อัลกรอลิทึมตรวจจับช่องทางเดินรถสำหรับยานยนต์อัตโนมัติ

มหาวิทยาลัยนเรศวร

LANE DETECTION FOR NU AUTONOMOUS VEHICLE

DRIVING CONTROL

นายกฤษณ์	สอนสำแดง	รหัสนิสิต	57361838
นายธนบัตร	ว่องวัฒนกูล	รหัสนิสิต	57362118
นายเรืองวิทย์	จันทะโก	รหัสนิสิต	57362408

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมเครื่องกล ภาควิชาวิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ใบรับรองโครงงานวิศวกรรมเครื่องกล

หัวข้อโครงงาน : อัลกอริทึมตรวจจับช่องทางเดินรถสำหรับยานยนต์อัตโนมัติ มหาวิทยาลัยนเรศวร

ผู้ดำเนินโครงงาน : นายกฤษณ์ สอนสำแดง รหัส 57361838
นายธนบัตร ว่องวัฒนกูล รหัส 57362118
นายเรืองวิทย์ จันทะโก รหัส 57362408

อาจารย์ที่ปรึกษาโครงงาน : ผู้ช่วยศาสตราจารย์ชูพงศ์ ช่วยเพ็ญ

ภาควิชา : วิศวกรรมเครื่องกล

ปีการศึกษา : 2560

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมเครื่องกล

คณะกรรมการสอบโครงงาน

..... ที่ปรึกษาโครงงาน
(ผู้ช่วยศาสตราจารย์ชูพงศ์ ช่วยเพ็ญ)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.อนันต์ชัย อยู่แก้ว)

..... กรรมการ
(ดร.ปองพันธ์ โอทกานนท์)

หัวข้อโครงงาน : อัลกรอลิทึมตรวจจับช่องทางเดินรถสำหรับยานยนต์อัตโนมัติ มหาวิทยาลัยนเรศวร

ผู้ดำเนินโครงงาน : นายกฤษณ์ สอนสำแดง รหัส 57361838
นายธนบัตร ว่องวัฒนกูล รหัส 57362118
นายเรืองวิทย์ จันทะโก รหัส 57362408

อาจารย์ที่ปรึกษาโครงงาน : ผู้ช่วยศาสตราจารย์ชูพงศ์ ช่วยเพ็ญ

ภาควิชา : วิศวกรรมเครื่องกล

ปีการศึกษา : 2560

บทคัดย่อ

โครงงานนี้มีวัตถุประสงค์เพื่อศึกษาและประยุกต์ใช้เรื่องของการประมวลผลภาพ พัฒนาอัลกอริทึมในการวิเคราะห์เส้นแบ่งแยกช่องทางเดินรถของนิสิตชั้นปีที่ 4 คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมเครื่องกล มหาวิทยาลัยนเรศวร เรื่องอัลกอริทึมตรวจจับช่องทางเดินรถสำหรับยานยนต์อัตโนมัติมหาวิทยาลัยนเรศวร โดยทำการทดลองกับถนนภายในมหาวิทยาลัย แต่เนื่องจากเส้นแบ่งช่องทางเดินรถภายในมหาวิทยาลัยนั้นมีสีที่ขาดหายและไม่ชัดเจน ทำให้ไม่สามารถนำมาใช้เป็นข้อมูลในการประมวลผลภาพได้ ดังนั้นผู้ดำเนินโครงงานจึงได้จัดทำแบบจำลองช่องทางเดินรถขึ้นมา 2 แบบ เพื่อใช้ในการดำเนินงาน โดยใช้หลักการและทฤษฎีของระบบช่วยขับ (Assistant Driving) การตรวจจับช่องทางการเดินรถ (Lane Detection) ระบบควบคุมรถให้อยู่ในช่องทาง (Lane Keeping System) ทำการทดลองที่ค่าความเข้มแสง 15 lux และ 241 lux เพื่อหาความแตกต่าง

ผลการทดลองพบว่าที่ค่าความเข้มแสงต่างกันและแบบถนนจำลองที่ต่างกันนั้นไม่มีผลต่อการตรวจจับช่องทางเดินรถของแบบจำลอง โปรแกรมประมวลผลภาพมีการทำงานค่อนข้างเสถียร และอาจมีการคลาดเคลื่อนของจุดกึ่งกลางเล็กน้อย เนื่องจากขนาดของเส้นช่องทางเดินรถที่ไม่เท่ากันและแสงที่ตกกระทบบนถนน แต่การคำนวณของการประมวลผลภาพจะอยู่ในระยะที่ปลอดภัยสามารถใช้งานได้ปกติ

Project Title	: Lane Detection for NU Autonomous Vehicle Driving Control
Names	: Mr. Krit Sonsamdang Student No. 57361838
	Mr. Tanabat Wongwattanakul Student No. 57362118
	Mr. Ruangwit Jantago Student No. 57362408
Project Advisor	: Assistant Professor Choopong Chuaypen
Department	: Mechanical Engineering
Academic	: 2017

Abstract

This project intended to study and apply the subject of image processing, Developing algorithms to analyze the lines separating lanes. This Project of the 4th year students of Naresuan University Faculty of Engineering And Major Mechanical Engineering. Title is "Lane Detection For Naresuan University Autonomous Vehicle Driving Control" by experimenting with the road within the University. However, The color of lines separating lanes not clear. It can't be used as image processing information. So, project managers have developed 2 models of lines separating lanes to be used in operations. Using the principles and theories are Assistant Driving, Lane Detection and Lane Keeping System. The experimental light intensity 15 lux and 241 lux for the difference.

The results showed that the difference between the light intensity and the different model roads did not affect the detection of the model Lanes. Image processing program is working relatively stable. But there may be a slight deviation of the midpoint. Because of lanes size are not equal and light that is incident on the road. But the calculation of the image processing is in a safe distance, can be used normally.

กิตติกรรมประกาศ

โครงงานอัลกอลิทึมตรวจจับช่องทางเดินรถสำหรับยานยนต์อัตโนมัติ มหาวิทยาลัยนเรศวรนี้สามารถประสบผลสำเร็จลงได้ด้วยดี คณะผู้ดำเนินโครงงานต้องของขอบพระคุณบุคคลที่มีส่วนช่วยเหลือ ให้คำปรึกษา แนะนำ และความอนุเคราะห์ในการดำเนินโครงงานตลอดมาจนสำเร็จ ดังนี้

1. พ่อและแม่ ที่อบรมสั่งสอนมาอย่างดี และคอยสนับสนุนในทุกๆด้าน
2. อาจารย์ชูพงศ์ ช่วยเพ็ญ ซึ่งเป็นอาจารย์ที่ปรึกษาโครงงาน ขอขอบพระคุณสำหรับการให้คำปรึกษาและคำแนะนำ ตลอดจนช่วยดูแลจนโครงงานนี้สำเร็จลุล่วง
3. อาจารย์ชูพงศ์ ช่วยเพ็ญ ที่ช่วยจัดหาอุปกรณ์สำคัญในการทำโครงงานนี้
4. คณาจารย์ บุคลากร และเพื่อนๆทุกคนสำหรับคำปรึกษา แนะนำ และให้ความอนุเคราะห์ในการดำเนินโครงงานนี้

คณะผู้ดำเนินโครงงาน

สารบัญ

	หน้า
ใบรับรองปริญญานิพนธ์	ก
บทคัดย่อภาษาไทย	ข
บทคัดย่อภาษาอังกฤษ	ค
กิตติกรรมประกาศ	ง
สารบัญ	จ
สารบัญตาราง	ซ
สารบัญรูปภาพ	ฌ
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของโครงงาน	1
1.2 วัตถุประสงค์ของโครงงาน	2
1.3 ขอบเขตของโครงงาน	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	3
1.5 ขั้นตอนและแผนการดำเนินงาน	3
บทที่ 2 หลักการและทฤษฎีเบื้องต้น	4
2.1 ตัวอย่างระบบช่วยเหลือในการขับขี่ของรถยนต์ Honda รุ่น Accord Hybrid Tech ปี 2016	4
2.1.1 ช่วยควบคุมรถให้อยู่ในช่องทางเดินรถ – Lane Keeping Assist System (LKAS)	4
2.1.2 ระบบแจ้งเตือนและช่วยเหลือเมื่อรถออกนอกช่องทางเดินรถ Road Departure Mitigation (RDM) กับ Lane Departure Warning (LDW)	5
2.1.3 ระบบช่วยควบคุมรถให้อยู่ในช่องทาง (Lane Keeping System)	6
2.2 การปรับเทียบภาพและการทำให้ภาพไม่บิดเบียน (Camera Calibration and Image Undistortion)	7
2.2.1 Barrel distortion	7
2.2.2 Pincushion distortion	8
2.2.3 Perspective distortion	8
2.3 ขั้นตอนการแก้ไขภาพที่เกิดการบิดเบี้ยว โดยใช้ MATLAB	9
2.4 การกรองภาพ (Image filtering)	11

2.4.1 การกำหนดเกณฑ์สี (Color thresholding) 11
2.4.2 การลดสัญญาณลบกวนของภาพ 17
2.4.2.1 สัญญาณรบกวนของภาพ (Noise) 17
2.4.2.2 การลดสัญญาณรบกวนด้วยการใช้การกรองค่ามัธยฐาน 18
(Median Filtering)
2.4.2.3 การลดสัญญาณรบกวนด้วยวิธี Median Filter โดยใช้ 19
โปรแกรม MATLAB
2.4.3 เกณฑ์ความกว้าง (Magnitude threshold) 20
2.5 การแปลงมุมมอง (Perspective transform) 22
2.6 การตรวจจับเลน (Lane detection) 22
2.7 แสดงเลนที่ตรวจพบ (Displaying the detected lane) 23

บทที่ 3 วิธีการดำเนินงาน 24
3.1 สมมติฐานในการดำเนินโครงงาน 25
3.2 นำกระบวนการที่เลือกใช้เขียนลง Simulink 26
3.2.1 สนใจเฉพาะที่เส้นแบ่งช่องทางการเดินรถสีเหลือง 27
3.2.1.1 การตัดภาพ 27
3.2.1.2 แปลงระบบสีของภาพ 28
3.2.1.3 ปรับค่า hue 28
3.2.1.4 แสดงขอบของวัตถุในภาพ 29
3.2.1.5 วิเคราะห์วัตถุที่สนใจ 30
3.2.2 สนใจเฉพาะที่เส้นแบ่งช่องทางการเดินรถสีขาว 31
3.2.2.1 การตัดภาพ 31
3.2.2.2 แปลงระบบสีของภาพ 31
3.2.2.3 ปรับค่า hue 32
3.2.2.4 แสดงขอบของวัตถุในภาพ 33
3.2.2.5 วิเคราะห์วัตถุที่สนใจ 34
3.2.3 หาจุดCentroid 35
3.2.4 สร้าง mark 35
3.2.5 ซ้อนทับภาพ 36

สารบัญ (ต่อ)

	หน้า
บทที่ 4 ผลการทดลองและวิเคราะห์ผลการทดลอง	41
4.1 ผลการทดลอง	41
4.2 วิเคราะห์ผลการทดลอง	44
บทที่ 5 สรุปผลโครงงานและข้อเสนอแนะ	45
5.1 สรุปผลโครงงาน	45
5.2 ข้อเสนอแนะ	46
บรรณานุกรม	47
ภาคผนวก	48 - 51
ประวัติผู้ดำเนินโครงงาน	52

สารบัญตาราง

หน้า

ตารางที่ 1.1 ขั้นตอนและแผนการดำเนินโครงงาน 3

ตารางที่ 3.1 แสดงระยะเบรกที่ปลอดภัย 25

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 ภาพที่เกิดการบิดเบี้ยว (Distortion) 8
รูปที่ 2.2 เรียกใช้งาน App Camera Calibrator 9
รูปที่ 2.3 เลือกภาพที่ต้องการ 9
รูปที่ 2.4 ทาการ Calibrate 10
รูปที่ 2.5 กดคาสั่ง Show undistorted 10
รูปที่ 2.6 แสดงภาพก่อนและหลังการ Calibrate 11
รูปที่ 2.7 เรียกใช้งาน APPS Color Thresholder 13
รูปที่ 2.8 เลือกภาพที่ต้องการ 14
รูปที่ 2.9 เลือกระบบสี 14
รูปที่ 2.10 แสดงFunction 15
รูปที่ 2.11 Function ที่ได้ออกมา 15
รูปที่ 2.12 ตัวอย่างการใช้ Function การแปลงระบบสี 16
รูปที่ 2.13 แสดงความแตกต่างของค่า hue ก่อนและหลัง แปลงระบบสี 17
รูปที่ 2.14 รูปภาพที่มี Noise 17
รูปที่ 2.15 รูปภาพที่ไม่มี Noise 17
รูปที่ 2.16 ระดับการเกิดสัญญาณรบกวน 18
รูปที่ 2.17 ฟังก์ชันที่ใช้เพิ่มสัญญาณรบกวนและลดสัญญาณรบกวน 19
รูปที่ 2.18 ตัวอย่างการใช้กระบวนการ sobel 20
รูปที่ 2.19 ภาพที่ได้ออกมาหลังจากผ่านกระบวนการ sobel 21
รูปที่ 2.20 ภาพที่ผ่านกระบวนการข้างต้นมาแล้ว 21
รูปที่ 2.21 ภาพที่ได้ออกมาหลังจากแปลงมุมมอง 22
รูปที่ 2.22 ภาพแสดงพิกัดของเส้นแบ่งช่องทางเดินรถ 22
รูปที่ 2.23 ภาพแสดงจุดกึ่งกลางที่หาได้ 23
รูปที่ 2.24 ภาพแสดงผลลัพธ์ที่ออกมาได้ 23
รูปที่ 3.1 แผนภาพการดาเนินงาน 24
รูปที่ 3.2 ภาพอ้างอิงระยะเบรกที่ปลอดภัย 26
รูปที่ 3.3 ฟังก์ชันที่ใช้ในการตัดภาพ 27
รูปที่ 3.4 ภาพต้นฉบับ 27

สารบัญรูปภาพ (ต่อ)

รูปที่ 3.5 ภาพที่ตัดออกมา 27
รูปที่ 3.6 ฟังก์ชันที่ใช้แปลงระบบสี 28
รูปที่ 3.7 ฟังก์ชันที่ใช้ปรับค่า Hue Saturation และValue 29
รูปที่ 3.8 ภาพที่ได้ออกมาหลังจากผ่านฟังก์ชัน 29
รูปที่ 3.9 เขียนฟังก์ชันเปลี่ยนภาพให้เป็น ภาพขาวดา 29
รูปที่ 3.10 Tool Box Edge Detection 30
รูปที่ 3.11 ภาพที่ได้หลังจากผ่าน Tool Box Edge Detection 30
รูปที่ 3.12 Toolbox Blob analysis 30
รูปที่ 3.13 ฟังก์ชันที่ใช้ในการตัดภาพ 31
รูปที่ 3.14 ภาพที่ตัดออกมา 31
รูปที่ 3.15 ฟังก์ชันที่ใช้แปลงระบบสี 32
รูปที่ 3.16 ฟังก์ชันที่ใช้ปรับค่า Hue Saturation และValue 32
รูปที่ 3.17 ภาพที่ได้ออกมาหลังจากผ่านฟังก์ชัน 33
รูปที่ 3.18 เขียนฟังก์ชันเปลี่ยนภาพให้เป็น ภาพขาวดา 33
รูปที่ 3.19 Toolbox Edge Detection 34
รูปที่ 3.20 ภาพที่ได้หลังจากผ่าน toolbox Edge Detection 34
รูปที่ 3.21 Toolbox Blob analysis 34
รูปที่ 3.22 ชุด Toolbox ในการหาจุด Centroid 35
รูปที่ 3.23 เขียนฟังก์ชันกาหนดความกว้างและความยาวของ mark 35
รูปที่ 3.24 Toolbox Draw Shapes 36
รูปที่ 3.25 ภาพที่ได้หลังจากผ่าน Toolbox Draw Shapes 36
รูปที่ 3.26 Toolbox Compositing 36
รูปที่ 3.27 ภาพที่ได้หลังจากผ่าน Toolbox Compositing 36
รูปที่ 3.28 แผนภาพกระบวนการตั้งแต่ 3.2.1 ถึง 3.2.5 37
รูปที่ 4.1 จุดสีแดงสิ้นสุดทางโค้ง 41
รูปที่ 4.2 จุดสีแดงเริ่มเข้าสู่ทางตรง 41
รูปที่ 4.3 จุดสีแดงเข้าสู่ทางตรง 41
รูปที่ 4.4 จุดสีแดงเริ่มเข้าสู่ทางโค้ง 41
รูปที่ 4.5 จุดสีแดงสิ้นสุดทางโค้ง 42
รูปที่ 4.6 จุดสีแดงเริ่มเข้าสู่ทางตรง 42
รูปที่ 4.7 จุดสีแดงเข้าสู่ทางตรง 42

สารบัญรูปภาพ (ต่อ)

