

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบอัตโนมัติสำหรับการจอดรถ

An Automatic System for Car Parking

โดย

นายประสงค์ สุข อนุวงศ์

นายปรีชา วชิรภราคร

ร.พ.

๔/๓๙/๕

๒๕๔๙

เลขหมู่.....

เลขทะเบียน 72239

วัน,เดือน,ปี 1.2 ส.ย. 2550

b. 1176559A

i.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AN AUTOMATIC SYSTEM FOR CAR PARKING

BY

MR. PRASONGSUK ANUWONG

MR. PRECHA WACHIRAPARADON




**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR THE DEGREE OF
BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2006

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญาโท ระบบอัตโนมัติสำหรับการจัดการที่จอดรถ
นักศึกษา นายประสงค์สุข อนุวงศ์ รหัสนักศึกษา 46010417
นายปรีชา วชิรภราดร รหัสนักศึกษา 46010432
อาจารย์ที่ปรึกษา ดร.พิทักษ์ ธรรมวาริน
ระดับการศึกษา ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมสารสนเทศ
ภาควิชา วิศวกรรมสารสนเทศ
ปีการศึกษา 2549

ปริญญาโทฉบับนี้ได้รับการอนุมัติเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



(ดร.พิทักษ์ ธรรมวาริน)
อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	ระบบอัตโนมัติสำหรับการที่จอดรถ	
ชื่อนักศึกษา	นายประสงค์สุข อนุวงศ์	รหัสนักศึกษา 46010417
	นายปรีชา วชิรภราดร	รหัสนักศึกษา 46010432
อาจารย์ที่ปรึกษา	ดร. พิทักษ์ ธรรมวาริน	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ภาควิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2549	

บทคัดย่อ

ในวิทยานิพนธ์ฉบับนี้ได้นำเสนอ ระบบอัตโนมัติสำหรับค้นหาตำแหน่งว่างสำหรับที่จอดรถ การค้นหาตำแหน่งว่างสำหรับที่จอดรถกระทำได้โดยใช้ภาพถ่ายสถานที่จอดรถมาทำการตรวจสอบด้วยโปรแกรมคอมพิวเตอร์ จากนั้นแสดงตำแหน่งที่จอดรถบนหน้าจอคอมพิวเตอร์สำหรับให้ผู้ใช้ตรวจสอบก่อนนำรถเข้าไปจอดในที่จอดรถ นอกจากนั้นในระบบที่ได้พัฒนายังมีระบบคิดค่าจอดรถ ระบบจองตำแหน่งจอดรถเพื่อให้การจัดสรรที่จอดรถ เป็นไปอย่างมีประสิทธิภาพ โดยประสิทธิภาพของระบบสามารถแสดงได้โดยนำระบบที่ได้พัฒนามา ทดลองใช้กับที่จอดรถแบบจำลองที่ได้สร้างขึ้นมา และแสดงผลไว้ในส่วนผลการทดลอง

Thesis Title An Automatic System for Car Parking.
Student Mr. Prasongsuk Anuwong ID. 46010417
Mr. Precha Wachiraparadon ID. 46010432
Advisor Dr. Pitak Thummawarin
Graduate Level Bachelor Degree of Information Engineering
Department Information Engineering
Academic Year 2006

ABSTRACT

This project presents an automatic system for car parking. In the proposed system, the available parking lot is searched by computer using an image obtained from digital camera. Then the position of the available spaces are shown to the user before entering the parking place. Moreover, the system also provide the parking fee and booking the parking lot. The simulation of the car parking system are given to show the effectiveness of the proposed system.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดี ด้วยคำแนะนำและคำปรึกษา จากอาจารย์พิทักษ์ ธรรมวาริน ซึ่งเป็นอาจารย์ที่ปรึกษา ข้าพเจ้ารู้สึกซาบซึ้งใจ ในความอนุเคราะห์จากท่านอาจารย์ และขอขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิศวกรรมสารสนเทศ และคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิศวกรรมสารสนเทศ ทุกคนที่ให้กำลังใจ และคอยให้คำแนะนำต่างๆเสมอมา

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และได้รับการสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี คุณค่าและประโยชน์อันพึงมาจากปริญญานิพนธ์ฉบับนี้ ข้าพเจ้าขอบอบแด่ผู้มีพระคุณทุกท่าน

นายประสงค์ สุข อนุวงศ์
นายปรีชา วชิรภารตร
8 กุมภาพันธ์ 2550

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
สารบัญ	ค
สารบัญรูปภาพ	ณ

บทที่ 1 บทนำ

1.1 ที่มาของโครงการ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	1
1.4 วิธีการดำเนินงาน	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2

บทที่ 2 ทฤษฎีและหลักการ

2.1 พื้นฐานการประมวลผลภาพดิจิทัล	3
2.2 การได้มาของภาพ (Image Acquisition)	3
2.3 พิกเซล (Pixels) จุดกำเนิดภาพในจอและชนิดของรูปภาพคอมพิวเตอร์	3
2.4 ประเภทการกระทำการภาพ (Types of Image Operation)	4
2.5 Color Gray Images	8
2.6 การหาค่าความสว่างของภาพ (Image Brightness)	9
2.7 การปรับปรุงภาพ (Image Enhancement)	10
2.8 การทำ Thresholding	14
2.9 การทำ Sobel Edge Detection	15

สารบัญ(ต่อ)

เรื่อง	หน้า
บทที่ 3 การออกแบบและการเขียนโปรแกรม	16
3.1 การออกแบบ	16
3.2 Data Flow Diagram ของระบบ	17
3.3 การออกแบบฐานข้อมูล	19
3.4 การออกแบบโปรแกรม	20
3.5 Flowchart การทำงานของระบบ	28
บทที่ 4 ผลการทดลอง	30
4.1 การทดลองเพื่อหาค่า Brightness ที่เหมาะสม	31
4.2 การทดลองโปรแกรมตรวจหาค่าแฉงที่ว่าง โดยให้สีของรถคงที่ และแสงเปลี่ยนแปลง	32
4.3 การทดลองโปรแกรมตรวจหาค่าแฉงที่ว่าง โดยให้แสงคงที่ และสีรถเปลี่ยนแปลง	36
บทที่ 5 สรุปการทดลอง	44
5.1 บทวิจารณ์และสรุป	44
5.2 ปัญหาและอุปสรรคที่พบ	45
5.3 แนวทางการพัฒนาต่อ	45
บรรณานุกรม	46
ภาคผนวก	47
ภาคผนวก ก	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

เรื่อง	หน้า
รูปที่ 2.1 การกระทำการจุดต่อจุด (Point Operations)	5
รูปที่ 2.2 การกระทำการเฉพาะบริเวณ (Local Operation)	6
รูปที่ 2.3 การกระทำการทั้งหมด (Global Operation)	7
รูปที่ 2.4 Gray Level ระดับต่างๆ	8
รูปที่ 2.5 ตัวอย่างการทำ วิธีการ Histogram Equalization	10
รูปที่ 2.6 ตัวอย่างการทำ วิธีการ Histogram Equalization	11
รูปที่ 2.7 ตัวอย่างการทำ วิธีการ Histogram Equalization	11
รูปที่ 2.8 ภาพที่จอตลอดขณะไม่มีรีดจอตก่อนและหลังทำ Histogram Equalization	12
รูปที่ 2.9 ภาพที่จอตลอดขณะมีรีดจอตก่อนและหลังทำ Histogram Equalization	13
รูปที่ 2.10 ภาพก่อนและหลังการทำ Sobel Edge Detection	15
รูปที่ 3.1 รูปแบบการทำงานของระบบ CarParking	16
รูปที่ 3.2 Context Diagram ของระบบ Car Parking	17
รูปที่ 3.3 DFD level 0 ของระบบ Car Parking	17
รูปที่ 3.4 DFD Level 1 of Process 1.0	18
รูปที่ 3.5 DFD Level 1 of Process 3.0	18
รูปที่ 3.6 แสดงแผนภาพ NIAM ของฐานข้อมูล	19
รูปที่ 3.7 หน้าจอหลักโปรแกรม	20
รูปที่ 3.8 หน้าจอกำหนดค่า IP Server	20
รูปที่ 3.9 หน้าจอกำหนดค่าพื้นที่จอตลอดแต่ละจุด	21
รูปที่ 3.10 หน้าจอกำหนดค่าอ้างอิง	22
รูปที่ 3.11 หน้าจอกำหนดแผนผังที่จอตลอด	23
รูปที่ 3.12 หน้าจอแสดงส่วนประมวลผล	23
รูปที่ 3.13 หน้าจอแสดงผลทางมอเนเตอร์	24
รูปที่ 3.14 หน้าจอแสดง ไฟล์ วีดีโอ	25
รูปที่ 3.15 หน้าจอเก็บข้อมูลลูกค้าทางเข้า	26
รูปที่ 3.16 หน้าจอเก็บข้อมูลลูกค้าทางออก	27
รูปที่ 3.17 Flowchart การทำงานแยกแยะว่ามีรถหรือไม่	28

สารบัญรูปภาพ(ต่อ)

เรื่อง	หน้า
รูปที่ 4.1 แบบจำลองที่จอตรด	30
รูปที่ 4.2 กราฟแสดงค่า Brightness ของภาพระหว่างมีรดและไม่มีรด	31
รูปที่ 4.3 ภาพที่จอตรดและจอมอนิเตอร์ก่อนมีรดจอตกรณีสว่าง	32
รูปที่ 4.4 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดจอตกรณีสว่าง	32
รูปที่ 4.5 ภาพที่จอตรดและจอมอนิเตอร์ก่อนมีรดจอตกรณีสว่างปกติ	33
รูปที่ 4.6 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดจอตกรณีสว่างปกติ	33
รูปที่ 4.7 ภาพที่จอตรดและจอมอนิเตอร์ก่อนมีรดจอตกรณีสว่างน้อยมาก	34
รูปที่ 4.8 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดจอตกรณีสว่างน้อยมาก	34
รูปที่ 4.9 ภาพที่จอตรดและจอมอนิเตอร์ก่อนมีรดจอตกรณีสว่างมืด	35
รูปที่ 4.10 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดจอตกรณีสว่างมืด	35
รูปที่ 4.11 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีขาวจอต	36
รูปที่ 4.12 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีแดงจอต	36
รูปที่ 4.13 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีฟ้าจอต	37
รูปที่ 4.14 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีเทาจอต	37
รูปที่ 4.15 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีดำจอต	38
รูปที่ 4.16 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีเขียวจอต	38
รูปที่ 4.17 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีเหลืองจอต	39
รูปที่ 4.18 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีน้ำเงินจอต	39
รูปที่ 4.19 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีขาวจอต	40
รูปที่ 4.20 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีแดงจอต	40
รูปที่ 4.21 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีฟ้าจอต	41
รูปที่ 4.22 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีเทาจอต	41
รูปที่ 4.23 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีดำจอต	42
รูปที่ 4.24 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีเขียวจอต	42
รูปที่ 4.25 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีเหลืองจอต	43
รูปที่ 4.26 ภาพที่จอตรดและจอมอนิเตอร์ขณะมีรดสีน้ำเงินจอต	43
รูปที่ ก-1 รูปกล้องซีทีสแกน OKER	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ที่มาของโครงการ

ในปัจจุบันปัญหาการจัดการที่จอดรถนับว่าเป็นปัญหาที่พบได้ทั่วไป เพราะที่จอดรถในปัจจุบันมีขนาดใหญ่และซับซ้อน เพื่อให้การจัดการที่จอดรถเป็นไปอย่างมีประสิทธิภาพ จึงจำเป็นต้องมีการพัฒนาระบบช่วยในการค้นหาที่จอดรถ

จึงได้มีแนวความคิดที่จะทำเทคโนโลยี มาประยุกต์เพื่อแก้ปัญหาดังกล่าวโดยอาศัยทฤษฎี Image Processing และ โปรแกรม Visual Basic โดยนำทฤษฎี Image Processing มาตรวจสอบเปรียบเทียบภาพจากกล้องที่ติดตั้ง แล้วประมวลผลโดยโปรแกรม Visual Basic แสดงผลออกทางหน้าจอคอมพิวเตอร์ อีกทั้งในโครงการยังมีระบบคิดค่าจอดรถ ระบบจองตำแหน่งจอดรถเพื่อป้องกันการแย่งกันจอดรถการเก็บสถิติการจอดรถเพื่อนำไปวิเคราะห์ในส่วนอื่นๆของสำนักอาคารที่จอดรถ

1.2 วัตถุประสงค์

- เพื่อตรวจสอบตำแหน่งว่ามีรถจอดอยู่หรือไม่ โดยวิเคราะห์จากภาพ แล้วแสดงผลทางมอนิเตอร์
- เพื่อศึกษาการวิเคราะห์และการออกแบบระบบ
- เพื่อศึกษาการเขียน โปรแกรม Application โดยอาศัยโปรแกรม Visual Basic และ โปรแกรมประยุกต์อื่นๆ
- เพื่อลดต้นทุนจากการใช้เซนเซอร์ โดยใช้การวิเคราะห์จากภาพแทน
- สามารถนำความรู้ที่ได้ศึกษามา ก่อประโยชน์สูงสุด

1.3 ขอบเขตของโครงการ

- ระบบสามารถตรวจหาตำแหน่งที่จอดรถได้
- ระบบสามารถจองตำแหน่งเพื่อป้องกันการเลือกที่จอดรถซ้ำซ้อน
- ระบบสามารถคำนวณค่าจอดรถ
- ระบบสามารถเก็บสถิติการจอดรถในแต่ละวันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 วิธีการดำเนินงาน

- ศึกษาระบบจอตกรดที่มีอยู่ในปัจจุบัน นำมาวิเคราะห์ปัญหาของระบบเดิม แล้วแก้ปัญหา ระบบเดิมให้มีประสิทธิภาพยิ่งขึ้น
- ศึกษาทฤษฎี Image Processing เช่น การได้มาของภาพ การปรับปรุงภาพ เป็นต้น และ ศึกษาโปรแกรม Visual Basic ซึ่งเป็นโปรแกรมที่ได้รับความนิยม และง่ายต่อการทำความเข้าใจ และโปรแกรมประยุกต์อื่นๆที่ใช้งานกับภาพ
- ลงมือเขียน โปรแกรม ทดลอง และหาปัญหาที่เกิดจากการทดลอง พร้อมทั้งแก้ปัญหาและปรับปรุงให้โปรแกรมมีประสิทธิภาพมากยิ่งขึ้น

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- เพื่อช่วยหาที่จอตกรดได้เร็วขึ้น ทำให้ประหยัดเวลา และประหยัดเชื้อเพลิง
- ได้ความรู้เกี่ยวกับเรื่อง Image Processing
- ได้ความรู้ในการเขียนโปรแกรมด้วยภาษา Visual Basic
- ได้ความรู้ในการวิเคราะห์ระบบและออกแบบฐานข้อมูล
- ทำให้ระบบเดิมมีประสิทธิภาพมากยิ่งขึ้น
- ผู้ศึกษาได้รับความรู้ แนวความคิด และทักษะการเขียน โปรแกรม

บทที่ 2

ทฤษฎีและหลักการ

การพัฒนาาระบบที่จอรรถให้มีประสิทธิภาพมากยิ่งขึ้น โดยเพิ่มระบบตรวจหาตำแหน่งที่ว่างสำหรับจอรรถ โดยโครงการงานอาศัยภาพที่ได้จากกล้องวงจรปิดมาวิเคราะห์ด้วยกระบวนการต่างๆทาง Image Processing ซึ่งในที่นี่จะกล่าวเฉพาะเนื้อหาที่ควรรู้เท่านั้นที่เกี่ยวกับโครงการงาน

2.1 พื้นฐานการประมวลผลภาพดิจิทัล

การประมวลผลภาพดิจิทัล (Image Processing) เป็นกระบวนการที่นำข้อมูลภาพดิจิทัลเข้ามาทำการประมวลผล และให้ข้อมูลออกมาเป็นข้อมูลภาพใหม่ ซึ่งการประมวลผลส่วนใหญ่จะเน้นในด้านการปรับปรุงภาพให้ดีขึ้น (Image Enhancement) การวิเคราะห์ภาพ (Image Analysis) การแบ่งภาพออกเป็นส่วนๆ (Image Segmentation)

2.2 การได้มาของภาพ (Image Acquisition)

การได้มาของภาพมาจากกล้องวงจรปิด โดย Capture ภาพนิ่งที่ได้ จะเก็บในรูปแบบ Digital Image เก็บลงในหน่วยความจำเพื่อนำไปประมวลผลภาพในขั้นต่อไป

2.3 พิกเซล (Pixels) จุดกำเนิดภาพในจอและชนิดของรูปภาพคอมพิวเตอร์

ภาพในคอมพิวเตอร์เกิดจากการนำจุดสีที่เรียกว่า “พิกเซล (Pixels)” มาวางเรียงต่อกันจนสุดต่อเนื่อง ภาพจะดูคมชัดเท่าใดก็ขึ้นอยู่กับว่าภาพนั้นมีพิกเซลอยู่มากเท่าใดเรามากจะนับความละเอียดของภาพโดยนับจำนวนพิกเซลต่อความยาวภาพหนึ่งนิ้วหรือพิกเซลต่อนิ้ว (Pixels/Inch) ยังมีจำนวนพิกเซลต่อนิ้วมากภาพก็ยิ่งละเอียด

ภาพยิ่งละเอียดก็ยิ่งดูเหมือนจริง ความละเอียดของภาพเรียกว่า “เรโซลูชัน (Resolution)” ภาพที่ละเอียดมากๆจึงเรียกว่าภาพที่มีเรโซลูชันสูงส่วนภาพที่หยาบๆ ก็เรียกว่า ภาพที่มีเรโซลูชันต่ำ

ชนิดของรูปภาพบนเครื่องคอมพิวเตอร์

รูปภาพที่ปรากฏและใช้งานบนเครื่องคอมพิวเตอร์นั้น มีอยู่ด้วยกันสองชนิดคือ รูปภาพแบบเวกเตอร์ (Vector Image) และรูปภาพแบบบิตแมป (Bitmap Image)

- ภาพแบบเวกเตอร์ (Vector Image)

ภาพชนิดเวกเตอร์ (Vector) เป็นภาพที่มีการบันทึกส่วนต่างๆ ของรูปภาพไม่ว่าจะเป็นเส้นตรงเส้นโค้ง รูปทรง และสีต่างๆ ด้วยสมการทางคณิตศาสตร์ เมื่อต้องการนำรูปภาพนี้แสดงออกทางจอภาพ หรือพิมพ์ออกทางเครื่องพิมพ์ โปรแกรมที่ใช้ในการเปิดรูปภาพดังกล่าวจะต้องนำสมการต่างๆ ที่บันทึกไว้มาคำนวณ และสร้างรูปทรงต่างๆ ขึ้นมาใหม่ จุดเด่นที่ถือว่าเป็นข้อดีของภาพแบบเวกเตอร์คือ ไม่ว่าจะขยายขนาดภาพให้ใหญ่โตแค่ไหนก็ตามเครื่องคอมพิวเตอร์จะคำนวณอัตราส่วนของสมการใหม่ทำให้ภาพที่เกิดขึ้นใหม่ที่มีความคมชัดทุกครั้ง

- ภาพแบบบิตแมป (Bitmap Image)

ภาพแบบบิตแมป (Bitmap Image) เป็นภาพที่เกิดจากการเรียงเม็ดสีขนาดเล็กประกอบกันขึ้นมาเป็นภาพใหญ่ที่เรามองเห็นกัน เม็ดสีเล็กๆ หรือจุดสีเล็กๆ ดังกล่าว จะบรรจุด้วยสีหนึ่งสี ซึ่งจุดนี้มีชื่อเรียกว่า พิกเซล (Pixels) ในการแก้ไขภาพแบบบิตแมปนั้นหมายความว่าเรากำลังแก้ไขเม็ดสีดังกล่าวเพื่อให้ภาพเปลี่ยนไปตามต้องการ

