kappa_l^t, d_r^t, \theta_r^t, \kappa_r^t, 0, 0] \quad \text{สมการที่ (14)}$$

3.2.7 การกำหนดค่าความแปรปรวนของสัญญาณรบกวน (Noise Covariance) ในที่นี้ผู้วิจัย
กำหนดให้ทุกตัวแปรมีค่าดังสมการที่ (15)

$$S_t = \begin{bmatrix} (\sigma d_l)^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & (\sigma d_r)^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & (\sigma \theta_l)^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & (\sigma \theta_r)^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & (\sigma \kappa_l)^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & (\sigma \kappa_r)^2 \end{bmatrix} \quad \text{สมการที่ (15)}$$

3.2.8 อัลกอริทึมของแบบจำลอง ในการประยุกต์ใช้ EKF แสดงดังนี้

1. Input Z_0 .
2. Calculate $\hat{X}_0$ and P_0
3. For $t=1, \dots, T$. do
 - (a) Predict $\hat{X}_t^- = f(\hat{X}_{t-1}, u_{t-1})$
 - (b) Calculate J_{ft} and Q_t
 - (c) Predict $P_t^- = J_{ft} P_{t-1} J_{ft}^T + Q_t$
 - (d) If Z_t is unavailable
 - i. Let $\hat{X}_t = \hat{X}_t^-$
 - ii. Let $P_t = P_t^-$
 - (e) otherwise
 - i. Calculate J_{ht} , S_t , and Kalman gain

$$K_t = P_t^- J_{ht}^T (J_{ht} P_t^- J_{ht}^T + S)^{-1}$$
 - ii. Estimate $\hat{X}_t = \hat{X}_t^- + K_t (Z_t - h_t(\hat{X}_t^-))$
 - iii. Update the error estimation

$$P_t = (I - K_t J_{ht}) P_t^-$$

3.3 การรวบรวมข้อมูล

ผู้วิจัยได้จำลองการขับเคลื่อนของรถยนต์ในลักษณะต่างๆ เพื่อใช้ในการทดลองระบบการจำลองที่ได้อ้างอิงจากข้อมูลที่เกิดขึ้นในสภาพแวดล้อมที่ควรจะเกิดขึ้นจากการเก็บรวบรวมข้อมูลมาจาก

- ภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง

- ภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นโค้งผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง
- ภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นตรงผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นโค้ง
- ภาพถ่ายวิดีโอของการขับเคลื่อนของรถยนต์ที่เป็นเส้นโค้งผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นโค้ง

3.4 การวิเคราะห์ข้อมูล

ในการสร้างแบบจำลองในครั้งนี้ผู้วิจัยได้ทำการจำลองข้อมูลที่ได้จากเซ็นเซอร์ในการรับข้อมูลตัวแปรต่างๆที่ใช้ในการจำลองการขับเคลื่อนดังได้อธิบายไปแล้วในข้อ 3.3 และได้จำลองการทำงานของแบบจำลองที่สร้างขึ้นด้วยโปรแกรม Matlab โดยอัลกอริทึมของตัวกรองคาลมานแสดงในข้อ 3.3.8 โดยแบบจำลองมีความสามารถในการประมาณค่าของสถานะอนาคตโดยใช้ข้อมูลจากเซ็นเซอร์ สถานะอดีต และสถานะปัจจุบัน

ในการประเมินประสิทธิภาพของแบบจำลองที่สร้างขึ้น ผู้วิจัยใช้ RMSE ซึ่งทฤษฎีเบื้องต้นได้อธิบายไว้ในบทที่ 2 หัวข้อความรู้พื้นฐาน โดยถ้าค่าเปอร์เซ็นต์ที่ได้จะเป็นตัววัดประสิทธิภาพในการลดสัญญาณรบกวนที่เกิดขึ้นของแบบจำลอง ซึ่งค่าที่ได้ยิ่งมีค่าน้อยก็แสดงถึงประสิทธิภาพในการลดสัญญาณรบกวนที่ดี แต่ถ้ามีค่ามากหรือเกือบเท่าสัญญาณรบกวนแสดงว่าประสิทธิภาพในการประมาณค่าต่ำเพราะค่าที่ได้คือค่าสัดส่วนระหว่างค่าของสัญญาณรบกวนกับค่าของการประมาณค่าที่ได้จากแบบจำลอง

บทที่ 4

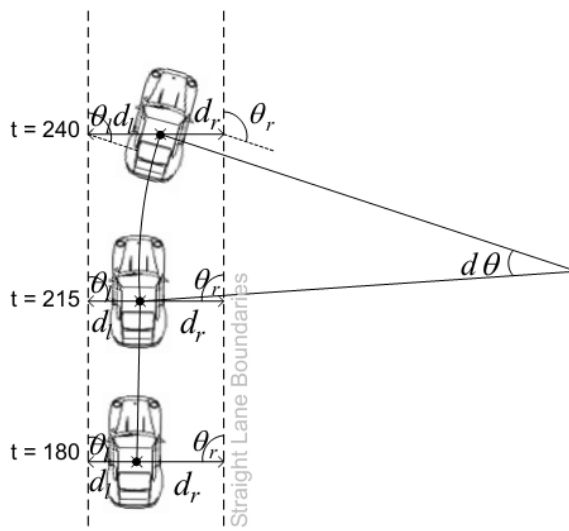
ผลการวิเคราะห์ข้อมูล

4.1 ผลการวิเคราะห์ข้อมูล

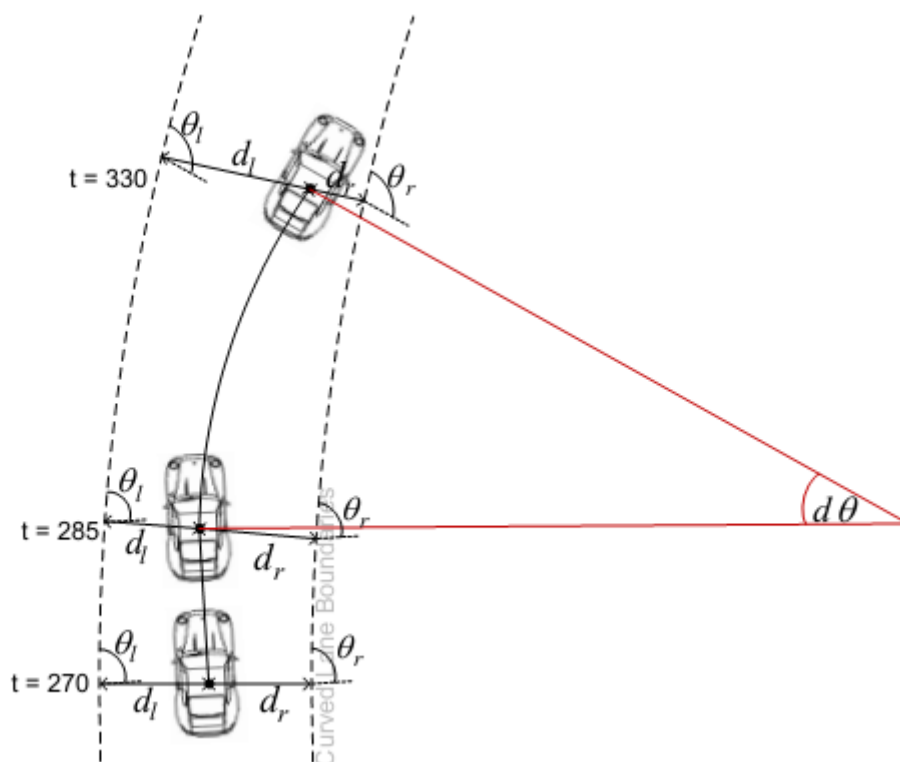
เพื่อประเมินประสิทธิภาพการทำงานของแบบจำลอง ผู้วิจัยได้จำลองการขับเคลื่อนของรถยนต์ผ่านเลนที่มีความกว้างประมาณ 3.5 เมตร โดยใช้ความเร็วประมาณ 60 กิโลเมตรต่อชั่วโมง และมีการประมวลผลภาพ 30 เฟรมต่อวินาที โดยการจำลองในแต่ละครั้งจะประกอบด้วยการขับเคลื่อนแบบตรงและแบบโค้งผ่านเส้นแบ่งเลนที่เป็นทั้งเส้นตรงและเส้นโค้ง

โดยการจำลองแบ่งออกเป็นสองแบบคือ การจำลองการขับเคลื่อนของรถยนต์ผ่านสภาพแวดล้อมที่มีสัญญาณรบกวนเพียงอย่างเดียวและการจำลองการขับเคลื่อนผ่านสภาพแวดล้อมที่มีสัญญาณรบกวนและการขาดหายไปของข้อมูลบางส่วนเพื่อทดสอบประสิทธิภาพในการประมาณค่าของแบบจำลอง

ผลการจำลองการขับเคลื่อนของรถยนต์ผ่านสภาพแวดล้อมที่มีสัญญาณรบกวนเพียงอย่างเดียวแสดงลักษณะการขับเคลื่อนแบบเส้นตรงและ โค้งผ่านเส้นแบ่งเลนที่มีลักษณะตรง แสดงดังภาพประกอบ 11 และ การขับเคลื่อนแบบเส้นตรงและโค้งผ่านเส้นแบ่งเลนที่มีลักษณะโค้ง แสดงดังภาพประกอบ 12