หน้า

รูปที่ 4.8 จุดสีแดงเริ่มเข้าสู่ทางโค้ง 42
รูปที่ 4.9 จุดสีแดงเริ่มเข้าสู่ทางโค้ง 43
รูปที่ 4.10 จุดสีแดงเข้าสู่ทางโค้ง 43
รูปที่ 4.11 จุดสีแดงเริ่มเข้าสู่ทางโค้ง 43
รูปที่ 4.12 จุดสีแดงเข้าสู่ทางตรง 43

ลำดับสัญลักษณ์ (การแปลงปริภูมิสี RGB มาเป็นปริภูมิสี HSV)

สัญลักษณ์	ความหมาย	หน่วย
H	ค่าสีของสีหลัก	-
S	ความเข้มของสี	-
V	ความมืดและสว่างของสี	-
R	ค่าของสีแดง	-
G	ค่าของสีเขียว	-
B	ค่าของสีน้ำเงิน	-

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ปัจจุบันยานพาหนะได้ผลิตออกมากันอย่างมากมาย หลากหลายรุ่นหลายยี่ห้อ เพื่อตอบสนองต่อผู้ใช้งาน ไม่ว่าจะเป็นการขนส่งสินค้า การค้าขาย การเดินทาง ต่างต้องใช้ยานพาหนะเป็นสิ่งจำเป็นที่บ้านทุกหลังต้องมีอย่างน้อยหลังละคัน ถึงแม้ว่าบ้านใดไม่มีก็ยังจะต้องไปใช้บริการรถร่วมหรือรถโดยสาร เห็นได้ว่ายานพาหนะเป็นสิ่งสำคัญมาก และก็ยังเป็นสิ่งที่สามารถก่อให้เกิดอุบัติเหตุได้ด้วยเช่นกัน สิ่งที่เป็นปัจจัยหลักของยาพาหนะคือ ระบบการควบคุมทิศทาง เพราะถ้าการควบคุมทิศทางไม่ดีอาจทำให้เกิดอุบัติเหตุได้ เช่น การขับผิดเลนไปยังเลนตรงข้าม การหลุดโค้ง การขับรถแบบไม่มีสติ เป็นต้น สิ่งเหล่านี้ล้วนทำให้เกิดความเสียหายอย่างมากและร้ายแรงที่สุดคือการเสียชีวิต ดังนั้นสิ่งที่สำคัญอย่างยิ่งที่ยาพาหนะต้องมีคือระบบความปลอดภัย เช่น ระบบเบรก ระบบทรงตัว ระบบขับเคลื่อนอัตโนมัติ เป็นต้น ระบบเบรกในปัจจุบันได้มีการพัฒนามาในระดับหนึ่งคือระบบกระจายแรงเบรก (ABS) ซึ่งจะสามารถป้องกันการเบรกแบบล้อล็อคตาย ซึ่งจะทำให้รถมีการลื่นไถลในขณะล้อล็อคทำให้ผู้ขับขี่ไม่สามารถควบคุมรถได้ชั่วขณะอาจทำให้เกิดอุบัติเหตุขึ้นได้ ระบบการทรงตัวทำให้สามารถควบคุมลักษณะที่หน้าดื้อ (Understeer) ท้ายปัด (Oversteer) ทำให้ควบคุมรถในเวลาคับขันช่วยลดปัญหาการเกิดอุบัติเหตุได้ และระบบขั้นเหนือกว่าคือระบบขับเคลื่อนอัตโนมัติ ระบบนี้เข้ามาควบคุมในหลายๆด้านเช่น การควบคุมความเร็วในระดับคงที่ โดยผู้ขับไม่ต้องเหยียบคันเร่ง การแจ้งเตือนเมื่อจะเกิดการชน โดยจะมีเซ็นเซอร์ติดอยู่รอบๆตัวรถ ขั้นสูงสุดของระบบนี้คือการขับเคลื่อนเองได้โดยไม่ต้องมีผู้ขับขี่ โดยการเดินทางด้วยระบบ GPS ตามพิกัดที่ได้กำหนดไว้ ทั้งจุดเริ่มต้นและจุดหมายปลายทาง และระบบรักษาช่องทางการเดินรถ (lane keeping) ซึ่งจะทำหน้าที่ค่อยบังคับให้รถวิ่งตามเลนที่ถูกต้อง การนำเอาเทคโนโลยีเข้ามาเป็นตัวช่วยในการควบคุมระบบบังคับทิศทาง (พวงมาลัย) เช่นการควบคุมพวงมาลัยขณะที่ผู้ใช้งานต้องการพักในขณะเมื่อยล้า ระบบดังกล่าวจำเป็นต้องรู้ลักษณะของถนนในพื้นที่การใช้งาน การตรวจจับช่องทางเดินรถ การตรวจจับแนวขอบทาง การตรวจจับสิ่งกีดขว้าง การตรวจจับรถอื่นๆบนท้องถนน เทคโนโลยีอย่างหนึ่งที่

สามารถนำมาประยุกต์ใช้ได้คือการประมวลผลภาพ (Image processing) ซึ่งมีทั้งขั้นตอนและเทคนิคการประยุกต์ใช้งานที่หลากหลาย ตลอดจนการประยุกต์ใช้ระบบอัจฉริยะ (artificially intelligent) ช่วยในการประมวลผลภาพเพื่อวัตถุประสงค์ต่างๆ เมื่อระบบสามารถวิเคราะห์ข้อมูลต่างๆในช่องทางการเดินรถแล้วจะส่งข้อมูลให้แก่ระบบบังคับเลี้ยวของรถ ซึ่งมีความจำเป็นในการพัฒนาตัวควบคุมให้สามารถทำงานได้อย่างสอดคล้องกับข้อมูลที่ได้จากการประมวลผลภาพดังกล่าว

ดังนั้นในโครงงานนี้จึงมุ่งเน้นการประยุกต์และพัฒนากระบวนการในการประมวลผลภาพเพื่อใช้ในการวิเคราะห์หาเส้นขอบทาง ช่องทางการเดินรถ ขนาดหรือระยะห่างของช่องทางเดินรถ ระยะห่างของรถที่อยู่ด้านหน้า ระยะห่างของสิ่งกีดขวาง กระบวนการและกลไกการควบคุมระบบบังคับเลี้ยวเพื่อการควบคุมให้สามารถเดินรถอยู่ในช่องทางเดินรถที่กำหนดมาให้ ตลอดจนการควบคุมความเร็วและระบบเบรกของรถอย่างอัตโนมัติ

1.2 วัตถุประสงค์ของโครงงาน

1) ศึกษาและประยุกษ์ใช้เรื่องของการประมวลผลภาพ
2) ศึกษาและพัฒนาอัลกรอลิทึมในการวิเคราะห์เส้นแบ่งแยกช่องทางจราจร

1.3 ขอบเขตของโครงงาน

1) ออกแบบการประมวลผลภาพ และวิเคราะห์หาเส้นขอบทางของถนน
2) ศึกษาหาตัวแปรที่มีผลต่อประสิทธิภาพของอัลกรอลิทึมที่ได้ทำขึ้นมา
3) ออกแบบอัลกรอลิทึมการควบคุมการเดินรถให้อยู่ในช่องทางเดินรถ

1.4 ประโยชน์ที่คาดว่าจะได้รับ

ได้อัลกอริทึมในการประมวลผลภาพจากกล้อง เพื่อพิสูจน์ภาพเส้นขอบทางและคำนวณหาจุดกึ่งกลางให้อยู่ในแบบจำลองช่องทางเดินรถ

- คำนวณหาจุดกึ่งกลาง
- ประมวลผลหาจุดกึ่งกลางเส้นแบ่งช่องทางเดินรถ

1.5 ขั้นตอนและแผนการดำเนินโครงงาน

ตารางที่ 1.1 ขั้นตอนและแผนการดำเนินโครงงาน

	การดำเนินโครงงาน	ช่วงเวลา									
		ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.
1.5.1	ศึกษาหลักการและทฤษฎีที่เกี่ยวข้อง										
1.5.2	พัฒนาและออกแบบ										
1.5.3	ประเมินผลทดสอบ										
1.5.4	ทดสอบประสิทธิภาพของระบบ										
1.5.5	วิเคราะห์และสรุปผลที่ได้										
1.5.6	นำเสนอโครงงาน										

บทที่ 2
หลักการและทฤษฎีเบื้องต้น

รถยนต์เป็นสิ่งที่จำเป็นต่อการดำรงชีวิต ทำให้สะดวกในการเดินทาง การขนส่งสินค้า ฯลฯ ถึงแม้จะสะดวกแต่ก็ยังมีความอันตรายอยู่เช่นกัน อันเนื่องจากตัวผู้ขับขี่ สภาพพื้นผิวการจราจร เป็นต้น ทำให้เกิดอุบัติเหตุได้ ดังนั้นสิ่งที่สำคัญอย่างยิ่งที่ในรถยนต์ที่ต้องมีคือระบบความปลอดภัย ในปัจจุบันได้มีการพัฒนาและติดตั้งระบบความปลอดภัยเข้าไปในรถยนต์ ในที่นี้จะยกตัวอย่างรถยนต์ของ Honda รุ่น Accord Hybrid Tech ปี 2016

2.1 ตัวอย่างระบบช่วยเหลือในการขับขี่ของรถยนต์ Honda รุ่น Accord Hybrid Tech ปี 2016[3]

ระบบช่วยควบคุมรถให้อยู่ในช่องทางเดินรถ Lane Keeping Assist (LKAS) ที่มีใน Honda Accord Hybrid Tech ปี 2016 โดยจะมาพร้อมระบบแจ้งเตือนและช่วยเหลือเมื่อรถออกนอกช่องทางเดินรถ Road Departure Mitigation (RDM) กับ Lane Departure Warning (LDW) ซึ่งจะทำงานควบคู่กันได้อย่างดี การทดสอบรถยนต์ Honda Accord Hybrid Tech ปี 2016 พร้อมกับระบบ Honda Sensing ด้วยเส้นทาง กรุงเทพ – เชียงใหม่ ซึ่งส่วนหนึ่งของระบบมี 2 ระบบที่เป็นประโยชน์มาก ตลอดการเดินทางและให้ความปลอดภัยได้ดี คือ

2.1.1 ช่วยควบคุมรถให้อยู่ในช่องทางเดินรถ – Lane Keeping Assist System (LKAS)[3]

เป็นระบบที่ใช้กล้องด้านหน้าในการตรวจจับเส้นแบ่งช่องทางเดินรถ โดยระบบจะช่วยเพิ่มแรงหน่วงของพวงมาลัยเพื่อช่วยผู้ขับขี่ให้ควบคุมรถอยู่ภายในช่องทางเดินรถปกติและลดอาการเหนื่อยล้าของผู้ขับขี่ กล้องที่ติดตั้งอยู่บริเวณกระจกบังลมด้านหน้า จะตรวจจับเส้นแบ่งช่องทางเดินรถที่อยู่บนถนน โดยจะเริ่มทำงานที่ช่วงความเร็ว 72 – 180 กิโลเมตร/ชั่งโมง ซึ่งผู้ขับขี่

สามารถเปิดหรือปิดการทำงานของระบบนี้ได้ด้วยการกดสวิตช์ควบคุมที่ติดตั้งอยู่ที่มุมล่างขวาของพวงมาลัย เนื่องจาก (LKAS) ไม่ใช่ระบบขับขี่อัตโนมัติ ดังนั้นระบบจะหยุดทำงานถ้าหากผู้ขับขี่ปล่อยมือออกจากพวงมาลัย โดยจะมีการเตือนด้วยภาพผ่านทางหน้าจอ MID เพื่อให้ผู้ขับขี่ควบคุมรถให้อยู่ภายในช่องทางอีกครั้ง

2.1.2 ระบบแจ้งเตือนและช่วยเหลือเมื่อรถออกนอกช่องทางเดินรถ Road Departure Mitigation (RDM) กับ Lane Departure Warning (LDW)[3]

เป็นระบบที่ใช้กล้องด้านหน้าในการตรวจจับเส้นแบ่งช่องทางจราจร หากพบว่ารถอยู่ในสภาวะเบี่ยงออกนอกช่องทางโดยไม่ตั้งใจ ระบบจะส่งสัญญาณเตือนที่หน้าจอแสดงข้อมูล พร้อมการสั่นเตือนของพวงมาลัย และในกรณีที่รถเริ่มเบี่ยงออกนอกช่องทางมากยิ่งขึ้น ระบบจะช่วยหน่วงพวงมาลัยเพื่อให้รถกลับเข้าสู่ช่องทาง หากรถยังคงเบี่ยงออกนอกช่องทางจนอาจเกิดอุบัติเหตุ ระบบเบรกจะทำงานเพื่อชะลอความเร็ว (ในกรณีเส้นแบ่งถนนเป็นเส้นทึบ) เพื่อช่วยลดความเสี่ยงที่รถจะออกนอกช่องทางจราจร ระบบ RDM ระรวมอยู่ในระบบควบคุมการทรงตัว หรือ VSA เพื่อให้ตัวรถมีการเบรกอย่างเหมาะสม และพวงมาลัยเพาเวอร์ไฟฟ้า หรือ EPS ช่วยให้การบังคับพวงมาลัยทำได้อย่างแม่นยำ การแจ้งเตือนผ่านทั้งภาพและเสียง โดยจะแสดงผลผ่านหน้าจอ Multi Information Display (MID) ช่วยให้ผู้ขับขี่ได้ทราบว่าระบบกำลังทำงานเพื่อให้รถกลับมาอยู่ในช่องทาง โดยสามารถตั้งค่าการแจ้งเตือนต่างๆ ผ่านทางฟังก์ชั่น Setting ในตัวรถได้ มาถึงวิธีการใช้งานระบบควบคุมความเร็วแปรผันตามรถคันหน้าก่อน เมื่อความเร็วถึงจุดที่ระบบทำงานได้นั้นคือ 72 – 180 กิโลเมตร/ชั่วโมง ให้กดปุ่ม “MAIN” ขวามือบนพวงมาลัย หน้าจอบอกการใช้งานจะสว่างขึ้น หลังจากนั้นก็ปรับตั้งความเร็วตามต้องการ ด้วยปุ่ม + / - เมื่อได้ความเร็วตามที่ตั้งไว้ หากต้องการใช้ระบบช่วยควบคุมรถให้อยู่ในช่องทางเดินรถ Lane Keeping Assist System (LKAS) ให้กดปุ่มล่างที่เป็นรูปรถกับเส้นขอบทางเพื่อเปิดการใช้งาน หลังจากนั้นระบบก็ตรวจจับเส้นช่องทางอัตโนมัติ พร้อมกับการบังคับเลี้ยวให้รถตรงช่องทางมากที่สุด แต่มีข้อแม้ว่าห้ามปล่อยมือจากพวกมาลัย เพราะจะมีการเตือนทันที ต่อมาเมื่อต้องการใช้ระบบแจ้งเตือนและช่วยเหลือเมื่อรถออกนอกช่องทางเดินรถ Road Departure Mitigation (RDM) กับ Lane Departure Warning (LDW) ระบบนี้สามารถเปิดใช้ได้จากปุ่มทางขวาใต้ช่องแอร์คนขับ (ขวาล่างรูปรถเอียงออกนอกเลน) เพียงแค่นี้ระบบก็ทำงานทันที

ข้อสังเกตสำหรับระบบช่วยควบคุมรถให้อยู่ในช่องทางเดินรถ Lane Keeping Assist System (LKAS) นั้นอาจต้องคอยดูบนจอหน้าปัดว่ารูปเส้นของทางนั้นต้องเป็นสีขาวทึบแสดงว่าระบบทำงานได้สมบูรณ์ แต่ถ้าเป็นเส้นโปร่งๆ หมายความว่าระบบไม่สามารถตรวจจับเส้นขอบทางได้ อาจเกิดจากเส้นแบ่งขอบทางนั้นไม่ชัดเจน ไม่มี หรืออาจซีดจาง เป็นต้น ส่วนระบบนี้หากเส้นขอบทางสภาพปกติจะสามารถตรวจจับได้ทั้งกลางวันและกลางคืนเลย นอกจากระบบจะช่วยเหลือการขับขี่และคอยเตือนเมื่อรถอยู่ในอาการที่ไม่ปลอดภัยแล้ว การใช้ระบบช่วยควบคุมรถออกนอกเลนและรักษาตำแหน่งให้อยู่ในเลนอัตโนมัติยังช่วยลดอาการเมื่อยล้าจากการต้องจับพวงมาลัยตลอดเวลา และลดการตึงเครียดในการขับขี่ระยะทางไกลได้เป็นอย่างดีอีกด้วย

อีกระบบหนึ่งที่ทันสมัยก็คือ ระบบควบคุมความแปรผันตามรถคันหน้า เมื่อเรดาร์ตรวจจับสัญญาณว่ามีรถด้านหน้าระบบจะปรับลดความเร็วให้เหมาะสมหรือเบรกให้เอง และเมื่อรถคันหน้าเพิ่มความเร็วหรือไม่มีรถอยู่ด้านหน้าแล้ว ระบบก็จะเพิ่มความเร็วให้ถึงจุดที่ผู้ขับตั้งเอาไว้โดยอัตโนมัติ และสามารถตั้งระยะห่างจากรถคันหน้าได้ 3 ระดับ

2.1.3 ระบบช่วยควบคุมรถให้อยู่ในช่องทาง (Lane Keeping System)