เนื่องจากภาพแบบบิตแมปนั้นบรรจุไปด้วยเม็ดสีในจำนวนคงที่ดังนั้นเมื่อขยายภาพขึ้นมามากๆ จะทำให้รายละเอียดของภาพสูญเสียไป

2.4 ประเภทการกระทำการภาพ (Types of Image Operation)

การประมวลผลภาพดิจิทัลจะเป็น กระบวนการที่ทำการกระทำการ (Operation) ภาพอย่างใดอย่างหนึ่งต่อภาพนำเข้า (Input Image) เพื่อให้ได้ภาพผลลัพธ์ (Output Image) มีลักษณะของภาพเป็นไปตามที่ต้องการ ซึ่งการกระทำการภาพที่ใช้ในการประมวลผลภาพดิจิทัลมีอยู่มากมายหลายแบบ ความเข้าใจเกี่ยวกับคุณลักษณะและการแยกแยะประเภทของการกระทำภาพ จะช่วยให้เราสามารถคาดคะเนภาพผลลัพธ์ที่จะได้จากการกระทำการแต่ละแบบ หรือการประมาณความซับซ้อนของการกระทำการภาพที่จะใช้

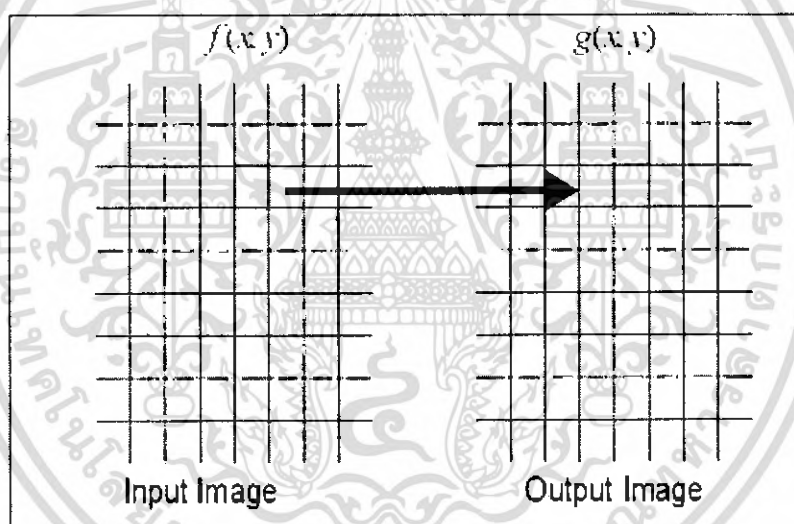
การกระทำการภาพในการประมวลผลภาพดิจิทัลสามารถแบ่งออกได้เป็นประเภทใหญ่ 3 ประเภท คือ

2.4.1 การกระทำการภาพจุดต่อจุด (Point Operations)

การกระทำแบบนี้ ค่าความเข้มแสงในแต่ละพิกเซลของภาพผลลัพธ์ จะขึ้นกับค่าความเข้มแสงของพิกเซลในภาพนำเข้า ณ ตำแหน่งที่สมนัยกันดังรูปที่ 1 ลักษณะการกระทำการภาพประเภทนี้ได้แก่ การปรับความสว่าง หรือ ความคมชัดของภาพดิจิทัล การ บวก ลบ คูณ และหาร ภาพดิจิทัล หรือการกระทำทางตรรกศาสตร์ต่างๆ เป็นต้น

ถ้า $f(x,y)$ และ $g(x,y)$ เป็นภาพนำเข้าและภาพผลลัพธ์ตามลำดับ ค่าของพิกเซล $g(x,y)$ จะมีค่าดังนี้

$$g(x,y) = f[f(x,y)] \quad ; \text{เมื่อ } f \text{ เป็นการกระทำภาพใดๆ}$$



รูปที่ 2.1 การกระทำการจุดต่อจุด (Point Operations)

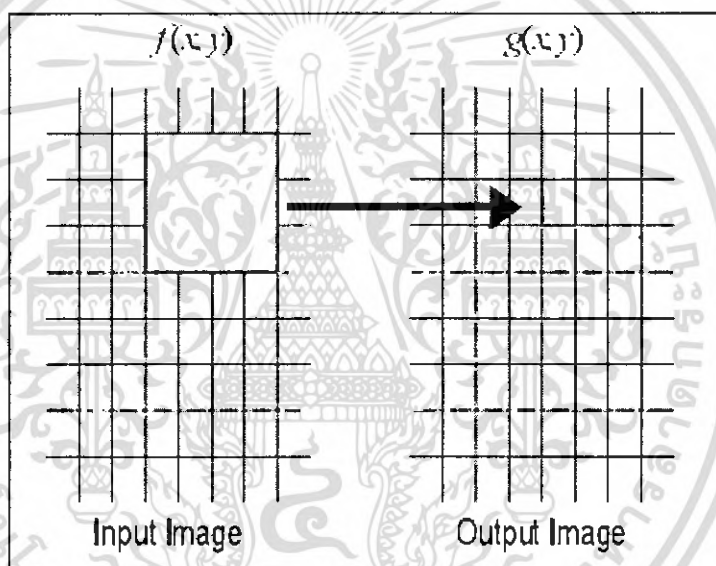
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 การกระทำการเฉพาะบริเวณ (Local Operation)

สำหรับการกระทำการแบบนี้ค่าความเข้มแสงของพิกเซลแต่ละจุดในภาพผลลัพธ์จะขึ้นกับค่าความเข้มแสงของกลุ่มพิกเซลที่อยู่บริเวณเดียวกัน (Neighborhood Pixels) ในภาพนำเข้า ดังรูปที่ 2 ลักษณะการกระทำการภาพประเภทนี้ได้แก่การหาขอบ (Edge Detection) การกรองสัญญาณใน โดเมนระยะทาง (Spatial Filtering) เป็นต้น

ถ้า $f(x,y)$ และ $g(x,y)$ เป็นภาพนำเข้าและภาพผลลัพธ์ตามลำดับ ค่าของพิกเซล $g(x,y)$ จะมีค่าดังนี้

$$g(x_i, y_j) = f[\text{neighborhood of } f(x_i, y_j)]$$



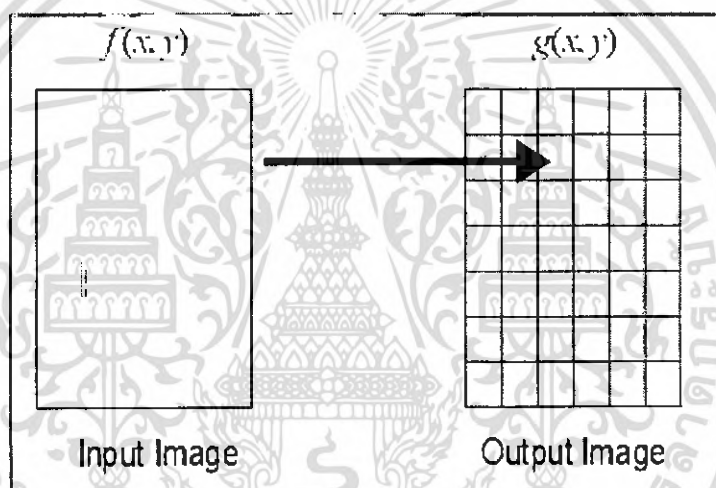
รูปที่ 2.2 การกระทำการเฉพาะบริเวณ (Local Operation)

2.4.3 การกระทำการทั้งหมด (Global Operations)

การกระทำแบบนี้ ค่าความเข้มแสงในแต่ละพิกเซลของภาพผลลัพธ์ (Output Image) จะขึ้นกับค่าความเข้มของแสงของพิกเซลทุกตัวในภาพนำเข้า ดังรูปที่ 2.3 ลักษณะการกระทำภาพประเภทนี้ได้แก่ การทresholding (Thresholding) การทำฮิสโทแกรม (Histogram) เป็นต้น

ถ้า $f(x,y)$ และ $g(x,y)$ เป็นภาพนำเข้าและภาพผลลัพธ์ตามลำดับ ค่าของพิกเซล $g(x,y)$ จะมีค่าดังนี้

$$g(x,y) = f(x,y) \text{ for all } i$$



รูปที่ 2.3 การกระทำการทั้งหมด (Global Operation)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 Color Gray Images

ในโครงการนี้ จะใช้ภาพ Color Gray มาใช้ในการทำงาน ซึ่งมีข้อดีหลายข้อ คือ ภาพ Color Gray เพียงพอที่จะนำมาทำการตรวจสอบการเปลี่ยนแปลงของภาพ ก็คือการเปลี่ยนแปลงของความเข้มของแสง และภาพ Color Gray ใช้หน่วยความจำในการเก็บข้อมูลน้อยกว่าภาพสี ทำให้ประหยัดหน่วยความจำ เมื่อเทียบกับการเก็บข้อมูลของภาพสี ทำให้ภาพ Color Gray ใช้เวลาในการประมวลผลเร็วกว่าภาพสี

Gray Level คือระดับความเข้มของเม็ดสียิ่งค่าความเข้มสูงสีที่แสดงผลออกทางจอคอมพิวเตอร์จะเป็นสีขาวและระดับความเข้มต่ำสีที่แสดงผลออกทางจอคอมพิวเตอร์จะเป็นสีดำ ซึ่งระดับ Gray Level มีหลายระดับดัง รูปที่ 4



รูปที่ 2.4 Gray Level ระดับต่างๆ

ในโครงการนี้ได้เลือก Gray Level 256 level เพราะมีความละเอียดเพียงพอที่จะนำมาใช้ในการวิเคราะห์ภาพ

2.6 การหาค่าความสว่างของภาพ (Image Brightness)

การหาค่า Brightness คือการหาค่าความสว่างของภาพ โดยหาได้จากหาค่าผลรวมของ gray level ทั้งหมดหารด้วยค่าของจำนวนพิกเซลทั้งหมด ดังสมการ

$$\text{Brightness} = \bar{X} = \sum_{i=0}^N \frac{X_i}{N} \quad 2.1$$

N = total of pixels

X_i = gray level of pixels

2.7 การปรับปรุงภาพ (Image Enhancement)

คือกระบวนการเปลี่ยนคุณภาพของภาพให้อยู่ในระดับที่ดีขึ้น ที่จะสามารถนำไปวิเคราะห์ให้ได้ การปรับปรุงภาพที่ใช้ คือวิธีการ Histogram Equalization เป็นการกระทำทั้งหมด (Global Operations) มีวิธีการดังนี้

2.7.1 หาจำนวน Pixel ของแต่ละ Gray level ($P_i(j)$)

2.7.2 ทำ Accumulated Sum มีสมการดังนี้

$$A_i(k) = \sum_{j \leq k} P_i(j)$$

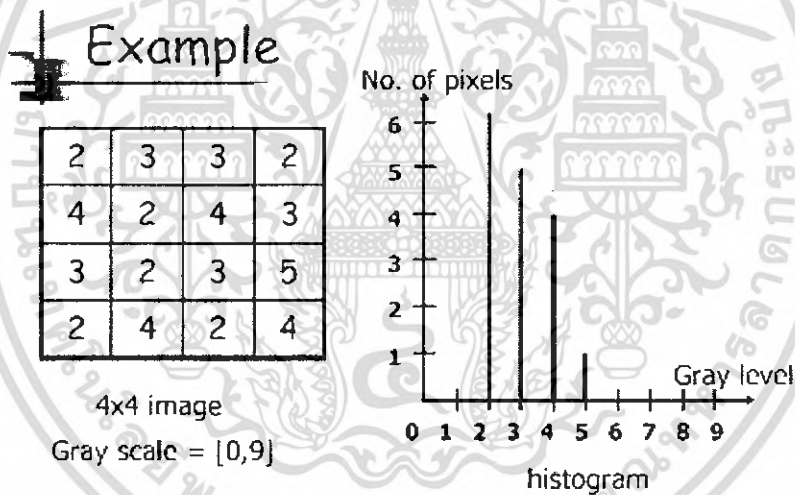
$A_i(k)$ = Accumulated Sum

$P_i(j)$ = จำนวน Pixel ของแต่ละ Gray level

2.7.3 ทำ Normalized มีสมการดังนี้

$$P_1(k) = A_1(k) / N$$

N = จำนวน Pixel ทั้งหมด

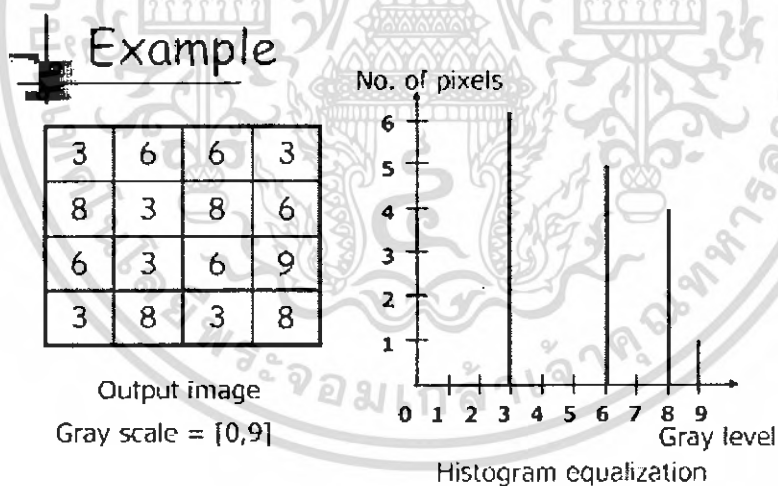
2.7.4 นำค่า max ของ gray level คูณค่า $P_1(k)$ 

รูปที่ 2.5 ตัวอย่างการทำ วิธีการ Histogram Equalization

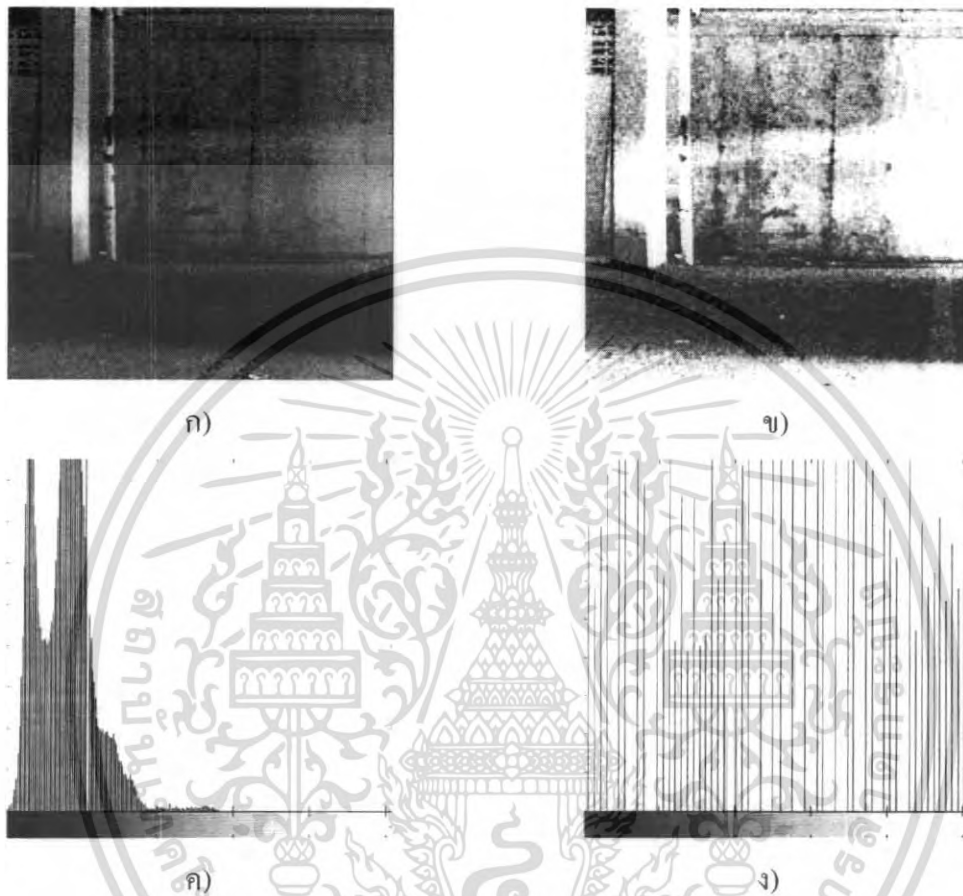
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Gray Level(j)	0	1	2	3	4	5	6	7	8	9
No. of pixels	0	0	6	5	4	1	0	0	0	0
$\sum_{i=0}^j n_i$	0	0	6	11	15	16	16	16	16	16
$\sum_{i=0}^j \frac{n_i}{n}$	0	0	/	/	/	/	/	/	/	/
			16	16	16	16	16	16	16	16
$s \times 9$	0	0	3.3	6.1	8.4	9	9	9	9	9
			≈3	≈6	≈8					

รูปที่ 2.6 ตัวอย่างการทำ วิธีการ Histogram Equalization



รูปที่ 2.7 ตัวอย่างการทำ วิธีการ Histogram Equalization



รูปที่ 2.8 ก) ภาพที่จอตรงขณะ ไม่มีรจจอดก่อนทำHistogram Equalization
 ข) ภาพที่จอตรงขณะ ไม่มีรจจอดหลังทำHistogram Equalization
 ค) ภาพ Histogram ขณะไม่มีรจจอด
 ง) ภาพ Histogram หลังทำ Histogram Equalization ขณะไม่มีรจจอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



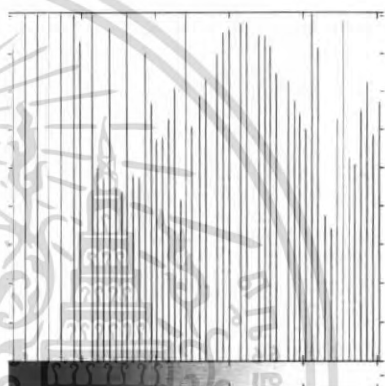
ก)



ข)



ค)



ง)

รูปที่ 2.9 ก) ภาพที่จอตลอดขณะมีรถจอดก่อนทำ Histogram Equalization

ข) ภาพที่จอตลอดขณะมีรถจอดหลังทำ Histogram Equalization

ค) ภาพ Histogram ขณะมีรถจอด

ง) ภาพ Histogram หลังทำ Histogram Equalization ขณะมีรถจอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 การทำ Thresholding

เป็นกระบวนการที่ใช้ในการแยกวัตถุที่มีความแตกต่างจากพื้นหลังของภาพ การจำแนกแต่ละกลุ่มของพิกเซลว่าเป็นวัตถุเดียวกันหรือว่าเป็น ภาพพื้นหลัง สามารถกำหนดค่าให้กับภาพต้นฉบับได้ถ้าอยู่ในช่วงที่เรากำหนด

การทำThresholdทั่วไป จะเป็นแบบตรงไปตรงมา กำหนดให้ $a \in R^x$ ของภาพต้นฉบับและให้ช่วงที่ต้องการทำThresholdเป็น $[h,k]$ Thresholdของภาพ $b \in \{0,1\}^x$ ให้

$$b(x) = \begin{cases} 1 & \text{ถ้า } h \leq a \leq k \\ 0 & \text{อื่นๆ} \end{cases}$$

ทุกค่าของ x มีสองกรณีที่ต้องให้ความสนใจคือ ค่าของภาพ b มีค่าสูงเพียงอย่างเดียว หรือมีค่าต่ำเพียงอย่างเดียว ในกรณีแรกของ Threshold ภาพ b จะให้

$$b(x) = \begin{cases} 1 & \text{ถ้า } a \geq k \\ 0 & \text{อื่นๆ} \end{cases}$$

กรณีที่สอง

$$b(x) = \begin{cases} 1 & \text{ถ้า } a < k \\ 0 & \text{อื่นๆ} \end{cases}$$

2.9 Sobel Edge Detection

เป็นวิธีการหาขอบของวัตถุในรูปภาพ โดยใช้ Mask ของ Sobel ไปทำการ Convolution เพื่อให้ได้ขอบของวัตถุในแนวแกนนอน และแนวแกนตั้ง

