ระบบตรวจหาตำแหน่งของป้ายทะเบียนยานพาหนะ

VEHICLE LICENSE PLATE LOCALIZATION SYSTEM

ปรีมนต์ รัตนาธรรมวัฒน์
PREEMON RATTANATHAMMAWAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2550

ระบบตรวจหาตำแหน่งของป้ายทะเบียนยานพาหนะ

VEHICLE LICENSE PLATE LOCALIZATION SYSTEM

ปรีมนต์ รัตนาธรรมวัฒน์

PREEMON RATTANATHAMMAWAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2550

VEHICLE LICENSE PLATE LOCALIZATION SYSTEM

PREEMON RATTANATHAMMAWAT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF SCIECE IN INFORMATION TECHNOLOGY

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2007

หัวข้อวิทยานิพนธ์	ระบบตรวจหาตำแหน่งของป้ายทะเบียนยานพาหนะ
นักศึกษา	นายปรีมนต์ รัตนาธรรมวัฒน์
รหัสนักศึกษา	44067431
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
พ.ศ.	2550
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ผศ.ดร.ธนารัตน์ ชลิดาพงศ์

บทคัดย่อ

ระบบรู้จำป้ายทะเบียนรถ สามารถแบ่งออกเป็นระบบย่อยสองส่วนด้วยกันคือ ระบบตรวจหาตำแหน่งของป้ายทะเบียน และระบบการอ่านป้ายทะเบียน วิทยานิพนธ์ฉบับนี้เสนอระบบตรวจหาตำแหน่งของป้ายทะเบียนยานพาหนะที่กำลังเคลื่อนที่ ด้วยเทคโนโลยีคอมพิวเตอร์วิชันจากลำดับภาพวีดิทัศน์ โดยใช้วิธีตรวจหาขอบด้วยตัวดำเนินการโซเบลแบบใช่ค่าจุดเริ่มเปลี่ยนชนิดเปลี่ยนแปลงได้ เพื่อให้ทนต่อสภาพการเปลี่ยนแปลงของแสง และใช้วิธีสแกนภาพด้วยวินโดว์เพื่อหาลักษณะการกระจายของขอบในแต่ละวินโดว์ แล้วนำค่าที่ได้มาคำนวณหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน จากนั้นจึงนำไปใช้ในการวิเคราะห์ตรวจหาตำแหน่งของป้ายทะเบียน รวมถึงนำเสนอวิธีในการลดลดการตรวจจับที่ผิดพลาดให้กับระบบ โดยใช้การวิเคราะห์ข้อมูลเชิงเวลาร่วมกับการวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน นอกจากนั้นยังได้นำเสนอวิธีการแก้ไขภาพเพอร์สเปคทีฟ เพื่อให้ได้ภาพที่พร้อมจะนำไปใช้ในระบบการอ่านป้ายทะเบียนต่อไป ซึ่งในวิทยานิพนธ์ฉบับนี้ได้ทำการทดลองการทำงานของระบบกับลำดับภาพวีดิทัศน์ของรถจำนวน 440 คัน ปรากฏว่าระบบสามารถทำงานได้ถูกต้อง 98% และสามารถลดการตรวจจับที่ผิดพลาดลงจาก 887 ป้ายให้เหลือ 74 ป้าย หรือคิดเป็น 8.3 % จากเดิม

Thesis Title	Vehicle License Plate Localization System
Student	Mr. Preemon Rattanathammawat
Student ID.	44067431
Degree	Master of Science
Program	Information Technology
Year	2007
Thesis Advisor	Asst. Prof. Dr. Thanarat Chalidabhongse

ABSTRACT

License Plate Recognition Systems (LPRS) can be separated into two subsystems, license plate detection system and license plate recognition system. This thesis proposes license plate detection system from recorded videos of moving vehicles using computer vision technology. Our approach employs Sobel operators and variable thresholds for edge detection to make it robust for changing in illumination. The license plate in the video frame is located by scanning a window over the entire frame. For each window location, the edge pixel distribution is analyzed to calculate the likelihood score of being a license plate then license plate window candidate can be extracted. After that, we analyze temporal data combine with likelihood score ranking to confirm the possibility of being license plate of the candidate to reduce false positive. In additions, perspective correction is also performed to make the output image suitable for sending to license plate recognition system. From the experiments on several video clips of 440 moving vehicles, the system can detect license plate 98% correctly and false positive can be reduced from 887 plates to 74 plates or 8.3% from origin.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงลงได้ด้วยดี โดยการชี้แนะแนวทางที่ดีตั้งแต่เริ่มต้น ตลอดจนให้คำปรึกษาเมื่อเกิดปัญหาจาก ผศ.ดร.ธนารัตน์ ชลิดาพงศ์ ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์

ขอขอบพระคุณคณาจารย์ทุกท่านในคณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโยลีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้กรุณามอบความรู้ทั้งหลายให้แก่ข้าพเจ้าตลอดเวลาที่ศึกษาอยู่

ขอขอบคุณเจ้าหน้าที่บริษัท และศูนย์การค้าที่เอื้อเฟื้อข้อมูลที่เป็นประโยชน์ รวมถึงสละเวลาอันมีค่าของท่านให้ข้าพเจ้าได้นำข้อมูลเหล่านั้นมาใช้ในการทำวิทยานิพนธ์

ขอขอบคุณเพื่อนนักศึกษารุ่น IS12.1 ทุกคนที่ช่วยเป็นกำลังใจและช่วยผลักดันให้ข้าพเจ้าทำวิทยานิพนธ์นี้จนสำเร็จ

ขอขอบคุณเพื่อนๆ และน้องๆ ใน PIC LAB ทุกคนที่คอยช่วยเหลือ และช่วยกันออกความคิดเห็นในการแก้ปัญหาต่างๆ ที่เกิดขึ้นขณะทำวิทยานิพนธ์

ขอขอบคุณเจ้าหน้าที่งานบัณฑิตศึกษาของคณะฯ และบัณฑิตวิทยาลัยทุกท่านที่ช่วยดำเนินงาน และให้คำปรึกษาต่างๆ

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณบิดา มารดา สมาชิกในครอบครัวทุกคน ที่อบรมเลี้ยงดูและให้การศึกษากับข้าพเจ้าจนเติบใหญ่ และคอยเป็นกำลังใจให้ข้าพเจ้าในการทำวิทยานิพนธ์เสมอมา

ปรีมนต์ รัตนาธรรมวัฒน์

สารบัญ

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญรูป

สารบัญรูป (ต่อ)

รูปที่ หน้า

สารบัญรูป (ต่อ)