ระบบช่วยควบคุมรถให้อยู่ในช่องทาง (Lane Keeping System) เป็นระบบที่ช่วยบังคับรถยนต์ให้อยู่ช่องทางการเดินรถ ซึ่งแตกต่างจากระบบแจ้งเตือนว่ารถกำลังจะออกนอกช่องทางการเดินรถ เพราะระบบนี้ไม่แค่แจ้งเตือนบนหน้าจอว่ารถกำลังจะออกจากช่องทางการเดินรถ แต่ระบบนี้ยังช่วยบังคับพวงมาลัยให้กลับมาอยู่ในช่องทางการเดินรถที่ถูกต้องอีกด้วย เป็นระบบเซฟตี้ที่ช่วยเพิ่มความปลอดภัย ช่วยลดอุบัติเหตุที่อาจเกิดขึ้นในเวลาที่ผู้ขับรถเผลอหรือวูบหลับไป แต่ระบบนี้ยังต้องพึ่งพาอาศัยเส้นจราจรบนถนนให้ตรวจจับอย่างน้อย 1 เส้น ระบบช่วยควบคุมรถให้อยู่ในช่องทางจะทำงานร่วมกับกล้องที่ติดตั้งอยู่บริเวณกระจกมองหลังเพื่อตรวจจับหาเส้นแบ่งช่องทางการเดินรถบนพื้นถนนด้านหน้า โดยระบบสามารถตรวจสอบได้ว่าผู้ขับขี่กำลังเบนรถออกจากเส้นแบ่งช่องทางการเดินรถโดยตั้งใจหรือไม่ จากนั้นระบบจะเปรียบเทียบข้อมูลจากกล้องกับมุมและแรงบิดของพวงมาลัย หากระบบพบว่าผู้ขับขี่กำลังเปลี่ยนช่องทางการเดินรถโดยไม่ได้ตั้งใจ ระบบจะเข้าควบคุมแรงบิดของพวงมาลัยพาวเวอร์ไฟฟ้าเพื่อดึงรถให้กลับเข้าสู่เส้นทางเดิม หากรถยังคงเคลื่อนออกนอกช่องทางการเดินรถ สัญญาณเตือนการเปลี่ยนช่องทางการเดินรถจะแจ้งเตือนผู้ขับขี่ด้วยระบบสั่นที่พวงมาลัย

องค์ประกอบสำคัญของระบบช่วยควบคุมรถให้อยู่ในช่องทาง (Lane Keeping System) คือ การประมวลผลภาพ (Image Processing) โดยการประมวลผลภาพเป็นกระบวนการการนำภาพมาประมวลผลด้วยคอมพิวเตอร์ เพื่อนำไปคำนวณหาเส้นกึ่งกลางของช่องทางเดินรถ แล้วทำการตรวจจับเส้นแบ่งช่องทางเดินรถ ซึ่งกระบวนการประมวลผลภาพมีอยู่ด้วยกันหลายอย่าง ยกตัวอย่าง เช่น การแปลงข้อมูลรูปภาพ (Image Transformation), การบอกลักษณะรูปภาพ (Image Description), การกรองรูปภาพ (Image Filter), การปรับปรุงคุณภาพของรูปภาพ (Image Enhancement), การแบ่งภาพและการหาขอบวัตถุในรูปภาพ (Image Segmentation and Edge Detection) เป็นต้น การประยุกต์ใช้วิธีการหรือการวิเคราะห์รูปภาพต่างๆ จะขึ้นอยู่กับข้อมูลของภาพที่ได้มา ซึ่งปัจจัยหลักของการเลือกใช้หรือพัฒนาวิธีการวิเคราะห์จะขึ้นอยู่กับองค์ประกอบและสภาพแวดล้อมของรูปภาพ ตลอดจนวัตถุประสงค์หรือข้อมูลที่ต้องการจากการวิเคราะห์ภาพถ่าย โดยคำอธิบายและหลักการประมวลผลภาพในแต่ละวิธีการมีดังต่อไปนี้

2.2 การปรับเทียบภาพและการทำให้ภาพไม่บิดเบียน (Camera Calibration and Image Undistortion)

Distortion คือ การบิดเบี้ยวของภาพ ในความเป็นจริงแล้วส่วนที่บิดเบี้ยวเหล่านั้นมันจะต้องเป็นเส้นตรงไม่ใช้เส้นที่โค้งงอ หรือเอียงไปด้านใดด้านหนึ่ง

Distortion แบ่งออกเป็น 3 ลักษณะ คือ

2.2.1 Barrel distortion

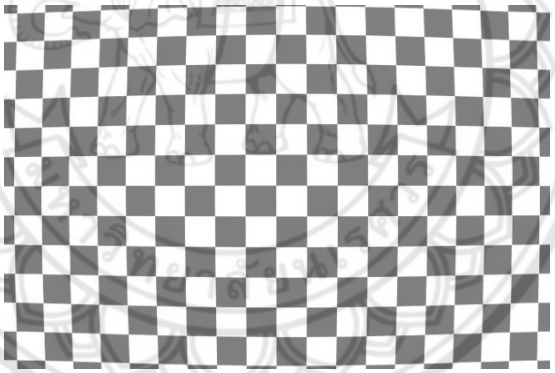
เป็นลักษณะการบิดเบี้ยวที่เหมือนภาพ ๆ นั้นมันบวม ป่อง ตรงกลางภาพ ภาพลักษณะนี้เราจะเห็นได้จากการใช้เลนส์ Fisheye ที่เมื่อถ่ายภาพออกมาแล้วจะได้ภาพลักษณะดังกล่าวที่ป่องขึ้นตรงกลางภาพอย่างรุนแรง อันที่จริงก็สวยไปอีกแบบหนึ่ง และไม่ได้ผิดอะไรถ้าเราต้องการให้ภาพ ๆ นั้นมีลักษณะดังกล่าว

2.2.2 Pincushion distortion

การบิดเบี้ยวลักษณะนี้ จะเป็นแบบตรงข้ามกันกับ Barrel distortion ภาพที่มีลักษณะนี้จะมีอาการเบี้ยว บวม ป่อง ตรงขอบภาพคล้าย ๆกับภาพนั้นมีตรงกลางภาพที่ยุบลงไปนั้นเอง

2.2.3 Perspective distortion

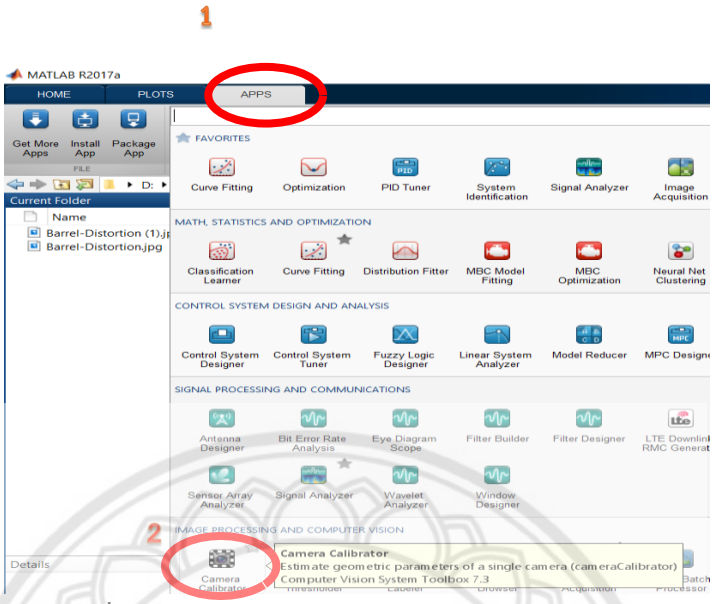
ภาพลักษณะนี้มักจะเกิดจากกรณีที่กล้องถ่ายภาพของเราไม่ได้ตั้งขนานไปกับวัตถุ เช่น ภาพตึก อาคารต่าง ๆ เรามักจะเห็นได้บ่อยครั้งที่ภาพดังกล่าวเหมือนคล้าย ๆกับว่าตึกอาคารเหล่านั้นกำลังล้มเอียงไปยังจุดรวมสายตา View point ของภาพ
เมื่อถ่ายภาพวัตถุ 3 มิติ ภาพจะเปลี่ยนเป็นภาพ 2 มิติ ทำให้เกิดการบิดเบี้ยวของภาพ (Distortion) ไปดังรูป



รูปที่ 2.1 ภาพที่เกิดการบิดเบี้ยว (Distortion)

จากรูปที่ 2.1 เป็นภาพตัวอย่างที่เกิดการบิดเบี้ยวแบบ Barrel distortion ซึ่งเป็นการบิดเบี้ยวที่มีลักษณะบวมป่องตรงกลางภาพ

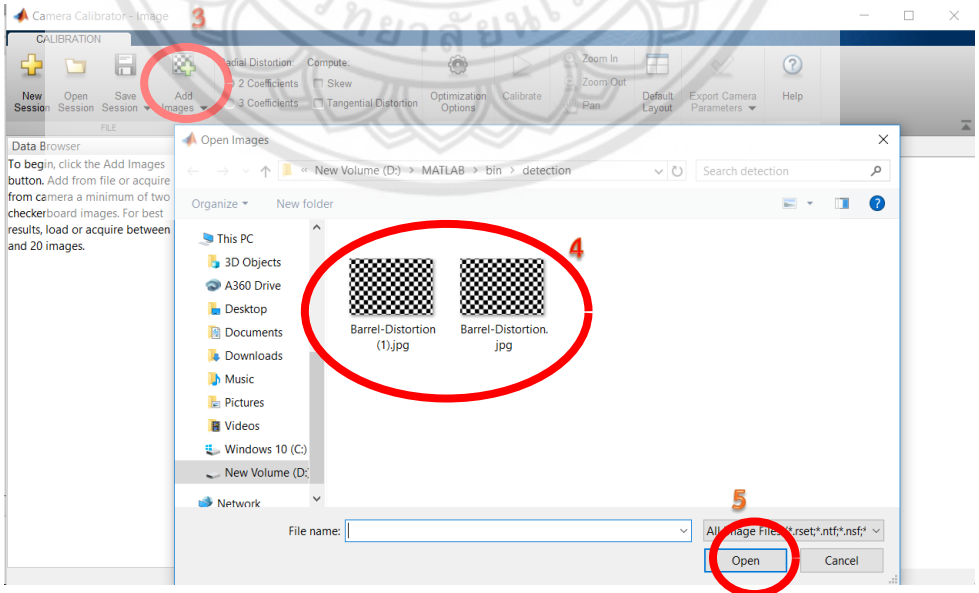
2.3 ขั้นตอนการแก้ไขภาพที่เกิดการบิดเบี้ยว โดยใช้ MATLAB



รูปที่ 2.2 เรียกใช้งาน APPS CAMERA CALIBRATOR

จากรูปที่ 2.2

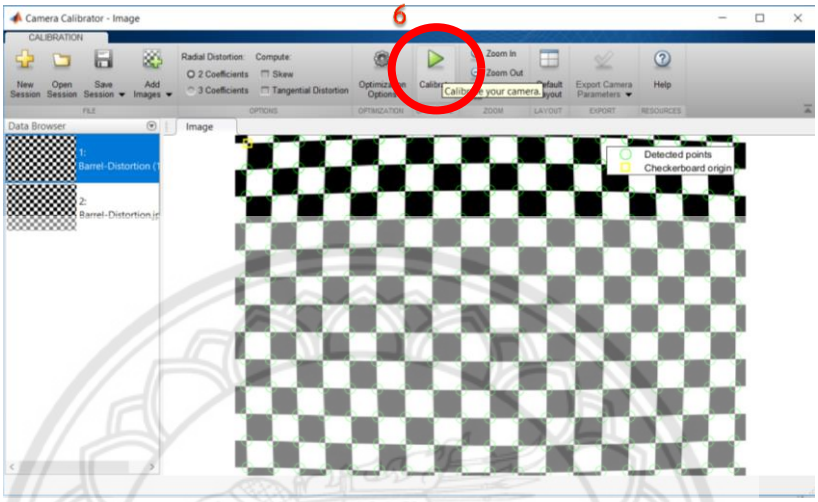
1. กดที่คำสั่ง APPS
2. เลือกที่ไอคอน Camera Calibrator



รูปที่ 2.3 เลือกภาพที่ต้องการ

จากรูปที่ 2.3

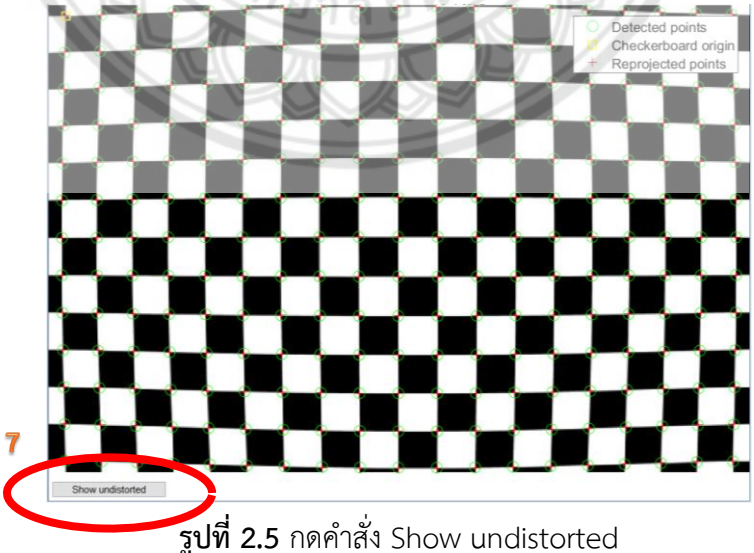
3. กด Add Images
4. เลือกรูปภาพ
5. เมื่อเลือกรูปภาพได้แล้วกด open



รูปที่ 2.4 ทำการ Calibrate

จากรูปที่ 2.4

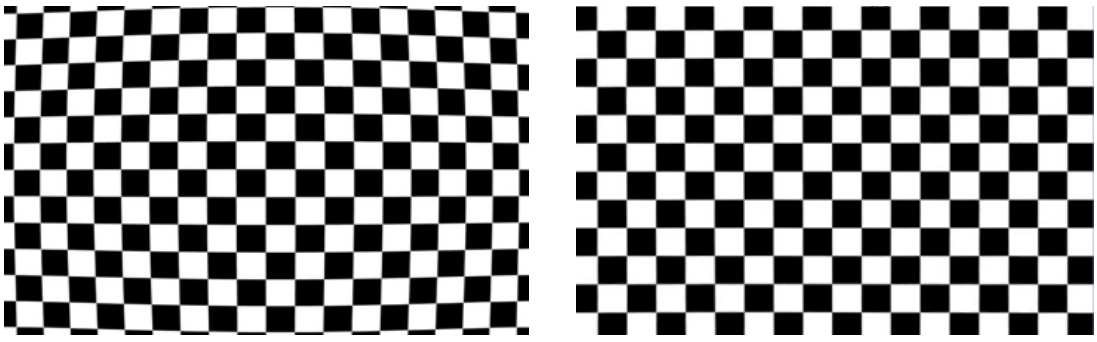
6. กดที่คำสั่ง Calibrate เพื่อทำการ Calibrate ภาพ



รูปที่ 2.5 กดคำสั่ง Show undistorted

จากรูปที่ 2.5

7. กดที่คำสั่ง Show undistorted เพื่อแสดงภาพที่แปลงแล้วออกมา



รูปที่ 2.6 แสดงภาพก่อนและหลังการ Calibrate

จากรูปที่ 2.6 จะได้ภาพเปรียบเทียบระหว่างก่อนและหลังการ Calibrate

2.4 การกรองภาพ (Image filtering)

การกรองภาพใช้เกณฑ์ที่แตกต่างกันสามแบบ คือ การกำหนดเกณฑ์สี การลดสัญญาณรบกวนของภาพ การแปลงมุมมองเพื่อกรองภาพ ให้เหลือแต่เส้นแบ่งช่องทางการเดินรถสีเหลืองและสีขาว

2.4.1 การกำหนดเกณฑ์สี (Color thresholding)

RGB เป็นระบบสีที่ประกอบด้วยแม่สี 3 สี คือ แดง (Red), เขียว (Green) และน้ำเงิน (Blue) เมื่อนำมาผสมผสานกันทำให้เกิดสีต่าง ๆ บนจอคอมพิวเตอร์มากถึง 16.7 ล้านสี ซึ่งใกล้เคียงกับสีที่ตาเรามองเห็นปกติ สีที่ได้จากการผสมสีขึ้นอยู่กับความเข้มของสี โดยถ้าสีมีความเข้มมาก เมื่อนำมาผสมกันจะทำให้เกิดเป็นสีขาว จึงเรียกระบบสีนี้ว่าแบบ Additive หรือการผสมสีแบบบวก

CMYK เป็นระบบสีที่ใช้กับเครื่องพิมพ์ที่พิมพ์ออกทางกระดาษหรือวัสดุผิวเรียบอื่น ๆ ซึ่งประกอบด้วย สีหลัก 4 สี คือ สีฟ้า (Cyan), สีม่วงแดง (Magenta), สีเหลือง (Yellow) และสีดำ