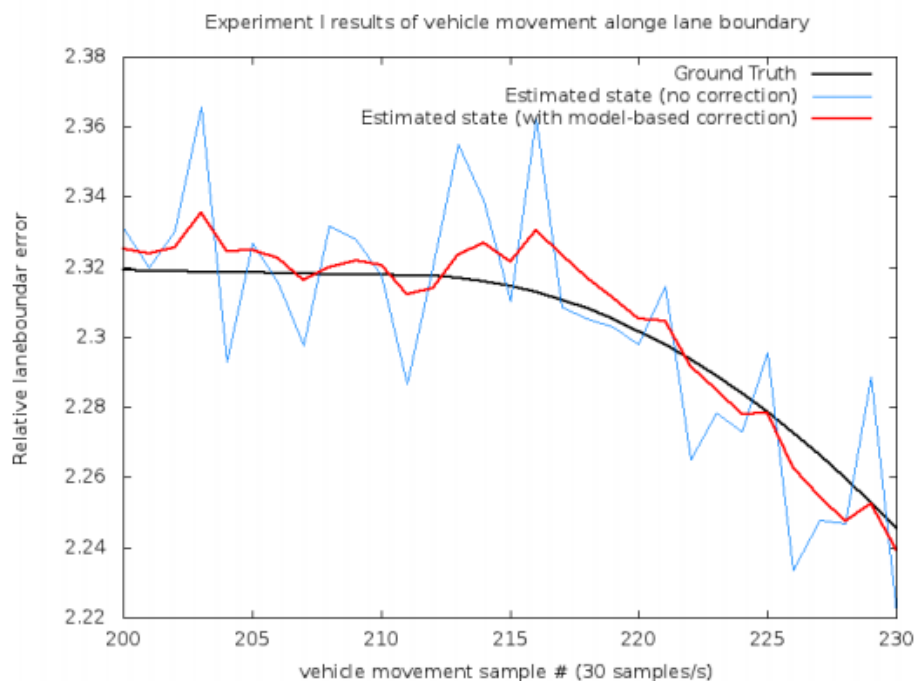
ภาพประกอบ 11 แสดงลักษณะการขับเคลื่อนของรถยนต์ผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นตรง ณ ช่วงเวลาต่างๆ



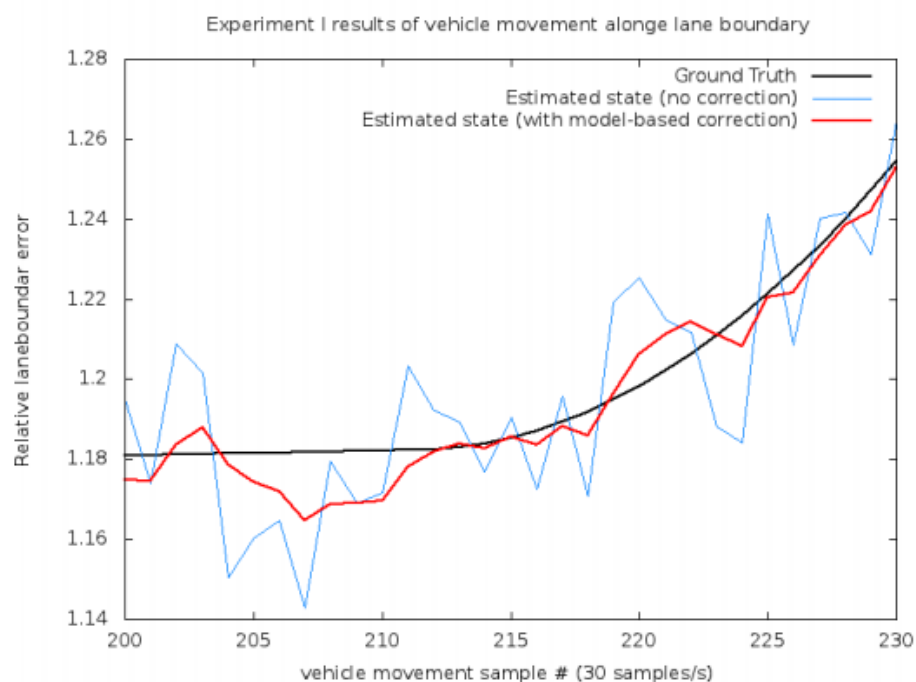
ภาพประกอบ 12 แสดงลักษณะการขับเคลื่อนของรถยนต์
ผ่านเส้นแบ่งเลนที่มีลักษณะเป็นเส้นโค้ง ณ ช่วงเวลาต่างๆ

ผลการจำลองแสดงดังภาพประกอบ 13 และ 14 โดยสีดำแสดงถึงข้อมูลจริง (Ground Truth) สีฟ้าแสดงถึงข้อมูลที่ได้จากเซ็นเซอร์ (Estimated state: no correction) และสีแดงคือข้อมูลที่ผ่านการประมาณค่าโดยแบบจำลอง (Estimated state: with model-based correction)

ซึ่งผลจากการทดลองทั้งสองแบบแสดงให้เห็นถึงประสิทธิภาพในการประมาณค่าของแบบจำลองที่สร้างขึ้น ในที่นี้ผู้วิจัยได้หาค่า Root Mean Square Error (RMSE) เพื่อแสดงการเปรียบเทียบค่าของข้อมูลที่มีสัญญาณรบกวนกับค่าประมาณที่ได้จากแบบจำลอง โดยในกรณีที่ไม่มี การขาดหายไปของข้อมูลแสดงดังภาพประกอบ 13 และ 14 แบบจำลองสามารถลดสัญญาณรบกวน ได้มากกว่า 50% และในกรณีที่มีสัญญาณรบกวนและการขาดหายไปของข้อมูลประสิทธิภาพของแบบจำลองลดลงตามจำนวนการขาดหายของสัญญาณที่เพิ่มมากขึ้น โดยกราฟเปรียบเทียบประสิทธิภาพการทำงานของระบบแสดงดังภาพประกอบ 15 และ 16

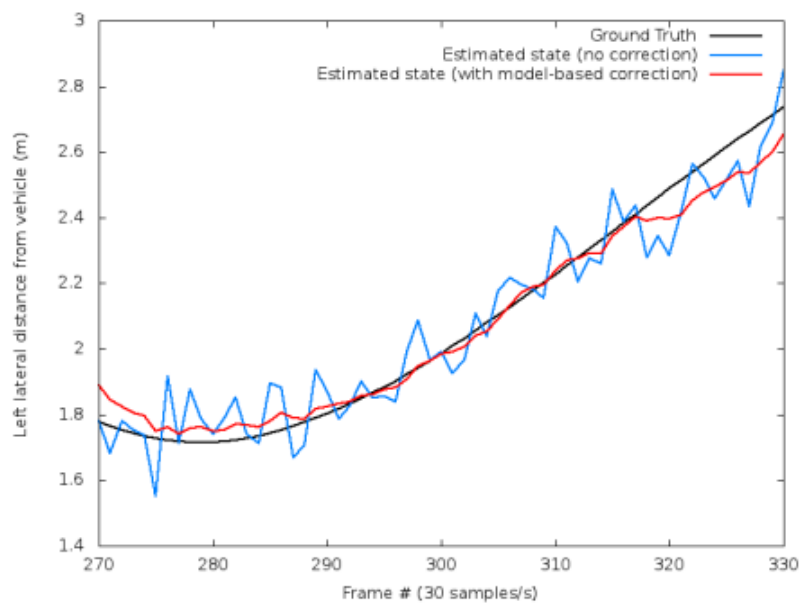


(a)

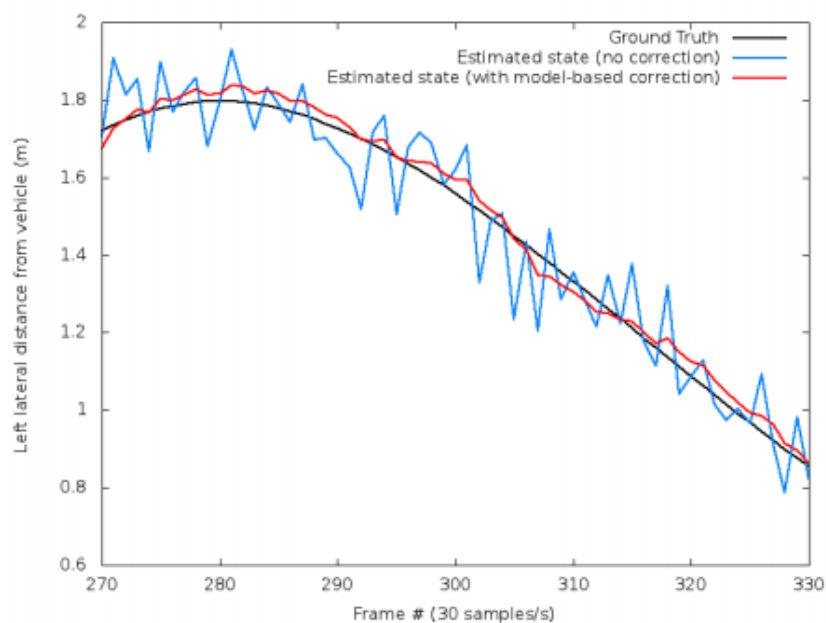


(b)