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

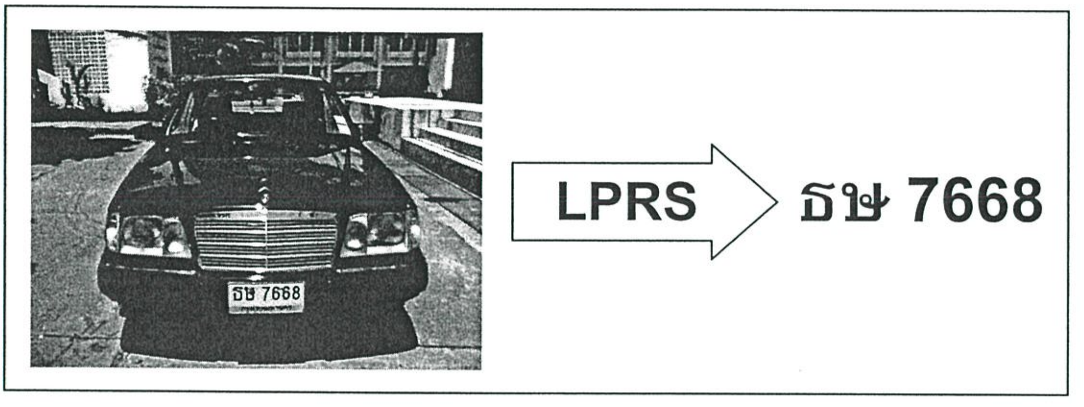
สภาพเศรษฐกิจสังคมในปัจจุบันรถกลายเป็นปัจจัยพื้นฐานของการดำเนินชีวิตของผู้คน ไม่ว่าจะเป็นรถส่วนตัวหรือรถรับจ้างสาธารณะ ล้วนแต่มีปริมาณเพิ่มขึ้นอย่างต่อเนื่อง เนื่องจากมีปริมาณมากจึงจำเป็นต้องมีการติดตามควบคุมกิจกรรมต่างๆของรถ เพื่อความเป็นระเบียบเรียบร้อยในการควบคุมนั้นเราต้องมีการระบุตัวตน (Identify) ของรถแต่ละคัน ซึ่งสิ่งที่จะใช้ในการระบุตัวตนนั้นที่นิยมกันมากก็คือการกำหนดเลขทะเบียนให้กับรถทุกคัน และกำหนดให้มีการติดป้ายทะเบียนที่รถทุกคัน สิ่งเหล่านี้เองได้ถูกนำมาใช้ในการติดตามควบคุมกิจกรรมต่างๆของรถ ไม่ว่าจะเป็นการควบคุมการเข้าออกอาคาร การเก็บค่าบริการจอดรถ การเก็บค่าผ่านทาง หรือแม้กระทั่งการควบคุมการปฏิบัติตามกฎจาจร ล้วนแต่เกี่ยวข้องกับทะเบียนรถทั้งสิ้น

ด้วยปริมาณรถที่เพิ่มสูงขึ้นเป็นอย่างมากในปัจจุบัน โดยเฉพาะเมืองใหญ่อย่างกรุงเทพมหานคร การติดตามควบคุมกิจกรรมของรถนั้นก็ต้องมีความรวดเร็วแม่นยำเพิ่มขึ้น เพื่อเป็นส่วนหนึ่งในการบรรเทาปัญหาจราจรติดขัดที่เป็นอยู่ทุกวันนี้ ซึ่งในปัจจุบันส่วนใหญ่ยังคงใช้มนุษย์ในการระบุตัวตนของรถแต่ละคัน ไม่ว่าจะเป็นการควบคุมการเข้าออกอาคารที่จะใช้คนอ่านป้ายทะเบียนแล้วจึงกำหนดว่าจะปิดหรือเปิดทางเข้าออกอาคารให้รถคันนั้นผ่านเข้ามา หรือจะเป็นการให้บริการที่จอดรถ ซึ่งจะต้องใช้คนอ่านป้ายทะเบียนแล้วบันทึกเลขทะเบียนและเวลาลงบนบัตรจอดรถ หรืออาจจะบันทึกลงบนบัตรอิเลคทรอนิกส์ (Electronic) รวมถึงการทำผิดกฎจราจรที่ตำรวจจารจรต้องอ่านและบันทึกทะเบียนรถที่กระทำผิดกฎจราจรแล้วออกใบสั่งให้ไปชำระค่าปรับเป็นต้น โดยปกติประสิทธิภาพการทำงานของมนุษย์นั้นจะลดลงเมื่อทำงานมากขึ้นซึ่งอาจจะก่อให้เกิดความผิดพลาดมากขึ้นรวมถึงปริมาณของงานก็จะลดลงด้วยเช่นกัน

หากเรามีเทคโนโลยีที่มาช่วยมนุษย์ทำงานด้านการอ่านป้ายทะเบียนได้ ก็จะส่งผลให้ระบบการติดตามควบคุมกิจกรรมของรถมีประสิทธิภาพมากขึ้นตามไปด้วย ซึ่งเทคโนโลยีที่เหมาะจะนำมาใช้ในงานนี้มีหลากหลายเทคโนโลยีด้วยกัน หนึ่งในนั้นก็คือ เทคโนโลยีทางด้านคอมพิวเตอร์วิชัน (Computer Vision) ซึ่งเป็นเทคโนโลยีที่เกี่ยวกับการมองเห็นของคอมพิวเตอร์ ที่ทำให้คอมพิวเตอร์นั้นสามารถมองเห็นและวิเคราะห์สิ่งที่มองเห็นนั้นได้ โดยที่เทคโนโลยีนี้สามารถพัฒนาขึ้นได้โดยไม่ต้องมีการดัดแปลงหรือเพิ่มเติมส่วนหนึ่งส่วนใดกับตัวรถ เพราะว่าเราใช้เพียงการอ่านภาพถ่ายรถที่มีป้ายทะเบียนติดอยู่ซึ่งมีอยู่ในรถทุกคันอยู่แล้ว ใช้เพียงกล้องเพื่อจับภาพรถและคอมพิวเตอร์เพื่อใช้ในการประมวลผลเท่านั้น ซึ่งระบบที่นำมาใช้เพื่อให้คอมพิวเตอร์สามารถ

มองเห็นและอ่านป้ายทะเบียนรถได้นั้นเรียกว่า ระบบรู้จำป้ายทะเบียนรถ (LPRS; License Plate Recognition System)

ระบบรู้จำป้ายทะเบียนรถนั้นสามารถแบ่งออกเป็นระบบย่อยได้สองระบบ ได้แก่ระบบค้นหาตำแหน่งป้ายทะเบียน (License Plate Detection System) และระบบการอ่านป้ายทะเบียน (License Plate Recognition System) สำหรับระบบค้นหาตำแหน่งป้ายทะเบียนนั้นเป็นการค้นหาว่าป้ายทะเบียนนั้นอยู่ในส่วนใดของภาพ เพื่อนำภาพส่วนนั้นส่งต่อให้ระบบการอ่านป้ายทะเบียนเพื่อนำไปเข้ากระบวนการรู้จำเพื่อให้ได้ผลออกมาเป็นหมายเลขทะเบียนของป้ายทะเบียนที่เราต้องการต่อไป ดังแสดงในรูปที่ 1.1



รูปที่ 1.1 ระบบรู้จำป้ายทะเบียนรถ

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วิทยานิพนธ์ฉบับนี้มุ่งหวังเพื่อศึกษาและพัฒนาระบบรู้จำป้ายทะเบียนรถในส่วนของระบบย่อยการค้นหาตำแหน่งป้ายทะเบียนโดยใช้เทคโนโลยีคอมพิวเตอร์วิชัน (Computer Vision) ซึ่งทำหน้าที่ในการวิเคราะห์ภาพที่ได้รับเข้ามาว่าป้ายทะเบียนควรอยู่ที่ตำแหน่งใดของภาพ เพื่อพร้อมที่จะส่งภาพส่วนที่เป็นป้ายทะเบียนนั่นให้กับระบบการอ่านป้ายทะเบียนต่อไป โดยสามารถทำงานได้กับป้ายทะเบียนรถหลากหลายประเภทและขณะที่รถกำลังเคลื่อนที่ รวมทั้งมีความทนทานต่อสภาพแสงที่เปลี่ยนแปลงไปในทุกช่วงเวลา ทั้งภายในและภายนอกอาคาร ซึ่งระบบรู้จำป้ายทะเบียนรถที่สมบูรณ์นั้นสามารถนำไปประยุกต์ใช้งานได้ในหลากหลายแอพพลิเคชัน (Application) เพื่อช่วยให้การติดตามควบคุมกิจกรรมของรถมีความถูกต้องแม่นยำ รวดเร็วและมีประสิทธิภาพมากกว่าการใช้มนุษย์ เช่นระบบเก็บค่าผ่านทางอัตโนมัติ ระบบเก็บค่าจอดรถอัตโนมัติ และระบบตรวจจับการกระทำผิดกฎจราจรของรถเป็นต้น เมื่อระบบควบคุมกิจกรรมต่างๆ ของรถทำงานได้มีประสิทธิภาพมากขึ้นก็ย่อมเป็นส่วนหนึ่งที่ส่งผลให้ สภาพการจราจรในบริเวณนั้นคล่องตัวขึ้นตามไปด้วย