ซึ่ง Mask ของ Sobel มีดังนี้

$$A = \begin{vmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{vmatrix} \quad B = \begin{vmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{vmatrix}$$

Mask หาขอบแกนตั้ง

Mask หาขอบแนวนอน



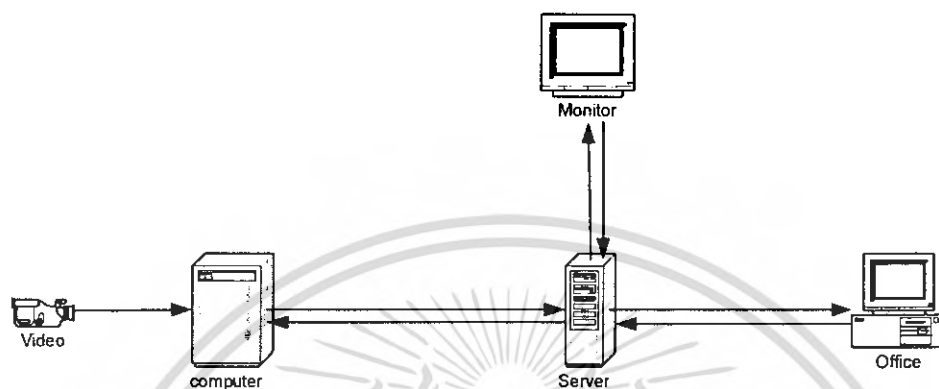
รูปที่ 2.10 ภาพก่อนและหลังการทำ Sobel Edge Detection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการเขียนโปรแกรม

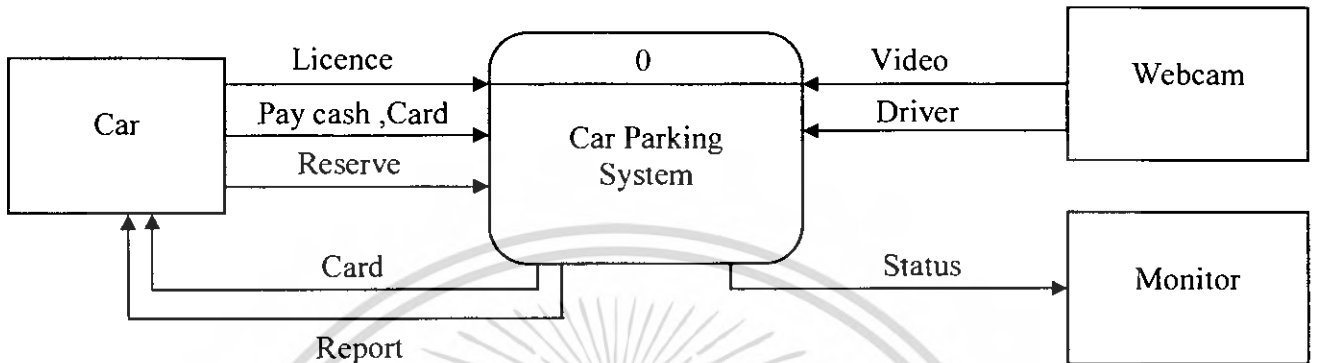
3.1 การออกแบบ



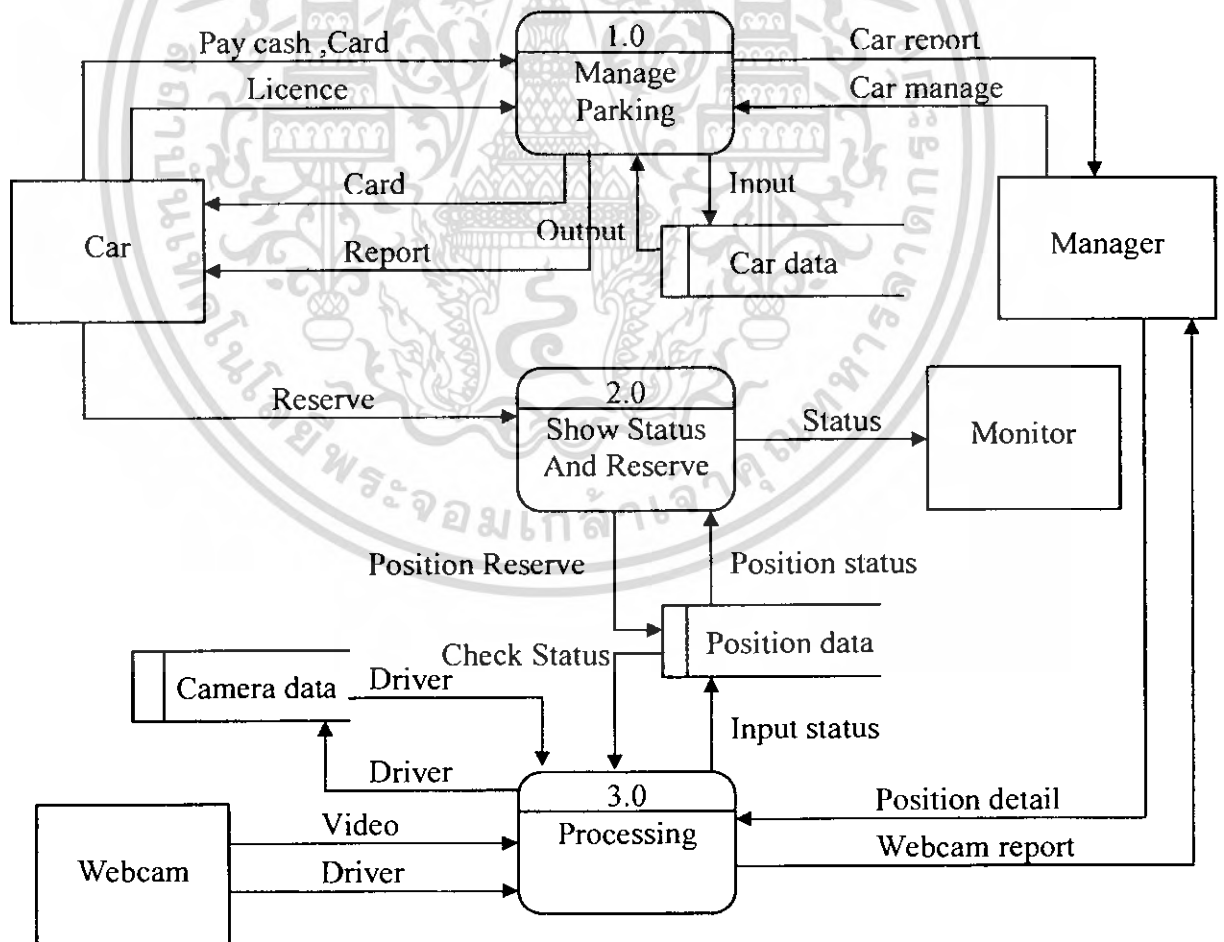
รูปที่ 3.1 รูปแบบการทำงานของระบบ CarParking

จากรูป video จะเป็นตัวเว็บแคมที่จับภาพในที่จอดรถ ซึ่งจะส่งภาพไปให้คอมพิวเตอร์ประมวลผล ซึ่งจะทำการประมวลผล ทุกๆ 15 วินาที เมื่อประมวลผลแล้วจะทำการส่งข้อมูลไปเก็บที่ Server ซึ่งจะเป็นตัวที่จะแสดงผลบนจอมอนิเตอร์ และที่มอนิเตอร์นี้จะมีการจอง ที่จอดรถได้ โดยเมื่อมีการจอง จะส่งตำแหน่งที่จองกลับมาเก็บยัง Server และเมื่อมีการมีการประมวลผลครั้งถัดไป คอมพิวเตอร์ที่ทำการวิเคราะห์ภาพ จะทำการดึงข้อมูลว่า ได้มีการจองที่ตำแหน่งใดบ้าง แล้วจะทำการยกเลิกการประมวลผลที่ ตำแหน่งที่ได้มีการจองเป็นเวลา 5 นาที เพื่อให้รถที่จอง ได้เข้าไปจอดแต่ถ้ารถเข้าไปจอดไม่ตรงกับช่องที่จอง หลังจาก 5 นาทีแล้ว status ก็จะได้กลับมาเป็นที่ว่างเหมือนเดิมให้คนอื่นจองได้ต่อไป ทางด้าน Office นั้นจะเป็นในส่วนที่ ขามจะแจกบัตรจอดรถ โดยที่จะเก็บข้อมูล เลขทะเบียนรถ,วันเวลาเข้าและออก,หมายเลขบัตรจอดรถ,และในส่วนคิดค่าจอดรถ และสามารถเรียกดู สถิติต่างๆได้

3.2 Data Flow Diagram ของระบบ

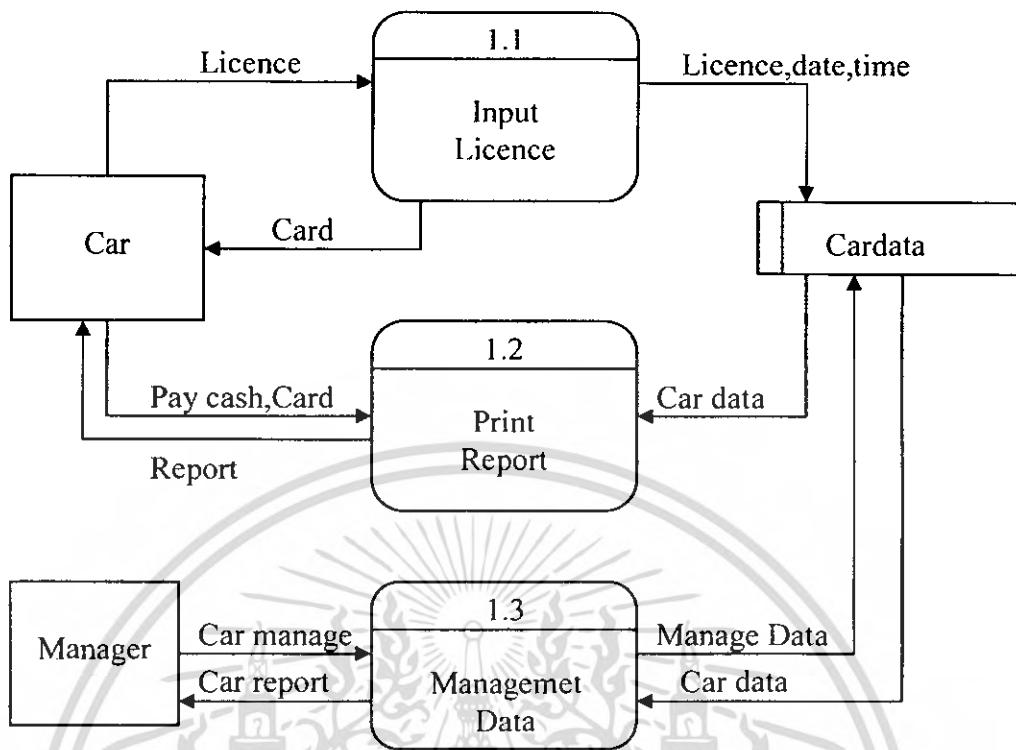


รูปที่ 3.2 Context Diagram ของระบบ Car Parking

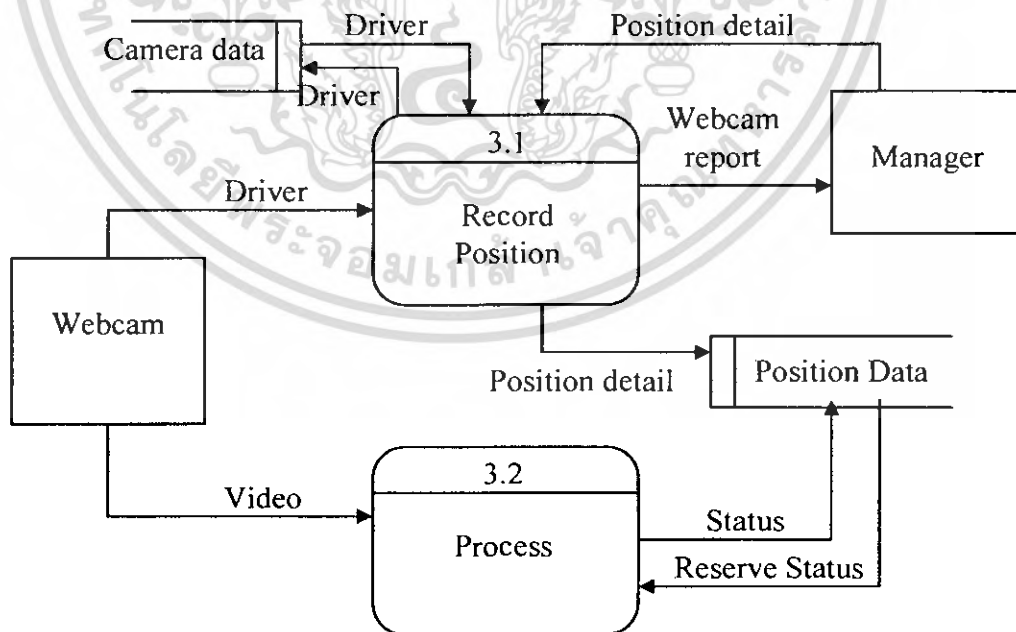


รูปที่ 3.3 DFD level 0 ของระบบ Car Parking

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอก 72239 ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 DFD Level 1 of Process 1.0

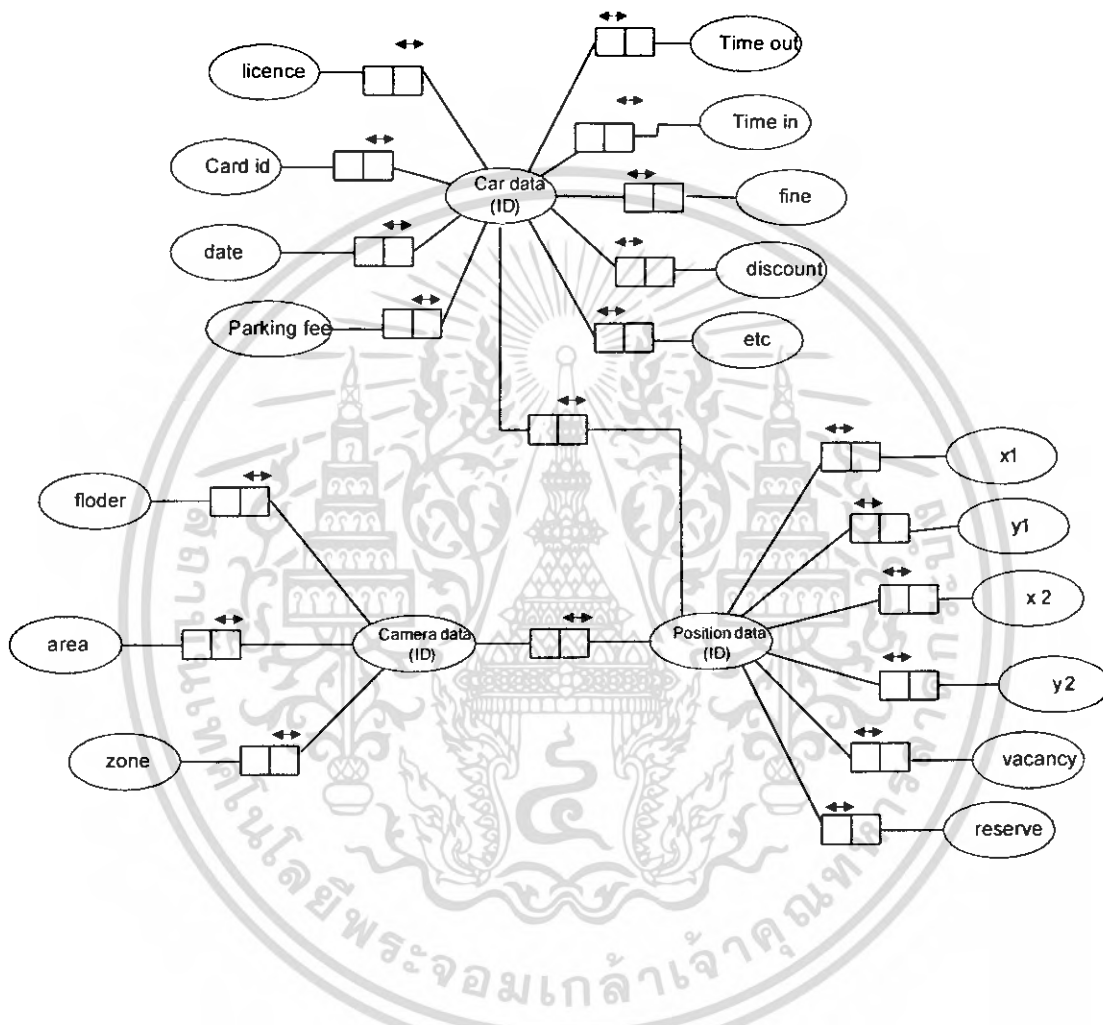


รูปที่ 3.5 DFD Level 1 of Process 3.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบฐานข้อมูล

จากการออกแบบ Data Flow Diagram จะได้ว่ามีทั้งหมด 3 ตาราง โดยมี ดังแผนภาพ NIAM

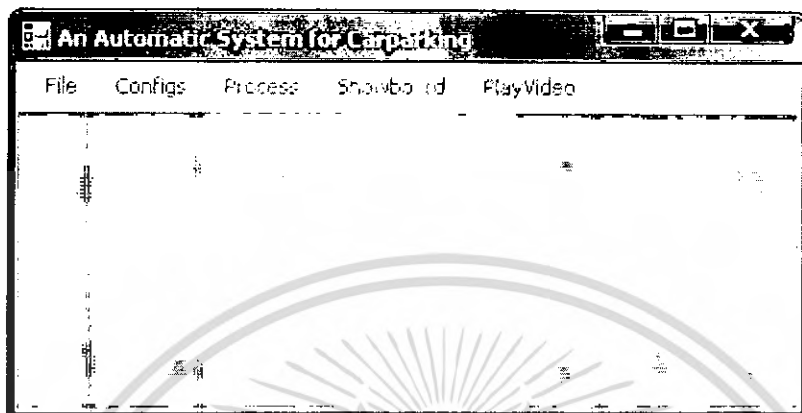


รูปที่ 3.6 แสดงแผนภาพ NIAM ของฐานข้อมูล

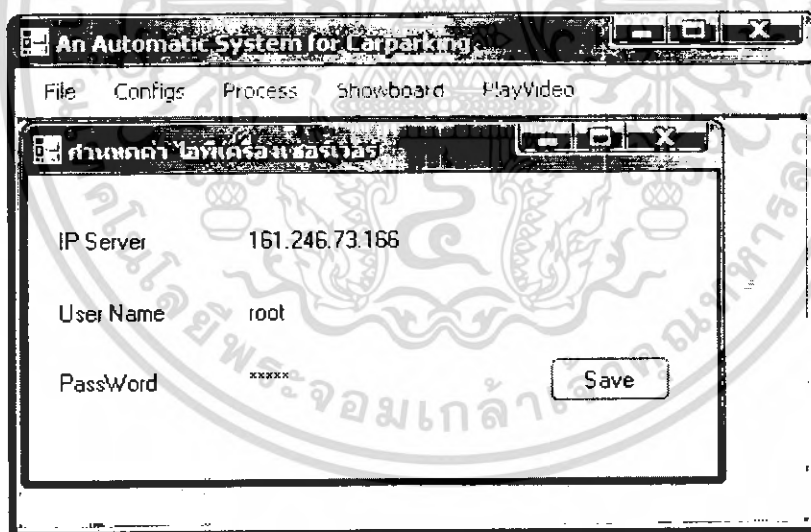
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การออกแบบโปรแกรม

ในตอนเริ่มต้นเขียน โปรแกรมจำเป็นจะต้องออกแบบ Graphic User Interface ก่อนเพื่ออำนวยความสะดวกในการเขียนโปรแกรม