(Black) เมื่อนำมาผสมกันจะเกิดสีเป็นสีดำ แต่จะไม่ดำสนิท เนื่องจากหมึกพิมพ์มีความไม่บริสุทธิ์ จึงเป็นการผสมสีแบบลบ (Subtractive) หลักการเกิดสีของระบบนี้ คือ หมึกสีหนึ่งจะดูด กลืนแสงจากสีหนึ่งแล้วสะท้อนกลับออกมาเป็นสีต่าง ๆ เช่น สีฟ้าดูดกลืนแสงของสีม่วงแล้วสะท้อนออกมาเป็นสีน้ำเงิน ซึ่งจะสังเกตได้ว่าสีที่สะท้อนออกมาจะเป็นสีหลัก ของระบบ RGB การเกิดสีในระบบนี้จึงตรงข้ามกับการเกิดสีในระบบ RGB ดังภาพ

HSB เป็นระบบสีแบบการมองเห็นของสายตามนุษย์ ซึ่งแบ่งออกเป็น 3 ส่วน คือ Hue คือ สีต่าง ๆ ที่สะท้อนออกมาจากวัตถุแล้วเข้าสู่สายตาของเรา ซึ่งมักจะเรียกสีตามชื่อสี เช่น สีเขียว สีเหลือง สีแดง เป็นต้น

Saturation คือ ความสดของสี โดยค่าความสดของสีจะเริ่มที่ 0 ถึง 100 ถ้ากำหนด Saturation ที่ 0 สีจะมีความสดน้อย แต่ถ้ากำหนดที่ 100 สีจะมีความสดมาก

Brightness คือ ระดับความสว่างของสี โดยค่าความสว่างของสีจะเริ่มที่ 0 ถึง 100 ถ้ากำหนดที่ 0 ความสว่างจะน้อยซึ่งจะเป็นสีดำ แต่ถ้ากำหนดที่ 100 สีจะมีความสว่างมากที่สุด

LAB เป็นระบบสีที่ไม่ขึ้นกับอุปกรณ์ใด ๆ (Device Independent) โดยแบ่งออกเป็น 3 ส่วน คือ "L" หรือ Luminance เป็นการกำหนดความสว่าง ซึ่งมีค่าตั้งแต่ 0 ถึง 100 ถ้ากำหนดที่ 0 จะกลายเป็นสีดำ แต่ถ้ากำหนดที่ 100 จะเป็นสีขาว "A" เป็นค่าของสีที่ไล่จากสีเขียวไปสีแดง "B" เป็นค่าของสีที่ไล่จากสีน้ำเงินไปเหลือง

ระบบสี HSV (Hue Saturation Value) เป็นการพิจารณาสีโดยใช้ Hue Saturation และ Value ซึ่ง Hue คือค่าสีของสีหลัก(แดง เขียวและน้ำเงิน)ในทางปฏิบัติจะอยู่ระหว่าง 0 และ 255 ซึ่งถ้า Hue มีค่าเท่ากับ 0 จะแทนสีแดงและเมื่อ Hue มีค่าเพิ่มขึ้นเรื่อย ๆ สีก็จะเปลี่ยนแปลงไปตามสเปกตรัมของสีจนถึง 256 จึงจะกลับมาเป็นสีแดงอีกครั้ง ซึ่งสามารถแทนให้อยู่ในรูปขององศาได้ ดังนี้คือ สีแดง = 0 องศา สีเขียวเท่ากับ 120 องศา สีน้ำเงินเท่ากับ 240 องศา

การแปลงปริภูมิสี RGB มาเป็นปริภูมิสี HSV ทำได้โดย

$$H = \begin{cases} \delta, & B \leq G \\ 2\pi - \delta, & \text{otherwise} \end{cases}$$

โดยที่ $\delta = \cos^{-1}\left(\frac{(R-G)+(R-B)}{2\sqrt{(R-G)^2+(R-B)\cdot(G-B)}}\right)$

$$S = 1 - 3 \cdot \frac{Min(R,G,B)}{R+G+B}$$

$$V = \frac{R+G+B}{3}$$

โดยที่

H (Hue) คือ ค่าสีของสีหลัก

S (Saturation) คือ ความเข้มของสี

V (Value) คือ ความมืดและสว่างของสี

R คือ ค่าของสีแดง

G คือ ค่าของสีเขียว

B คือ ค่าของสีน้ำเงิน

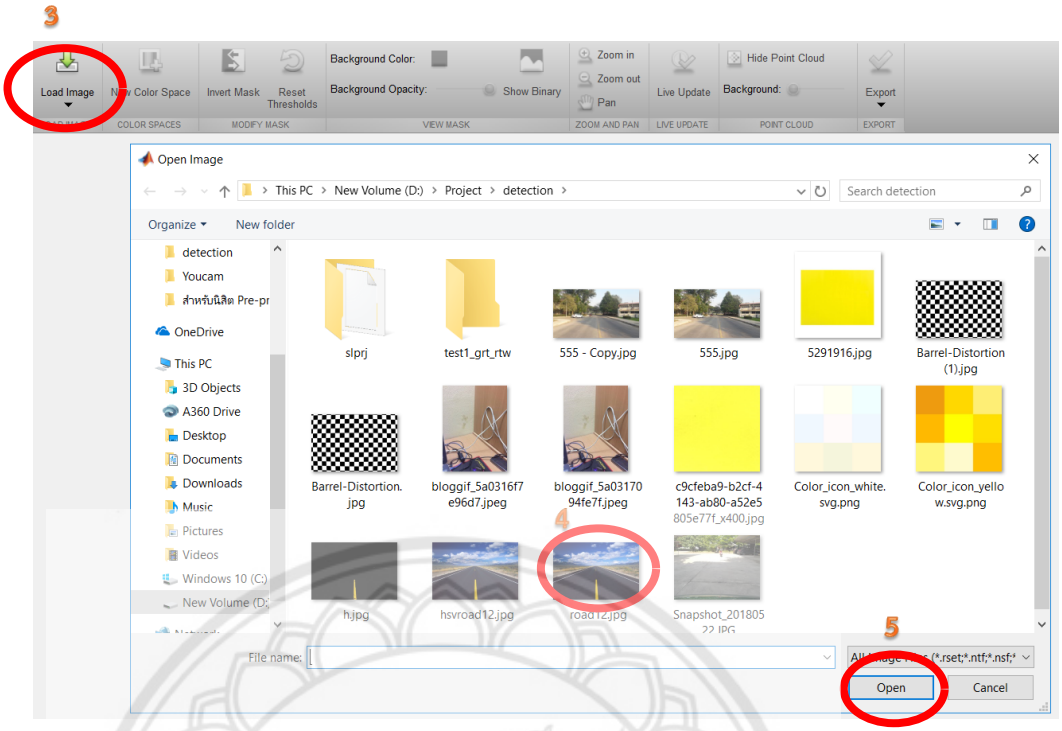
เราแปลงภาพของเราเป็นระบบสี HSV โดยสนใจไปที่ค่าของสี(hue)และความเข้มของพิกเซล (สีที่ใช้และความเข้มของสี) เพื่อกำหนดสีที่จะคงไว้ในภาพ โดยการกำหนดค่าสีที่ต้องการและถ้าค่าของสีมีค่ามากหรือน้อยกว่าที่กำหนดไว้ก็ทำให้เป็นสีดำ

การแปลงภาพให้เป็นระบบสี HSV โดยใช้ MATLAB

รูปที่ 2.7 เรียกใช้งาน APPS Color Thresholder

จากรูปที่ 2.7

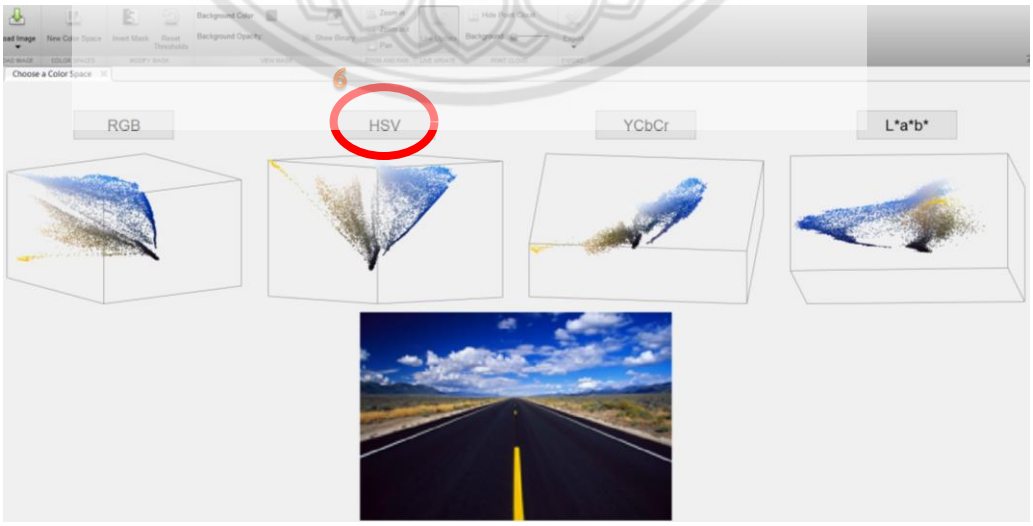
1. กดที่คำสั่ง APPS
2. เลือกที่ไอคอน Color Thresholder



รูปที่ 2.8 เลือกภาพที่ต้องการ

จากรูปที่ 2.8

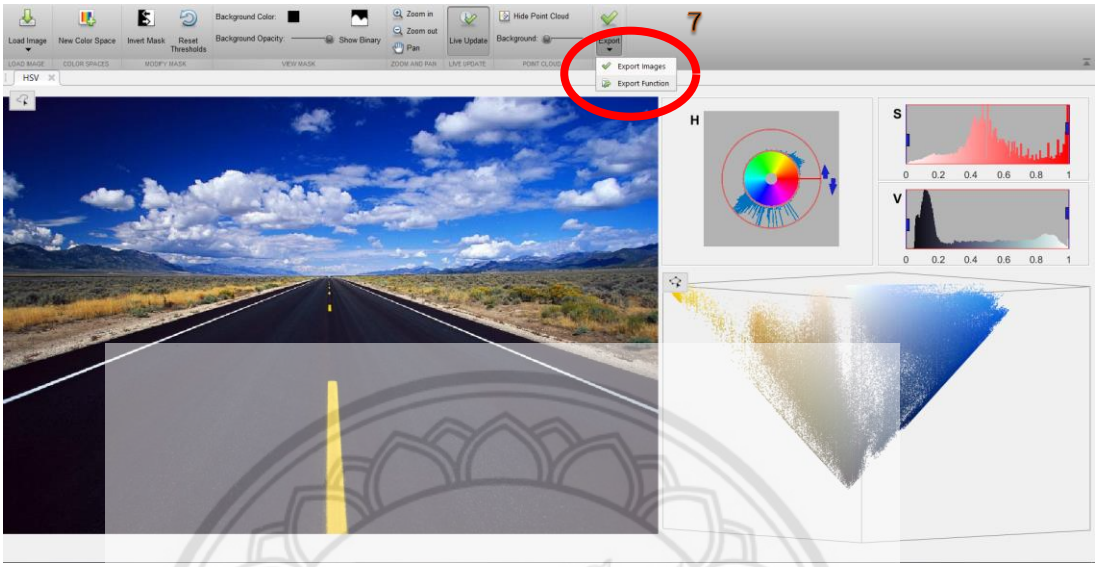
3. กด Add Images
4. เลือกรูปภาพ
5. เมื่อเลือกรูปภาพได้แล้วกด open



รูปที่ 2.9 เลือกระบบสี

จากรูปที่ 2.9

6. เลือกระบบสีเป็น HSV



รูปที่ 2.10 แสดงFunction

จากรูปที่ 2.10

7. กดคำสั่ง Export และเลือก Export Function

```
function [BW,maskedRGBImage] = createMask(RGB)
I = rgb2hsv(RGB);

% Define thresholds for channel 1 based on histogram settings
channel1Min = 0.000;
channel1Max = 1.000;

% Define thresholds for channel 2 based on histogram settings
channel2Min = 0.000;
channel2Max = 1.000;

% Define thresholds for channel 3 based on histogram settings
channel3Min = 0.000;
channel3Max = 1.000;

% Create mask based on chosen histogram thresholds
sliderBW = (I(:,:,1) >= channel1Min ) & (I(:,:,1) <= channel1Max) & .
    (I(:,:,2) >= channel2Min ) & (I(:,:,2) <= channel2Max) & ...
    (I(:,:,3) >= channel3Min ) & (I(:,:,3) <= channel3Max);
BW = sliderBW;

% Initialize output masked image based on input image.
maskedRGBImage = RGB;

% Set background pixels where BW is false to zero.
maskedRGBImage(repmat(~BW,[1 1 3])) = 0;

end
```

รูปที่ 2.11 Function ที่ได้ออกมา

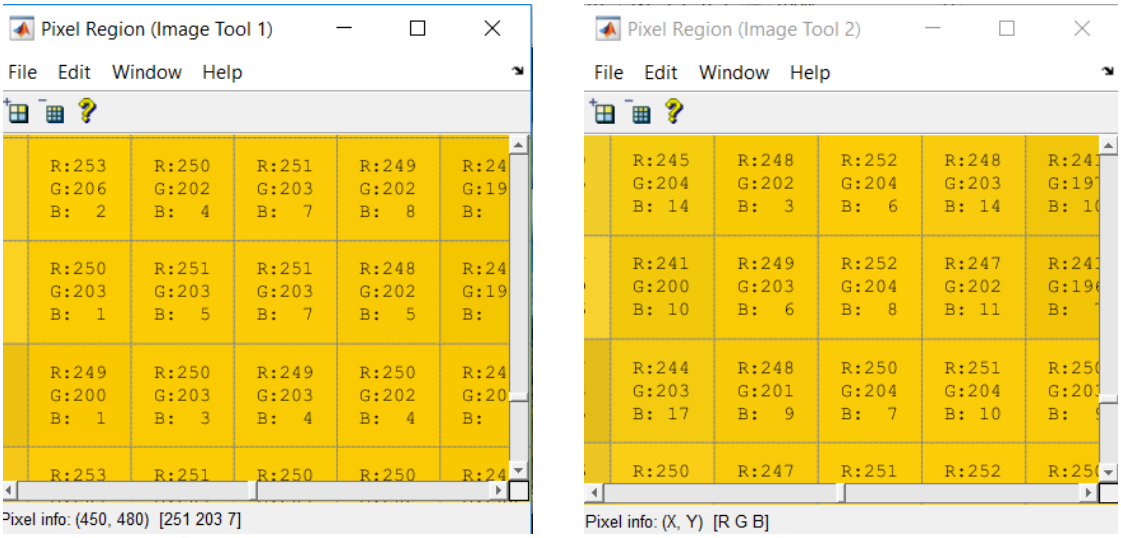
จากรูปที่ 2.11 บันทึก Function ในชื่อ createMask เพื่อนำมาเรียกใช้ต่อไป

```
1 -    im1 = imread('road12.jpg');
2 -    imtool (im1)
3 -    [bw,mask] = createMask(im1);
4 -    imwrite(mask,'hsvroad12.jpg');
5 -    im2 = imread('hsvroad12.jpg');
6 -    imtool (im2)
```

รูปที่ 2.12 ตัวอย่างการใช้ Function การแปลงระบบสี

จากรูปที่ 2.12 แสดงตัวอย่างการนำ Function มาใช้แปลงระบบสีของภาพ

- บรรทัดที่1 เป็นการอ่านข้อมูลของภาพที่มีชื่อว่า road12 และเก็บอยู่ในรูปตัวแปรที่มีชื่อว่า im1
- บรรทัดที่2 คำสั่งแสดงข้อมูลของค่าสีของแต่ละพิกเซล ของตัวแปร im1
- บรรทัดที่3 เรียกใช้ฟังก์ชัน createMask กับตัวแปล im1
- บรรทัดที่4 เป็นการใช้คำสั่ง imwrite เพื่อเขียนภาพและบันทึกภาพขึ้นมาใหม่ที่เป็นระบบสี HSV
- บรรทัดที่5 อ่านข้อมูลของภาพที่มีชื่อว่า hsvroad12 และเก็บอยู่ในรูปตัวแปรที่มีชื่อว่า im2
- บรรทัดที่6 คำสั่งแสดงข้อมูลของค่าสีของแต่ละพิกเซล ของตัวแปร im2



Before After

รูปที่ 2.13 แสดงความแตกต่างของค่า hue ก่อนและหลัง แปลงระบบสี

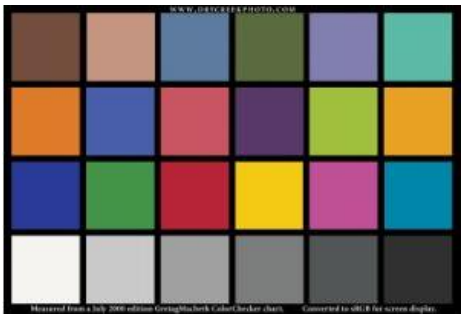
จากรูปที่ 2.13 แสดงความแตกต่างของค่า hue ของภาพก่อนแปลงและหลังแปลงเป็น HSV ณ ตำแหน่งเดียวกันของภาพ