ภาพประกอบ 13 แสดงผลการประมาณค่าของแบบจำลองเมื่อรถยนต์ขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นตรง โดยรถขับเคลื่อนแบบเส้นตรงจากเฟรมที่ 200 ถึง 215 และเปลี่ยนลักษณะการขับเคลื่อนแบบโค้งจากเฟรมที่ 216 ถึง 230 (a) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางซ้าย (b) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางขวา

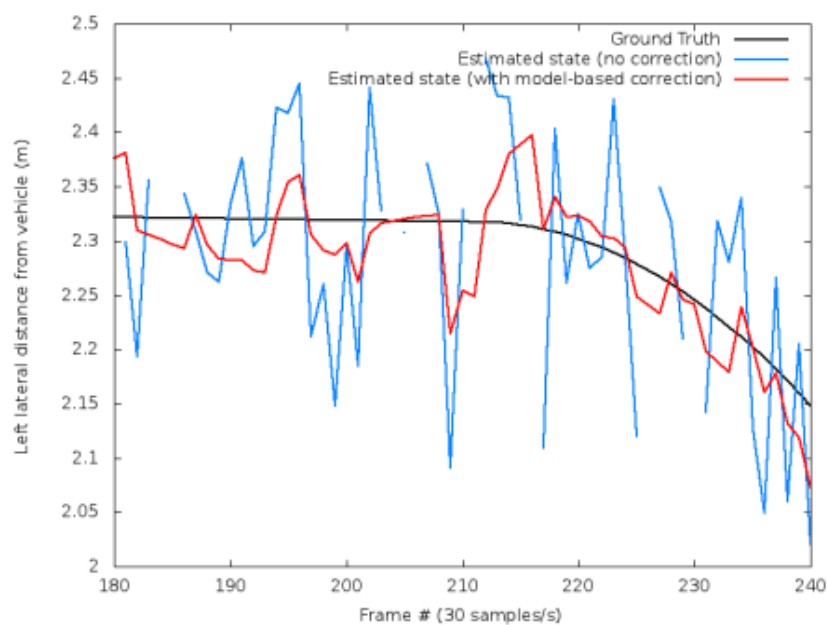


(a)

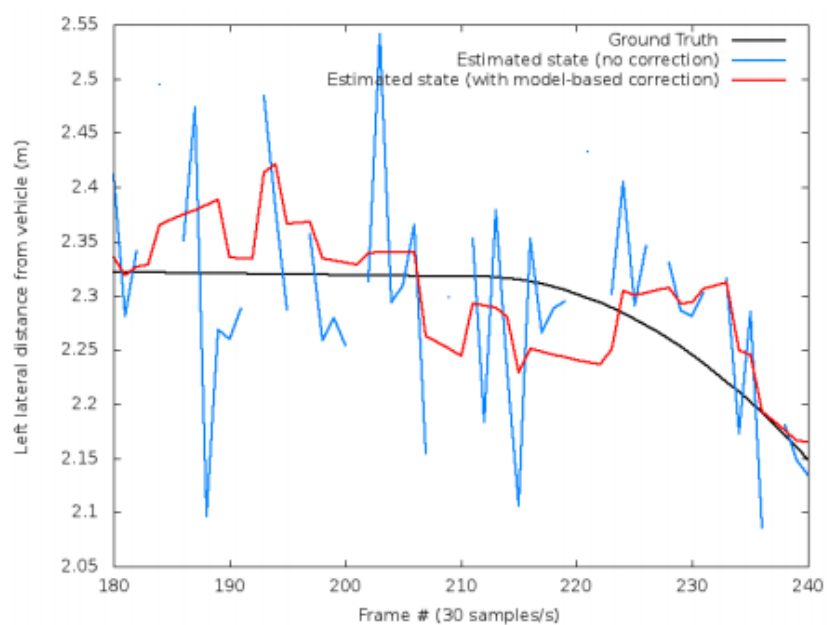


(b)

ภาพประกอบ 14 แสดงผลการประมาณค่าของแบบจำลองเมื่อรถยนต์ขับเคลื่อนผ่านเส้นแบ่งเลนที่เป็นเส้นโค้ง โดยรถขับเคลื่อนแบบเส้นตรงจากเฟรมที่ 200 ถึง 215 และเปลี่ยนลักษณะการขับเคลื่อนแบบโค้งจากเฟรมที่ 216 ถึง 230 (a) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางซ้าย (b) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางขวา



(a)



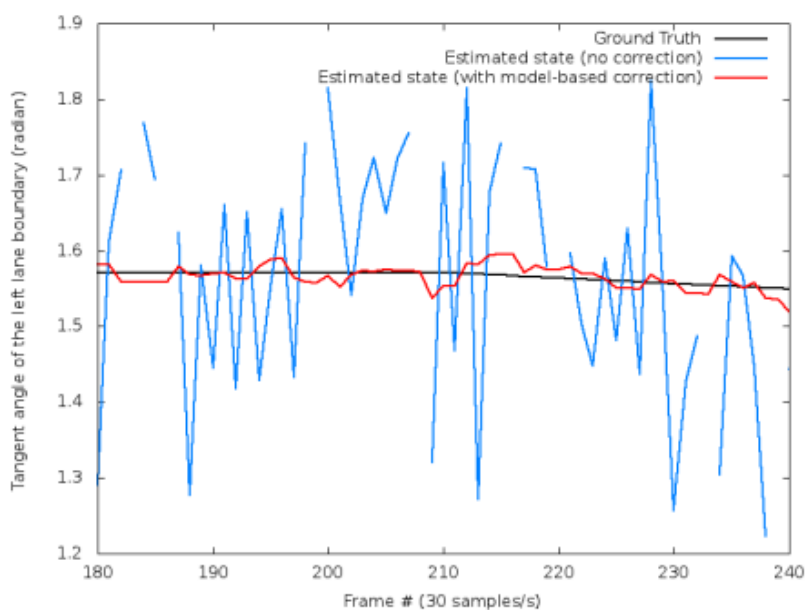
(b)

ภาพประกอบ 15 แสดงผลการประมาณค่าของแบบจำลอง

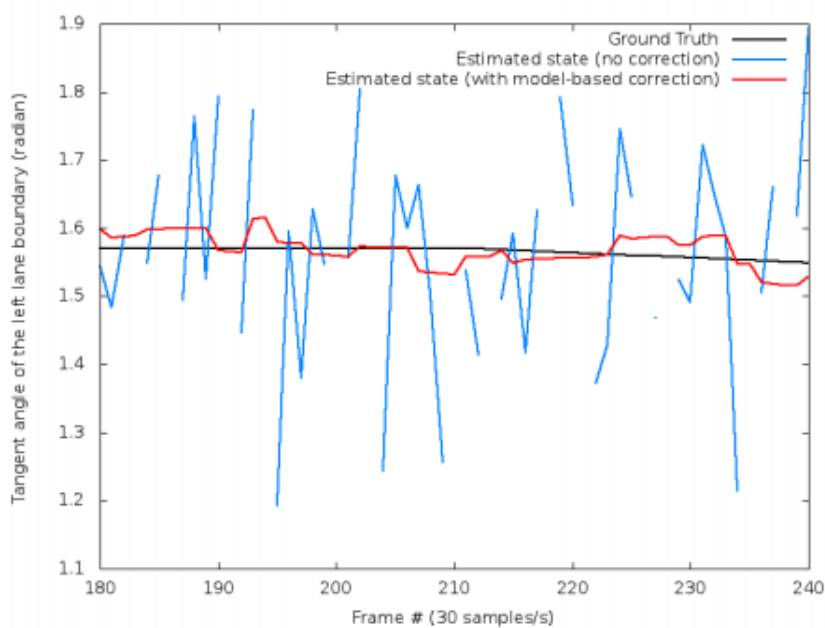
เช่นเดียวกับภาพประกอบ 13 แต่มีการเพิ่มการขาดหายไปของข้อมูลเข้าไป

(a) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางซ้ายที่มีการขาดหายไปของข้อมูล 15 %

(b) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางขวาที่มีการขาดหายไปของข้อมูล 25%



(a)