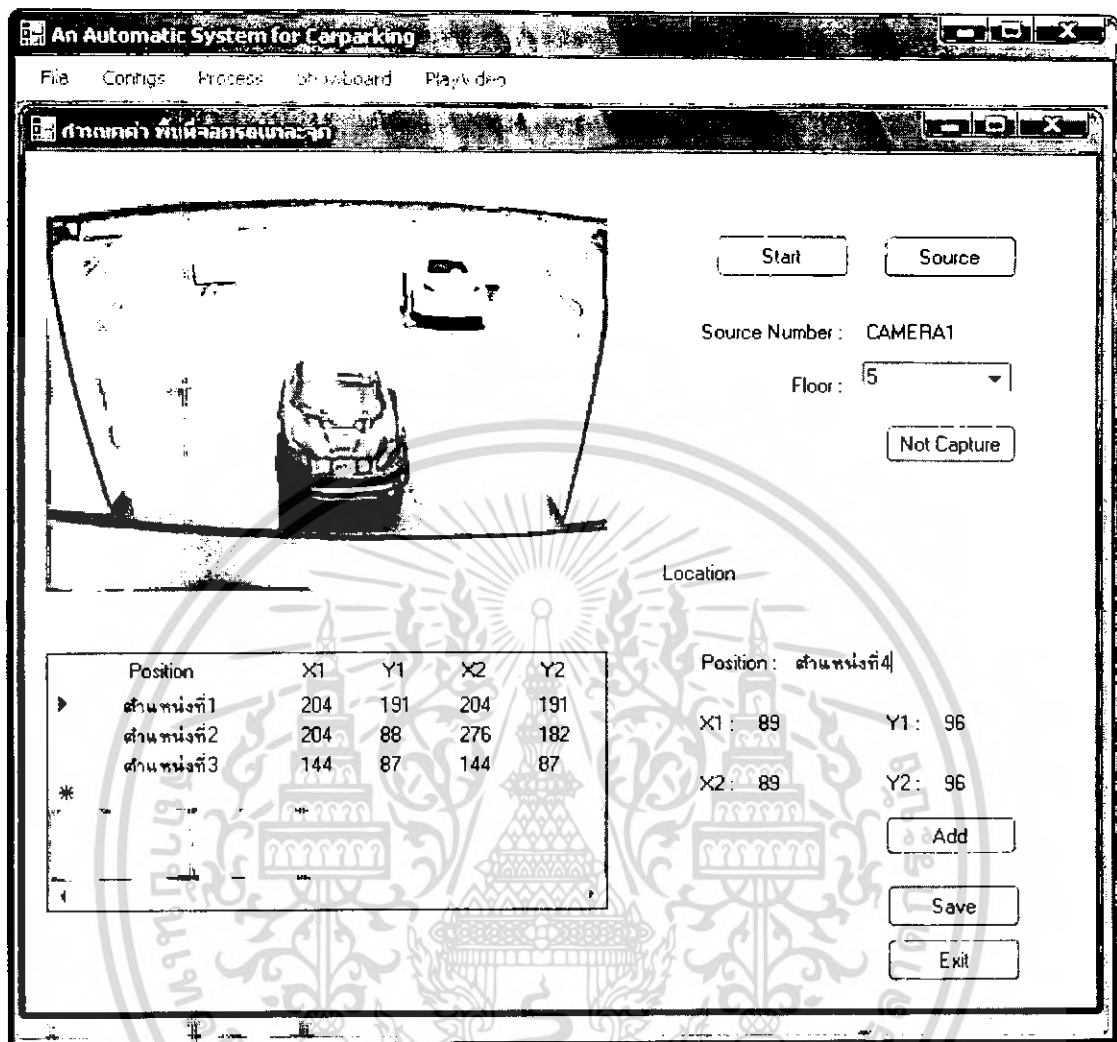


รูปที่ 3.7 หน้าจอหลักโปรแกรม



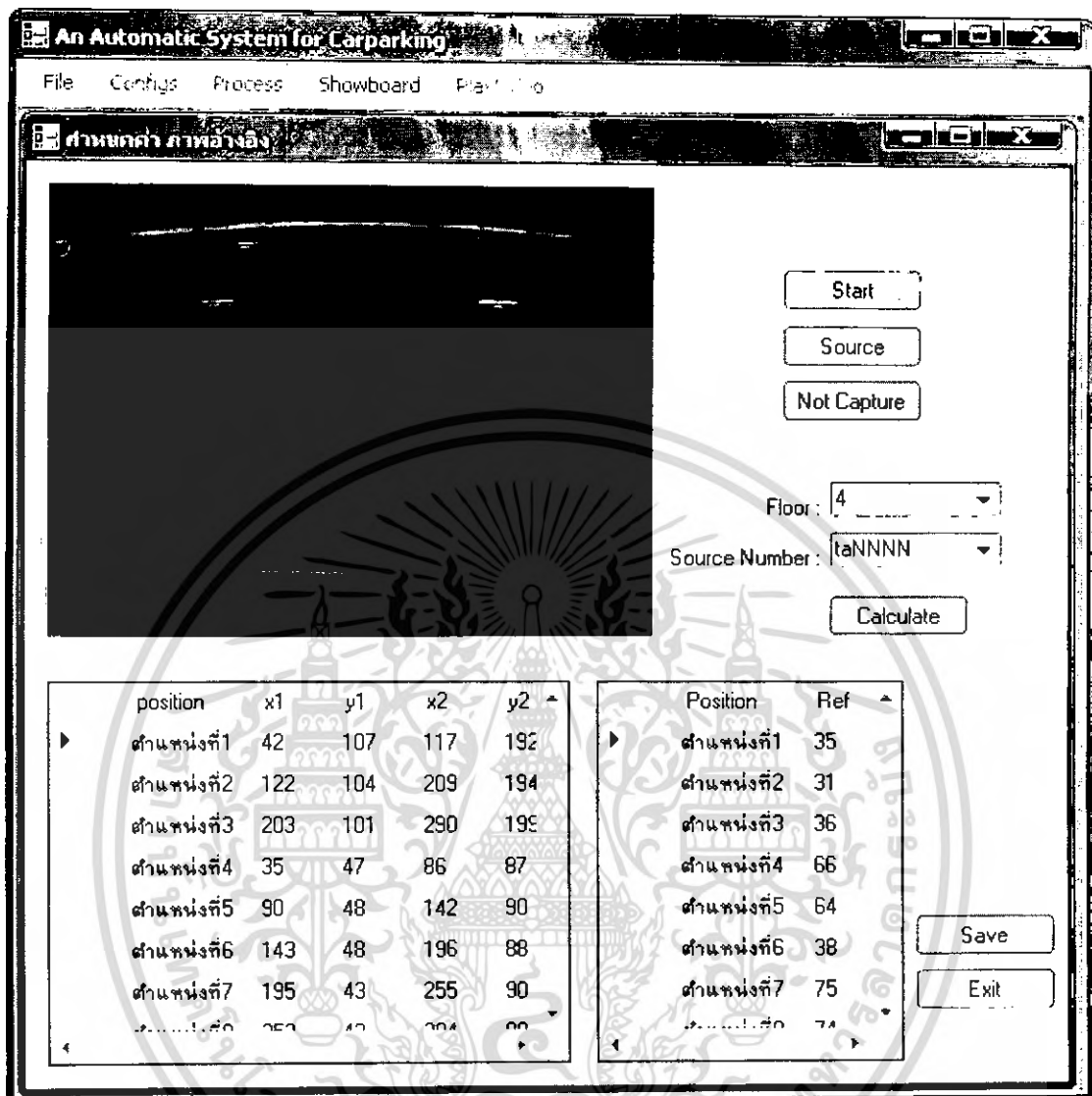
รูปที่ 3.8 หน้าจอกำหนดค่า IP Server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



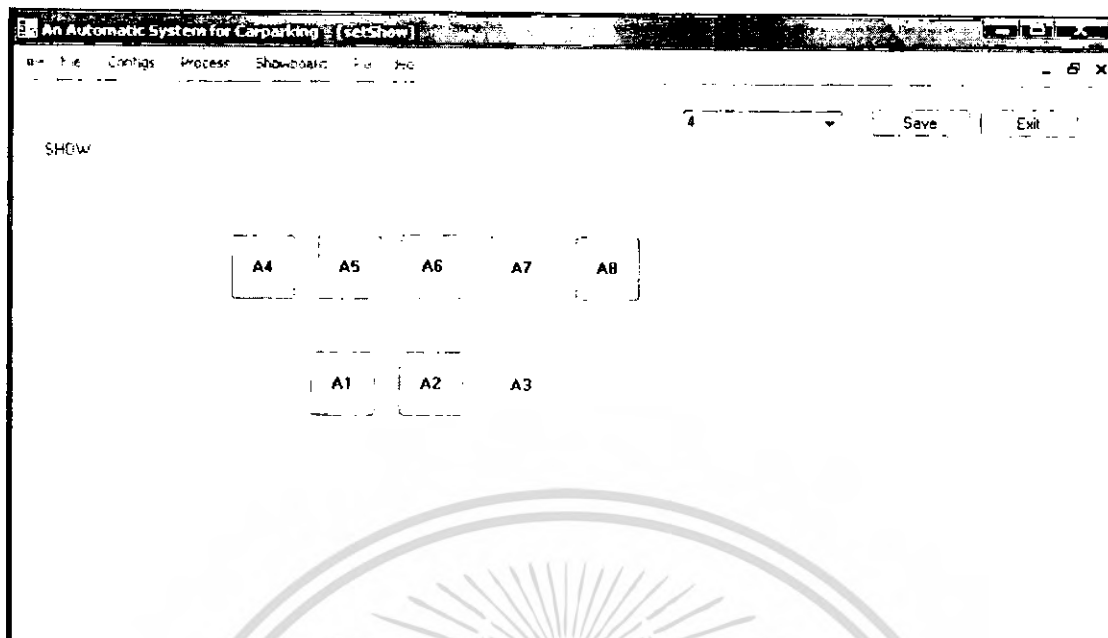
รูปที่ 3.9 หน้าจอกำหนดค่าพื้นที่จอดรถแต่ละจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

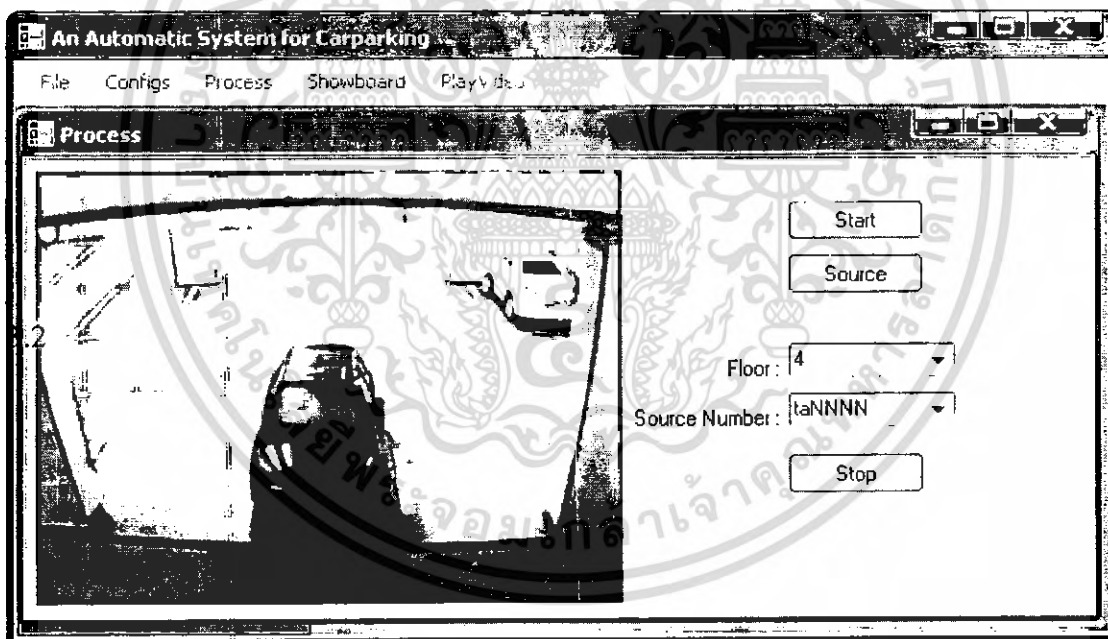


รูปที่3.10 หน้าจอกำหนดค่าอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

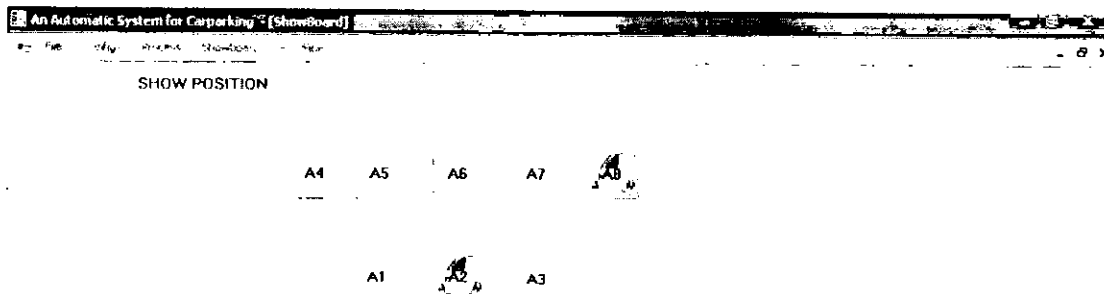


รูปที่ 3.11 หน้าจอกำหนดแผนผังที่จอดรถ



รูปที่ 3.12 หน้าจอแสดงส่วนประมวลผล

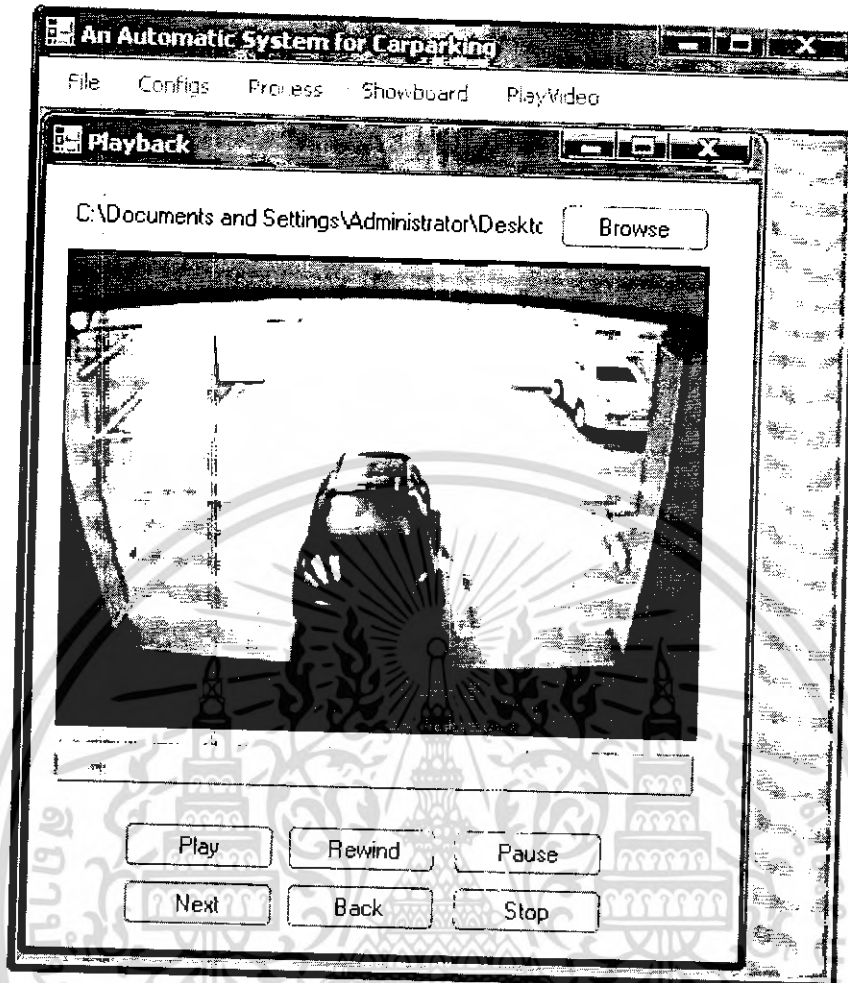
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.13 หน้าจอแสดงผลทางมอนิเตอร์