2.4.2 การลดสัญญาณลบกวนของภาพ

2.4.2.1 สัญญาณรบกวนของภาพ (Noise)

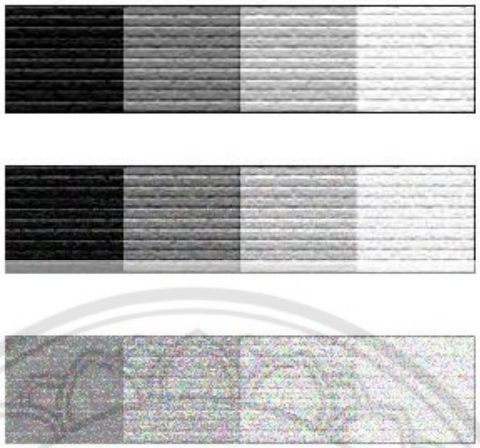
Noise คือสัญญาณรบกวนที่เกิดขึ้นในภาพ ทำให้เห็นความไม่สม่ำเสมอของความสว่างและสี ซึ่งบ่งบอกถึงคุณภาพของกล้องและระบบการเกิดภาพแสดงดังรูปที่ 2.14

รูปที่ 2.14 รูปภาพที่มี Noise



รูปที่ 2.15 รูปภาพที่ไม่มี Noise

ระดับการเกิดสัญญาณรบกวนที่มากขึ้นจะทำให้รายละเอียดของภาพหายไป ระดับสัญญาณรบกวนที่เท่ากันแต่จะมองเห็น grain ต่างกัน ขึ้นอยู่กับระดับของสัญญาณภาพส่วนสว่างและส่วนเงา ยิ่งภาพมีความสว่างน้อยก็จะยิ่งเห็นสัญญาณรบกวนชัดขึ้นดังแสดงในรูปที่ 2.16

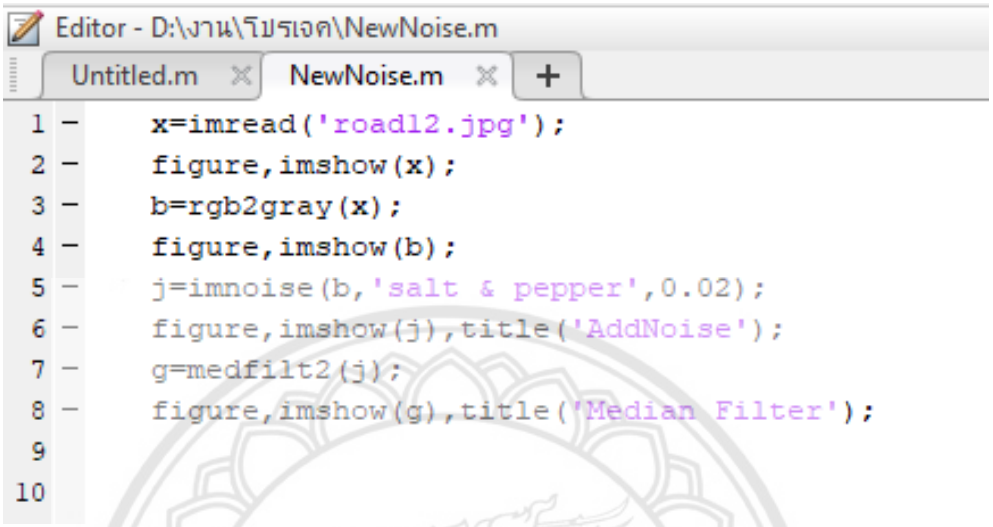


รูปที่ 2.16 ระดับการเกิดสัญญาณรบกวน

2.4.2.2 การลดสัญญาณรบกวนด้วยการใช้การกรองค่ามัธยฐาน (Median Filtering)

การกรองภาพแบบค่ามัธยฐาน (Median Filtering) จะเป็นการกำจัดสัญญาณรบกวน หรือที่เรียกกันว่า Noise โดยที่มีความสามารถในการลดความเบลอของภาพได้ดีกว่าการกรองภาพแบบค่าเฉลี่ย (Mean Filtering) เพราะการกรองภาพแบบ Median Filter จะไม่ลดความคมชัดของภาพ ทำให้ภาพไม่เบลอ

2.4.2.3 การลดสัญญาณรบกวนด้วยวิธี Median Filter โดยใช้โปรแกรม MATLAB



รูปที่ 2.17 ฟังก์ชันที่ใช้เพิ่มสัญญาณรบกวนและลดสัญญาณรบกวน

จากรูปที่ 2.17 เป็นการแสดงขั้นตอนการลดสัญญาณรบกวนของภาพ โดยมีขั้นตอนดังนี้

- **บรรทัดที่ 1** เป็นคำสั่งให้โปรแกรมทำการอ่านรูปภาพที่มีชื่อว่า road12.jpg โดยกำหนดให้เป็นตัวแปร x
- **บรรทัดที่ 2** เป็นการสั่งให้โปรแกรมทำการแสดงรูปภาพจากตัวแปร x ขึ้นมา
- **บรรทัดที่ 3** เป็นคำสั่งให้โปรแกรมทำการแปลงรูป road12.jpg ให้กลายเป็นรูปสี ขาว เทา ดำ โดยกำหนดให้เป็นตัวแปร b
- **บรรทัดที่ 4** เป็นการสั่งให้โปรแกรมทำการแสดงรูปภาพจากตัวแปร b ขึ้นมา
- **บรรทัดที่ 5** เป็นคำสั่งให้โปรแกรมเพิ่มสัญญาณรบกวนลงไปในภาพ โดยใช้ภาพจากตัวแปร b และทำการเพิ่มสัญญาณรบกวนประเภท salt & pepper กำหนดค่าสัญญาณรบกวนให้มีค่าเท่ากับ 0.02 (ค่าสัญญาณรบกวนมีค่าอยู่ระหว่าง 0 – 1) ยิ่งค่าสัญญาณรบกวนมีค่ามากรูปภาพที่ได้ก็จะมีสัญญาณรบกวนมาก
- **บรรทัดที่ 6** เป็นการสั่งให้โปรแกรมแสดงรูปภาพจากตัวแปร j ขึ้นมา โดยตั้งชื่อหัวข้อว่า Add Noise

- **บรรทัดที่ 7** เป็นคำสั่งลดสัญญาณรบกวนแบบ Median Filter โดยกำหนดให้เป็นตัวแปร g
- **บรรทัดที่ 8** เป็นการสั่งให้โปรแกรมแสดงรูปภาพจากตัวแปร g ขึ้นมา โดยตั้งชื่อหัวข้อว่า Median Filter

2.4.3 เกณฑ์ความกว้าง (Magnitude threshold)

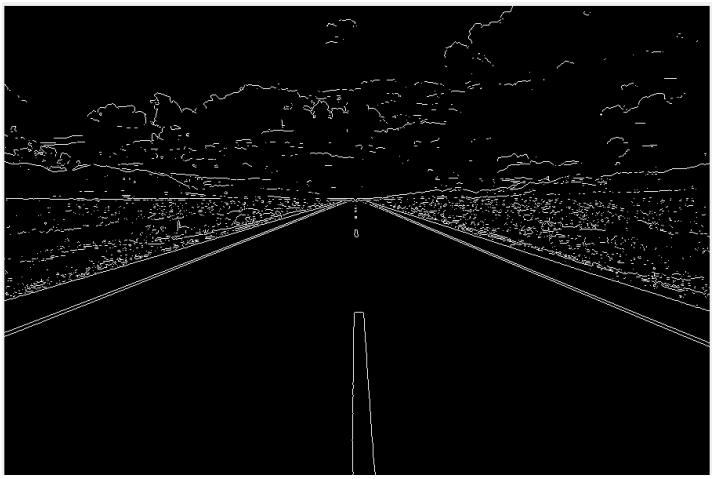
ในขั้นตอนการตรวจหาขอบของเส้นแบ่งช่องทางเดินรถ และวัตถุภายในภาพ โดยใช้กระบวนการ Sobel (เพื่อหาความแตกต่างกันของสี)

```
1 -    im1 = imread('road12.jpg');
2 -    im2 = im1(:,:,2);
3 -    im3=edge(im2,'sobel');
```

รูปที่ 2.18 ตัวอย่างการใช้กระบวนการ sobel

จากรูปที่ 2.18 แสดงรายระเอียดของการใช้กระบวนการ sobel

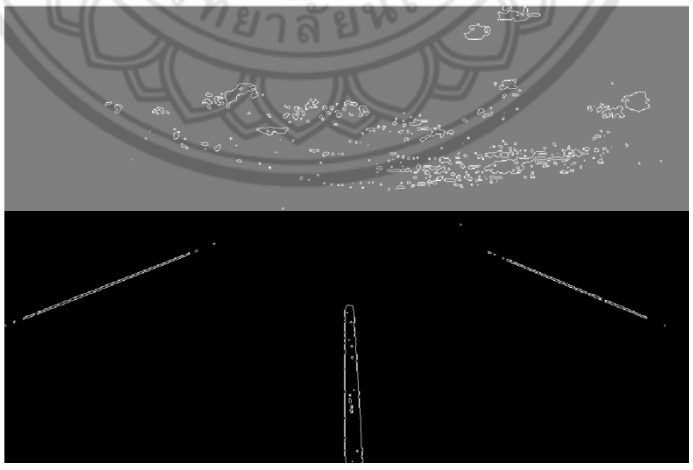
- บรรทัดที่1 อ่านข้อมูลของภาพที่มีชื่อว่า road12 และเก็บอยู่ในรูปตัวแปรที่มีชื่อว่า im1
- บรรทัดที่2 แปลงภาพให้เป็นภาพสีเทา
- บรรทัดที่3 เป็นการใช้คำสั่ง edge และเลือกใช้กระบวนการ sobel เพื่อหาขอบของวัตถุภายในภาพ



รูปที่ 2.19 ภาพที่ได้ออกมาหลังจากผ่านกระบวนการ sobel

จากรูปที่ 2.19 ภาพที่ผ่านกระบวนการ sobel แล้วนั้นจะแสดงขอบของวัตถุทั้งหมดภายในภาพรวมไปถึงขอบของเส้นแบ่งช่องทางเดินรถ

เรารวมเกณฑ์ที่ทั้ง 3 ก่อนหน้านี้นั้น คือ กำหนดเกณฑ์สี การลดสัญญาณลบกวนของภาพ เกณฑ์ความกว้าง เพื่อลดวัตถุภายในภาพให้เหลือน้อยที่สุด แต่ยังคงสภาพเส้นของเส้นแบ่งช่องทางเดินรถ ไว้ให้มากที่สุด และหาขอบของเส้นแบ่งช่องทางเดินรถ นั้น เพื่อที่จะนำไปใช้ในกระบวนการถัดไป



รูปที่ 2.20 ภาพที่ผ่านกระบวนการข้างต้นมาแล้ว

จากรูปที่ 2.20 แสดงขอบของเส้นแบ่งช่องทางเดินรถ และขอบของวัตถุที่เหลืออยู่ในภาพ

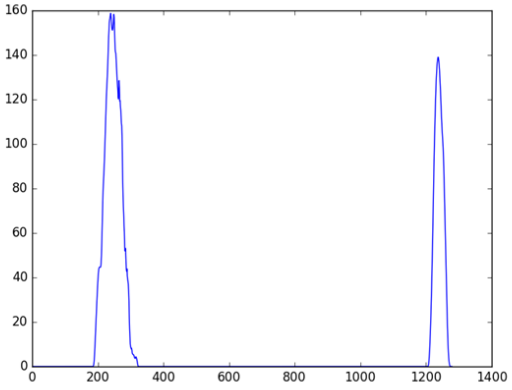
2.5 การแปลงมุมมอง (Perspective transform)

จากภาพที่ได้จนมาถึงขณะนี้จะเห็นได้ว่าเมื่อมองในภาพระยะยิ่งไกลขึ้นความกว้างของช่องทางเดินรถ ก็จะดูแคบลง แต่ในความเป็นจริงความกว้างของช่องทางเดินรถ นั้นมีระยะเท่าเดิม ดังนั้นจึงต้องทำการเปลี่ยนมุมมองของภาพให้เหมือนมองจากด้านบนลงมา โดยการเปลี่ยนแปลงพิกัดบนภาพ ผลที่ได้ออกมาแสดงดังรูปที่ 2.21

รูปที่ 2.21 ภาพที่ได้ออกมาหลังจากแปลงมุมมอง

2.6 การตรวจจับเลน (Lane detection)

คำนวณฮิสโตแกรมของภาพครึ่งล่างเพื่อหาตำแหน่งคร่าวๆของเส้นแบ่งช่องทางเดินรถแต่ละเส้น



รูปที่ 2.22 ภาพแสดงพิกัดของเส้นแบ่งช่องทางเดินรถ

แสดงพิกัดของเส้นแบ่งช่องทางเดินรถของแต่ละเส้นจากรูปที่ 2.22

จากนั้นหาตำแหน่งของจุดกึ่งกลางของเส้นแบ่งช่องทางเดินรถ ในแต่ละเส้น หลังจากนั้นใช้ตำแหน่งเหล่านี้เพื่อคำนวณ polylines ของแต่ละช่องทางเดินรถ โดยใช้หลักการ linear regression กับกลุ่มจุดกึ่งกลางของทั้ง 2 กลุ่มและจะแสดงผลเป็นเส้นแบ่งช่องทางเดินรถ ใหม่ออกมาตามรูปที่ 2.23

รูปที่ 2.23 ภาพแสดงจุดกึ่งกลางที่หาได้

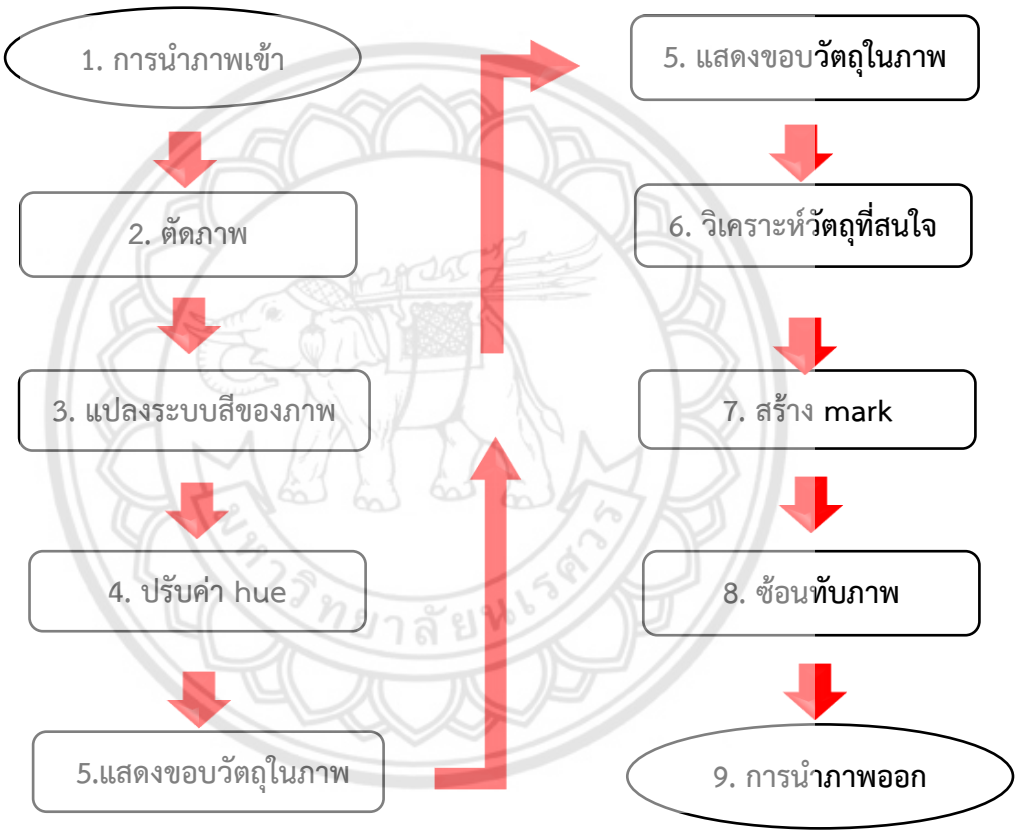
2.7 แสดงเลนที่ตรวจพบ (Displaying the detected lane)

นำผลลัพธ์ที่ได้มาซ้อนทับกับภาพหรือวีดีโอก็จะได้ผลลัพธ์ดังรูปที่ 2.24

รูปที่ 2.24 ภาพแสดงผลลัพธ์ที่ออกมาได้

บทที่ 3
วิธีการดำเนินงาน

จากขั้นตอนต่างๆที่กล่าวมาจากบทที่ 3 สามารถสร้างเป็นแผนภาพการดำเนินงานดังแสดงใน รูปที่ 3.1



รูปที่ 3.1 แผนภาพการดำเนินงาน

1. การนำภาพเข้า ภาพที่นำเข้ามาเป็นระบบสี RGB
2. ตัดภาพ เลือกพื้นที่ๆสนใจ
3. แปลงระบบสีของภาพ แปลงภาพจากระบบสี RGB เป็นระบบสี HSV
4. ปรับเปลี่ยนค่า Hue Saturation และValue ให้ภาพนั้นเหลือเพียงสีที่สนใจ ส่วนสีอื่นๆทำให้เป็นสีดำทั้งหมด