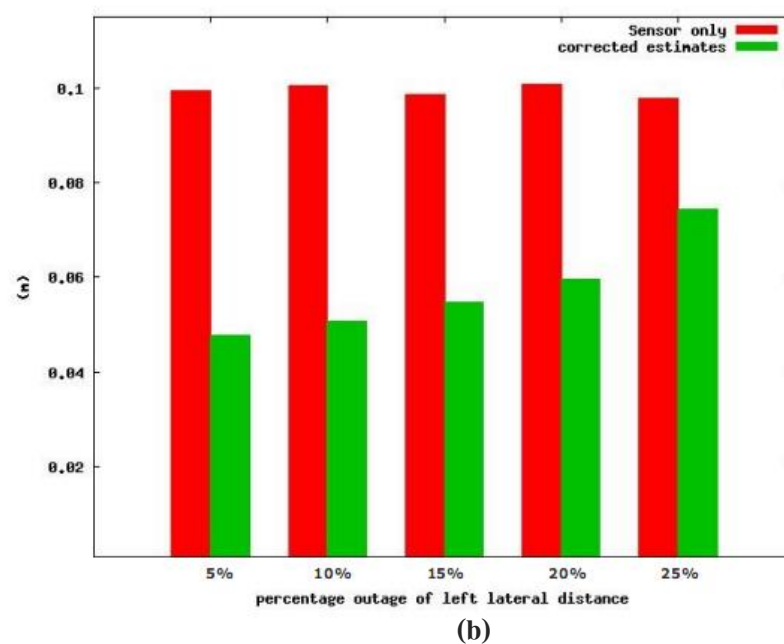
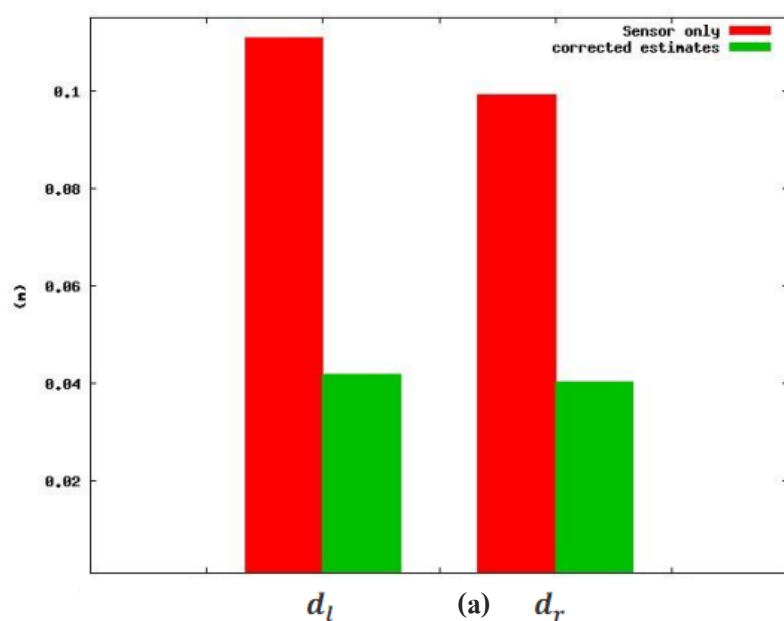
(b)

ภาพประกอบ 16 แสดงผลการประมาณค่าของแบบจำลอง

เช่นเดียวกับภาพประกอบ 14 แต่มีการเพิ่มการขาดหายไปของข้อมูลเข้าไป

(a) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางซ้ายที่มีการขาดหายไปของข้อมูล 15 %

(b) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางขวาที่มีการขาดหายไปของข้อมูล 25%



ภาพประกอบ 17 แสดงค่า RMSE

โดยสีแดงคือค่าของเซ็นเซอร์ส่วนสีเขียวคือค่าที่ได้ผ่านการประมาณค่าโดยแบบจำลอง

a) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางซ้ายและขวา

(b) ค่าระยะห่างระหว่างขอบเส้นเลนกับรถยนต์ทางซ้ายที่มีการขาดหายไปของข้อมูลจาก 5 ถึง 25%

การจำลองการขับเคลื่อนผ่านสภาพแวดล้อมที่มีสัญญาณรบกวนและการขาดหายไปของข้อมูลบางส่วนเพื่อทดสอบประสิทธิภาพในการประมาณค่าของแบบจำลอง แสดงดังภาพประกอบ 15 และ 16

บทที่ 5

สรุป อภิปราย และข้อเสนอแนะ

5.1 สรุป

บทความวิจัยนี้ได้นำเสนอการสร้างแบบจำลองในการประมาณค่าระยะห่างระหว่างรถยนต์และขอบของเส้นแบ่งเลนเพื่อช่วยในการเพิ่มประสิทธิภาพการทำงานของระบบตรวจจับเลน ซึ่งถือว่าเป็นส่วนที่มีความสำคัญเป็นอย่างมากเพราะค่าที่ได้จากระบบจะถูกนำไปใช้ต่อไปในระบบการขนส่งและจราจรอัจฉริยะ โดยแบบจำลองนี้สามารถรองรับการขับเคลื่อนของรถยนต์ทั้งแบบที่เป็นเส้นตรงและเส้นโค้งผ่านเส้นแบ่งเลนทั้งที่เป็นเส้นตรงและเส้นโค้งเช่นเดียวกัน ซึ่งผลจากการทดสอบประสิทธิภาพของแบบจำลองแสดงให้เห็นว่าเป็นแบบจำลองที่มีประสิทธิภาพสูง เพราะสามารถลดสัญญาณรบกวนได้มากกว่า 50 % และยังคงประสิทธิภาพอยู่แม้ว่าข้อมูลที่ได้จะมีทั้งสัญญาณรบกวนและสัญญาณที่ขาดหายไปโดยประสิทธิภาพของแบบจำลองแปรผกผันกับจำนวนสัญญาณที่ขาดหายไป โดยผู้วิจัยได้วางแผนที่จะนำแบบจำลองนี้ไปประยุกต์ใช้งานจริงร่วมกับระบบตรวจจับเลนต่อไป

5.2 อภิปราย

งานวิจัยชิ้นนี้ได้สร้างแบบจำลองที่รองรับการขับเคลื่อนของรถยนต์ส่วนบุคคลที่มีลักษณะแบบไม่เป็นเชิงเส้นและถูกจำลองแบบ 2 มิติ ในการคำนวณหาตัวแปรที่จำเป็นสำหรับสถานะปัจจุบันที่จะถูกนำไปใช้ร่วมกับค่าที่ได้จากเซ็นเซอร์ในการประมาณค่าความเป็นไปได้ของสถานะอนาคต โดยใช้ความรู้ทางเลขาคณิตวิเคราะห์ในการสร้างแบบจำลองการทำนายวงโคจรของรถยนต์ส่วนบุคคลขึ้นมาเพื่อช่วยให้การประมาณค่าของตัวกรองคาลมาน ซึ่งเป็นส่วนที่ถือว่าเป็นหัวใจสำคัญในการประยุกต์ใช้การตัวกรองคาลมานให้มีความถูกต้องแม่นยำและเหมาะสมกับการใช้งานในสภาพแวดล้อมที่มีการรบกวนสูงอย่างระบบตรวจจับเลน

5.3 ข้อเสนอแนะ

ควรนำแบบจำลองที่สร้างขึ้นไปใช้งานร่วมกับระบบตรวจจับเลน เพื่อช่วยลดสัญญาณรบกวนที่เกิดขึ้นกับระบบเพื่อพัฒนาให้ระบบการขนส่งและจราจรอัจฉริยะให้มีประสิทธิภาพมากยิ่งขึ้น

บรรณานุกรม

บรรณานุกรม

- ประมวล ตียะพานิชย์. 2553. “การศึกษาทฤษฎีและหลักการทำงานของ Kalman Filter ร่วมกับระบบการจราจรและขนส่งอัจฉริยะ” วิทยานิพนธ์ปริญญาบัณฑิตสาขาวิชาวิศวกรรมศาสตร์คอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยรังสิต.
- ปัทม์ วัลยะเพ็ชร. 2553. “Lane detection and tracking for driving aid system” (ออนไลน์) เข้าถึงเมื่อ 2 กุมภาพันธ์ 2556 จาก <http://fivedots.coe.psu.ac.th/~kom/?p=1322>
- อรธนันท์ เล็กอุทัย. 2535. “การประยุกต์ใช้เทคนิคคาลแมนฟิลเตอร์ในการพยากรณ์ปริมาณน้ำหลากเข้าเขื่อนอุบลรัตน์” วิทยานิพนธ์ปริญญาโทบัณฑิตสาขาวิชาวิศวกรรมศาสตร์(วิศวกรรมโยธา) คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย.
- A. Borkar, M. Hayes, and M. Smith. 2012. “A novel lane detection system with efficient ground truth generation,” *Intelligent Transportation Systems, IEEE Transactions*, 13, 1: 365–374.
- A. Wright. 2011. “Automotive autonomy,” *Commun. ACM*, 54, 7:16–18.
- B. Ma, S. Lakshmanan, and A. O. Hero. 2000. “Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion,” *IEEE Transactions on Intelligent Transportation Systems*.
- B. Ristic, S. Arulampalam, and N. Gordon. 2004. “Beyond the Kalman Filter: Particle Filters for Tracking Applications” (Artech House Radar Library). Artech Print on Demand.
- C. Yu, and C. Su. 2012. “An improved lane detection algorithm and the definition of the error rate standard”, *Proc. SPIE 8334, Fourth International Conference on Digital Image Processing*.
- G. Welch and G. Bishop. 1997. *An Introduction to the Kalman Filter*. Department of Computer Science University of North Carolina at Chapel Hill Chapel Hill: 3-312.
- J. C. McCall and M. M. Trivedi. 2006. “Video based lane estimation and tracking for driver assistance: Survey, system, and evaluation”
- J. Goldbeck, B. Huertgen, S. Ernst, and L. Kelch. 2000. “Lane following combining vision and dgps,” *Image Vision Comput.*, 18, 5: 425–433.