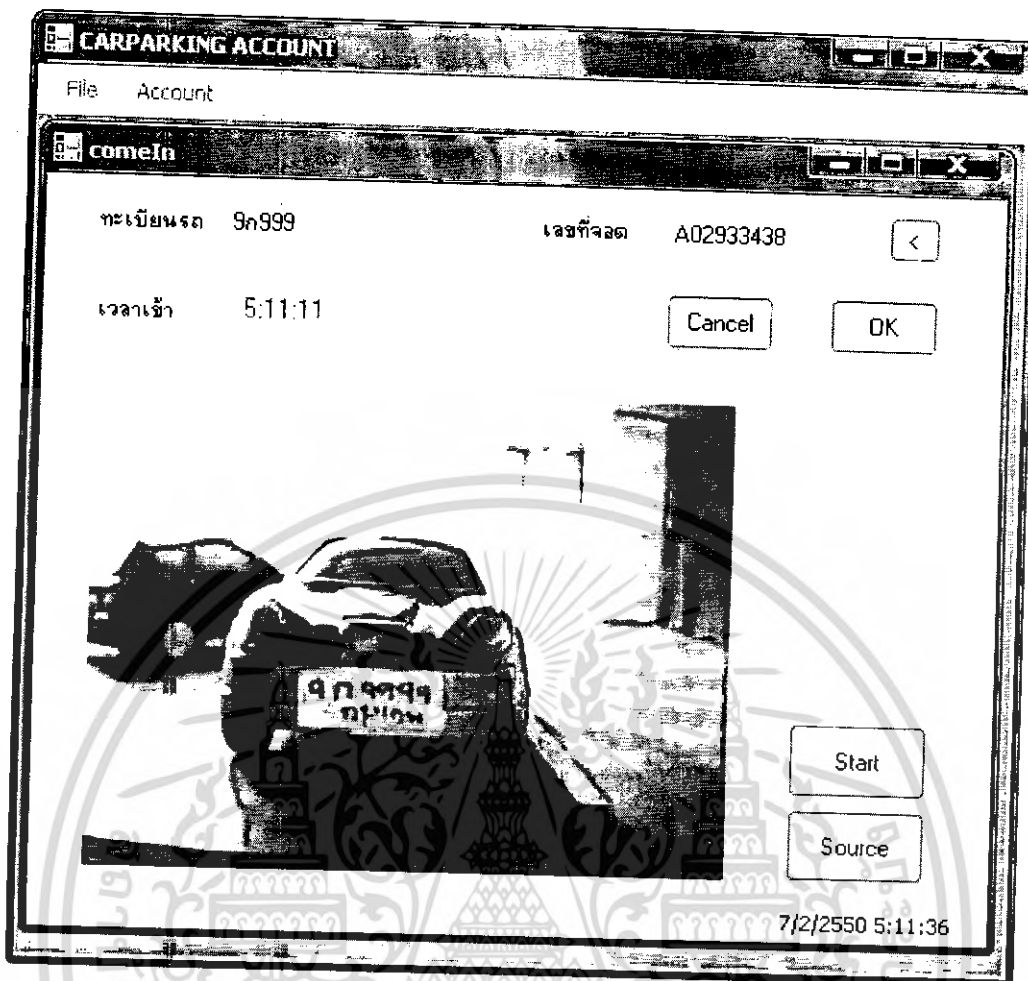


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



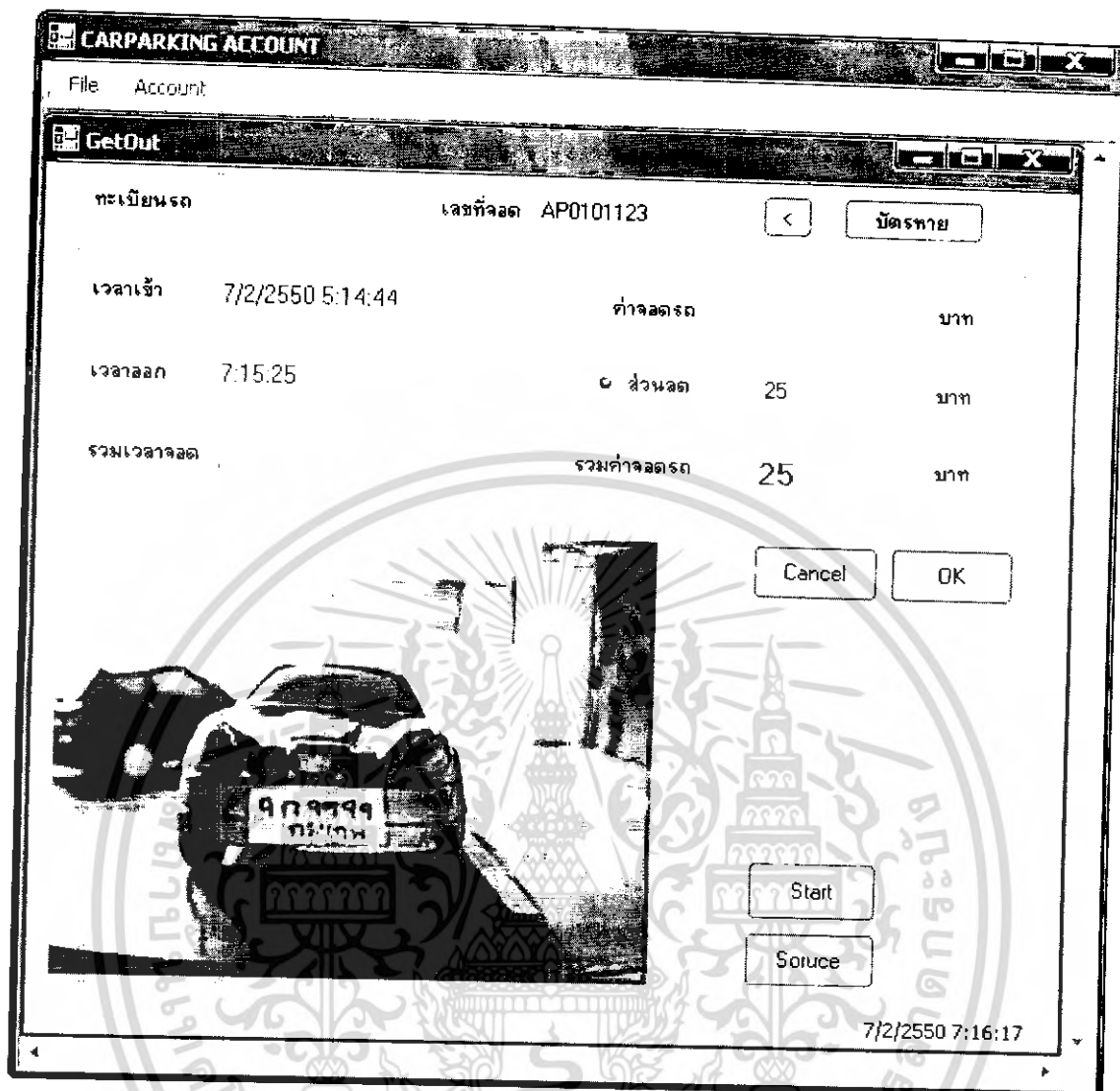
รูปที่ 3.14 หน้าจอแสดง ไฟล์ วิดีโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.15 หน้าจอเก็บข้อมูลลูกค้าทางเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

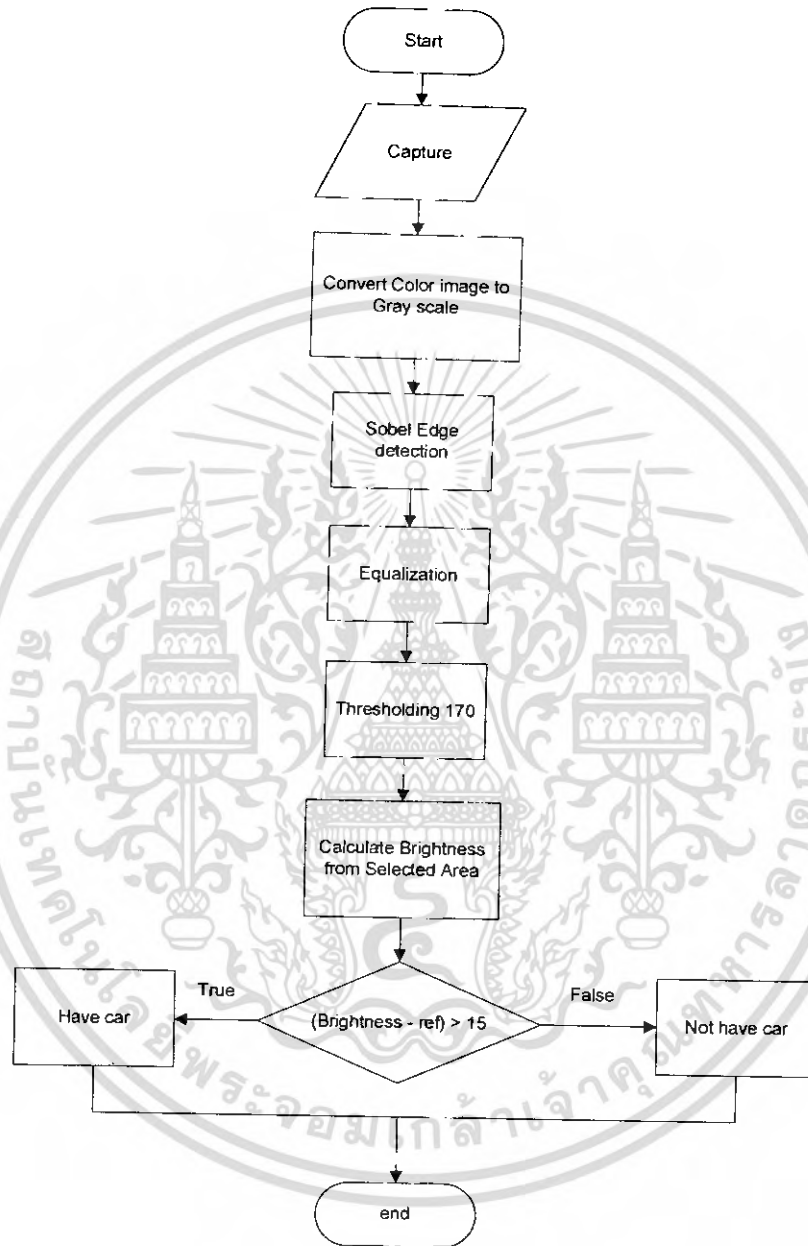


รูปที่ 3.16 หน้าจอเก็บข้อมูลลูกค้าทางออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 Flowchart การทำงานของระบบ

เป็นลำดับขั้นตอนในการประมวลผลว่าภาพที่จอตลอดนั้นว่างหรือไม่



รูปที่ 3.17 Flowchart การทำงานแยกแยะว่ามีรถหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อขั้นตอน	คำอธิบาย
1. Capture	เก็บภาพจากกล้องเป็นรูปแบบภาพสี เพื่อใช้ในการวิเคราะห์
2. Convert Color Image To Gray Scale	เปลี่ยนภาพที่ได้จากข้อ 1 เป็นภาพ Gray Scale ซึ่งใช้ในการวิเคราะห์
3. Sobel Edge Detection	นำภาพที่ได้จากข้อ 2 มาหาขอบของภาพโดยวิธีการ Sobel Edge Detection
4. Equalization	นำภาพที่ได้ จากข้อ 3 มาทำการปรับปรุงภาพ เพื่อให้ ได้ขอบภาพที่ชัดเจนยิ่งขึ้น โดยวิธี Equalization
5. Thresholding 170	ทำการ Thresholding ที่ระดับ 170 เพื่อ เปลี่ยนภาพจาก Gray Scale เป็นภาพ Binary โดยเลือกเอาเฉพาะขอบของภาพ
6. Calculate Brightness From Select Area	นำภาพที่ได้จากข้อ 5 มาคำนวณหาค่า Brightness ของภาพ โดยหาเฉพาะบริเวณที่ช่องที่จอดรถแต่ละช่อง
7. Brightness - ref	นำค่าที่ได้จากข้อ 6. ทำการ Normalize เพื่อให้ได้ค่าที่อยู่ในมาตรฐานเดียวกันจะได้ใช้ค่าในการขีดแบ่งค่าเดียวกัน เท่ากับ 15 โดยถ้ากรณีที่ค่าที่ได้จากการลบกัน มีค่ามากกว่า 15 จะถือว่า มีรถจอด และกรณีที่ลบกันแล้วได้ค่าน้อยกว่า 15 ถือว่า ไม่มีรถจอด
8. Have Car	จากข้อ 7. ถ้าเป็นกรณีที่มากกว่า 15 จะถือว่า มีรถจอดจะทำการเก็บค่า Status เป็น 1 คือ มีรถจอด
9. Not Have Car	จากข้อ 7. ถ้าเป็นกรณีที่น้อยกว่า 15 จะถือว่า ไม่มีรถจอดจะทำการเก็บค่า Status เป็น 0 คือ ไม่มีรถจอด

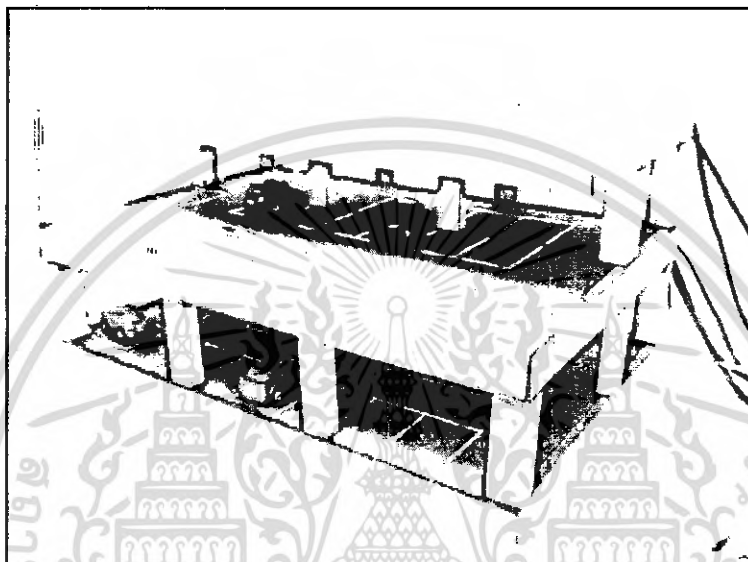
จากตารางจะอธิบายถึงแต่ละขั้นตอนของการทำงานในการประมวลผลว่ามีรถจอดอยู่หรือไม่ มีรถจอด ซึ่งหลังจากการประมวลผลจะทำการเก็บค่า Status ลงยังฐานข้อมูลเพื่อนำผลที่ได้ไปโชว์ที่ว่างต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

การทดลองโปรแกรมในส่วนนี้เป็นการทดลองหาค่าแรงที่จอตลอด แล้วแสดงผลทาง มอนิเตอร์ โดยในการทดลองนี้ได้จำลองโมเดลที่จอตลอด เพื่อง่ายในการทดลองและควบคุมตัวแปร ที่สำคัญคือ แสงและสีของรถยนต์



รูปที่ 4.1 แบบจำลองที่จอตลอด

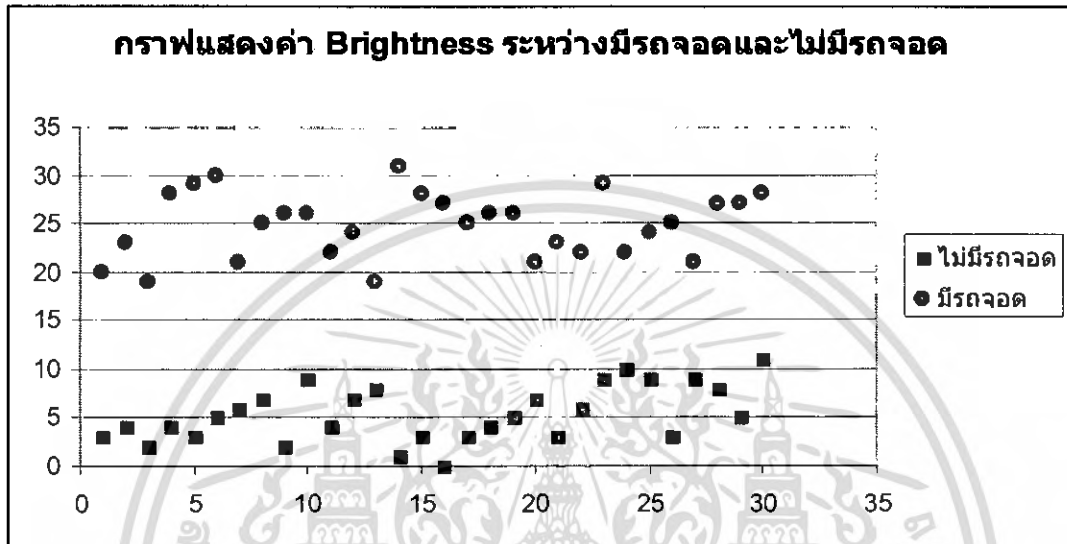
ขั้นตอนการทดลอง

1. หาค่า Brightness ที่เหมาะสม ที่จะนำไปใช้ Threshold เพื่อแยกแยะว่าภาพนั้นมีรถจอดหรือไม่
2. ทดลองโปรแกรมตรวจหาค่าแรงที่ว่าง โดยใช้สีของรถคงที่ และแสงเปลี่ยนแปลง
3. ทดลองโปรแกรมตรวจหาค่าแรงที่ว่าง โดยใช้แสงคงที่ และสีรถเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 การทดลองเพื่อหาค่า Brightness ที่เหมาะสม

ในการทดลองนี้ได้ทดลองวัดค่า Brightness ของตำแหน่งต่างๆ ระหว่างมีรถจอดและไม่มีการจอดแล้วนำมาพล็อตกราฟ ได้ผลดังนี้ ซึ่งค่า Brightness หาได้ตามหัวข้อที่ 2.6

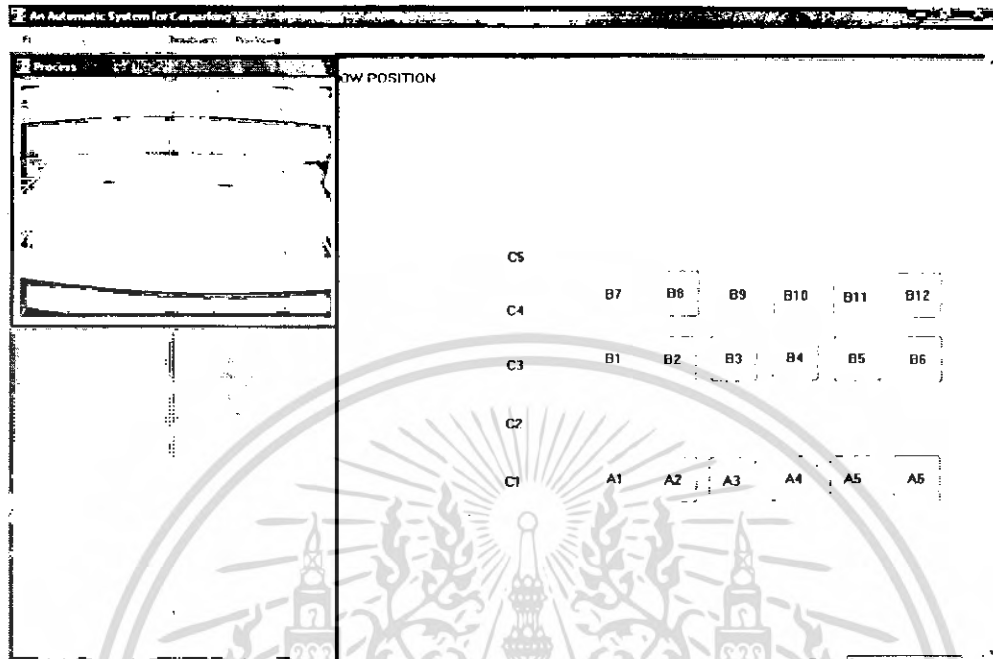


รูปที่ 4.2 กราฟแสดงค่า Brightness ของภาพระหว่างมีรถจอดและไม่มีการจอด

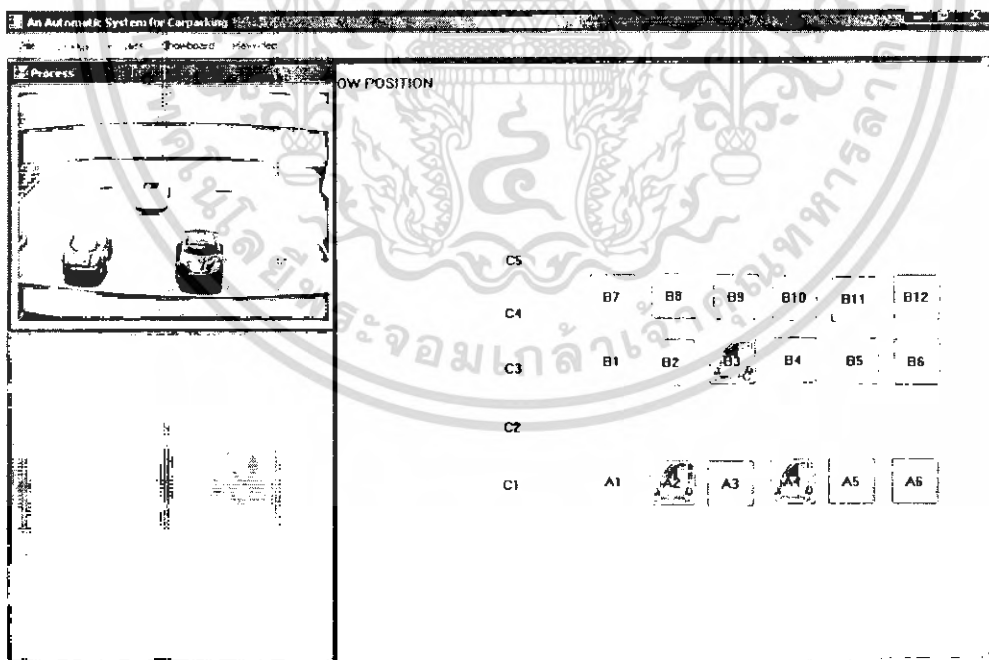
จากกราฟ สังเกตได้ว่าค่าของ Brightness ระหว่างมีรถจอดและไม่มีการจอดสามารถจัดกลุ่มแยกได้ โดยใช้ค่า Brightness = 15 สำหรับเป็นค่าขีดแบ่งเพื่อแยกแยะว่ามีรถจอดหรือไม่มีการจอด