1.3 สมมติฐานของการศึกษา

ในการค้นหาว่าส่วนใดของภาพควรจะเป็นป้ายทะเบียนนั้น ต้องหาความแตกต่างระหว่างส่วนของป้ายทะเบียนและส่วนอื่นๆของภาพ เพื่อใช้แยกแยะว่าส่วนใดของภาพคือป้ายทะเบียนและส่วนใดของภาพคือฉากหลัง (Background) เมื่อพิจารณาส่วนที่เป็นป้ายทะเบียนจะพบว่า เป็นบริเวณที่มีความแตกต่างกันของสีในแนวตั้งในบริเวณใกล้เคียงกัน ชัดเจนมากกว่าส่วนอื่นๆของภาพ กล่าวคือจะมีส่วนที่เป็นขอบแนวตั้งอยู่หนาแน่นมากกว่าบริเวณอื่นๆ ซึ่งเกิดจากขอบระหว่างสีของตัวอักษรบนป้ายทะเบียนและสีของพื้นหลังของป้ายทะเบียน โดยที่ขอบที่เกิดขึ้นนั้นจะมีการกระจายตัวในรูปแบบเฉพาะ อันมีลักษณะแตกต่างจากขอบที่เกินขึ้นโดยสภาพแวดล้อมอื่นๆ ทำให้สามารถแยกแยะบริเวณที่เป็นป้ายทะเบียนออกจากส่วนอื่นของภาพได้

1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

จากการที่บริเวณป้ายทะเบียนมีส่วนที่เป็นขอบชัดเจนมากกว่าบริเวณอื่น ในวิทยานิพนธ์ฉบับนี้จึงใช้วิธีตรวจจับขอบ (Edge Detection) ในการหาขอบที่เกิดขึ้นในภาพ และพิจารณาการกระกระจายของขอบที่เกิดขึ้น เพื่อมองหาบริเวณของภาพที่มีการเกาะกลุ่มของขอบที่คาดว่าน่าจะเป็นป้ายทะเบียนโดยการใช้วินโดว์ (Window) ที่มีขนาดใกล้เคียงกับขนาดของป้ายทะเบียน สแกน (Scan) ไปตลอดทั้งแนวตั้งและแนวนอน แล้วคำนวณหาการกระจายของขอบในแต่ละวินโดว์เพื่อนำมาเป็นข้อมูลในการหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน แล้วนำค่าที่ได้มาใช้ในการคัดเลือกแคนดิเดทวินโดว์ (Candidate Windows) ซึ่งเป็นวินโดว์ที่คาดว่าจะเป็นป้ายทะเบียน จากนั้นจะวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนของแต่ละแคนดิเดทวินโดว์ในอันดับต้นๆ ซึ่งมีค่ามาก ว่ามีการเกาะกลุ่มกันอย่างไร แล้วจึงนำข้อมูลที่ได้มาใช้พิจารณาว่าแคนดิเดทวินโดว์ใดควรเป็นป้ายทะเบียนหรือควรคัดออกไป

1.5 ขอบเขตการวิจัย

วิทยานิพนธ์ฉบับนี้นำเสนอการพัฒนาระบบค้นหาตำแหน่งป้ายทะเบียนรถซึ่งประกอบด้วยรถเก๋ง รถกระบะ รถตู้ รถบรรทุก รถโดยสาร รวมถึงรถจักรยานยนต์ ซึ่งเป็นระบบงานย่อยของระบบรู้จำป้ายทะเบียนรถ จากภาพนิ่งที่ถูกตัด(Capture) มาจากไฟล์ลำดับภาพวีดิทัศน์ซึ่งถูกบีบอัดเพื่อประหยัดพื้นที่ในการจัดเก็บข้อมูลโดยใช้เทคโนโลยีทางด้านคอมพิวเตอร์วิชัน (Computer Vision) และประมวลผลด้วยโปรแกรม MATLAB® โดยเป็นการประมวลผลแบบออฟไลน์ (Offline Processing) ซึ่งผลที่ได้คืออัลกอริธึมที่ใช้ในการคำนวณหาตำแหน่งของป้าย

ทะเบียนรถจากภาพนิ่งที่ถูกตัดมาจากไฟล์ภาพวีดิทัศน์โดยอัตโนมัติ และพร้อมที่จะส่งภาพส่วนที่เป็นป้ายทะเบียนนั้นเข้าสู่กระบวนการอ่านป้ายทะเบียนต่อไป

1.6 ขั้นตอนของการศึกษา

1. ศึกษาขั้นตอนวิธีการ และแบบจำลองที่ใช้ในการตรวจหาป้ายทะเบียน
2. ศึกษาถึงคุณลักษณะที่เหมาะสมของป้ายทะเบียนรถ ที่จะสามารถนำมาใช้ในการตรวจหาได้
3. เลือกขั้นตอนวิธีการ และเลือกคุณลักษณะที่เหมาะสมของป้ายทะเบียนที่จะนำมาใช้ในการตรวจหาป้ายทะเบียนรถจากภาพนิ่งที่จับได้จากลำดับภาพวีดิทัศน์
4. ทดลองนำขั้นตอนวิธีการ แบบจำลอง และคุณลักษณะที่เหมาะสมของป้ายทะเบียนที่เลือกใช้ มาทำการตรวจหาป้ายทะเบียนรถจากภาพนิ่งที่จับได้จากลำดับภาพวีดิทัศน์
5. สรุปผลการทดลองและเขียนวิทยานิพนธ์

1.7 เค้าโครงวิทยานิพนธ์

ในวิทยานิพนธ์ฉบับนี้แบ่งออกเป็นทั้งหมด 5 บทด้วยกัน คือ

บทที่ 1 กล่าวถึงความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ ขอบเขตของการวิจัย และขั้นตอนการศึกษา

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานทางด้านคอมพิวเตอร์วิชันที่เกี่ยวข้องกับงานวิจัย ทฤษฎีของการเคลื่อนที่ที่เกี่ยวกับตำแหน่ง ระยะทาง ความเร็ว และความเร่ง รวมถึงงานวิจัยที่เกี่ยวข้องกับหัวข้อวิจัย

บทที่ 3 กล่าวถึงขั้นตอนและวิธีต่างๆ ที่ผู้วิจัยใช้ในการคำนวณหาตำแหน่งของป้ายทะเบียนรถซึ่งประกอบไปด้วย ขั้นตอนการตรวจหาขอบ (Edge Detection) การหาจุดเริ่มเปลี่ยนแบบเปลี่ยนแปลงได้ (Variable Threshold) การหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน การหาแคนดิเดทวินโดว์ (Candidate Windows) การวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนเพื่อนำมาใช้ในการคัดแยกส่วนที่เป็นป้ายทะเบียนจริงออกมาจากแคนดิเดทวินโดว์

บทที่ 4 กล่าวถึงผลการทดลองหาตำแหน่งของป้ายทะเบียนรถจากภาพนิ่งที่ตัด (Capture) ได้จากไฟล์ลำดับภาพวีดิทัศน์ ด้วยขั้นตอนที่เสนอไว้ในวิทยานิพนธ์ฉบับนี้ พร้อมทั้งเปรียบเทียบผลที่ได้จากการทดลองด้วยวิธีต่างๆ