- J. K. Patel and C. B. Read. 1996. "Handbook of the Normal Distribution, Second Edition (Statistics: A Series of Textbooks and Monographs)", CRC Press, USA: 120-131.
- K. H. Lim, K. P. Seng, and L.-M. Ang. 2012. "River flow lane detection and kalman filtering-based b-spline lane tracking,"*International Journal of Vehicular Technology*.
- P. K. Wang, P. A. Torrion, L. M. Collins, and Kenneth D. Morton. 2012. "Rapid position estimation and tracking for autonomous driving", *Proc. SPIE 8387, Unmanned System Technology*.
- Pindyck, Robert S. 1998. "Irreversibilities and timing of environmental policy," Working papers WP, Massachusetts Institute of Technology(MIT), Sloan School of Management: 47-98.
- Kalman, R.E. 1960. "A new approach to linear filtering and prediction problems", *Transaction of the ASME, Journal of Basic Engineering* 82: 35 – 45.
- Kailath, T. 1969. "A general likelihood-ratio formula for random signals in Gaussian noise," *Information Theory, IEEE Transactions* 15, 3: 350- 361.
- R. Bishop. 2000. "Intelligent vehicle applications worldwide," *IEEE Intelligent Systems* 15, 1: 78–81.
- R. Jurgen and S. of Automotive Engineers. 2007. "A Simple CCD Based Lane Tracking System", PT (Series) (Warrendale, Pa.). Society of Automotive Engineers, 2.
- S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niek-erk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney. 2006. "Winning the darpa grand challenge," *Journal of Field Robotics*, accepted for publication.
- T. Chiku and J. Miura. 2012. "On-line road boundary estimation by switching multiple road models using visual features from a stereo camera." in *IROS. IEEE*: 4939–4944.

U. Handmann, I. Leefken, C. Tzomakas, W. v.Seelen, and L. F. T. Biologie. 1999. “A flexible architecture for driver assistance,” In Proceedings of SPIE.

W. M. centre. 2012. “Road traffic injuries” from
<http://www.who.int/mediacentre/factsheets/fs358/en/index.html>.

Y.-C. Leng and C.-L. Chen. 2010. “Vision-based lane departure detection system in urban traffic scenes.” in ICARCV. IEEE: 1875–1880.

_____, Subject M163. 2008. “Kalman Filter Applications” 1st July 2014, from
www.cs.unc.edu/~welch/kalman/media/pdf/kftool_models.pdf

ภาคผนวก

ภาคผนวก ก

โค้ด Matlab

```

function [X,mu,Sigma,rmse_obs,rmse_est] = test_model_NaN()

X = gen_sim_data();
Z = get_obs(X);

[mu{1}, Sigma{1}, Sigma_x] = initial_distribution(Z);

for T = 2:size(Z,2)
    thisz = Z{T};
    if isnan(thisz(1)) | isnan(thisz(2)) | isnan(thisz(3)) | isnan(thisz(4))
        %prediction state for NaN values
        muprev = mu{T-1};
        [pred_x,Jf] = sysf(muprev);
        [pred_z,H,Sigma_z] = sysh(pred_x);
        pred_sig = Jf*Sigma{T-1}*Jf' + Sigma_x;
        %residual = Z{T}-pred_z;
        %e(T) = norm(residual);
        %K = pred_sig*H'*inv(H*pred_sig*H'+Sigma_z);
        %Sigma{T} = (eye(size(K,1)) - K*H)*pred_sig;
        Sigma{T} = pred_sig;
        munew = pred_x;
        mu{T} = munew;
    else
        muprev = mu{T-1};
        [pred_x,Jf] = sysf(muprev);
        [pred_z,H,Sigma_z] = sysh(pred_x);
        pred_sig = Jf*Sigma{T-1}*Jf' + Sigma_x;
    end
end

```

```

    residual = Z{T}-pred_z;
    e(T) = norm(residual);
    K = pred_sig*H'*inv(H*pred_sig*H'+Sigma_z);
    Sigma{T} = (eye(size(K,1)) - K*H)*pred_sig;
    munew = pred_x + K*residual;
    mu{T} = munew;
end
end;

%Root mean square error calculation
rmse_obs = cal_rms(X,Z);
rmse_est = cal_rms(X,mu);

figure(2);
plot_sim_data( X, Z, mu);
%plot_sim_data(X);

creatdatfile(X,mu,Z);
%    creatdatfile1(rmse_obs,rmse_est);

```

```

function creatdatfile(X,mu,Z)

    myfile = 'simdatas_NaN25.dat';
    fid = fopen(myfile,'w');

%   for i = 1:size(X,2)
%       x = X{i};
%       z = Z{i};
%       m = mu{i};
%
%       fprintf(fid, '%d %d %d %d %d %d %d %d %d %d %d %d %d \n', i,
x(1),x(2),x(3),x(4),z(1),z(2),z(3),z(4), m(1),m(2),m(3),m(4));
%   end

%range of data
% 180-240 => straight lane boundary
% 270-300 => curve lane boundary

for i = 180:240
    x = X{i};
    z = Z{i};
    m = mu{i};

    fprintf(fid, '%d %d %d %d %d %d %d %d %d %d %d %d %d \n', i,
x(1),x(2),x(3),x(4),z(1),z(2),z(3),z(4),m(1),m(2),m(3),m(4));

end

fclose(fid);

```

```
function creatdatfile1(rmse_obs,rmse_est)
```

```
    myfile = 'simdata_rmse.dat';
```

```
    fid = fopen(myfile,'a');
```

```
    x = rmse_obs;
```

```
    fprintf(fid, '%d %d %d %d \n', x(1),x(2),x(3),x(4));
```

```
    x = rmse_est;
```

```
    fprintf(fid, '%d %d %d %d \n', x(1),x(2),x(3),x(4));
```

```
    fclose(fid);
```

```
function [Z,H,Sigma_z] = sysh(X)
```

```
    H = [ eye(6), zeros(6,2) ];
```

```
    Z = H * X;
```

```
    Sigma_z = diag( [ 0.1 0.1 10/180*pi 10/180*pi 0.01 0.01].^2 );
```

```
function [xnew,J] = sysf(xold)
```

```
    dl = xold(1);
```

```
    dr = xold(2);
```

```
    thetal = xold(3);
```

```
    thetar = xold(4);
```

```
    kappal = xold(5);
```

```
    kappar = xold(6);
```

```
v = xold(7);
```

```
av = xold(8);
```

```
delta_t = 1/30;
```

```
xnew = predict_X(dl,dr,thetal,thetar,kappal,kappar,v,av,delta_t);
```

```
% Jacobian
```

```
J = Jacob(xold)
```

```
function J = Jacob(xold)
```

```
X = xold;
```

```
for i=1:size(X,1)
```

```
    for j=1:size(X,1)
```

```
        old = beforeadding_h(X,i);
```

```
        newX = adding_h(X,j);
```

```
        new = afteradding_h(newX,i);
```

```
        J(i,j) = finite(old,new);
```

```
    end
```

```
end
```

```
function new = afteradding_h(X,i)
```

```
    dl = X(1);
```

```
    dr = X(2);
```

```
    thetal = X(3);
```

```
    thetar = X(4);
```

```
    kappal = X(5);
```

```
    kappar = X(6);
```

```

v = X(7);
av = X(8);

deltat = 1/30;

sl = pre_arc_length(dl,thetal,kappal,v,av,deltat);
sr = pre_arc_length(dr,thetar,kappar,v,av,deltat);

if(i==1)
    new = cal_dl(dl,thetal,kappal,v,av,deltat);
elseif(i==2)
    new = cal_dr(dr,thetar,kappar,v,av,deltat);
elseif(i==3)
    new = thetal+(kappal*sl)+(av*deltat);
elseif(i==4)
    new = thetar-(kappar*sr)-(av*deltat);
elseif(i==5)
    new = kappal;
elseif(i==6)
    new = kappar;
elseif(i==7)
    new = v;
else
    new = av;
end

```

```

function new = cal_dl(dl,theta_l,kappa_l,v,av,delta_t)

    sl = pre_arc_length(dl,theta_l,kappa_l,v,av,delta_t);

    new = pre_leteral_distance(sl,dl,theta_l,kappa_l,v,av,delta_t);

```

```

function new = cal_dr(dr,theta_r,kappa_r,v,av,delta_t)

    sr = pre_arc_length(dr,theta_r,kappa_r,v,av,delta_t);

    new = pre_leteral_distance(sr,dr,theta_r,kappa_r,v,av,delta_t);

```

```

function f = finite(old,new)

    h = 1.0e-5;

    f = (new-old)/h;

```

```

function X = adding_h(X,j)

    h = 1.0e-5;

    if(j==1)

        X(1)= X(1)+h;

    elseif(j==2)

        X(2)= X(2)+h;

    elseif(j==3)

        X(3)= X(3)+h;

    elseif(j==4)

        X(4)= X(4)+h;

    elseif(j==5)

        X(5)= X(5)+h;

    elseif(j==6)

        X(6)= X(6)+h;

    elseif(j==7)