5. แสดงขอบของวัตถุในภาพ ทำเพื่อแสดงขอบของวัตถุที่สนใจ นั้นก็คือเส้นแบ่งช่องทางเดินรถ
6. วิเคราะห์วัตถุที่สนใจ เป็นการวิเคราะห์วัตถุที่สนใจโดยวัตถุที่สน ใจที่นี้คือขอบของเส้นแบ่งช่องทางเดินรถ
7. สร้าง mark เพื่อแสดงให้เห็นตำแหน่งของเส้นเลนส์ที่ค้นหาได้
8. ซ้อนทับภาพ เป็นกระบวนการสุดท้าย คือการนำภาพที่ผ่านกระบวนการขั้นต้นมาทั้งหมดมาซ้อนทับบนภาพต้นฉบับ
9. นำภาพออกมาแสดงผลภาพที่ผ่านกระบวนการทั้งหมดมาแล้ว

3.1 สมมติฐานในการดำเนินโครงงาน

จากการเก็บรูปภาพของเส้นทางเดินรถในมหาวิทยาลัยนเรศวรพบว่าถนนของมหาวิทยาลัยนเรศวรมีสีที่ขาดหายและไม่ชัดเจน ทำให้ไม่สามารถนำมาใช้เป็นข้อมูลในการประมวลผลภาพได้ ทางผู้ดำเนินโครงงานจึงได้จัดทำแบบจำลองช่องทางเดินรถขึ้นมาเพื่อใช้ในการดำเนินงาน ดังแสดงในรูปที่ 3.4

อัตราความเร็ว (กม/ชม.)	รถนั่งทั่วไป ระยะคิด + ระยะเบรก = ระยะหยุดรถ			รถบรรทุกขนาดใหญ่ ระยะคิด + ระยะเบรก = ระยะหยุดรถ		
20	4	3	7	14	9	23
30	6	6	12	21	17	38
40	8	10	18	28	27	55
50	10	15	25	35	38	73
60	12	22	34	42	55	97
70	14	29	43	49	74	123
80	16	38	54	56	102	158
90	18	48	66	63	122	185
100	20	60	80	69	145	214

ตารางที่ 3.1 แสดงระยะเบรกที่ปลอดภัย ที่มา : www.motortravel.in.th [5]

จากตารางที่ 3.1 จาก www.motortravel.in.th [5] แสดงระยะเบรกที่ปลอดภัยตามความเร็วของรถยนต์ โดยระยะคิดและระยะเบรกของรถบรรทุกจะมีค่ามากกว่ารถทั่วไปประมาณ 3 เท่า เพราะรถบรรทุกมีขนาดที่ใหญ่กว่า มีน้ำหนักที่มากกว่าจึงต้องมีระยะคิดและระยะเบรกที่มากขึ้น ถ้าในกรณีที่มีการเบรกกะทันหันอาจจะทำให้เบรกไม่อยู่ ทำให้เสียการควบคุมของรถและมีโอกาสเกิดการพลิกคว่ำ ซึ่งต่างกับรถทั่วไปที่มีขนาดเล็กและน้ำหนักที่ไม่มากจึงทำให้ระยะคิดและระยะเบรกมีระยะทางที่สั้นกว่า ในโครงงานนี้เลือกใช้ความเร็วรถทั่วไปที่ 40 กิโลเมตรต่อชั่วโมง ซึ่งจะได้ระยะเบรกปลอดภัยที่ 10 เมตร เมื่อบวกกับระยะคิด 8 เมตร จะได้ระยะหยุดรถที่ 18 เมตร ในการดำเนินโครงงานนี้เลือกใช้กล้อง LOGITECH C930E[6] ระยะเบรกที่ปลอดภัยอ้างอิงดังแสดงในรูปที่ 3.2

รูปที่ 3.2 ภาพอ้างอิงระยะเบรกที่ปลอดภัย

จากรูปที่ 3.2 เลือกใช้ความละเอียดที่ 800 x 600 เมื่อทำการติดตั้งกล้องกับรถเก๋งรุ่น Toyota corolla ณ ตำแหน่งหลังกระจกมองหลังตรงกลางหน้ารถ ที่ระยะเบรก 10 เมตร ตรงกับพิกัดแกน X ที่ 200 จากข้อมูลนี้จะเป็นกรอบของภาพที่จะใช้งานกับแบบจำลอง

3.2 นำกระบวนการที่เลือกใช้เขียนลง Simulink

กระบวนการประมวลผลภาพ (image processing) ที่นำมาใช้ มีดังนี้

3.2.1สนใจเฉพาะที่เส้นแบ่งช่องทางการเดินรถสีเหลือง

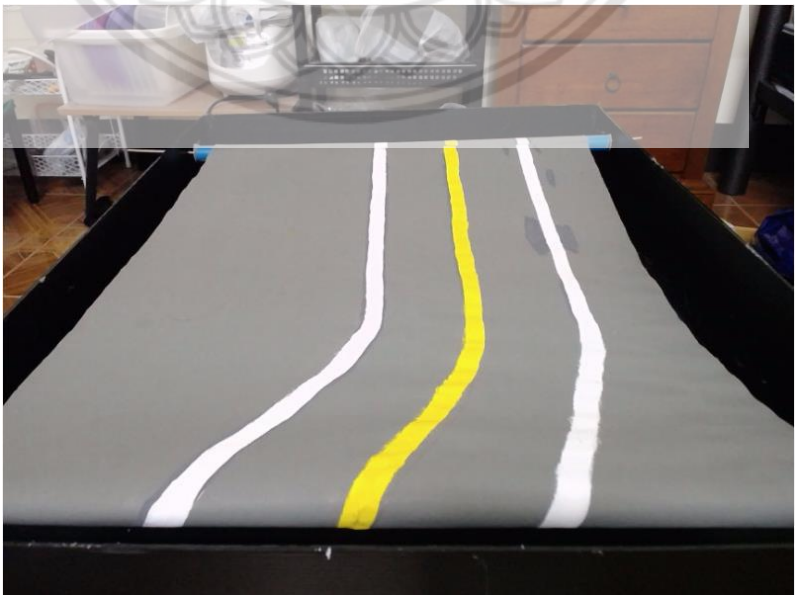
ทำตามกระบวนการดังนี้

3.2.1.1 การตัดภาพ

เลือกพื้นที่ ๆสนใจโดยเลือกระยะของแกน X ที่ 180 ถึง 200 ระยะของแกน Y ที่ 200 ถึง 500 โดยที่แกน X ใช้ระยะตรงกับระยะห่างอ้างอิงจากรูปที่ 3.2 ที่ตำแหน่ง 200 ของภาพ

```
1    function y = fcn(in1)
2 -  B=in1(180:200,200:500,1:3);
3 -  y = B;
```

รูปที่ 3.3 ฟังก์ชันที่ใช้ในการตัดภาพ



รูปที่ 3.4 ภาพต้นฉบับ

รูปที่ 3.5 ภาพที่ตัดออกมา

3.2.1.2 แปลงระบบสีของภาพ

แปลงภาพจากระบบสี RGB เป็นระบบสี HSV (Hue Saturation Value) โดยเมื่อแปลงเป็นระบบสี HSV สามารถปรับค่า Hue Saturation Value ได้

```
1   function y = fcn(RGB,I)
2   % Define thresholds for channel 1 based on histogram settings
3   channel1Min = 0.000;
4   channel1Max = 1.000;
5
6   % Define thresholds for channel 2 based on histogram settings
7   channel2Min = 0.000;
8   channel2Max = 1.000;
9
10  % Define thresholds for channel 3 based on histogram settings
11  channel3Min = 0.000;
12  channel3Max = 1.000;
```

Channel1

Channel2

Channel3

รูปที่ 3.6 ฟังก์ชันที่ใช้แปลงระบบสี

จากรูปที่ 3.6 เป็นการแปลงระบบสีเป็น HSV โดยอ้างอิงจากหัวข้อที่ 2.4.1 หน้า 11 แสดงรายละเอียดของ Channel ดังนี้

Channel1 ค่าสีของสีหลัก (Hue)

Channel2 ค่าความเข้มของสี (Saturation)

Channel3 ค่าความมืดและสว่างของสี (Value)

3.2.1.3 ปรับค่า hue

ปรับเปลี่ยนค่า Hue Saturation และ Value ให้ภาพนั้นเหลือเพียงสีที่สนใจ สีที่สนใจคือสีของเส้นเลนที่เป็นสีเหลือง ส่วนสีอื่นๆทำให้เป็นสีดำทั้งหมด

```
function y = fcn(RGB,I)
% Define thresholds for channel 1 based on histogram settings
channel1Min = 0.104;
channel1Max = 0.167;

% Define thresholds for channel 2 based on histogram settings
channel2Min = 0.196;
channel2Max = 1.000;

% Define thresholds for channel 3 based on histogram settings
channel3Min = 0.937;
channel3Max = 1.000;
```

Channel1

Channel2

Channel3

รูปที่ 3.7 ฟังก์ชันที่ใช้ปรับค่า Hue Saturation และValue

จากรูปที่ 3.7 เป็นการปรับค่าของ Channel ต่างๆดังนี้

Channel1 เป็นการปรับค่าสีของสีหลัก (Hue) น้อยสุดอยู่ที่ 0.104 มากสุดอยู่ที่ 0.167

Channel2 เป็นการปรับค่าความเข้มของสี (Saturation) น้อยสุดอยู่ที่ 0.196 มากสุดอยู่ที่ 1.000

Channel3 เป็นการปรับค่าความมืดและสว่างของสี (Value) น้อยสุดอยู่ที่ 0.937 มากสุดอยู่ที่ 1.000

รูปที่ 3.8 ภาพที่ได้ออกมาหลังจากผ่านฟังก์ชัน

จากรูปที่ 3.8 เป็นภาพที่ได้ออกมา เห็นได้ว่าสามารถทำให้สีที่ไม่สนใจในภาพกลายเป็นสีดำได้ และเหลือเพียงเฉพาะสีเหลือง ที่ต้องการ

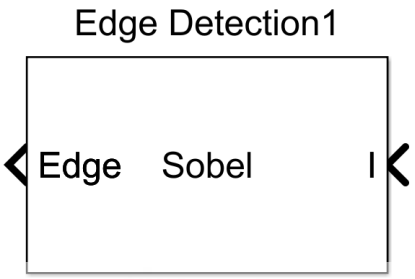
3.2.1.4 แสดงขอบของวัตถุในภาพ

กระบวนการนี้ทำเพื่อแสดงขอบของวัตถุที่สนใจ นั้นก็คือเลนส์ของถนน การแสดงขอบนั้นจะแสดงขอบเขตของวัตถุนั้นได้ชัดเจนมากขึ้น

```
function y = fcn(maskedRGBImage)
td=maskedRGBImage(:,:,2);
y = td;
```

รูปที่ 3.9 เขียนฟังก์ชันเปลี่ยนภาพให้เป็น ภาพขาวดำ

จากรูปที่ 3.9 เปลี่ยนภาพให้เป็นภาพขาวดำโดยคำสั่งที่สำคัญคือคำสั่งในบรรทัดที่ 2 ที่จะนำภาพที่แสดงเพียงค่าของ ความเข้มของสี และส่งไปยัง toolbox Edge Detection ดังรูปที่ 3.10 เพื่อเข้ากระบวนการ Sobel



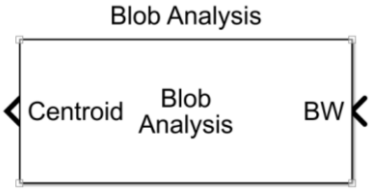
รูปที่ 3.10 Tool Box Edge Detection

รูปที่ 3.11 ภาพที่ได้หลังจากผ่าน Tool Box Edge Detection

จากรูปที่ 3.11 เป็นภาพที่ได้หลังผ่านกระบวนการ Sobel ภาพนี้จะเหลือเพียงขอบของเส้นแบ่งช่องทางการเดินรถสีเหลืองเท่านั้น

3.2.1.5 วิเคราะห์วัตถุที่สนใจ

กระบวนการนี้ใช้ Toolbox Blob analysis เป็นการวิเคราะห์วัตถุที่สนใจโดยวัตถุที่สน ใจที่นี้คือขอบของเส้นเลนส์ถนนที่ผ่านตัวของ toolbox Edge Detection



รูปที่ 3.12 Toolbox Blob analysis

3.2.2 สนใจเฉพาะที่เส้นแบ่งช่องทางการเดินรถสีขาว

ทำตามกระบวนการดังนี้

3.2.2.1 การตัดภาพ

เลือกพื้นที่ ๆสนใจโดยเลือกระยะของแกน X ที่ 180 ถึง 200 ระยะของแกน Y ที่ 200 ถึง 400 เพื่อให้เห็นเฉพาะเส้นแบ่งช่องทางเดินรถสีขาว โดยที่แกน X ใช้ระยะตรงกับระยะห่างอ้างอิงจากรูปที่ 35 ที่ตำแหน่ง 200 ของภาพ

```
1    function y = fcn(in2)
2 -  Z=in2(180:200,200:400,1:3);
3 -  y = Z;
```

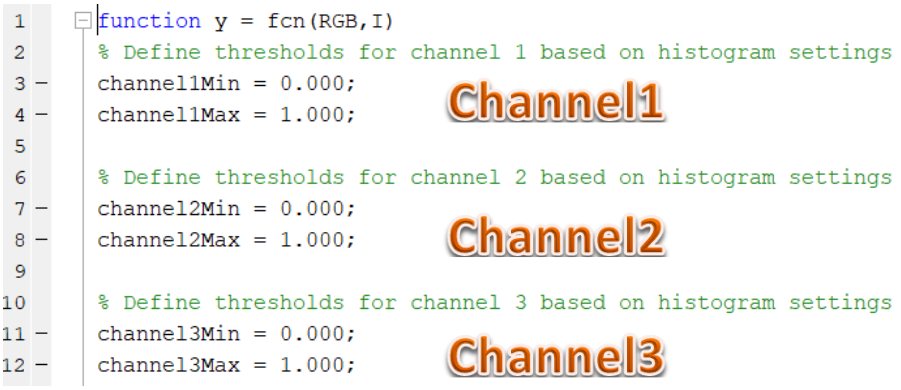
รูปที่ 3.13 ฟังก์ชันที่ใช้ในการตัดภาพ

จากรูปที่ 3.4 ตัดภาพออกมาได้ดัง รูปที่ 3.14

รูปที่ 3.14 ภาพที่ตัดออกมา

3.2.2.2 แปลงระบบสีของภาพ

แปลงภาพจากระบบสี RGB เป็นระบบสี HSV (Hue Saturation Value) โดยเมื่อแปลงเป็นระบบสี HSV สามารถปรับค่า Hue Saturation Value ได้



รูปที่ 3.15 ฟังก์ชันที่ใช้แปลงระบบสี

จากรูปที่ 3.15 เป็นการแปลงระบบสีเป็น HSV โดยอ้างอิงจากหัวข้อที่ 2.4.1 หน้า 11 แสดงรายละเอียดของ Channel ดังนี้

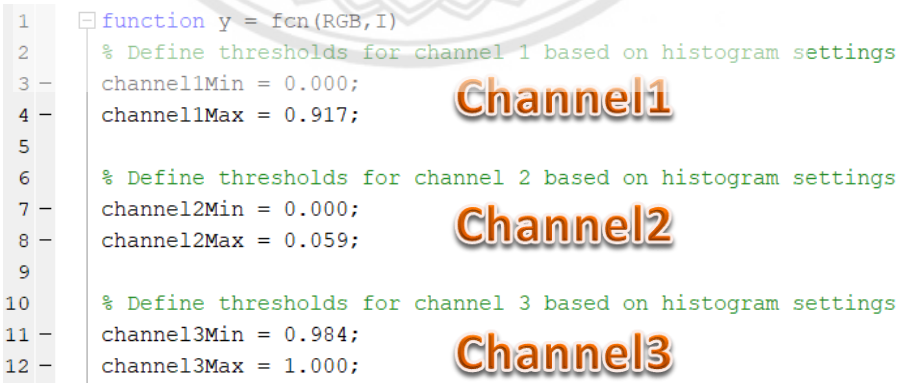
Channel1 ค่าสีของสีหลัก (Hue)

Channel2 ค่าความเข้มของสี (Saturation)

Channel3 ค่าความมืดและสว่างของสี (Value)

3.2.2.3 ปรับค่า hue

ปรับเปลี่ยนค่า Hue Saturation และValue ให้ภาพนั้นเหลือเพียงสีที่สนใจ สีที่สนใจคือสีของเส้นเลนที่เป็นสีขาว ส่วนสีอื่นๆทำให้เป็นสีดำทั้งหมด



รูปที่ 3.16 ฟังก์ชันที่ใช้ปรับค่า Hue Saturation และValue

จากรูปที่ 3.16 เป็นการปรับค่าของ Channel ต่างๆดังนี้

Channel1 เป็นการปรับค่าสีของสีหลัก (Hue) น้อยสุดอยู่ที่ 0.000 มากสุดอยู่ที่ 0.917