บทที่ 5 เป็นบทสรุปผลการวิจัยและข้อเสนอแนะในการพัฒนางานวิจัยขั้นต่อไป

บทที่ 2

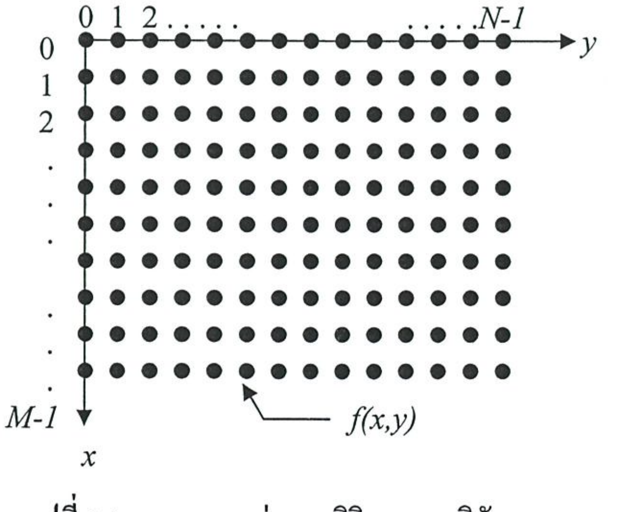
ทฤษฎีพื้นฐาน งานวิจัยที่เกี่ยวข้องกับการตรวจหาป้ายทะเบียน และระบบป้ายทะเบียนรถที่ใช้ในประเทศไทย

ในบทนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆเกี่ยวกับการประมวลผลภาพดิจิตอล และการเคลื่อนที่ของวัตถุในแนวเส้นตรงที่นำมาใช้ในวิทยานิพนธ์ฉบับนี้ รวมถึงงานวิจัยต่างๆที่เกี่ยวข้องกับการตรวจหาป้ายทะเบียน รวมถึงประเภทของป้ายทะเบียนรถที่มีใช้ในประเทศไทย

2.1 ทฤษฎีพื้นฐานกรรมวิธีประมวลภาพและคอมพิวเตอร์วิชัน

2.1.1 การแทนค่าภาพดิจิตอล (Digital Image Representation)

เราสามารถแทนภาพดิจิตอลด้วยฟังก์ชั่นสองมิติ [1] (Two-Dimensional Function) $f(x,y)$ ซึ่ง x และ y คือ ระบบพิกัดระนาบ (Spatial Coordinators) และแอมปลิจูด (Amplitude) ของ f คือความค่าความสว่าง (Brightness) หรือความเข้ม (Intensity) หรืออาจจะถูกเรียกว่าค่าระดับเทา (Gray Level) ซึ่งถูกใช้ในกรณีที่ภาพนั้นเป็นภาพสีเดียว (Monochrome Image) ในกรณีที่เป็นภาพสีจะใช้ภาพสีเดียวหลายๆภาพมารวมกันเพื่อแสดงถึงแต่ละองค์ประกอบของสี เช่น โมเดลสี (Color Model) RGB ก็จะประกอบไปด้วยสามองค์ประกอบได้แก่ ภาพองค์ประกอบสีแดง ภาพองค์ประกอบสีเขียว และภาพองค์ประกอบสีน้ำเงิน เป็นต้น ซึ่งจะกล่าวโดยละเอียดต่อไปในส่วนโมเดลสี RGB เมื่อพิจารณาถึงภาพดิจิตอลที่มีขนาด M แถวและ N หลัก เราเรียกภาพนั้นว่ามีขนาด $M \times N$ แล้วเราสามารถแทนภาพนั้นด้วยฟังก์ชั่น $f(x,y)$ โดยแต่ละคู่ของ x และ y เรียกว่าพิกเซล (Pixel) ของภาพในตำแหน่งพิกัด (x,y) และมีขนาดความสว่างที่ค่าของ $f(x,y)$ ดังแสดงในสมการที่ (2.1) และรูปที่ 2.1



รูปที่ 2.1 การแทนค่าภาพดิจิตอลบนพิกัดระนาบ

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix} \tag{2.1}$$

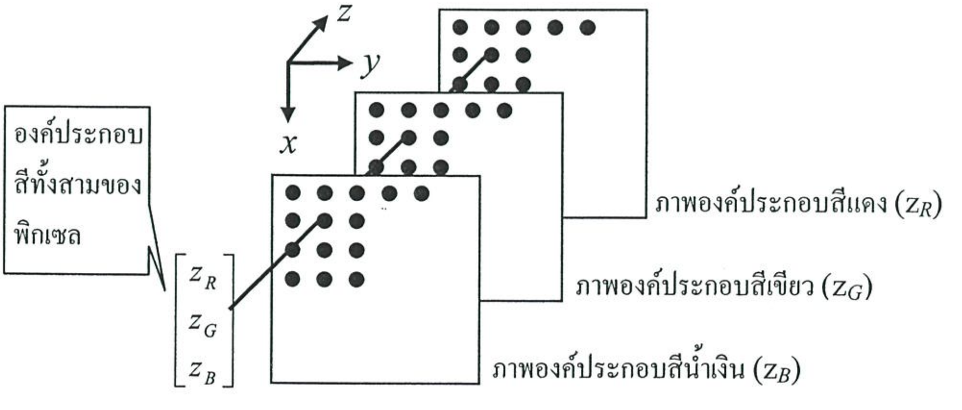
เนื่องจากวิทยานิพนธ์ฉบับนี้ประมวลผลด้วยโปรแกรม MATLAB® ดังนั้นจากสมการที่ (2.1) จะเห็นว่าเราสามารถแทนค่าภาพดิจิตอลให้อยู่ในรูปเมตริก (Metric) ได้ดังสมการที่ (2.2)

$$\mathrm{f}=\begin{bmatrix} \mathrm{f}(1,1) & \mathrm{f}(1,2) & \dots & \mathrm{f}(1,\mathrm{N}) \\ \mathrm{f}(2,1) & \mathrm{f}(2,2) & \cdots & \mathrm{f}(2,\mathrm{N}) \\ \vdots & \vdots & & \vdots \\ \mathrm{f}(\mathrm{M},1) & \mathrm{f}(\mathrm{M},2) & \cdots & \mathrm{f}(\mathrm{M},\mathrm{N}) \end{bmatrix} \tag{2.2}$$

โดยที่ f(1,1) = *f(0,0)* ซึ่งทั้งสมการที่ (2.1) และ (2.2) นั้นเหมือนกัน แต่แตกต่างกันตรงที่การเลือนตำแหน่งจุดออริจิน (Origin) จากจุด (0,0) ไปเป็นจุด (1,1) เท่านั้น เนื่องจากเมตริกในโปรแกรม MATLAB® นั้นจุดมีออริจินอยู่ที่ตำแหน่ง (1,1)

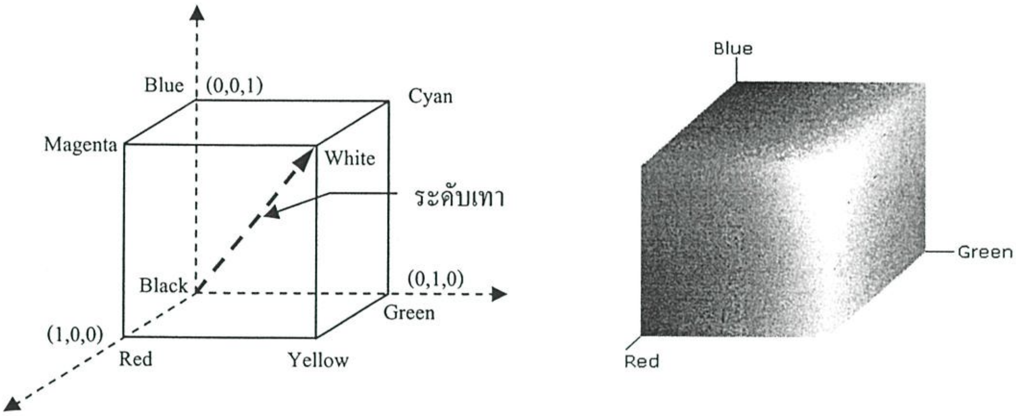
2.1.2 การแทนค่าภาพสีและโมเดลสี RGB

ดังที่ได้กล่าวมาแล้วข้างต้นว่าเราสามารถสร้างภาพสีขึ้นจากภาพองค์ประกอบสีพื้นฐานอันได้แก่ แดง เขียว และน้ำเงิน ในกรณีของโมเดลสี RGB โดยความสว่างของแต่ละสีพื้นฐานที่แตกต่างกันเมื่อนำมารวมกันทำให้เกิดสีต่างๆ ดังนั้นภาพสีขนาด $M \times N$ พิกเซล (Pixel) จึงเกิดจากการนำภาพองค์ประกอบสีพื้นฐานทั้งสามมาวางซ้อนกันในแนวแกน z ดังรูปที่ 2.2 ทำให้เกิดอาร์เรย์ (Array) สามมิติขนาด $M \times N \times 3$ ที่ใช้ในการแทนค่าภาพสีที่ใช้ในการประมวลผลภาพดิจิตอล