```

X(7)= X(7)+h;

else

X(8)= X(8)+h;

end

function old = beforeadding_h(X,i)

dl = X(1);

dr = X(2);

thetal = X(3);

thetar = X(4);

kappal = X(5);

kappar = X(6);

v = X(7);

av = X(8);

deltat = 1/30;

sl = pre_arc_length(dl,thetal,kappal,v,av,deltat);

sr = pre_arc_length(dr,thetar,kappar,v,av,deltat);

if(i==1)

old = cal_dl(dl,thetal,kappal,v,av,deltat);

elseif(i==2)

old = cal_dr(dr,thetar,kappar,v,av,deltat);

elseif(i==3)

old = thetal+(kappal*sl)+(av*deltat);

elseif(i==4)

old = thetar-(kappar*sr)-(av*deltat);

```

elseif(i==5)
    old = kappal;
elseif(i==6)
    old = kappar;
elseif(i==7)
    old = v;
else
    old = av;
end

```

```

function newX = predict_X(dl,dr,theta_l,theta_r,kappa_l,kappa_r,v,av,delta_t)

```

```

    sl = pre_arc_length(dl,theta_l,kappa_l,v,av,delta_t);
    sr = pre_arc_length(dr,theta_r,kappa_r,v,av,delta_t);

    new_dl = pre_lateral_distance(sl,dl,theta_l,kappa_l,v,av,delta_t);
    new_dr = pre_lateral_distance(sr,dr,theta_r,kappa_r,v,av,delta_t);

    [new_theta_l,new_theta_r] =
pre_tangent_angle(theta_l,theta_r,kappa_l,kappa_r,av,delta_t,sl,sr);

    newX = [new_dl new_dr new_theta_l new_theta_r kappa_l kappa_r v av]';

```

```

function s = pre_arc_length(ld,theta,kappa,v,av,delta_t)

if(kappa <= abs(0.001) && av <= abs(0.001))

    lamda = sqrt((ld^2+(v*delta_t)^2)-(2*ld*(v*delta_t)*cos(theta)));

    theta_1 = acos((ld^2+lamda^2-(v*delta_t)^2)/(2*ld*lamda));

```

```

s = lamda*sin(theta_1);

%disp('Straight lane boundary with straight movement');

elseif(kappa <= abs(0.001) && av > abs(0.001))

    lamda = sqrt((ld^2+ (2*v/av*sin(av*delta_t/2))^2)-
(2*ld*(2*v/av*sin(av*delta_t/2))*cos(theta)));

    theta_1 = acos((ld^2+lamda^2-(2*v/av*sin(av*delta_t/2))^2)/((2*ld)*lamda));

    s = lamda*sin(theta_1);

    %disp('Straight lane boundary with circular movement');

elseif(kappa > abs(0.001) && av <= abs(0.001))

    lamda = sqrt((((1/kappa)+ld)^2 + (v*delta_t)^2) -
(2*((1/kappa)+ld)*(v*delta_t)*cos(theta)));

    theta_1 = acos((((1/kappa)+ld)^2 + lamda^2 - (v*delta_t)^2)/(2*((1/kappa)+ld)*lamda));

    s = theta_1/kappa;

else

    lamda = sqrt((((1/kappa)+ld)^2 + (2*v/av*sin(av*delta_t/2))^2) -
(2*((1/kappa)+ld)*(2*v/av*sin(av*delta_t/2)*cos(theta))));

    theta_1 = acos((((1/kappa)+ld)^2 + lamda^2 -
(2*v/av*sin(av*delta_t/2))^2)/(2*((1/kappa)+ld)*lamda));

    s = theta_1/kappa;

end

```

```

function ld = pre_leteral_distance(s,ld,theta,kappa,v,av,delta_t)

```

```

    if(kappa <= abs(0.001) && av <= abs(0.001))

        theta1 = pi/2;

        theta2 = abs((theta) - (pi/2));

        theta3 = pi-theta2;
    
```

```

theta4 = pi - theta1 - theta3;
theta5 = theta3 + theta4;

beta1 = s;
beta2 = sqrt((beta1^2 + ld^2 -(2*beta1*ld*cos(theta5))));

theta6 = acos(((beta1^2+beta2^2)-ld^2)/(2*beta1*beta2));
theta7 = pi-theta5-theta6;
theta8 = (theta)-theta7;
ld = sqrt((beta2^2 + (v*delta_t)^2) - (2*beta2*(v*delta_t)*cos(theta8)));

elseif(kappa <= abs(0.001) && av > abs(0.001))
    theta1 = pi/2;
    theta2 = abs((theta) - (pi/2));
    theta3 = pi-theta2;
    theta4 = pi - theta1 - theta3;
    theta5 = theta3 + theta4;

    beta1 = s;
    beta2 = sqrt((beta1^2 + ld^2 -(2*beta1*ld*cos(theta5))));

    theta6 = acos(((beta1^2+beta2^2)-ld^2)/(2*beta1*beta2));
    theta7 = pi-theta5-theta6;
    theta8 = (theta)-theta7;

    ld = sqrt(beta2^2 + (2*v/av*sin(av*delta_t/2))^2 -
(2*beta2*((2*v/av*sin(av*delta_t/2))))*cos(theta8));

```

```

elseif(kappa > abs(0.001) && av <= abs(0.001))

    theta1 = (pi-(kappa*s))/2;
    theta2 = abs((theta) - (pi/2));
    theta3 = pi-theta2;
    theta4 = pi - theta1 - theta3;
    theta5 = theta3 + theta4;

    beta1 = 2*(1/kappa)*sin((kappa*s)/2);
    beta2 = sqrt((beta1^2 + ld^2 -(2*beta1*ld*cos(theta5))));

    theta6 = acos(((beta1^2+beta2^2)-ld^2)/(2*beta1*beta2));
    theta7 = pi-theta5-theta6;
    theta8 = (theta)-theta7;
    ld = sqrt(beta2^2 + (v*delta_t)^2 - (2*beta2*(v*delta_t)*cos(theta8)));

else

    theta1 = (pi-(kappa*s))/2;
    theta2 = abs((theta) - (pi/2));
    theta3 = pi-theta2;
    theta4 = pi - theta1 - theta3;
    theta5 = theta3 + theta4;

    beta1 = 2*(1/kappa)*sin((kappa*s)/2);
    beta2 = sqrt((beta1^2 + ld^2 -(2*beta1*ld*cos(theta5))));

    theta6 = acos(((beta1^2+beta2^2)-ld^2)/(2*beta1*beta2));
    theta7 = pi-theta5-theta6;

```

```

    theta8 = (theta)-theta7;

    ld = sqrt(beta2^2 + (2*v/av*sin(av*delta_t/2))^2 -
(2*beta2*(2*v/av*sin(av*delta_t/2))*cos(theta8)));

end

function [new_theta_l new_theta_r] =
pre_tangent_angle(theta_l,theta_r,kappa_l,kappa_r,av,delta_t,sl,sr)

    new_theta_l = theta_l+(kappa_l*sl)+(av*delta_t);
    new_theta_r = theta_r-(kappa_r*sr)-(av*delta_t);

function [mu, Sigma, Sigma_x] = initial_distribution(Z)

    thisz = Z{1};

    if isnan(thisz(1)) | isnan(thisz(2)) | isnan(thisz(3)) | isnan(thisz(4))

        mu = [1.75 1.75 pi/2 pi/2 0 0 11.11 0.000]';

    else

        mu = [ thisz(1) thisz(2) thisz(3) thisz(4) thisz(5) thisz(6) 5 0]';

    end

Sigma = diag( [0.1 0.1 10/180*pi 10/180*pi 0.01 0.01 0.01 0.01].^2 );

deltat = 1/30;

Sigma_x = diag( (deltat * [0.2 0.2 10/180*pi 10/180*pi 0.05 0.05 0.5 0.5]).^2);

function X = gen_sim_data()

%X{1} = [1.75 1.75 pi/2 pi/2 0 0 11.11 0.000]';

X{1} = [1.75 1.75 90.75*pi/180 89.25*pi/180 0 0 16.66 0]';

```

```

t=1;

for i=1:600

    t = t+1;

    prevX = X{t-1};

    dl = prevX(1);

    dr = prevX(2);

    thetal = prevX(3);

    thetar = prevX(4);

    kappal = prevX(5);

    kappar = prevX(6);

    v = prevX(7);

    av = prevX(8);


    deltat = 1/30;


    if(i <= 60)

        kappal = 0;

        kappar = 0;

        av = 0;

        compass = 0; % compass = 1 => turn right , 2 => turn left

        newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);

    elseif(i > 60 && i <= 100)

        kappal = 0.00;

        kappar = 0.00;

        av = 0.010;

        compass = 2;

```

```

    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 100 && i <= 210)

    kappal = 0.00;

    kappar = 0.00;

    av = 0.0;

    compass = 0;

    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 210 && i <= 240)

    kappal = 0.00;

    kappar = 0.00;

    av = 0.022;

    compass = 2;

    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 240 && i <= 270)

    kappal = 0.00;

    kappar = 0.00;

    av = 0.0;

    compass = 0;

    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 270 && i <= 285)

    kappal = 0.005;

    kappar = 0.005;

    av = 0.0;

    compass = 1;

    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 285 && i <= 300)

    kappal = 0.005;