4.2 การทดลองโปรแกรมตรวจหาตำแหน่งที่ว่างโดยใช้สีของรถคงที่ และแสงเปลี่ยนแปลง

4.2.1 กรณีแสงสว่าง



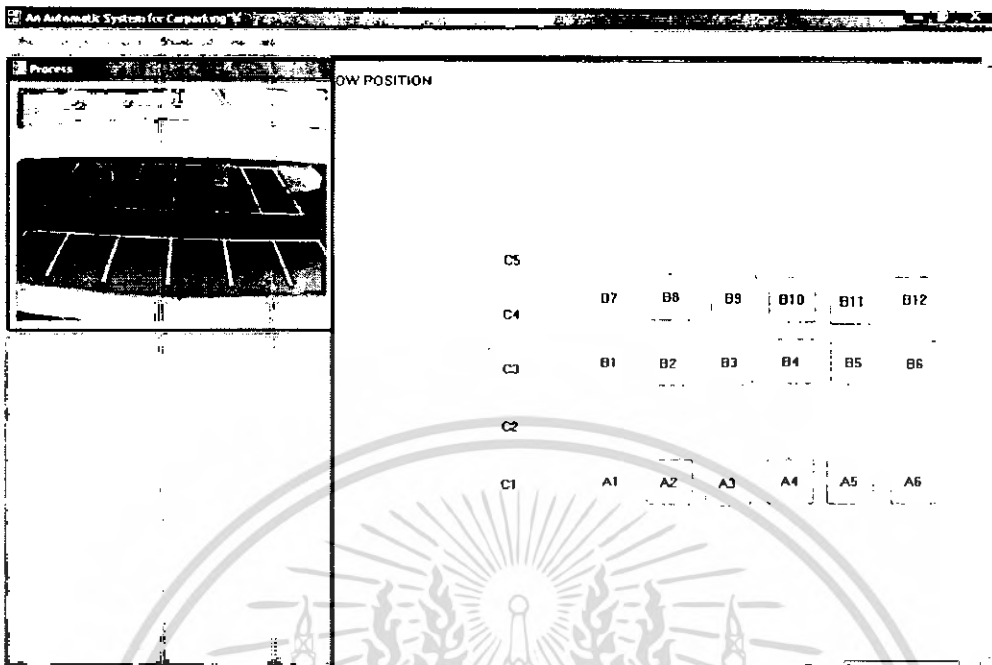
รูปที่ 4.3 ภาพที่จอครดและจอมอนิเตอร์ก่อนมีรถจอดกรณีแสงสว่าง



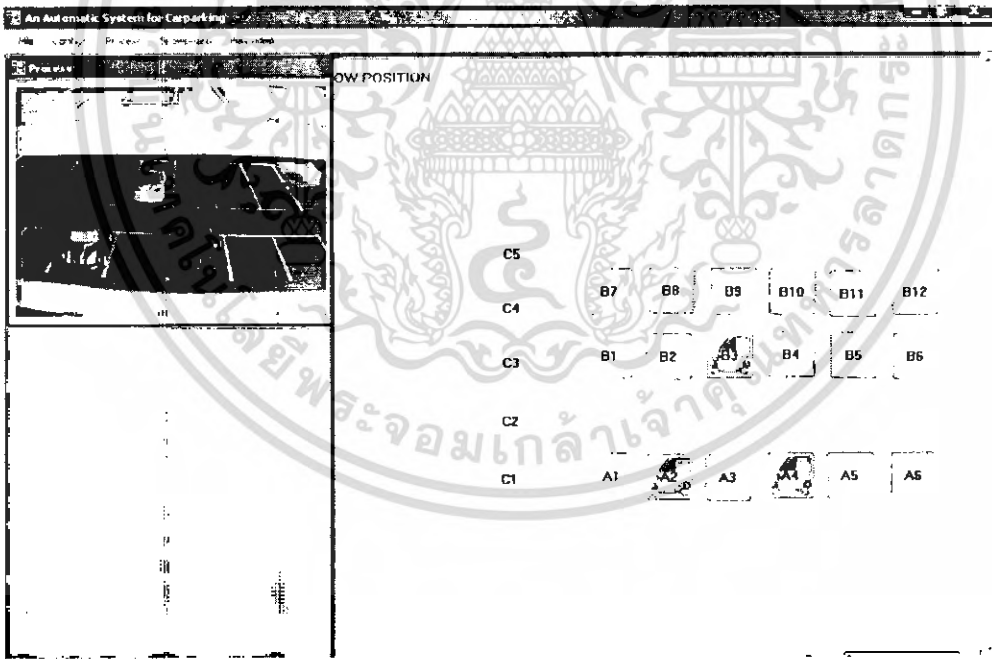
รูปที่ 4.4 ภาพที่จอครดและจอมอนิเตอร์ขณะมีรถจอดกรณีแสงสว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 กรณีแสงปกติ



รูปที่ 4.5 ภาพที่จอตรงและจอมอนิเตอร์ก่อนมีรถจอดกรณีแสงปกติ



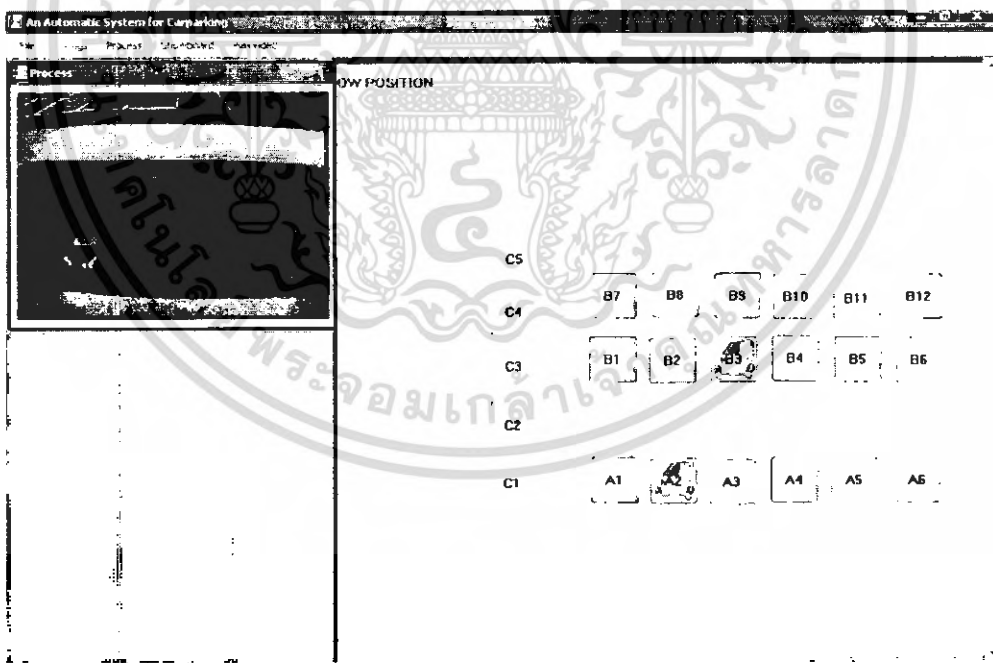
รูปที่ 4.6 ภาพที่จอตรงและจอมอนิเตอร์ขณะมีรถจอดกรณีแสงปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 กรณีแสงน้อยมาก



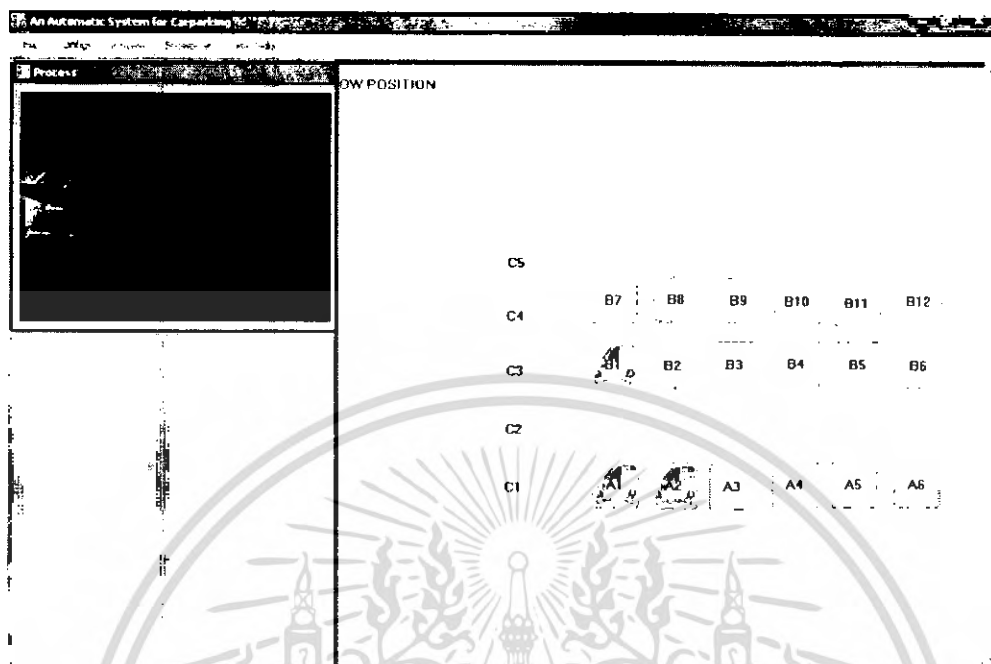
รูปที่ 4.7 ภาพที่จอร์รถและจอมอนิเตอร์ก่อนมีรถจอดกรณีแสงน้อยมาก



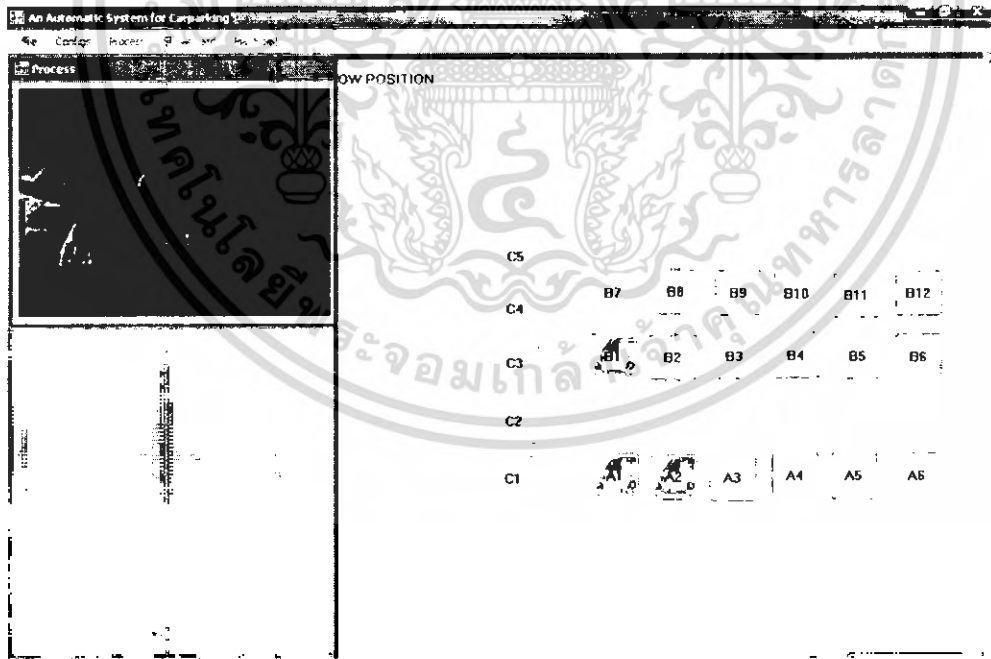
รูปที่ 4.8 ภาพที่จอร์รถและจอมอนิเตอร์ขณะมีรถจอดกรณีแสงน้อยมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 กรณีแสงมืด



รูปที่ 4.9 ภาพที่จอตรงและจอมอนิเตอร์ก่อนมีรถจอดกรณีแสงมืด



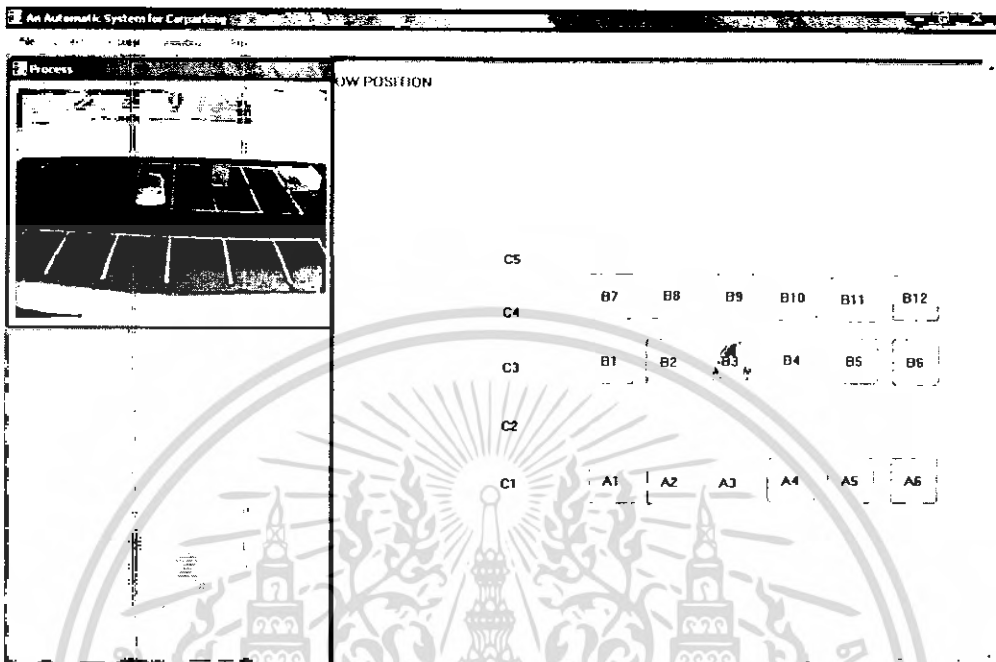
รูปที่ 4.10 ภาพที่จอตรงและจอมอนิเตอร์ขณะมีรถจอดกรณีแสงมืด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองโปรแกรมตรวจหาตำแหน่งที่ว่างโดยใช้แสงคงที่ และสวิตช์เปลี่ยนแปลง

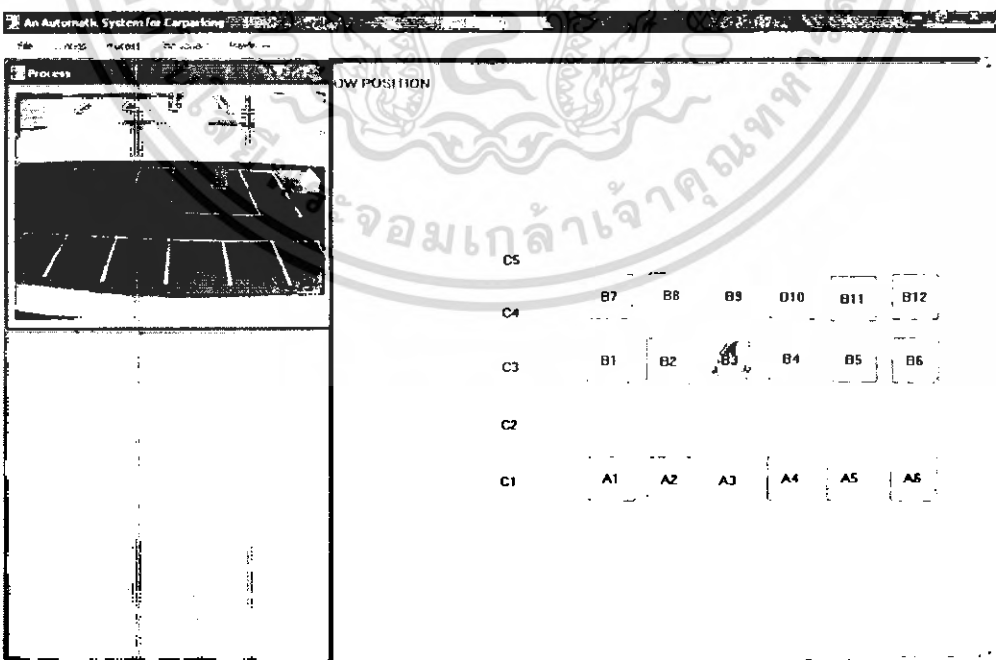
4.3.1 กรณีแสงปกติ

4.3.1.1 รถสีขาว



รูปที่ 4.11 ภาพที่จอครดและจอมอนิเตอร์ขณะมีรถสีขาวจอด

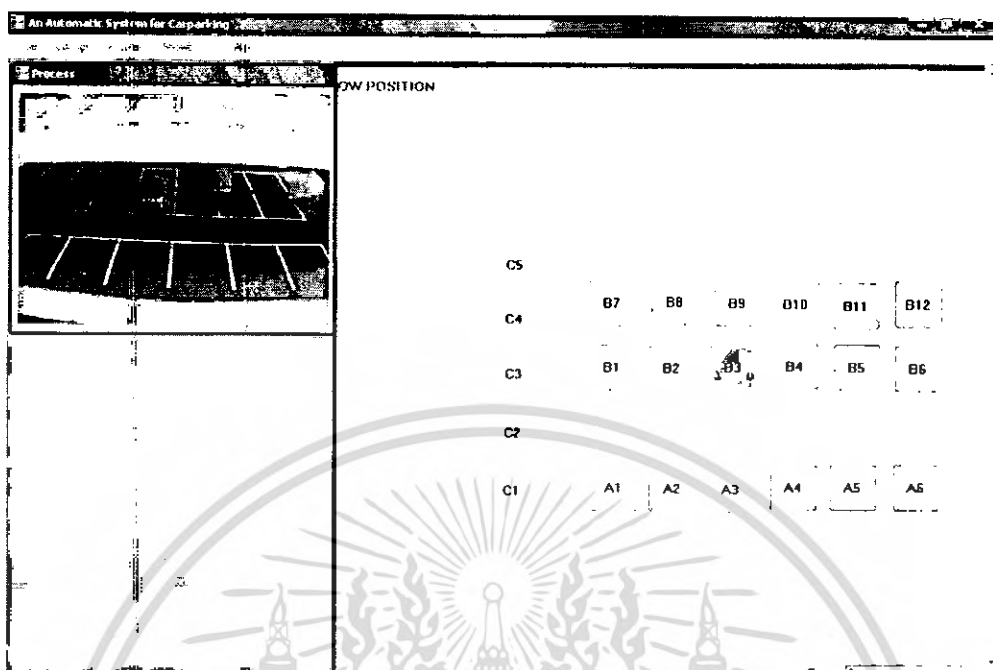
4.3.1.2 รถสีแดง



รูปที่ 4.12 ภาพที่จอครดและจอมอนิเตอร์ขณะมีรถสีแดงจอด

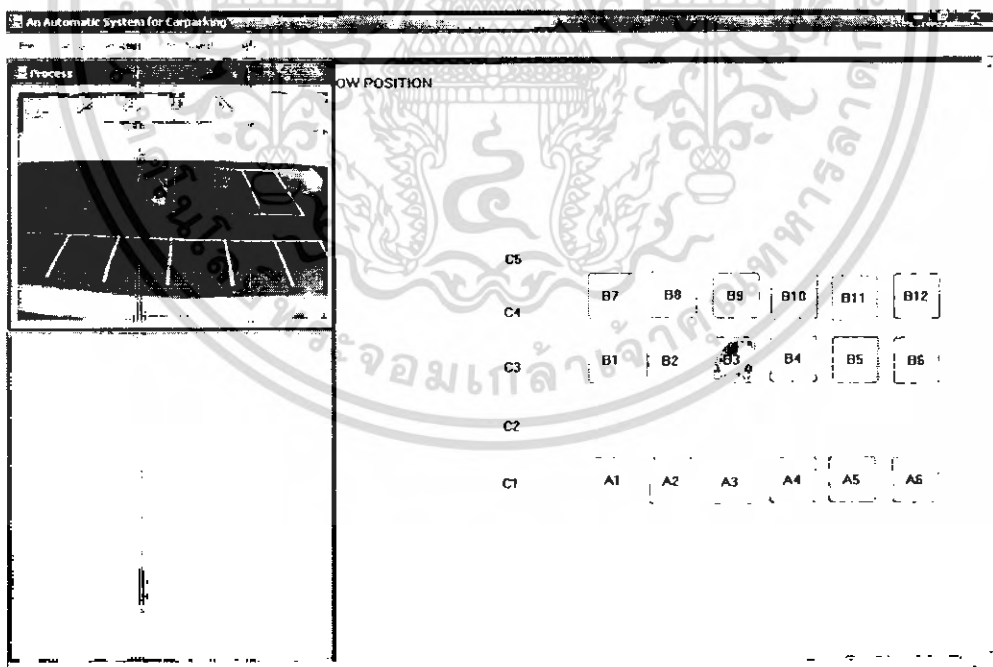
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1.3 รถสีฟ้า