รูปที่ 2.2 การสร้างภาพสีจากภาพองค์ประกอบสีทั้งสามของพิกเซล

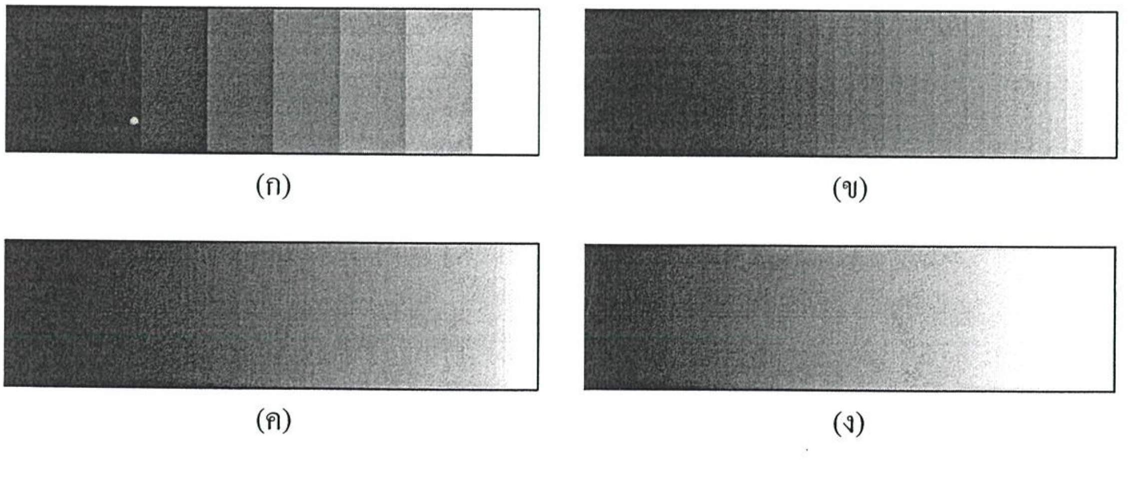
การแสดงภาพสีที่เกิดจากการรวมสีพื้นฐานทั้งสามสีในแนวแกน *x,y* และ *z* เข้าด้วยกันเป็นการใช้โมเดลสีที่เรียกว่า โมเดลสี RGB (RGB Color Model) ซึ่งเมื่อความสว่างของแต่ละสีพื้นฐานมีค่าต่างๆกันก็จะทำให้เกิดสีต่างกันดังแสดงในรูป 2.3



รูปที่ 2.3 ลูกบาศก์โมเดลสี RGB

2.1.3 ภาพระดับเทา (Gray level Image)

ภาพระดับเทาหรืออาจจะเรียกว่าภาพระดับความสว่าง (Brightness Image) เป็นภาพที่พิกเซลแต่ละพิกเซลของภาพจะบ่งบอกถึงระดับความสว่างของภาพที่ตำแหน่งนั้น โดยมีค่าตั้งแต่น้อยสุด (สีดำ) ไปจนถึงมากสุด (สีขาว) ซึ่งความละเอียดขึ้นอยู่กับจำนวนบิตของข้อมูลที่ใช้เก็บค่าระดับความสว่าง เช่น 3 บิตมีได้ 2^3=8 ระดับ 5 บิตมีได้ 2^5=32 ระดับ 6 บิตมีได้ 2^6=64 ระดับ และ 8 บิตมีได้ 2^8=256 ระดับ ดังแสดงในรูป 2.4 ซึ่งปกติทั่วไปรวมถึงในวิทยานิพนธ์ฉบับนี้ใช้ภาพระดับเทาขนาด 8 บิตซึ่งมีทั้งหมด 256 ระดับ ตั้งแต่ค่า 0 จนถึง 255



รูปที่ 2.4 ภาพระดับเทาในละความละเอียด (ก) 8 ระดับ (ข) 32 ระดับ (ค) 64 ระดับ (ง) 256 ระดับ

ในการแปลงจากภาพสีเป็นภาพระดับเทานั้นสามารถทำได้โดยการใช้สมการที่ (2.3) หรือสมการที่ (2.4)

$$gray(x,y) = Avg(R(x,y) + G(x,y) + B(x,y)) \tag{2.3}$$

$$gray(x,y) = \frac{R(x,y) + G(x,y) + B(x,y)}{3} \tag{2.4}$$

โดยที่ $gray(x,y)$ คือค่าระดับเทาของพิกเซลตำแหน่งพิกัด (x,y)

$R(x,y)$ คือระดับความสว่างของสีพื้นฐานสีแดงของพิเซลตำแหน่งพิกัด (x,y)

$G(x,y)$ คือระดับความสว่างของสีพื้นฐานสีเขียวของพิเซลตำแหน่งพิกัด (x,y)

$B(x,y)$ คือระดับความสว่างของสีพื้นฐานสีน้ำเงินของพิเซลตำแหน่งพิกัด (x,y)

ซึ่งจะได้ผลดังรูปที่ 2.5

(ก) (ข)

รูปที่ 2.5 การแปลงจากภาพสีไปเป็นภาพระดับเทา (ก) ภาพสี (ข) ภาพระดับเทา

2.1.4 ภาพสองระดับ (Binary Image)

เป็นภาพที่แต่ละพิกเซลจะเป็นได้แค่สีดำหรือสีขาวเท่านั้น โดยแต่ละพิกเซลจะใช้พื้นที่ในการเก็บข้อมูลเพียงหนึ่งบิตเท่านั้น โดยถ้ามีค่าเท่ากับ 1 พิกเซลนั้นจะเป็นสีขาว และถ้ามีค่าเท่ากับ 0 พิกเซลนั้นจะเป็นสีดำ ในการแปลงจากภาพระดับเทาไปเป็นภาพสองระดับนั้นทำได้โดยการกำหนดค่าจุดเริ่มเปลี่ยน (Threshold) เพื่อใช้ในการพิจารณาว่าพิกเซลนั้นมีค่าเท่ากับ 1 หรือ 0 จากสมการที่ 2.5

$$binary(x,y) = \begin{cases} 1 & \text{ถ้า} \quad gray(x,y) > T \\ 0 & \text{ถ้า} \quad gray(x,y) \le T \end{cases} \tag{2.5}$$

โดยที่ $binary(x,y)$ คือค่าสองระดับของพิกเซลตำแหน่งพิกัด (x,y)

$gray(x,y)$ คือค่าระดับเทาของพิกเซลตำแหน่งพิกัด (x,y)

T คือค่าจุดเริ่มเปลี่ยน (Threshold)

ซึ่งจะได้ผลดังรูปที่ 2.6

(ก) (ข)

รูปที่ 2.6 การแปลงภาพระดับเทาไปเป็นภาพสองระดับ (ก) ภาพระดับเทา (ข) ภาพสองระดับ

2.1.5 การประมวลผลเชิงจุด (Point Processing)

เป็นการประมวลผลในระดับพิกเซลต่อพิกเซลโดยไม่เกี่ยวข้องกับพิกเซลข้างเคียงซึ่งสามารถทำได้ในหลายกรณีด้วยกัน ไม่ว่าจะเป็นการนำค่าคงที่ใดๆ มาดำเนินการทางคณิตศาสตร์ เช่น การบวก ลบ คูณ หาร รวมทั้งการดำเนินการทางตรรกะซึ่งจะทำกับภาพสองระดับ เช่น AND, OR, XOR และ NOT [2]