```

```

    kappal = 0.005;
    av = 0.044;
    compass = 1;
    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 300 && i <= 315)
    kappal = 0.005;
    kappar = 0.005;
    av = 0.011;
    compass = 1;
    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 315 && i <= 360)
    kappal = 0.005;
    kappar = 0.005;
    av = 0.011;
    compass = 2;
    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 360 && i <= 450)
    kappal = 0.005;
    kappar = 0.005;
    av = 0.022;
    compass = 2;
    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 360 && i <= 450)
    kappal = 0.005;
    kappar = 0.005;
    av = 0.044;
    compass = 1;

```

```

        newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 450 && i <= 480)

    kappal = 0.01;

    kappar = 0.01;

    av = 0.088;

    compass = 1;

    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 480 && i <= 500)

    kappal = 0.01;

    kappar = 0.01;

    av = 0.088;

    compass = 2;

    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
elseif(i > 500 && i <= 600)

    kappal = 0.0;

    kappar = 0.0;

    av = 0;

    compass = 0;

    newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
end

%
%     X{t} = newX;
% end
%
%     if(i <= 30)
%         kappal = 0;
%         kappar = 0;

```

```

%      av = 0;
%      compass = 0; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 30 & i <= 45)
%      thetal = 1.553;
%      thetar = 1.588;
%      kappal = 0;
%      kappar = 0;
%      av = 0;
%      compass = 0; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 45 & i <= 60)
%      thetal = pi/2;
%      thetar = pi/2;
%      kappal = 0;
%      kappar = 0;
%      av = 0;
%      compass = 0; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 60 & i <= 90)
%      thetal = 1.588;
%      thetar = 1.553;
%      kappal = 0;
%      kappar = 0;
%      av = 0;
%      compass = 0; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);

```

```

% elseif(i > 90 & i <= 120)
%     av = 0.003;
%     compass = 1; % compass = 1 => turn right , 2 => turn left
%     newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
% elseif(i > 120 & i <= 150)
%     av = 0.006;
%     compass = 1; % compass = 1 => turn right , 2 => turn left
%     newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
% elseif(i > 150 & i <= 180)
%     av = 0.01;
%     compass = 2; % compass = 1 => turn right , 2 => turn left
%     newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
% elseif(i > 180 & i <= 210)
%     av = 0.05;
%     compass = 2; % compass = 1 => turn right , 2 => turn left
%     newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
% elseif(i > 210 & i <= 240)
%     av = 0.07;
%     compass = 2; % compass = 1 => turn right , 2 => turn left
%     newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
% elseif(i > 240 & i <= 270)
%     av = 0.1;
%     compass = 1; % compass = 1 => turn right , 2 => turn left
%     newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
% elseif(i > 270 & i <= 300)
%     av = 0.1;
%     compass = 1; % compass = 1 => turn right , 2 => turn left

```

```

%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 300 & i <= 330)
%      av = 0.1;
%      compass = 2; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 330 & i <= 360)
%      av = 0.1;
%      compass = 2; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 360 & i <= 390)
%      av = 0.15;
%      compass = 1; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 390 & i <= 410)
%      av = 0.15;
%      compass = 1; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 410 & i <= 420)
%      av = 0.15;
%      compass = 2; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 420 & i <= 450)
%      av = 0.15;
%      compass = 2; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%  elseif(i > 450 & i <= 540)
%      av = 0;

```

```

%      compass = 0; % compass = 1 => turn right , 2 => turn left
%      newX = sim_model(dl,dr,thetal,thetar,kappal,kappar,v,av,deltat,compass);
%      end

X{t} = newX;

end;

function newX = sim_model(dl,dr,theta_l,theta_r,kappa_l,kappa_r,v,av,delta_t,compass)

sl = cal_arc_length(dl,theta_l,kappa_l,v,av,delta_t);
sr = cal_arc_length(dr,theta_r,kappa_r,v,av,delta_t);

new_dl = cal_leteral_distance(sl,dl,theta_l,kappa_l,v,av,delta_t);
new_dr = cal_leteral_distance(sr,dr,theta_r,kappa_r,v,av,delta_t);

[new_theta_l,new_theta_r] =
cal_tangent_angle(theta_l,theta_r,kappa_l,kappa_r,av,delta_t,sl,sr,compass);

newX = [new_dl new_dr new_theta_l new_theta_r kappa_l kappa_r v av]';

function s = cal_arc_length(ld,theta,kappa,v,av,delta_t)
if(kappa <= abs(0.001) && av <= abs(0.001))
    lamda = sqrt((ld^2+(v*delta_t)^2)-(2*ld*(v*delta_t)*cos(theta)));
    theta_1 = acos((ld^2+lamda^2-(v*delta_t)^2)/(2*ld*lamda));
    s = lamda*sin(theta_1);
elseif(kappa <= abs(0.001) && av > abs(0.001))

```

```

        lamda = sqrt((ld^2+ (2*v/av*sin(av*delta_t/2))^2)-
(2*ld*(2*v/av*sin(av*delta_t/2))*cos(theta)));

        theta_1 = acos((ld^2+lamda^2-(2*v/av*sin(av*delta_t/2))^2)/((2*ld)*lamda));

        s = lamda*sin(theta_1);

elseif(kappa > abs(0.001) && av <= abs(0.001))

        lamda = sqrt((((1/kappa)+ld)^2 + (v*delta_t)^2) -
(2*((1/kappa)+ld)*(v*delta_t)*cos(theta)));

        theta_1 = acos((((1/kappa)+ld)^2 + lamda^2 - (v*delta_t)^2)/(2*((1/kappa)+ld)*lamda));

        s = theta_1/kappa;

else

        lamda = sqrt((((1/kappa)+ld)^2 + (2*v/av*sin(av*delta_t/2))^2) -
(2*((1/kappa)+ld)*(2*v/av*sin(av*delta_t/2)*cos(theta))));

        theta_1 = acos((((1/kappa)+ld)^2 + lamda^2 -
(2*v/av*sin(av*delta_t/2))^2)/(2*((1/kappa)+ld)*lamda));

        s = theta_1/kappa;

end

```

```

function ld = cal_leteral_distance(s,ld,theta,kappa,v,av,delta_t)

```

```

    if(kappa <= abs(0.001) && av <= 0.001)

        theta1 = pi/2;

        theta2 = abs((theta) - (pi/2));

        theta3 = pi-theta2;

        theta4 = pi - theta1 - theta3;

        theta5 = theta3 + theta4;

        beta1 = s;

```

```

    beta2 = sqrt((beta1^2 + ld^2 -(2*beta1*ld*cos(theta5))));

    theta6 = acos(((beta1^2+beta2^2)-ld^2)/(2*beta1*beta2));
    theta7 = pi-theta5-theta6;
    theta8 = (theta)-theta7;
    ld = sqrt((beta2^2 + (v*delta_t)^2) - (2*beta2*(v*delta_t)*cos(theta8)));
    0.5+0.5
elseif(kappa <= abs(0.001) && av > 0.001)
    theta1 = pi/2;
    theta2 = abs((theta) - (pi/2));
    theta3 = pi-theta2;
    theta4 = pi - theta1 - theta3;
    theta5 = theta3 + theta4;

    beta1 = s;
    beta2 = sqrt((beta1^2 + ld^2 -(2*beta1*ld*cos(theta5))));

    theta6 = acos(((beta1^2+beta2^2)-ld^2)/(2*beta1*beta2));
    theta7 = pi-theta5-theta6;
    theta8 = (theta)-theta7;
    ld = sqrt(beta2^2 + (2*v/av*sin(av*delta_t/2))^2 -
(2*beta2*((2*v/av*sin(av*delta_t/2)))*cos(theta8)));
    1+1
elseif(kappa > abs(0.001) && av <= 0.001)
    theta1 = (pi-(kappa*s))/2;
    theta2 = abs((theta) - (pi/2));
    theta3 = pi-theta2;

```

theta4 = pi - theta1 - theta3;

theta5 = theta3 + theta4;

beta1 = 2*(1/kappa)*sin((kappa*s)/2);

beta2 = sqrt((beta1^2 + ld^2 -(2*beta1*ld*cos(theta5))));

theta6 = acos(((beta1^2+beta2^2)-ld^2)/(2*beta1*beta2));

theta7 = pi-theta5-theta6;

theta8 = (theta)-theta7;

ld = sqrt(beta2^2 + (v*delta_t)^2 - (2*beta2*(v*delta_t)*cos(theta8)));

1.5+1.5

else

theta1 = (pi-(kappa*s))/2;

theta2 = abs((theta) - (pi/2));

theta3 = pi-theta2;

theta4 = pi - theta1 - theta3;

theta5 = theta3 + theta4;

beta1 = 2*(1/kappa)*sin((kappa*s)/2);

beta2 = sqrt((beta1^2 + ld^2 -(2*beta1*ld*cos(theta5))));

theta6 = acos(((beta1^2+beta2^2)-ld^2)/(2*beta1*beta2));

theta7 = pi-theta5-theta6;

theta8 = (theta)-theta7;