Channel2 เป็นการปรับค่าความเข้มของสี (Saturation) น้อยสุดอยู่ที่ 0.000 มากสุดอยู่ที่ 0.059

Channel3 เป็นการปรับค่าความมืดและสว่างของสี (Value) น้อยสุดอยู่ที่ 0.984 มากสุดอยู่ที่ 1.000

รูปที่ 3.17 ภาพที่ได้ออกมาหลังจากผ่านฟังก์ชัน

จากรูปที่ 3.17 เป็นภาพที่ได้ออกมา เห็นได้ว่าสามารถทำให้สีที่ไม่สนใจในภาพกลายเป็นสีดำได้ และเหลือเพียงเฉพาะสีขาว ที่ต้องการ

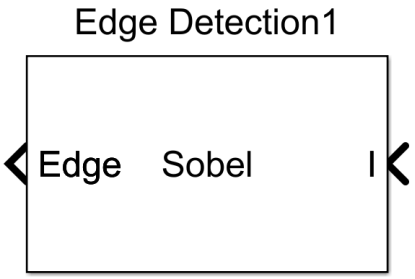
3.2.2.4 แสดงขอบของวัตถุในภาพ

กระบวนการนี้ทำเพื่อแสดงขอบของวัตถุที่สนใจ นั้นก็คือเลนส์ของถนน การแสดงขอบนั้นจะแสดงขอบเขตของวัตถุนั้นได้ชัดเจนมากขึ้น

```
1     function y = fcn(maskedRGBImage)
2 -     td=maskedRGBImage(:,:,2);
3 -     y = td;
```

รูปที่ 3.18 เขียนฟังก์ชันเปลี่ยนภาพให้เป็น ภาพขาวดำ

จากรูปที่ 3.18 เปลี่ยนภาพให้เป็นภาพขาวดำโดยคำสั่งที่สำคัญคือคำสั่งในบรรทัดที่ 2 ที่จะนำภาพที่แสดงเพียงค่าของ ความเข้มของสี และส่งไปยัง toolbox Edge Detection ดังรูปที่ 3.19 เพื่อเข้ากระบวนการ Sobel



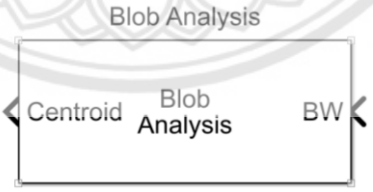
รูปที่ 3.19 Toolbox Edge Detection

รูปที่ 3.20 ภาพที่ได้หลังจากผ่าน toolbox Edge Detection

จากรูปที่ 3.20 เป็นภาพที่ได้หลังผ่านกระบวนการ Sobel ภาพนี้จะเหลือเพียงขอบของเส้นแบ่งช่องทางการเดินรถสีขาวเท่านั้น

3.2.2.5 วิเคราะห์วัตถุที่สนใจ

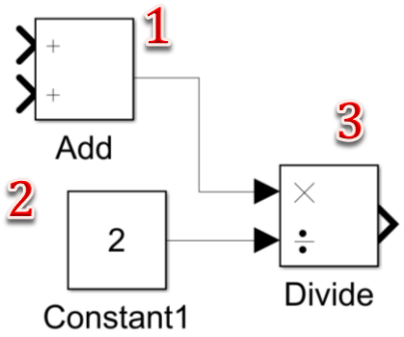
กระบวนการนี้ใช้ Toolbox Blob analysis เป็นการวิเคราะห์วัตถุที่สนใจโดยวัตถุที่สนใจที่นี้คือขอบของเส้นเลนส์ถนนที่ผ่านตัวของ toolbox Edge Detection



รูปที่ 3.21 Toolbox Blob analysis

3.2.3 หาจุดCentroid

หาจุด Centroid ระหว่าง Centroid ของเส้นแบ่งช่องทางเดินรถสีเหลืองกับสีขาว เพื่อเป็นตำแหน่งในการวาด Mark



รูปที่ 3.22 ชุด Toolbox ในการหาจุด Centroid

จากรูปที่ 3.22

1. Toolbox Add เป็นตัวรับตำแหน่งของ Centroid ของเส้นแบ่งช่องทางเดินรถสีขาวและสีเหลือง
2. Toolbox Constant เป็น Toolbox สำหรับใส่ค่าคงที่
3. Toolbox Divide รับตัวแปรและนำมาหารกับค่าที่กำหนดไว้

เมื่อข้อมูลของภาพออกมาจากรูปที่ 4.12 และรูปที่ 4.21 จะเป็นข้อมูลของจุด Centroid ของเส้นแบ่งช่องทางเดินรถสีเหลืองกับสีขาว และผ่าน Toolbox ในรูปที่ 4.22 ข้อมูลที่ออกมาจะเป็นตำแหน่งจุด ระหว่าง Centroid ของเส้นแบ่งช่องทางเดินรถสีขาวและสีเหลือง

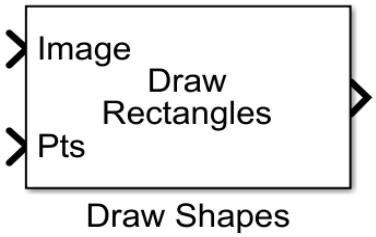
3.2.4 สร้าง mark

สร้าง mark เพื่อแสดงให้เห็นตำแหน่งของเส้นเลนส์ที่ค้นหาได้

```
1   function x = fcn(Pts,width,height)
2
3 -   x = [uint32(Pts(1,1)-10),uint32(Pts(1,2)-10),uint32(width),uint32(height)];
```

รูปที่ 3.23 เขียนฟังก์ชันกำหนดความกว้างและความยาวของ mark

Mark ที่ใช้เป็นรูปสี่เหลี่ยมจากรูปที่ 3.23 เป็นฟังก์ชันกำหนดความกว้างและความยาวของ mark จากนั้นส่งไปยัง Toolbox Draw Shapes ในรูปที่ 3.24 เพื่อวาด Mark ออกมา



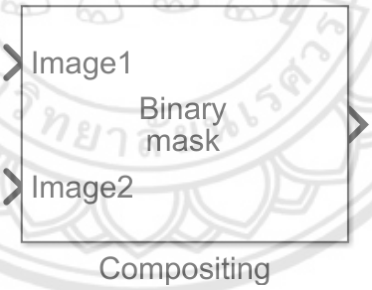
รูปที่ 3.24 Toolbox Draw Shapes

รูปที่ 3.25 ภาพที่ได้หลังจากผ่าน Toolbox Draw Shapes

จากรูปที่ 3.25 แสดง mark ที่วาดออกมาอยู่ระหว่างเส้นแบ่งช่องทางเดินรถสีเหลืองกับสีขาว

3.2.5 ซ้อนทับภาพ

เป็นกระบวนการสุดท้าย คือการนำภาพที่ผ่านกระบวนการขั้นต้นมาทั้งหมดตั้งแต่ 3.2.1 ถึง 3.2.2 แล้วมาซ้อนทับบนภาพต้นฉบับรูปที่ 3.4 โดยใช้ Toolbox Compositing

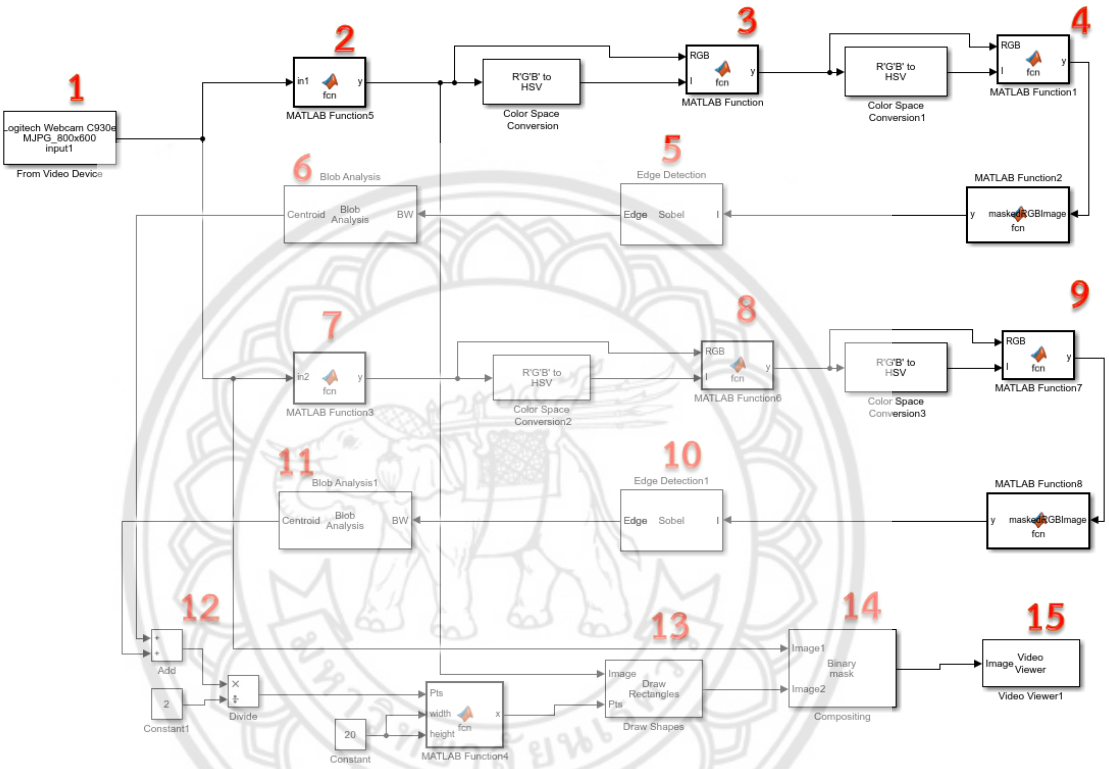


รูปที่ 3.26 Toolbox Compositing

รูปที่ 3.27 ภาพที่ได้หลังจากผ่าน Toolbox Compositing

จากรูปที่ 3.27 แสดง mark สี่เหลี่ยมสีแดงอยู่ระหว่างเส้นแบ่งช่องทางเดินรถสีเหลืองและสีขาว

สรุป จากกระบวนการตั้งแต่ 3.2.1 ถึง 3.2.5 สามารถสร้าง model ได้ดังรูปที่ 3.28



รูปที่ 3.28 แผนภาพกระบวนการตั้งแต่ 3.2.1 ถึง 3.2.5

การรวมกระบวนการทั้งหมดเป็นไปตามลำดับขั้นตอนตามรูปที่ 3.28

1. **Inputภาพ เข้ามา**

ภาพที่นำเข้ามาเป็นระบบสี RGB ขนาด 800×600

2. ตัดภาพ

เลือกพื้นที่ ๆสนใจเลือกระยะของแกน X ที่ 150 ถึง 200 ระยะของแกน Y ที่ 200 ถึง 500 โดยที่แกน X ใช้ระยะตรงกับระยะห่างอ้างอิงที่ตำแหน่ง 200 ของภาพ ภาพที่ได้ยังคงเป็นระบบสี RGB

3. แปลงระบบสีของภาพ

แปลงภาพจากระบบสี RGB เป็นระบบสี HSV (Hue Saturation Value) โดยเมื่อแปลงเป็นระบบสี HSV สามารถปรับค่า Hue Saturation Value ได้

4. ปรับเปลี่ยนค่า Hue Saturation และValue

ปรับเปลี่ยนค่า Hue Saturation และ Value ให้ภาพนั้นเหลือเพียงสีที่สนใจ สีที่สนใจคือสีของเส้นแบ่งช่องทางเดินรถสีเหลือง ส่วนสีอื่นๆทำให้เป็นสีดำทั้งหมด

5. แสดงขอบของวัตถุในภาพ

กระบวนการนี้ทำเพื่อแสดงขอบของวัตถุที่สนใจ นั้นก็คือเส้นแบ่งช่องทางเดินรถสีเหลือง การแสดงขอบนั้นจะแสดงขอบเขตของวัตถุนั้นได้ชัดเจนมากขึ้น ภาพที่ได้ออกมาเป็นภาพขาวดำ

6. วิเคราะห์วัตถุที่สนใจ

เป็นการวิเคราะห์วัตถุที่สนใจโดยวัตถุที่สนใจที่นี้คือขอบของเส้นแบ่งช่องทางเดินรถที่ผ่าน toolbox Edge Detection

7. ตัดภาพ

เลือกพื้นที่ ๆสนใจเลือกระยะของแกน X ที่150 ถึง 200 ระยะของแกน Y ที่ 200 ถึง 400 โดยที่แกน X ใช้ระยะตรงกับระยะห่างอ้างอิงที่ตำแหน่ง 200 ของภาพ ภาพที่ได้ยังคงเป็นระบบสี RGB

8. แปลงระบบสีของภาพ

แปลงภาพจากระบบสี RGB เป็นระบบสี HSV (Hue Saturation Value) โดยเมื่อแปลงเป็นระบบสี HSV สามารถปรับค่า Hue Saturation Value ได้

9. ปรับเปลี่ยนค่า Hue Saturation และValue

ปรับเปลี่ยนค่า Hue Saturation และ Value ให้ภาพนั้นเหลือเพียงสีที่สนใจ สีที่สนใจคือสีของเส้นแบ่งช่องทางเดินรถสีขาว ส่วนสีอื่นๆทำให้เป็นสีดำทั้งหมด

10. แสดงขอบของวัตถุในภาพ

กระบวนการนี้ทำเพื่อแสดงขอบของวัตถุที่สนใจ นั้นก็คือเส้นแบ่งช่องทางเดินรถสีขาว การแสดงขอบนั้นจะแสดงขอบเขตของวัตถุนั้นได้ชัดเจนมากขึ้น ภาพที่ได้ออกมาเป็นภาพขาวดำ

11. วิเคราะห์วัตถุที่สนใจ

เป็นการวิเคราะห์วัตถุที่สนใจโดยวัตถุที่สนใจที่นี้คือขอบของเส้นแบ่งช่องทางเดินรถที่ผ่าน toolbox Edge Detection

12. หาจุดCentroid

หาจุด Centroid ระหว่าง Centroid ของเส้นแบ่งช่องทางเดินรถสีเหลืองกับสีขาว เพื่อเป็นตำแหน่งในการวาด Mark

13. สร้าง mark

สร้าง mark เพื่อแสดงให้เห็นตำแหน่งของเส้นเลนส์ที่ค้นหาได้

14. ซ้อนทับภาพ

เป็นกระบวนการสุดท้าย คือการนำภาพที่ผ่านกระบวนการขั้นต้นมาทั้งหมดตั้งแต่ 2 ถึง 13 แล้วมาซ้อนทับบนภาพต้นฉบับ โดยใช้ Toolbox Compositing

15. นำภาพออกมา

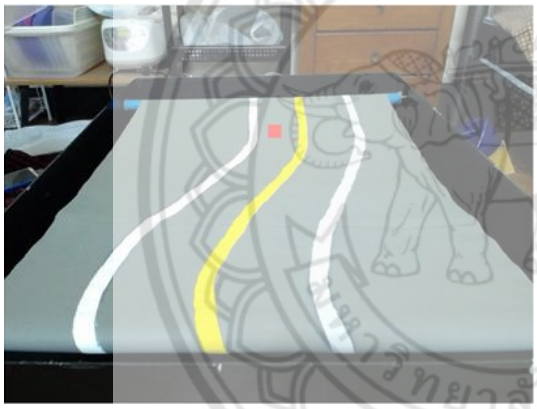
แสดงผลภาพตามลำดับขั้นตอนตามรูปที่ 3.28

บทที่ 4
ผลการทดลองและวิเคราะห์ผลการทดลอง

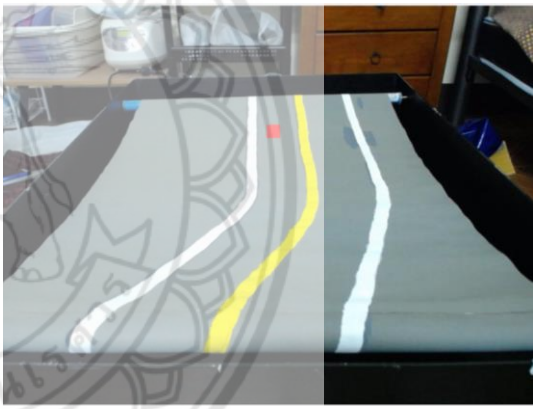
4.1 ผลการทดลอง

การทดลองโดยอ้างอิงระยะเบรกจากรูปที่ 3.2 ที่ระยะ 10 เมตร ในระดับความเข้มแสงต่างกัน

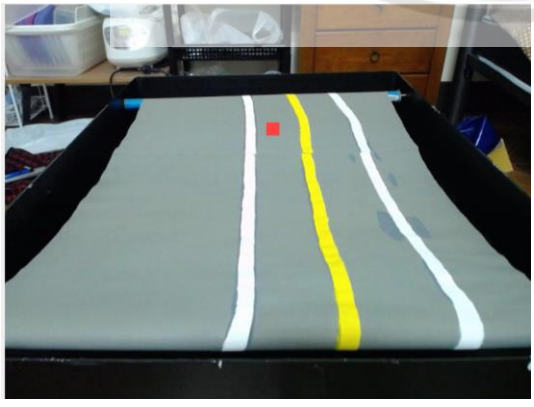
ที่ความเข้มแสง (LUX) เท่ากับ 15 lux Model1