2.1.5.1 การดำเนินการบวก ลบ ภาพ

การดำเนินการทางคณิตศาสตร์เหล่านี้ อาจจะถูกใช้ในการปรับค่าความสว่างของภาพ การบวกด้วยค่าคงที่จะช่วงเพิ่มความสว่างแต่ไม่ได้เพิ่มความคมชัดของภาพ การลบด้วยค่าคงที่จะช่วยเพิ่มความมืด ซึ่งจะเหมาะกับภาพที่มีความสว่างมาก ในกรณีที่เป็นการบวกลบระหว่างภาพด้วยกันเอง ผลลัพธ์จะกลายเป็นภาพใหม่ การบวกและการลบระหว่างภาพด้วยกันสามารถทำได้โดยสมการที่ (2.6)

$$g(x,y) = f_1(x,y) \pm f_2(x,y) \tag{2.6}$$

โดยที่ $g(x,y)$ คือภาพผลลัพท์ที่ได้

$f_1(x,y)$ คือค่าภาพที่นำมาดำเนินการภาพแรก

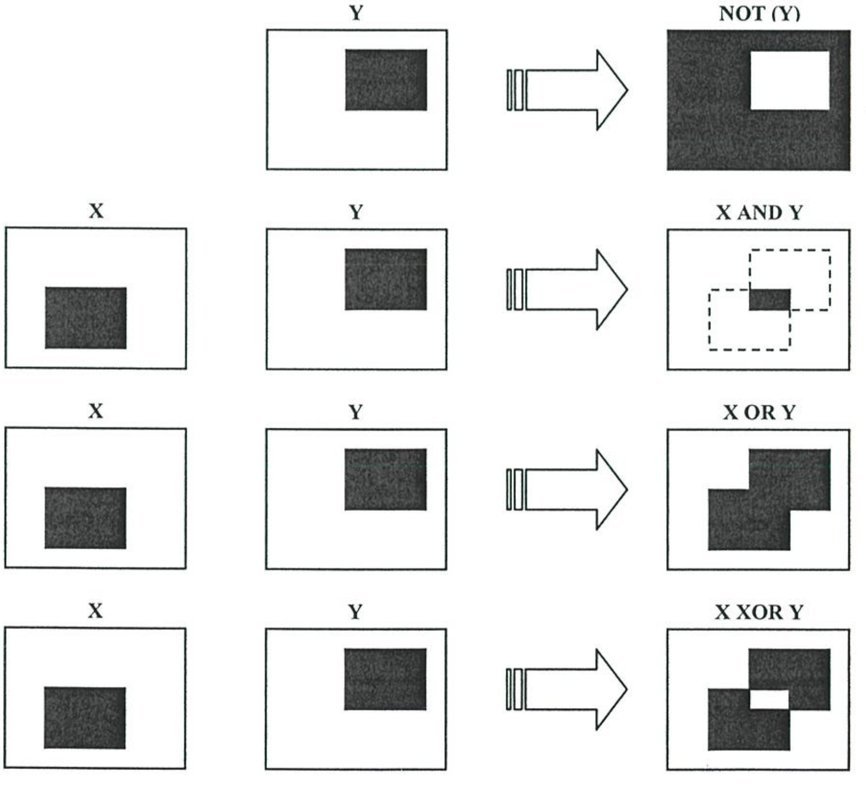
$f_2(x,y)$ คือค่าภาพที่นำมาดำเนินการภาพที่สอง

2.1.5.2 การดำเนินการทางตรรกะ

หลักของการดำเนินการทางตรรกะที่ถูกนำมาใช้ในการประมวลผลภาพ คือ AND, OR, XOR และ NOT ซึ่งจะกระทำกับพิกเซลระหว่าง 2 ภาพหรือมากกว่านั้น (ยกเว้น NOT ที่เป็นตัวดำเนินการทางตรรกะของภาพเพียงภาพเดียว) สำหรับการประมวลผลที่ซับซ้อนกว่านี้ เมื่อพิจารณาอย่างถ่องแท้แล้วจะพบว่าเกิดจากการนำตัวดำเนินการพื้นฐานเหล่านี้มาเรียงกันเป็นลำดับขั้นแตกต่างกันไป ซึ่งแสดงไว้ดังตารางที่ 2.1 และตัวอย่างผลลัพธ์ที่ได้จากการดำเนินการทางตรรกะระหว่างภาพสองภาพแสดงดังรูปที่ 2.7

ตารางที่ 2.1 การดำเนินการทางตรรกะพื้นฐานทั้ง 4 ตัวดำเนินการ

P	Q	P AND Q	P OR Q	P XOR Q	NOT(P)
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

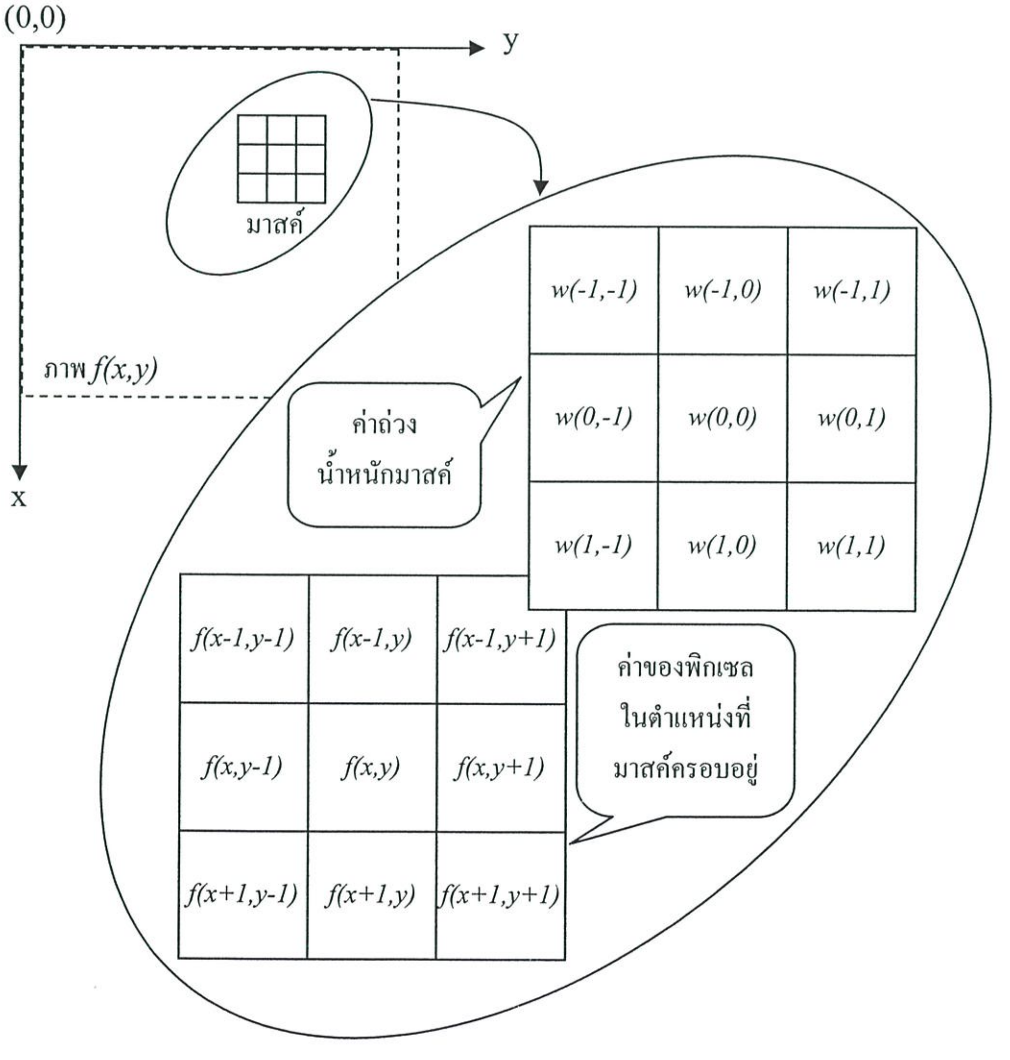


รูปที่ 2.7 ตัวอย่างของการดำเนินการทางตรรกะระหว่างรูปสองระดับ

2.1.6 การประมวลผลเชิงพื้นที่ (Spatial Processing)