**ld = sqrt(beta2^2 + (2*v/av*sin(av*delta_t/2))^2 -
(2*beta2*(2*v/av*sin(av*delta_t/2))*cos(theta8)));**

2+2

end

```
function [new_theta_l new_theta_r] =  
cal_tangent_angle(theta_l,theta_r,kappa_l,kappa_r,av,delta_t,sl,sr,compass)  
  
if(kappa_l <= abs(0.001) && kappa_r <= abs(0.001) && av <= abs(0.001) && compass == 0)  
    new_theta_l = theta_l;  
    new_theta_r = theta_r;  
elseif(kappa_l > abs(0.001) && kappa_r > abs(0.001) && av <= abs(0.001) && compass ==  
1)  
    new_theta_l = theta_l+(kappa_l*sl);  
    new_theta_r = theta_r-(kappa_r*sr);  
elseif(kappa_l > abs(0.001) && kappa_r > abs(0.001) && av <= abs(0.001) && compass ==  
2)  
    new_theta_l = theta_l-(kappa_l*sl);  
    new_theta_r = theta_r+(kappa_r*sr);  
elseif(av > 0.001 && compass ==1)  
    new_theta_l = theta_l+(av*delta_t);  
    new_theta_r = theta_r-(av*delta_t);  
elseif(av > 0.001 && compass == 2)  
    new_theta_l = theta_l-(av*delta_t);  
    new_theta_r = theta_r+(av*delta_t);  
end
```

```
function Z = get_obs(X)
```

```
%    z_dl = [];
```

```
%    z_dr = [];
```

```

%   z_thetal = [];
%   z_thetar = [];
    for i = 1:size(X,2)
        xthis = X{i};
        obs_dl = xthis(1)+0.1*randn(1);
%       z_dl(end+1) = obs_dl;
        obs_dr = xthis(2)+0.1*randn(1);
%       z_dr(end+1) = obs_dr;
        obs_thetal = xthis(3)+0.01*randn(1);
        obs_thetar = xthis(3)+0.1745*randn(1);
%       z_thetal(end+1) = obs_thetal;
        obs_thetar = xthis(4)+0.1745*randn(1);
%       z_thetar(end+1) = obs_thetar;
        obs_kappar = xthis(5)+0.01*randn(1);
        obs_kappal = xthis(6)+0.01*randn(1);

        Z{i} = [obs_dl obs_dr obs_thetal obs_thetar obs_kappar obs_kappal]';
    end;
%   std(z_dl)
%   std(z_dr)
%   std(z_thetal)
%   std(z_thetar)

%   % Random NaN for dl
%   n_index = rand_NaN(0.3,Z);
%   Z = set_NaN(n_index,Z,1);
%   % Random NaN for dr

```

```

%   n_index = rand_NaN(0.5,Z);
%   Z = set_NaN(n_index,Z,2);
%   % Random NaN for thetal
%   n_index = rand_NaN(0.5,Z);
%   Z = set_NaN(n_index,Z,3);
%   % Random NaN for thetar
%   n_index = rand_NaN(0.3,Z);
%   Z = set_NaN(n_index,Z,4);

```

```

% Random NaN for dl
n_index = rand_NaN(0.3,Z);
Z = set_NaN(n_index,Z,1);
% Random NaN for dr
n_index = rand_NaN(0.3,Z);
Z = set_NaN(n_index,Z,2);
% Random NaN for thetal
n_index = rand_NaN(0.3,Z);
Z = set_NaN(n_index,Z,3);
% Random NaN for thetar
n_index = rand_NaN(0.3,Z);
Z = set_NaN(n_index,Z,4);

```

```

function new_Z = set_NaN(n_index,Z,z)

for i = 1:size(n_index,2)

    index = n_index(i);

    thisz = Z{index};

```

```

    thisz(z) = NaN;

    Z{index} = thisz;

end

new_Z = Z;

```

```

function index = rand_NaN(time,Z)

```

```

    %time => finding inveral for ramdom missing value if time = 1 means

```

```

    %ramdom range = 30

```

```

    time = time * 30;

```

```

    numberofIteration = fix(size(Z,2)/time);

```

```

    numindexforrand = [];

```

```

    index = [];

```

```

    r = [];

```

```

    n=1;

```

```

    m=time;

```

```

    for i=1:numberofIteration

```

```

        for j=n:m

```

```

            numindexforrand(end+1) = j;

```

```

        end

```

```

    % r = number of missing values number

```

```

    r = round(rand(1,2)*time);

```

```

    while(r(1)== 0 | r(2) == 0 | r(1)> size(Z,2) |r(2)> size(Z,2) )

```

```

        r = round(rand(1,2)*time);

```

```

        index1 = r(1); % missing index number 1

```

```

        index2 = r(2); % missing index number 2 => there are 2 missing values in very 30 frames

```

```

    end

```

```

    index1 = r(1);

```

```

    index2 = r(2);

    index(end+1) = numindexforrand(index1);
    index(end+1) = numindexforrand(index2);
    numindexforrand = [];

    n=n+time;
    m=m+time;
end

```

```

function plot_sim_data(X, Z, Xest)

```

```

%ploting of dl

```

```

subplot(2,2,1);

```

```

%figure(1)

```

```

plotx = plot_array(X,1);

```

```

plot(plotx,'-.*r');

```

```

hold on;

```

```

plotz = plot_array(Z,1);

```

```

plot(plotz,'-.og');

```

```

plote = plot_array(Xest,1);

```

```

plot(plote,'-.+b');

```

```

title('left lateral distance');

```

```

%ploting of dr

```

```

subplot(2,2,2);

```

```

%figure(2)

```

```

plotx = plot_array(X,2);

```

```
plot(plotx,'-.*r');  
hold on;  
plotz = plot_array(Z,2);  
plot(plotz,'-.*og');  
plote = plot_array(Xest,2);  
plot(plote,'-.*b');  
  
title('right lateral distance');
```

```
%ploting of theta_l  
subplot(2,2,3);  
%figure(3)  
plotx = plot_array(X,3);  
plot(plotx,'-.*r');  
hold on;  
plotz = plot_array(Z,3);  
plot(plotz,'-.*og');  
plotz = plot_array(Xest,3);  
plot(plotz,'-.*b');  
title('right tangent angle')
```

```
%ploting of theta_r  
subplot(2,2,4);  
% figure(4)  
plotx = plot_array(X,8);  
plot(plotx,'-.*r');  
hold on;
```

```

% plotz = plot_array(Z,4);
% plot(plotz,'-.og');
plotz = plot_array(Xest,8);
plot(plotz,'-.+b');
title('Angular Velocity')

```

```

function [rmse] = cal_rms(Xold,Xnew)

```

```

    err_dl = [];
    err_dr = [];
    err_thetal = [];
    err_thetar = [];

```

```

for i = 1:size(Xold,2)
    thisx = Xold{i};
    dl_x = thisx(1);
    dr_x = thisx(2);
    thetal_x = thisx(3);
    thetar_x = thisx(4);
    thisn = Xnew{i};
    dl_n = thisn(1);
    dr_n = thisn(2);
    thetal_n = thisn(3);
    thetar_n = thisn(4);

```

```

    if ~isnan(dl_n)
        err_dl(end+1) = dl_x - dl_n;
    end

```

```

if ~isnan(dr_n)
    err_dr(end+1) = dr_x - dr_n;
end
if ~isnan(thetal_n)
    err_thetal(end+1) = thetal_x - thetal_n;
end
if ~isnan(thetar_n)
    err_thetar(end+1) = thetar_x - thetar_n;
end
mse_dl = mean(err_dl.^2);
mse_dr = mean(err_dr.^2);
mse_thetal = mean(err_thetal.^2);
mse_thetar = mean(err_thetar.^2);

rmse_dl = sqrt(mse_dl);
rmse_dr = sqrt(mse_dr);
rmse_thetal = sqrt(mse_thetal);
rmse_thetar = sqrt(mse_thetar);
rmse = [rmse_dl rmse_dr rmse_thetal rmse_thetar];
end

```

```

function plotx = plot_array(X,x)
for i=1:size(X,2)
    thisx = X{i};
    plotx(i) = thisx(x);
end

```

ประวัติย่อผู้วิจัย

ชื่อ	นางสาวชฎาพร เกตุมณี
วัน เดือน ปี เกิด	วันที่ 29 มิถุนายน 2523
สถานที่เกิด	อำเภอเมือง จังหวัดชุมพร
สถานที่อยู่ปัจจุบัน	หอพัก ST7 Ro.S28 สถาบันเอเชีย ถนนพหลโยธิน คลองหลวง ปทุมธานี 12120
ตำแหน่งหน้าที่การงานปัจจุบัน	อาจารย์ประจำ สาขาเทคโนโลยีสารสนเทศและการสื่อสาร
สถานที่ทำงาน	คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม
ประวัติการศึกษา	พ.ศ. 2545 คอ.บ. จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณ- ทหารลาดกระบัง พ.ศ. 2554 วศ.ม. จากสถาบันเอเชีย