รูปที่ 4.13 ภาพที่จอครดและจอมอนิเตอร์ขณะมีรถสีฟ้าจอด

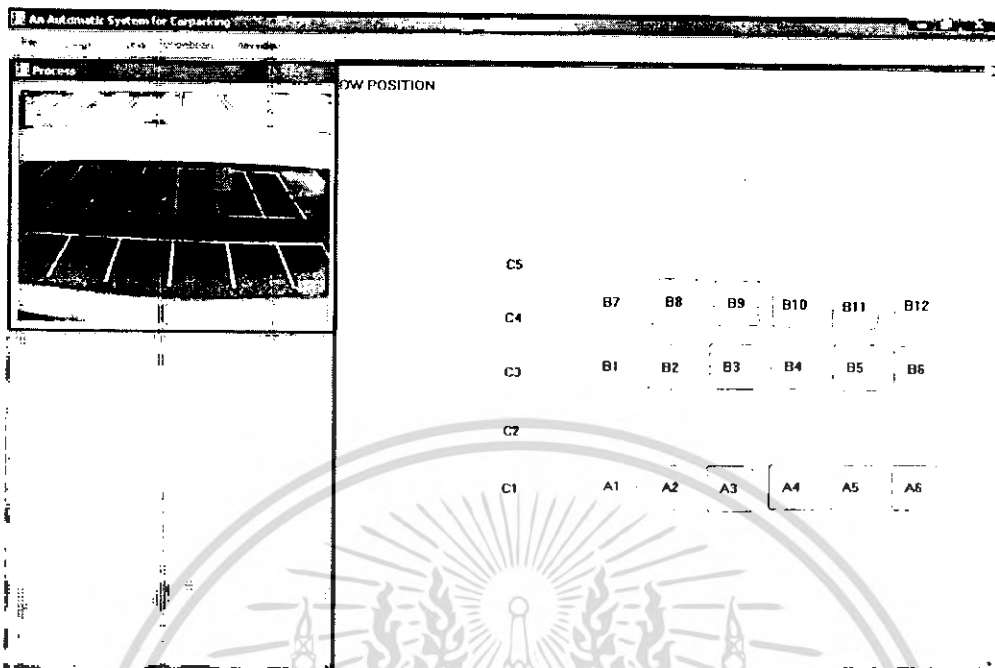
4.3.1.4 รถสีเทา



รูปที่ 4.14 ภาพที่จอครดและจอมอนิเตอร์ขณะมีรถสีเทาจอด

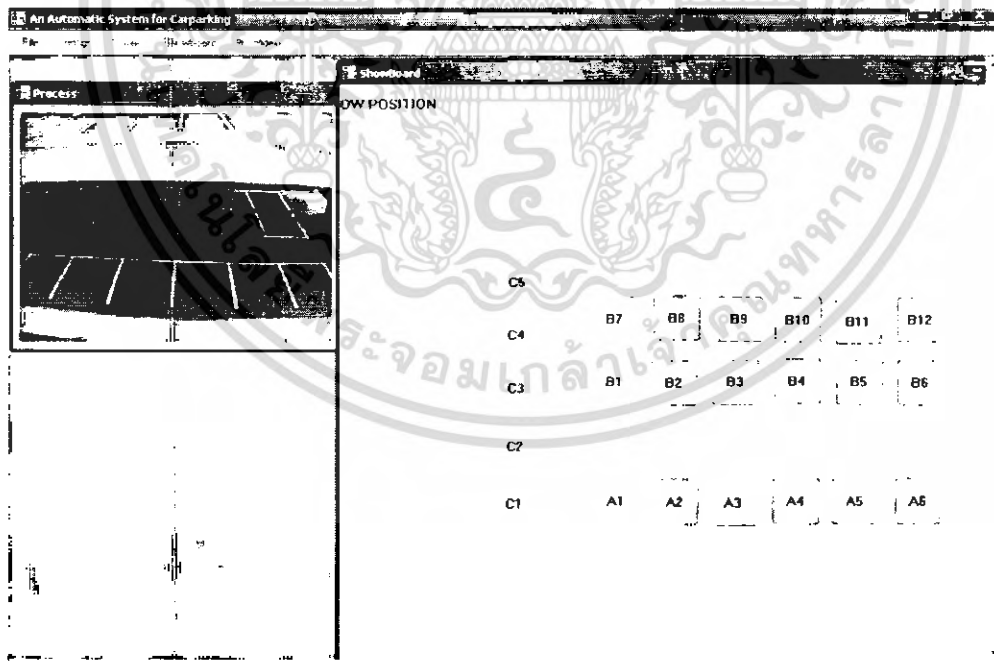
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1.5 รถสีดำ



รูปที่ 4.15 ภาพที่จอรถและจอมอนิเตอร์ขณะมีรถสีดำจอด

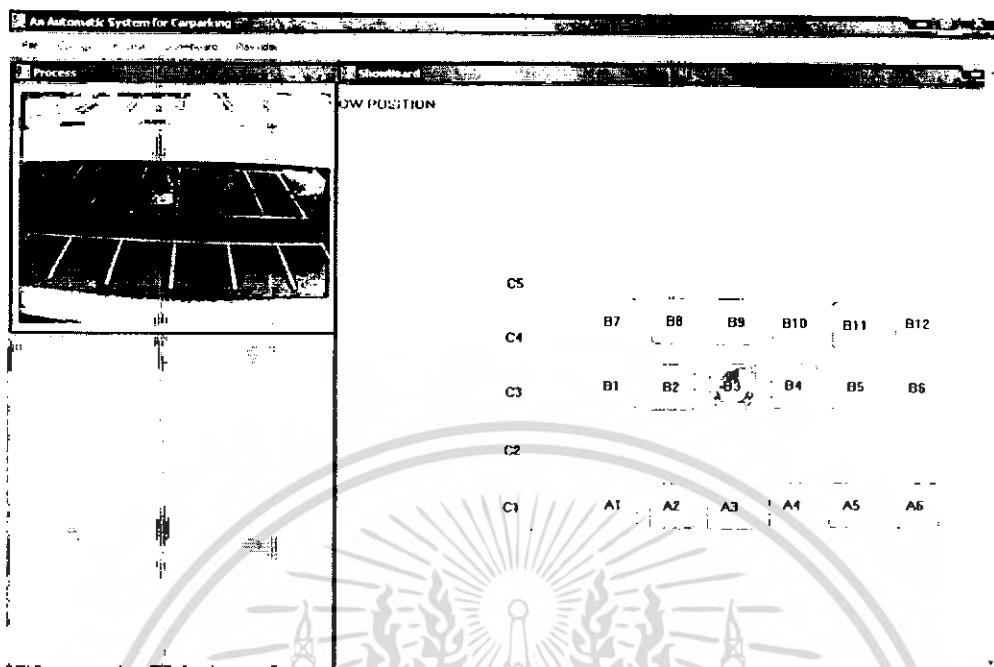
4.3.1.6 รถสีเขียว



รูปที่ 4.16 ภาพที่จอรถและจอมอนิเตอร์ขณะมีรถสีเขียวจอด

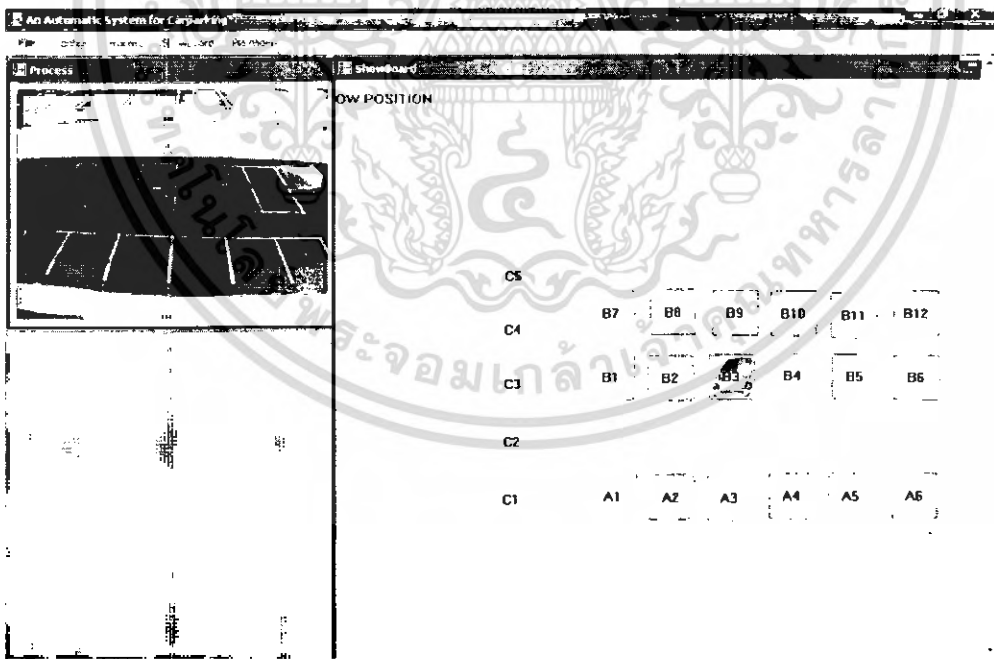
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1.7 รถสีเหลือง



รูปที่ 4.17 ภาพที่จอรถและจอมอนิเตอร์ขณะมีรถสีเหลืองจอด

4.3.1.8 รถสีน้ำเงิน

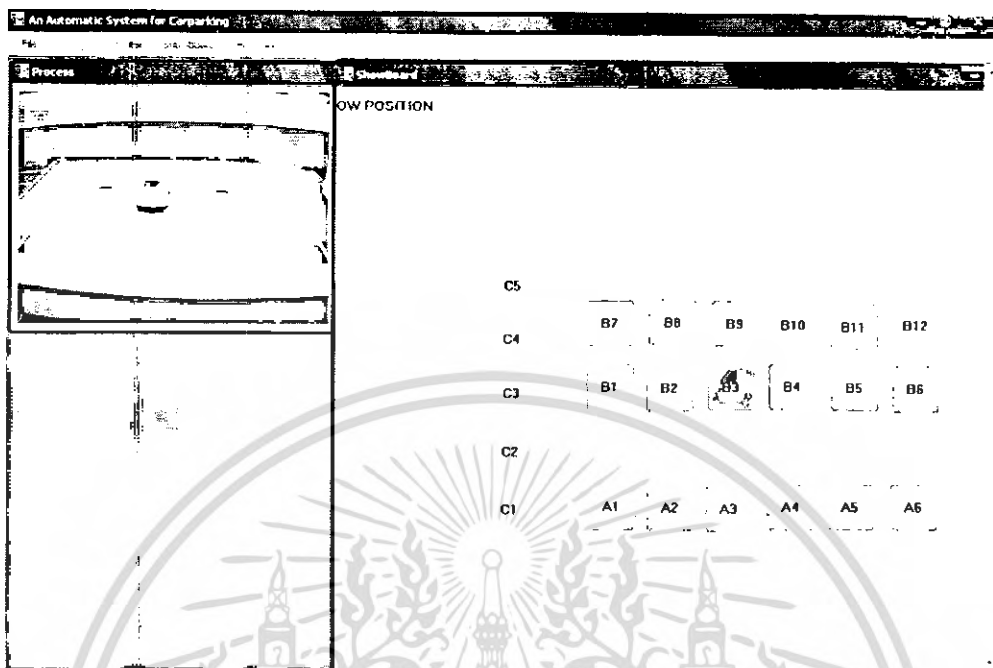


รูปที่ 4.18 ภาพที่จอรถและจอมอนิเตอร์ขณะมีรถสีน้ำเงินจอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

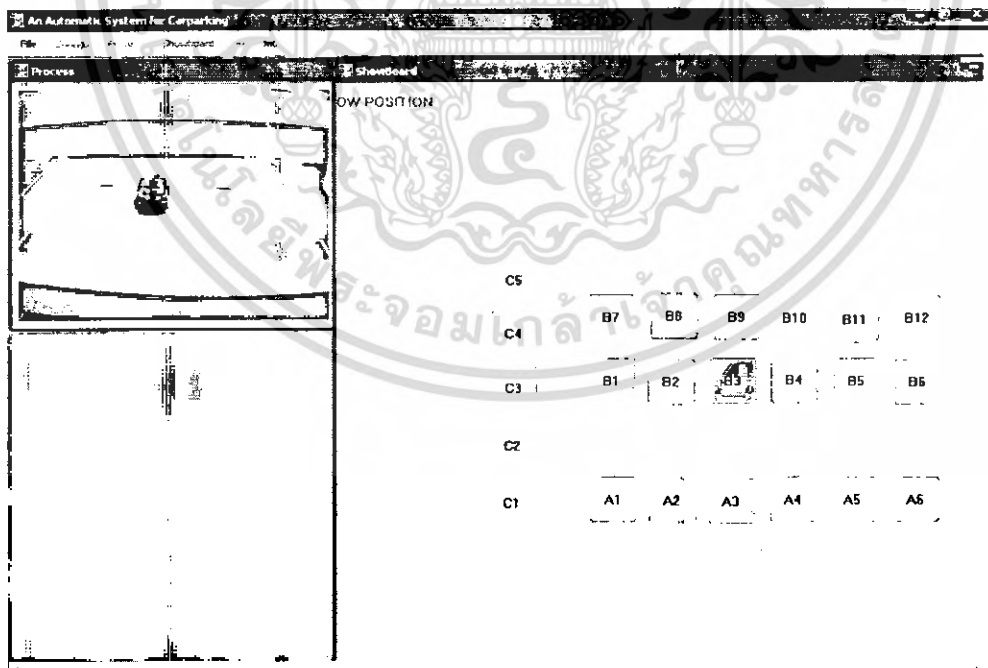
4.3.2 กรณีแสงสว่าง

4.3.2.1 รถสีขาว



รูปที่ 4.19 ภาพที่จอรถและจอมอนิเตอร์ขณะมีรถสีขาวจอด

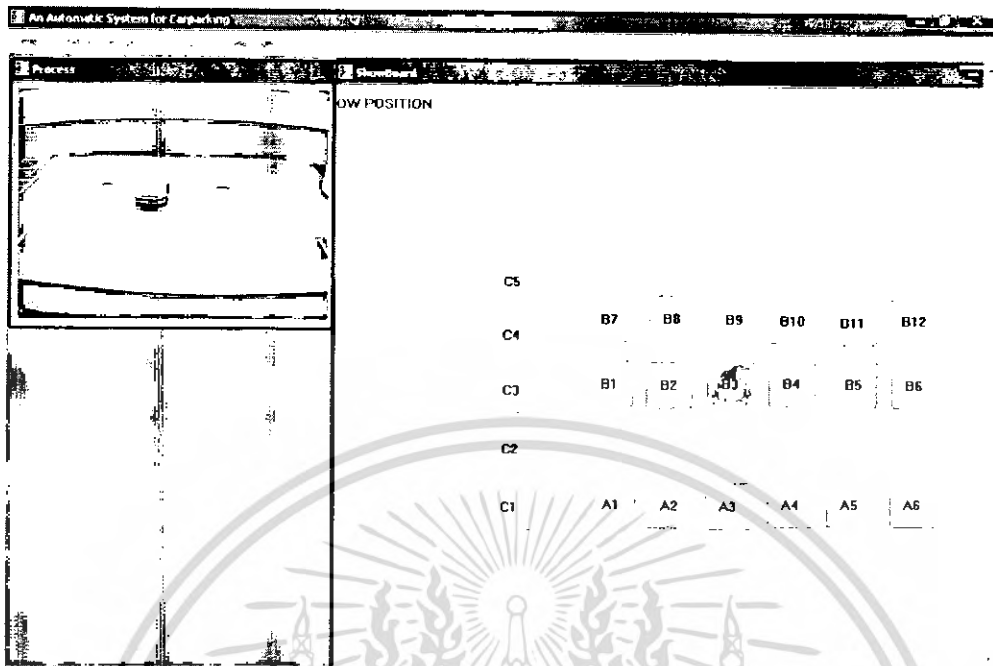
4.3.2.2 รถสีแดง



รูปที่ 4.20 ภาพที่จอรถและจอมอนิเตอร์ขณะมีรถสีแดงจอด

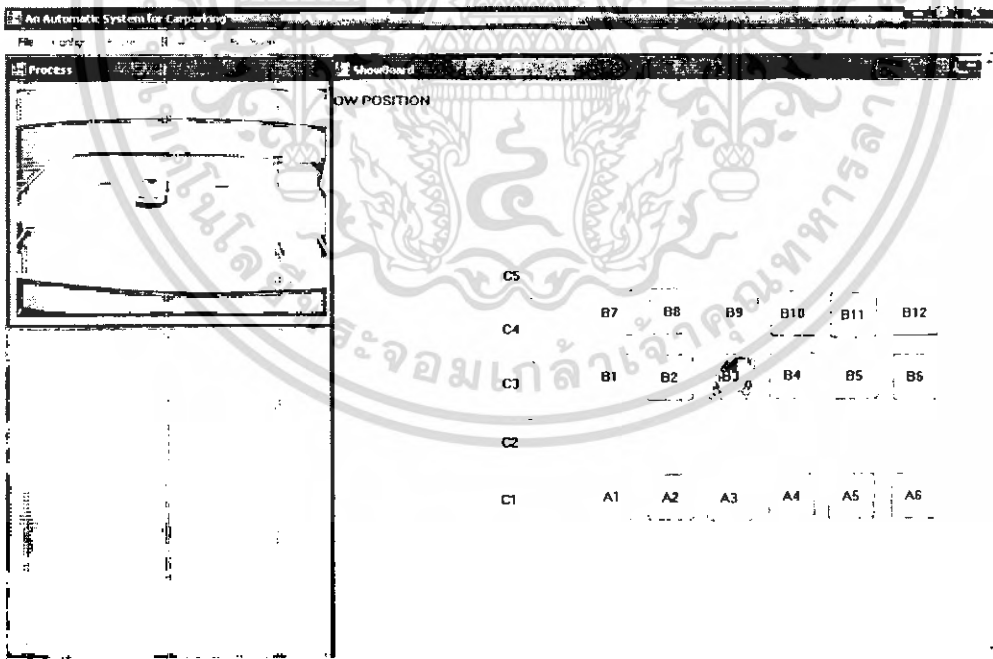
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2.3 รหัสสีฟ้า