การประมวลผลเชิงพื้นที่นี้จะเป็นการนำค่าของพิกเซลบริเวรรอบพิกเซลที่เราต้องการดำเนินการมาคำนวณ แล้วนำมาแทนค่าพิกเซลเดิม ซึ่งพิกเซลบริเวณรอบๆนี้มีชื่อเรียกหลายแบบ เช่น ตัวกรอง (Filter) วินโดว์ (Window) เคอร์แนล (Kernel) หรือมาสค์ (Mask) การคำนวณบริเวณรอบๆพิกเซลนี้เรียกว่าการทำคอนโวลูชัน (Convolution)

การทำคอนโวลูชันเป็นการหาผลรวมค่าของพิกเซลรอบๆที่ได้ผ่านการถ่วงน้ำหนักแล้วของพิกเซลที่กำลังพิจารณา โดยค่าที่ใช้ถ่วงน้ำหนักนั้นเรียกว่าค่าถ่วงน้ำหนักมาสค์ (Mask Coefficient) โดยปกติจะกำหนดค่าถ่วงน้ำหนักเป็นเมตริกขนาด $m \times n$ โดยที่ m และ n จะเป็นเลขคี่ เช่น 3, 5, 7, 9 … เป็นต้น ในการทำคอนโวลูชันนี้สามารถทำได้โดยการคำนวณตามสมการที่ (2.7) ดังแสดงในตัวอย่างการคำนวณในรูป 2.8 ซึ่งใช้เมตริกขนาด 3×3 มาเป็นค่าถ่วงน้ำหนักมาส์ค [3]



รูปที่ 2.8 ตัวอย่างขั้นตอนในการทำคอนโวลูชันโดยใช้มาส์คขนาด 3×3

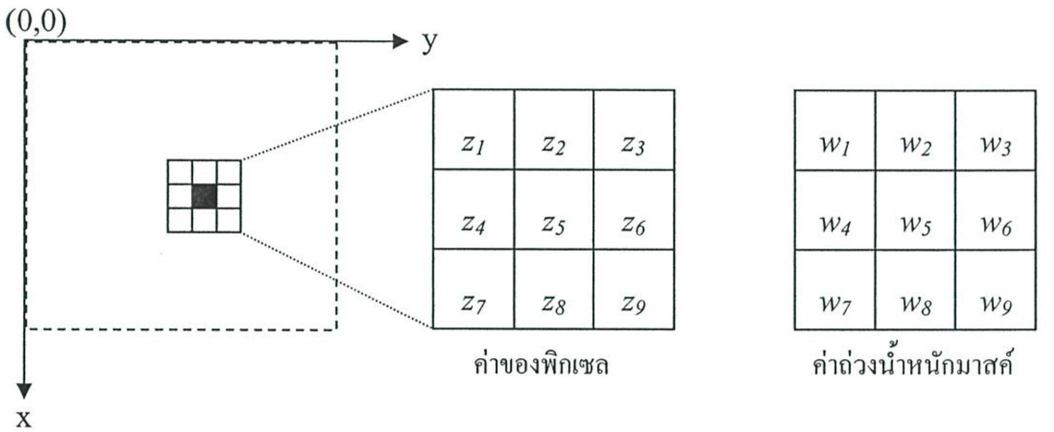
$$R = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \ldots + w(0,0)f(x,y) + \ldots + w(1,0)f(x+1,y) + w(1,1)f(x+1,y+1) \quad (2.7)$$

โดยที่ R คือภาพผลลัพธ์ที่ได้จากการทำคอนโวลูชัน
w คือค่าถ่วงน้ำหนักมาสค์ตำแหน่งต่างๆ ในรูปที่ 2.8
f คือค่าของพิกเซลตำแหน่งต่างๆ ในรูปที่ 2.8

ซึ่งการทำคอนโวลูชันกับภาพขนาด $M \times N$ ด้วยมาสค์ขนาด $m \times n$ จากสมการที่ (2.7) สามารถเขียนให้อยู่ในรูปทั่วไปได้ว่า

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t) \quad (2.8)$$

โดยที่ $a = (m-1)/2$ และ $b = (n-1)/2$ ขณะที่ w คือค่าถ่วงน้ำหนักมาสค์และ f คือค่าของพิกเซล และเพื่อให้ง่ายต่อความเข้าใจให้พิจารณารูปที่ 2.9 ซึ่งจะทำให้เราสามารถใช้สมการที่ (2.9) แทนสมการที่ (2.8) ในการคำนวณหาค่า R ได้



รูปที่ 2.9 การทำคอนโวลูชัน

$$R = w_1 z_1 + w_2 z_2 + \ldots + w_{mn} z_{mn} \quad (2.9)$$

$$R = \sum_{i=1}^{mn} w_i z_i \quad (2.10)$$

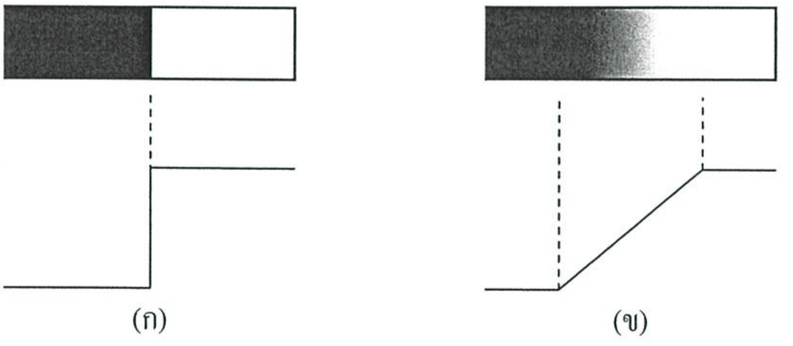
โดยที่ w คือค่าถ่วงน้ำหนักมาสค์ตำแหน่งต่างๆ ในรูปที่ 2.9
z คือค่าของพิกเซลตำแหน่งต่างๆ ในรูปที่ 2.9

2.1.7 การตรวจหาขอบ (Edge Detection)

การตรวจหาขอบ ถือเป็นการะบวนการประมวลผลภาพที่สำคัญมากอันหนึ่ง ถูกนำมาใช้ในการวิเคราะห์รูปภาพเพื่อกำหนดขอบเขตของบริเวณที่ต้องการ ซึ่งกระบวนการนี้จะกระทำโดยตรวจสอบว่าพิกเซลจุดใดที่มีการเปลี่ยนแปลงของความสว่างอย่างทันทีทันใดเมื่อเปรียบเทียบกับบริเวณใกล้เคียงก็จะกำหนดให้พิกเซลนั้นเป็นขอบ

วิธีการที่ใช้หาการเปลี่ยนแปลงของฟังก์ชันต่อเนื่องคือ การหาอนุพันธ์ แต่ในฟังก์ชันของรูปภาพจะขึ้นอยู่กับสองตัวแปรในระบบพิกัดระนาบ ดังนั้นวิธีการที่นำมาใช้คือ การหาอนุพันธ์ย่อย โดยการเปลี่ยนแปลงของฟังก์ชันรูปภาพสามารถอธิบายด้วยค่าเกรเดียนต์ในทิศทางสูงสุด