รูปที่ 4.1 จุดสีแดงสิ้นสุดทางโค้ง



รูปที่ 4.2 จุดสีแดงเริ่มเข้าสู่ทางตรง

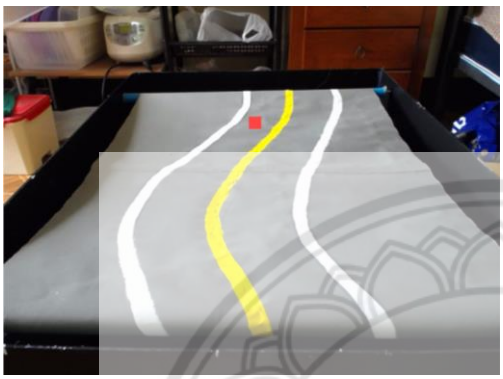


รูปที่ 4.3 จุดสีแดงเข้าสู่ทางตรง

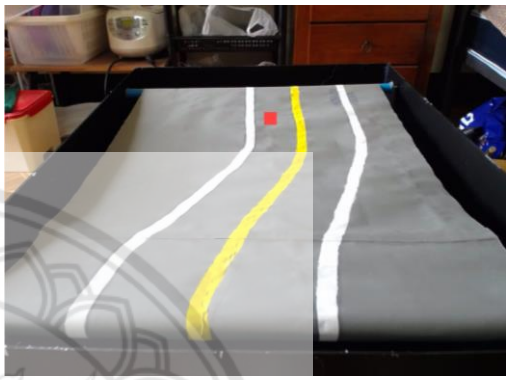
รูปที่ 4.4 จุดสีแดงเริ่มเข้าสู่ทางโค้ง

จากการทดลอง Model1 ที่ค่าความเข้มแสง 15 lux เห็นได้ว่ากล้องสามารถตรวจจับช่องทางการเดินรถได้อย่างไม่มีปัญหาทั้งทางตรงและทางโค้ง

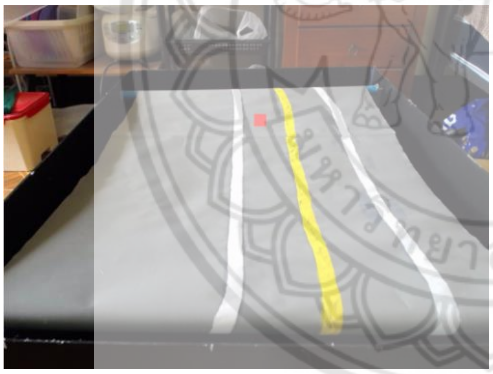
ที่ความเข้มแสง (LUX) เท่ากับ 241 lux Model1



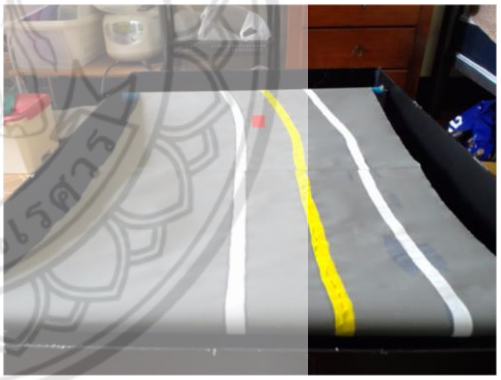
รูปที่ 4.5 จุดสีแดงสิ้นสุดทางโค้ง



รูปที่ 4.6 จุดสีแดงเริ่มเข้าสู่ทางตรง



รูปที่ 4.7 จุดสีแดงเข้าสู่ทางตรง

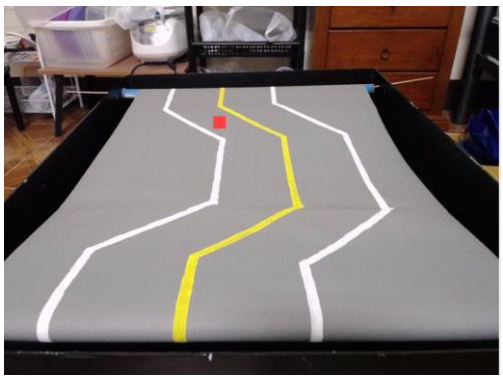


รูปที่ 4.8 จุดสีแดงเริ่มเข้าสู่ทางโค้ง

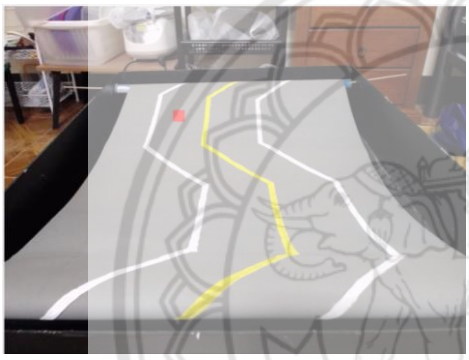
จากการทดลอง Model1 ที่ค่าความเข้มแสง 241 lux เห็นได้ว่ากล้องก็ยังสามารถตรวจจับช่องทางเดินรถได้เช่นเดียวกันกับค่าความเข้มแสง 15 lux ได้โดยไม่พบปัญหาใดๆ

ที่ความเข้มแสง (LUX) เท่ากับ 15 model 2

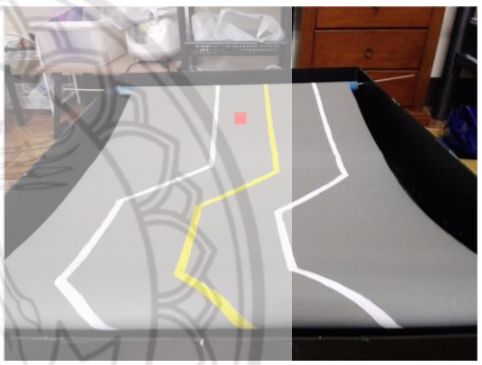
รูปที่ 4.9 จุดสีแดงเริ่มเข้าสู่ทางโค้ง



รูปที่ 4.10 จุดสีแดงเข้าสู่ทางโค้ง



รูปที่ 4.11 จุดสีแดงเริ่มเข้าสู่ทางโค้ง



รูปที่ 4.12 จุดสีแดงเข้าสู่ทางตรง

จากการทดลอง Model2 ที่ค่าความเข้มแสง 15 lux กล้องสามมารถตรวจจับช่องทางเดินรถได้เช่นเดียวกับ Model1 โดยไม่พบปัญหาใดๆ

วิเคราะห์ผลการทดลอง

แบบจำลองที่ 1 ทำการทดลองที่ค่าความเข้มแสง (Lux) เท่ากับ 15 lux เทียบกับค่าความเข้มแสงที่ 241 lux จากผลการทดลองจะเห็นว่าการทดลองที่ค่าความเข้มแสงของทั้งสองค่านั้นไม่มีความแตกต่าง เนื่องจากกล้องสามารถจับช่องทางเดินรถทั้งสองค่าได้เสถียรทั้งทางตรงและทางโค้ง

แบบจำลองที่ 1 เทียบกับ แบบจำลองที่ 2 ที่มีค่าความเข้มแสงเท่ากับ 15 lux จากผลการทดลองของทั้งสองแบบถนนจำลองจะเห็นว่าไม่เกิดความแตกต่างเช่นกัน เนื่องจากกล้องสามารถจับช่องทางเดินรถของถนนจำลองได้ทั้งสองแบบทั้งทางตรงและทางโค้ง

จากการทดลองนี้ทำให้เห็นว่าค่าความเข้มแสงที่ต่างกันและแบบถนนจำลองที่ต่างกันนั้นไม่มีผลต่อการตรวจจับช่องทางเดินรถของถนนจำลอง เนื่องจากการทดลองนี้ทดลองกับถนนจำลอง ไม่ได้ทดลองกับถนน และสภาพแวดล้อมของจริง ทำให้ผลการทดลองที่ได้นี้ไม่น่าเชื่อถือเท่าที่ควร

บทที่ 5
สรุปผลโครงงานและข้อเสนอแนะ

5.1 สรุปผลโครงงาน

โครงงานนี้ทำการศึกษาข้อมูลเกี่ยวกับระบบช่วยควบคุมรถให้อยู่ในช่องทาง (Lane Keeping) และ ศึกษาการประมวลผลภาพ (Images Processing) ซึ่งสามารถนำมาเป็นระบบต้นแบบเพื่อที่จะนำไปพัฒนาเป็นรถไฟฟ้าประจำทางแบบไร้คนขับภายในมหาวิทยาลัยนเรศวร เพราะทางมหาวิทยาลัยนเรศวรมีโครงการที่จะลดปริมาณมลพิษ จึงได้รณรงค์ให้มีการงดใช้ยานพาหะที่มีการปล่อยมลพิษออกมา และนำไปพัฒนารถขับเคลื่อนอัตโนมัติที่ใช้วิ่งรับนิสิตภายในมหาลัยนเรศวร หรือรถขนส่งเอกสารภายในมหาลัยนเรศวร เพื่อลดอุบัติเหตุทางถนน แต่ก่อนหน้านั้นต้องทำการทดสอบช่องทางเดินรถภายในมหาวิทยาลัยนเรศวรว่าถนนภายในมหาวิทยาลัยนเรศวรสามารถรองรับระบบนี้ได้หรือไม่

จากที่ได้ศึกษาข้อมูลเกี่ยวกับการประมวลผลภาพ ซึ่งมีอยู่หลายขั้นตอนจนได้เส้นกึ่งกลางของช่องทางเดินรถ ข้อดีของการประมวลผลภาพ คือ ใช้งานได้ง่ายเมื่อได้ทำการตั้งค่าของโปรแกรมไว้แล้วสามารถใช้งานได้แบบต่อเนื่อง ข้อเสียของการประมวลผลภาพ คือ มีความซับซ้อนในโปรแกรมควรศึกษามาเป็นอย่างดี มีความตอบสนองไวต่อสภาพแวดล้อมอาจทำให้เกิดความผิดพลาดของการคำนวณจุดกึ่งกลางของช่องทางเดินรถ

จากที่ได้ทำการทดลองกับถนนจำลองทั้ง 2 แบบ พบว่าโปรแกรมประมวลผลภาพมีการทำงานค่อนข้างเสถียร และอาจจะมีการคลาดเคลื่อนของจุดกึ่งกลางเล็กน้อย เนื่องจากขนาดของเส้นช่องทางเดินรถที่ไม่เท่ากันและแสงที่ตกกระทบบนถนน แต่การคำนวณของการประมวลผลภาพก็จะคงอยู่ในระยะที่ปลอดภัยสามารถใช้งานได้ปกติ

5.2 ข้อเสนอแนะ

ปัญหาที่พบ

1. โปรมแกรมที่ใช้งานมีการประมวลผลสูง ทำให้คอมพิวเตอร์ที่ใช้งานเกิดอาการค้างขณะทำการประมวลผล
2. เส้นแบ่งช่องทางเดินรถภายในมหาวิทยาลัยนเรศวรมีสีที่ไม่ชัดเจน ทำให้ไม่สามารถตรวจจับช่องทางเดินรถได้
3. แบบถนนจำลองไม่ค่อยมีความแข็งแรง ทำให้เกิดความคลาดเคลื่อนของจุดกึ่งกลางเล็กน้อย
4. แหล่งการค้นหาวัสดุอุปกรณ์ในการทำโครงงานไม่เพียงพอกับความต้องการ

ข้อเสนอแนะ

1. เพิ่มสเป็คของเครื่องคอมพิวเตอร์ หรืออาจเปลี่ยนไปใช้งานเครื่องคอมพิวเตอร์ที่มีสเป็คสูงกว่า
2. ทำแบบถนนจำลองขึ้นมาเพื่อทำการทดลอง
3. ควรปรับปรุงแบบถนนจำลองให้มีความแข็งแรงมากกว่านี้
4. ดัดแปลงวัสดุขึ้นมาใช้งานให้เพียงพอกับความต้องการ

บรรณานุกรม

1. Mehdi Sqalli. (2559). **Lane detection.** สืบค้นเมื่อ 26 สิงหาคม 2560, จาก https://medium.com/@MSqalli/lane-detection-446986c44021.
2. Mehdi Sqalli. (2559). **Advanced Lane detection.** สืบค้นเมื่อ 26 สิงหาคม 2560, จาก https://medium.com/@MSqalli/advanced-lane-detection-6a769de0d581.
3. สินธนุ จำปีศรี. (2559). **เจาะลึกระบบและการใช้งาน Lane Keeping Assist System ใน Honda Sensing.** สืบค้นเมื่อ 1 กันยายน 2560, จาก https://www.checkraka.com/knowledge/รถยนต์ใหม่-14-158/ข้อมูลความรู้/1668477/.
4. **What is Noise.** สืบค้นเมื่อ 11 กันยายน 2560, จาก http://www.fotofile.net/learning/noise/noise.html.
5. **ระยะเบรกที่ปลอดภัย.** สืบค้นเมื่อ 14 กันยายน 2560, จาก https://motortravel.in.th/?p=2323.
6. Logitech. (2561). **LOGITECH C930E.** สืบค้นเมื่อ 5 พฤษภาคม 2561, จาก https://www.logitech.com/th-th/product/c930e-webcam.
7. **HP Pavilion 15 bc207TX.** สืบค้นเมื่อ 5 พฤษภาคม 2561, จาก https://notebookspec.com/notebook/7639-HP-Pavilion-15-bc207TX.html.
8. K-Technics Solutions. **2v 200rpm gear motor Zhengk High Torque Electric Motor Low Noise.** สืบค้นเมื่อ 5 พฤษภาคม 2561, จาก https://ktechnics.com/shop/motor-zhengk-12v-200rpm-gear-motor/.

ภาคผนวก

รายละเอียดข้อมูลอ้างอิงต่างๆ

ตารางที่ ก.1 ข้อมูลกล้อง Webcam [www.logitech.com]

	การสนทนาผ่านวิดีโอ Full HD 1080p (สูงสุด 1920 x 1080 พิกเซล)
	การบีบอัดวิดีโอ H.264
	มุมมอง 90°
	ซูม 4 เท่า ความละเอียด 1080p
	เทคโนโลยี Right light 2 เพื่อความชัดเจนในสภาพแสงที่หลากหลาย แม้ในสภาวะแสงน้อย
	โฟกัสอัตโนมัติ
	รับรองการใช้งาน Hi-Speed USB 2.0 (USB 3.0 พร้อมใช้งาน)

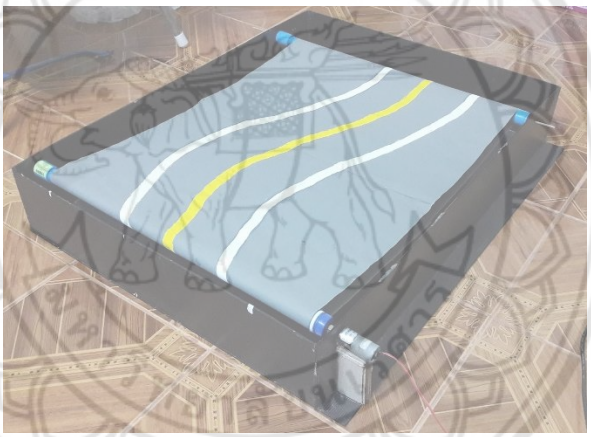
ตารางที่ ก.2 ข้อมูลเครื่องคอมพิวเตอร์ [www. notebookspec.com]

	Intel Core i7-7700HQ (2.80 - 3.80 GHz)
	NVIDIA GeForce GTX 1050 (4GB GDDR5)
	8 GB DDR4
	1 TB 7200 RPM
	15.6 inch (1920x1080) Full HD IPS
	DOS Operating System

ตารางที่ ก.3 ข้อมูลมอเตอร์ [www. ktechnics.com]

	Motor Diameter 25mm
	Gearbox Length19mm
	Voltage 12V
	Reduction Ratio1/23.4
	No Load Speed200RPM
	Rated Speed140RPM
	Rated Torque(Kg-cm)0.51
	Rated Current(AMP)0.22

รูปที่ ก.1 ข้อมูลแบบถนนจำลอง



ทำจากวัสดุแผ่นฟิวเจอร์บอร์ดสีดำ มีขนาดกว้าง 60 เซนติเมตร ยาว 122 **เซนติเมตร สูง** 10 เซนติเมตร มีลักษณะเป็นรูปสี่เหลี่ยมผืนผ้า ส่วนตัวถนนเป็นกระดาษสีเท่าน้ำมาวาดรูปเส้นแบ่งช่องทางการเดินรถตามภาพ ใช้ท่อน้ำประปา (PVC) เป็นแกนหมุนของถนนขนาดความยาวที่ 55 เซนติเมตร ใช้มอเตอร์ (ภาพ ก.3) เป็นทำหน้าที่หมุนแกนเพื่อให้ถนนนเคลื่อนตัวได้

รูปที่ ก.2 แหล่งจ่ายไฟ