รูปที่ 4.21 ภาพที่จอตรงและจอมอนิเตอร์ขณะมีรถสีฟ้าจอด

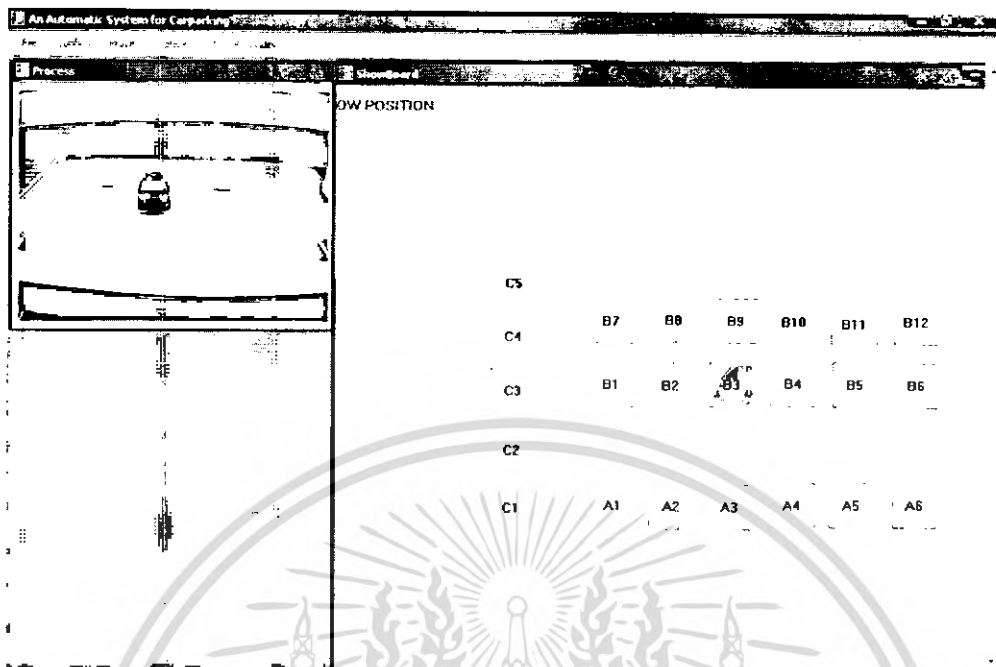
4.3.2.4 รหัสสีเทา



รูปที่ 4.22 ภาพที่จอตรงและจอมอนิเตอร์ขณะมีรถสีเทาจอด

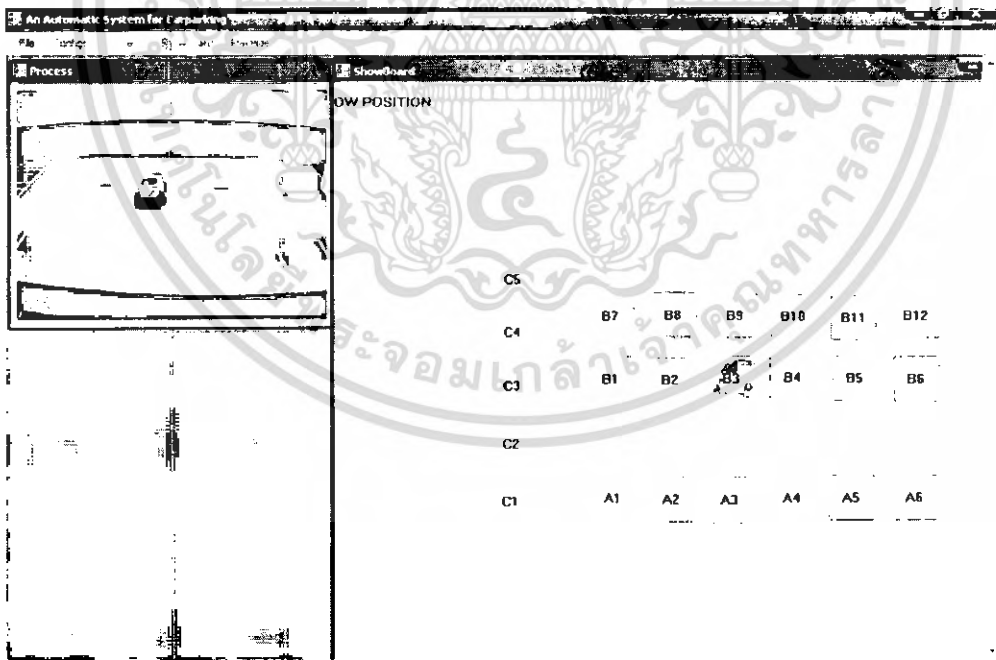
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2.5 รถสีดำ



รูปที่ 4.23 ภาพที่จอครดและจอมอนิเตอร์ขณะมีรถสีดำจอด

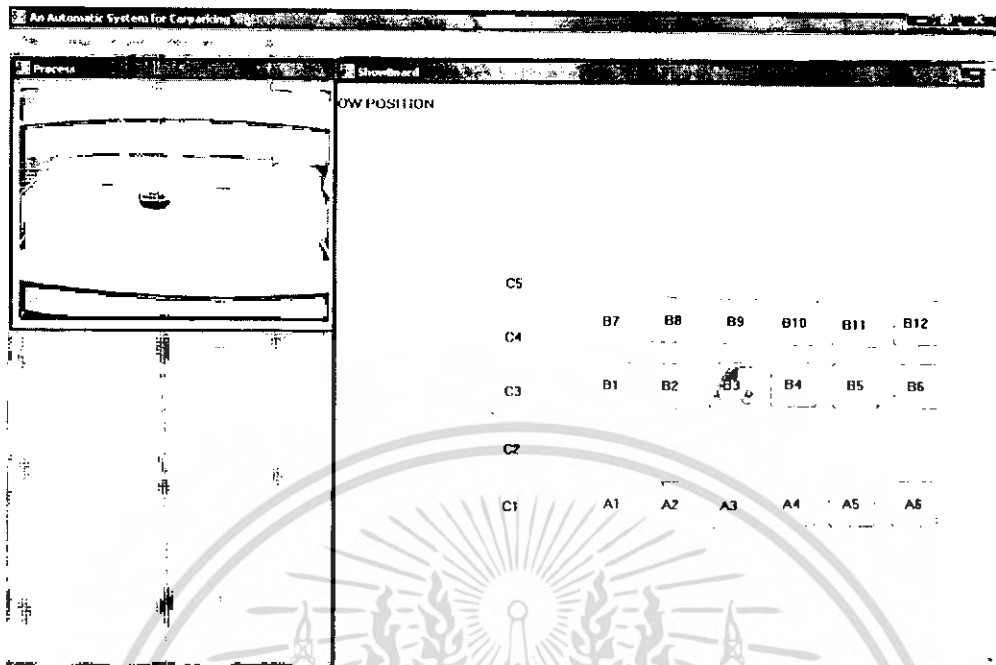
4.3.2.6 รถสีเขียว



รูปที่ 4.24 ภาพที่จอครดและจอมอนิเตอร์ขณะมีรถสีเขียวจอด

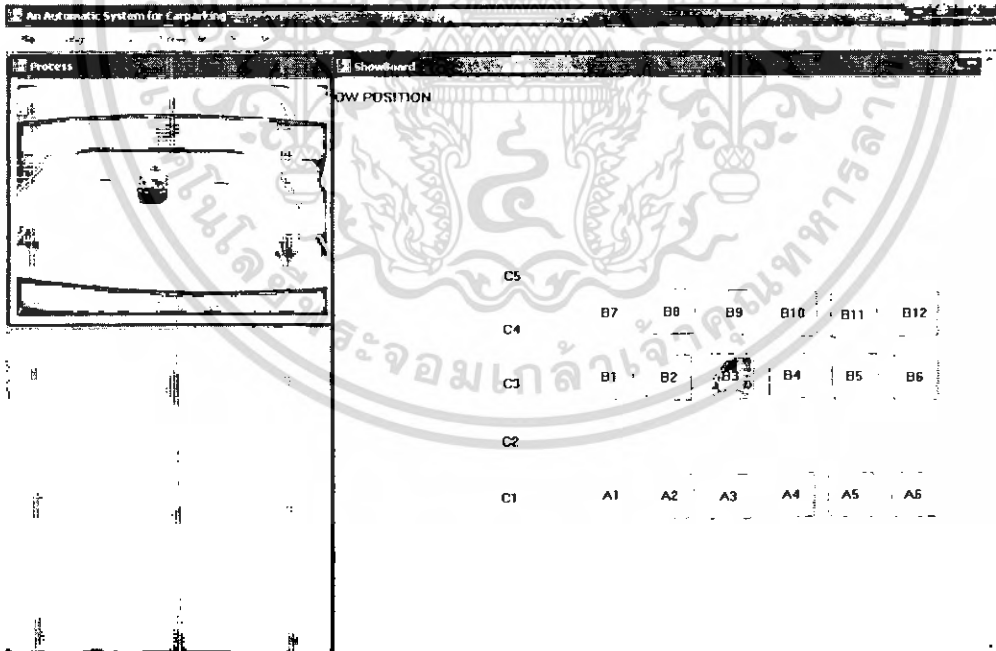
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2.7 รถสีเหลือง



รูปที่ 4.25 ภาพที่จอรถและจอมอนิเตอร์ขณะมีรถสีเหลืองจอด

4.3.2.8 รถสีน้ำเงิน



รูปที่ 4.26 ภาพที่จอรถและจอมอนิเตอร์ขณะมีรถสีน้ำเงินจอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และสรุป

5.1 บทวิจารณ์และสรุป

จากการทดลองที่ได้แบ่งเป็น 3 ตอน โดยในตอนที่ 1 ได้สรุปผลของค่า Brightness ที่จะนำไปใช้ในการแยกแยะภาพว่ามีรอยจุดหรือไม่มีรอยจุดดังนี้

$$\text{Status} = \begin{cases} 1 \text{ (มีรอยจุด)} & ; \text{Brightness} > = 15 \\ 0 \text{ (ไม่มีรอยจุด)} & ; \text{Brightness} < 15 \end{cases}$$

ในตอนที่ 2 ได้ทำการทดลองโปรแกรมโดยให้แสงมีการเปลี่ยนแปลง 4 กรณี คือ แสงสว่าง แสงปกติ แสงน้อย และ แสงมืด โดยควบคุมให้สีของรอยจุดที่ ผลการทดลองพบว่าแสงมีผลต่อการแยกแยะภาพว่ามีรอยจุดหรือไม่ โดยปัญหาที่พบคือแสงที่ทำให้โปรแกรมไม่สามารถแยกแยะภาพได้ว่ามีรอยจุดหรือไม่คือ แสงน้อยและแสงมืด

ในตอนที่ 3 ได้แบ่งการทดลองย่อยเป็น 2 ตอน คือ ตอนที่ 1 ได้เลือกใช้แสงปกติและควบคุมแสงให้คงที่ หลังจากนั้นจึงนำรถที่มีสีต่างๆมาทดสอบได้แก่รถสี ขาว แดง ฟ้า เทา ดำ เขียว เหลือง และ น้ำเงิน ผลการทดลองพบว่า สามารถแยกแยะได้ว่ามีรอยจุดหรือไม่ แต่มีรถสีเขียวและดำที่ไม่สามารถแยกแยะได้ว่ามีรอยจุดหรือไม่ ตอนที่ 2 ได้เปลี่ยนไปใช้แสงสว่างและควบคุมให้คงที่เช่นเดียวกัน ผลการทดลองพบว่า สามารถแยกแยะรถได้ทุกสีว่ามีรอยจุดหรือไม่

จากการทดลองทั้ง 3 ตอนทำให้สรุปได้ว่าในการที่จะแยกแยะรถได้ว่ามีรอยจุดหรือไม่มีรอยจุด จะต้องคำนึงถึงปัจจัย 2 อย่างด้วยกันคือ แสงในที่จอดรถ หากแสงไม่เพียงพอก็จะทำให้ไม่สามารถแยกแยะว่ามีรอยจุดหรือไม่ อย่างที่สองคือ สีของรถ ถ้าหากเป็นรถสีมืดและนำไปจอดในที่มืดก็จะไม่สามารถแยกแยะได้เช่นกันว่ามีรอยจุดหรือไม่ ซึ่งได้มีการแก้ปัญหาโดยการปรับปรุงภาพด้วยวิธีการ Histogram Equalization ตามหัวข้อที่ 2.7 ซึ่งจากการทดลองพบว่าวิธี Histogram Equalization สามารถแก้ปัญหาเรื่องความแตกต่างของแสงในที่จอดรถได้อย่างมีประสิทธิภาพ ทำให้ประสิทธิภาพของโปรแกรมดีขึ้น

5.2 ปัญหาและอุปสรรคที่พบ

- ปัญหาเรื่องแสง ซึ่งเป็นตัวแปรที่เปลี่ยนไปตามสิ่งแวดล้อมตามช่วงเวลาซึ่งถ้าแสงน้อยมากจะทำให้ประสิทธิภาพของระบบลดลง
- ปัญหาเรื่องสีของรถที่เข้ามาจอด ซึ่งถ้าแสงน้อยและรถที่จอดมีสีดำหรือสีมืดจะทำให้กลมกลืนกับฉากหลัง ทำให้ประสิทธิภาพของระบบลดลง
- ปัญหาการวางมุมกล้อง การวางมุมกล้องที่สูงจะทำให้ประสิทธิภาพดีกว่าการวางมุมกล้องที่ต่ำ เพราะจะลดการบังกล้องในที่จอดรถ

5.3 แนวทางการศึกษาต่อ

- นำไปทดสอบติดตั้งกับสถานที่จริง เพื่อหาค่าพารามิเตอร์ต่างๆที่เหมาะสมเพื่อให้โปรแกรมมีความสมบูรณ์มากยิ่งขึ้น
- เปลี่ยนจากกล้อง Web Cam ไปเป็นกล้องวงจรปิดเพื่อให้ได้ความคงทนและความรวดเร็วในการตอบสนองการใช้งาน
- ศึกษาหาวิธีอื่นที่ทำให้สามารถตรวจสอบวัตถุที่เป็นรถได้
- เพิ่มเติมในส่วนการใช้งานการเก็บค่าจอดรถให้เป็นระบบอัตโนมัติมากยิ่งขึ้น

บรรณานุกรม

Alasdair McAndrew . **Introduction to Digital Image Processing with Matlab** .United States of America:Course Technology.

พร้อมเลิศ หล่อวิจิตร .2549.**คู่มือเรียน Visual Basic 2005** . กรุงเทพฯ : โปรวิชั่น.

พงศธร เกียรติ์ และ พัฒนวัฒน์ เรืองวัฒน์ไพศาล “ระบบลานจอดรถยนต์ไร้ผู้ควบคุม”

วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2548.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

กล้อง

กล้องที่เลือกใช้เป็นกล้อง Webcam เพราะมีราคาถูกและสามารถใช้ในการเก็บภาพได้คุณภาพดีพอสมควร ขีห้อที่เลือกใช้คือ OKER ดังรูป



รูปที่ ก-1 รูปกล้องขีห้อ OKER

คุณสมบัติของกล้อง

- High-quality VGA CMOS sensor
- Manual focus
- Video capture: up to 640*480 pixels
- Still image capture: up to 640*480 pixels
- Frame rate: up to 30 frame per second
- Price : 550

คุณสมบัติที่เลือกใช้

- Video capture: 320*240 pixels
- Frame rate: 5 frame per second
- Picture Grayscale 256 level

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ActiveX Control (VideoOCX และ VideoOCXTools)

ในปัจจุบันกล้อง web cam ถือได้ว่าเป็นอุปกรณ์ที่ได้รับความนิยมจากผู้ใช้เป็นจำนวนมาก จึงได้มีผู้พัฒนาเครื่องมือต่างๆ ที่ช่วยในการเขียน โปรแกรมที่ใช้ในการติดต่อกับกล้อง web cam ออกมามากมาย เพื่อตอบสนองต่อความต้องการของผู้ใช้ ซึ่งในโครงการนี้ได้เลือกใช้เครื่องมือ VideoOCX ซึ่งเป็น ActiveX control ที่ช่วยให้ผู้เขียน โปรแกรมนั้น สามารถเขียน โปรแกรมติดต่อกับกล้อง web cam เพื่อใช้ในการจับภาพและทำ image processing ได้อย่างง่ายดาย โดยเพียง กำหนดค่าตัวแปรให้กับ VideoOCX และเรียกใช้งานผ่าน method ที่มีให้มา ตัวอย่าง method ที่สำคัญได้แก่

Method ที่สำคัญ

Syntax : **BOOL** *object*.**Start()**

Description : initiate the background capture process

Parameter : none

Return Type : true if successful

Syntax : **BOOL** *object*.**Stop()**

Description : end the internal capturing process

Parameter : none

Return Type : true if the internal capture process was stopped

Syntax : **BOOL** *object*.**ShowDriverDlg()**

Description : open dialog that enables the user to choose a video driver

Parameter : none

Return Type : true if successful

Syntax : void *object*.**SetErrorMessages(BOOL flag)**

Description : display every error that occurs

Parameter :true to turn error messages on

Return Type : none

Syntax : **BOOL** *object*.**Init()**

Description : initialize VideoOCX and connect to the specified image source

Parameter : none

Return Type : true if successful

Syntax : **LPCTSTR** *object*.**GetLastErrorString()**

Description : returns the message of the last occurred error

Parameter : none

Return Type : error message

Syntax : **long** *object*.**GetGrayImageHandle()**

Description : receive an empty grayscale image

Parameter : none

Return Type : handle to the new allocated image

Syntax : **long** *object*.**GetColorImageHandle()**

Description : receive an empty color image

Parameter : none

Return Type : handle to the new allocated image

Syntax : **BOOL** *object*.**ToGray(long colorimagehandle, long grayimagehandle)**

Description : convert a color image to grayscale

Parameter : color image handle and grayscale image handle

Return Type : true if successful

Syntax : **long** *object*.**GetDataPointer(long imagehandle)**

Description : get a pointer to the image data

Parameter : image handle

Return Type : address of image data

Syntax : long *object*.**Capture**(long *imagehandle*)

Description : capture the current video image

Parameter : current video image is copied into the handle

Return Type : address to image data array

Syntax : VOID *object*.**Show**(long *imagehandle*)

Description : display any image in the control

Parameter : handle to the image

Return Type : none

Syntax : VARIANT *object*.**GetMatrix**(long *imagehandle*)

Description : get dimensional array to the image pixels

Parameter : handle to an image

Return Type : 2 or 3 dimensional matrix containing the image pixels

Syntax : void *object*.**ReleaseMatrixToImageHandle**(long *imagehandle*)

Description : release memory for the last matrix

Parameter : handle to an image

Return Type : none

Syntax : void *object*.**ReleaseImageHandle**(long *imagehandle*)

Description : release the memory for an image

Parameter : handle to an image

Return Type : none

Syntax : BOOL *object*.**Close**()

Description : closes the current connection

Parameter : none

Return Type : true if successful

Syntax : **BOOL** *object*.**ToPicture**(*long imagehandle*)

Description : assign image handle any picture to a picture box

Parameter : imagehandle Handle to an image

Return Type : True if successfull

Syntax : **BOOL** *object*.**ShowSourceDlg**()

Description : user to choose the video source and to adjust some image parameters

Parameter : none

Return Type : true if successfull

Syntax : **BOOL** *object*.**AVISaveMovieInit**(*String filename*)

Description : Initialize continuoud capturc into an AVI movie

Parameter : String filename

Return Type : True if SaveMovie mode was initialized

Syntax : **BOOL** *object*.**AVISaveMovieStart**()

Description : Start contunous capture into an AVI file

Parameter : None

Return Type : True if Capture was started

Syntax : **BOOL** *object*.**AVISaveMovieStop**()

Description : Stop contunous capture into an AVI file

Parameter : None

Return Type : True if Capture was stopped

Syntax : **BOOL** *object*.**AVISaveMovieClose**()

Description : Closc continuous AVI-Movie capture mode

Parameter : None

Return Type : True if AVI capture was closed successfully

Syntax : **BOOL object.AVIPlaybackMovieInit(String filename)**

Description : Initialize AVI Movie playback mode

Parameter : String filename

Return Type : True if AVI movie was opened successfully

Syntax : **BOOL object.AVIPlaybackMovieStart()**

Description : Start to playback an AVI movie.

Parameter : None

Return Type : True if movie started successfully

Syntax : **BOOL object.AVIPlaybackMovieRewind()**

Description : Allows you to restart AVI movie playback (seek to the first frame).

Parameter : None

Return Type : True if successfully

Syntax : **BOOL object.AVIPlaybackMovieStop()**

Description : Stop AVI playback started with

Parameter : None

Return Type : True if movie was stopped

Syntax : **BOOL object.AVIPlaybackMovieGetLen()**

Description : Returns the length (in number of frames) of the current AVI movie opened

Parameter : None

Return Type : Long Number of frames in the AVI

Syntax : **BOOL object.SobelMagnitude**(*ImageHandle image, ImageHandle magnitude*)

Description : Use **SobelMagnitude** to detect edges

Parameter : image The original image (has to be grayscale)

Magnitude Sobel magnitude (has to be grayscale)

Return Type : True if successfully

Syntax :

BOOL object.Threshold (*ImageHandle image, long threshold, ImageHandle binary*)

Description : The **Threshold** method sets all pixel in *image* to 0 all 255

Parameter : image The original image (has to be grayscale)

Threshold Threshold value

Binary The binary result image (has to be grayscale)

Return Type : True if successfully