ขอบ คือคุณสมบัติที่เกี่ยวกับพิกเซลใดพิกเซลหนึ่งซึ่งนำไปคำนวณร่วมกับพฤติกรรมของพิกเซลรอบข้าง และประกอบไปด้วยตัวแปรชนิดเวคเตอร์สองตัวแปรคือ ขนาดและทิศทาง ขนาดของขอบสามารถหาได้จากขนาดของเกรเดียนต์ และทิศทางของขอบถูกหมุนในทิศเกรเดียนต์ ซึ่งมีค่าสูงสุด จากพิกเซลสีดำ [$f(x,y)=0$] ไปพิกเซลสีขาว [$f(x,y)=255$] ในรูปที่ 2.10 แสดงแบบจำลองขอบในอุดมคติซึ่งมีการเปลี่ยนแปลงอย่างทันทีทันใด และขอบที่เปลี่ยนลักษณะทางลาดซึ่งจะมีการเปลี่ยนแปลงในลักษณะไล่ระดับ



รูปที่ 2.10 แบบจำลองขอบ (ก) ขอบในอุดมคติ (ข) ขอบที่เปลี่ยนลักษณะทางลาด

อนุพันธ์อันดับหนึ่งของรูปดิจิตอล จะหาได้จากการประมาณของเกรเดียนต์ใน 2 มิติ ค่าเกรเดียนต์ของรูปภาพ $f(x,y)$ ที่ตำแหน่ง (x,y) ถูกกำหนดด้วยเวคเตอร์ ∇f ดังแสดงในสมการที่ (2.11)

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix} \qquad (2.11)$$

ในการวิเคราะห์เวคเตอร์ก็คือจุดของเวคเตอร์เกรเดียนต์ ซึ่งมีอัตราการเปลี่ยนแปลงสูงสุดของทิศทางของ f ที่พิกัด (x,y) โดยมีค่าสำคัญที่นำมาใช้ตรวจหาขอบ คือ ค่าขนาดของเวคเตอร์แทนด้วย $\nabla \mathrm{f}$ และสามารถคำนวณได้จากสมการที่ (2.12)

$$\nabla f = mag(\nabla \mathrm{f}) = [G_x^2 + G_y^2]^{1/2} \tag{2.12}$$

ส่วนค่าทิศทางคำนวณได้จาก

$$\psi = \tan^{-1}\left(\frac{G_x}{G_y}\right) \tag{2.13}$$

โดยที่ ψ คือมุมที่วัดจากแกน x ไปยังจุด $f(x,y)$ ที่มีหน่วยเป็นเรเดียน

การคำนวณค่าเกรเดียนต์ของรูปภาพด้วยอนุพันธ์ย่อยในทุกๆ พิกเซล ทำได้โดยให้บล็อกขนาด 3×3 รูปที่ 2.11 แสดงค่าระดับเทาที่จุด z_5 ของพิกเซลและบริเวณใกล้เคียง อนุพันธ์อันดับหนึ่งที่จุด z_5 สามารถหาได้ด้วยตัวดำเนินการที่ต่างๆ กัน ดังแสดงในสมการที่ (2.14)-(2.19)

- ตัวดำเนินการไขว้เกรเดียนต์แบบโรเบิร์ต (Robert Cross-gradient Operators)

$$G_x = (z_9 - z_5) \tag{2.14}$$

$$G_y = (z_8 - z_6) \tag{2.15}$$

- ตัวดำเนินการแบบพรีวิตต์ (Prewitt Operators)

$$G_x = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3) \tag{2.16}$$

$$G_y = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7) \tag{2.17}$$

- ตัวดำเนินการแบบโซเบล (Sobel Operators)

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \tag{2.18}$$

$$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \tag{2.19}$$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

บล๊อกขนาด 3×3

$$G_x = \begin{array}{|c|c|}\hline -1 & 0 \\ \hline 0 & 1 \\ \hline\end{array} \qquad G_y = \begin{array}{|c|c|}\hline 0 & -1 \\ \hline 1 & 0 \\ \hline\end{array}$$

Roberts

$$G_x = \begin{array}{|c|c|c|}\hline -1 & -1 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 1 & 1 \\ \hline\end{array} \qquad G_y = \begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline -1 & 0 & 1 \\ \hline -1 & 0 & 1 \\ \hline\end{array}$$

Prewitt

$$G_x = \begin{array}{|c|c|c|}\hline -1 & -2 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 2 & 1 \\ \hline\end{array} \qquad G_y = \begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline\end{array}$$

Sobel

รูปที่ 2.11 มาสค์ต่างๆ ที่ใช้ในการคำนวณหาค่าเกรเดียนต์ที่จุด z_5

ส่วนการหาด้วยวิธีไขว้ศูนย์ (Zero crossing) หรืออนุพันธ์อันดับสอง สามารถทำได้โดยการใช้สมการที่ (2.20)

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \tag{2.20}$$

2.1.8 ฮิสโตแกรม (Histogram)

ฮิสโตแกรมของภาพสามารถอธิบายถึงความมืด ความสว่าง ความคมชัด (Contrast) ว่าสูงหรือต่ำ อีกทั้งยังสามารถประยุกต์ใช้กับการทำอีควอลไลซ์ (Equalized) ภาพได้ ซึ่งเรียกว่า ฮิสโตแกรม อีควอลไลซ์ (Histogram Equalized) นอกจากนั้นแล้วยังสามารถนำมาประยุกต์ใช้ในงานด้านการบีบอัดภาพ (Image Compression) และการคัดแยกส่วนภาพ (Image Segmentation) อีกด้วย

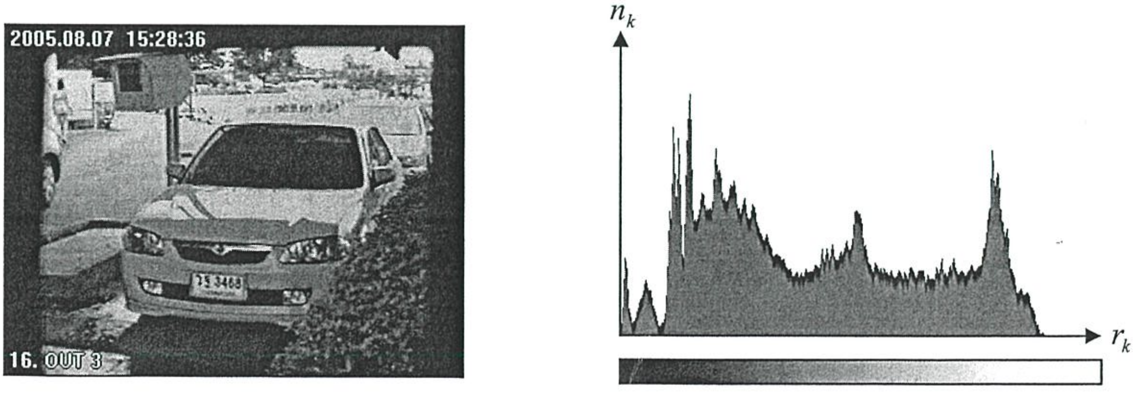
2.1.8.1 ฮิสโตแกรมของภาพระดับเทา (Gray Level Image)

ฮิสโตแกรมของภาพระดับเทาที่มีช่วงระหว่าง [0, *L-1*] คือฟังก์ชันไม่ต่อเนื่อง $h(r_k) = n_k$ เมื่อ r_k คือค่าระดับเทาในระดับที่ k และ n_k คือจำนวนของพิกเซลในภาพที่มีค่าระดับเทาเท่ากับ r_k ซึ่งเรานิยมทำนอร์มอลไลซ์ (Normalized) ฮิสโตแกรมนี้ด้วยการหารแต่ละค่าของฮิสโตแกรมด้วยจำนวนพิกเซลทั้งหมดในภาพซึ่งก็คือค่า n หากจะพูดให้ชัดเจนก็คือค่า คือความน่าจะเป็นในการเกิดพิกเซลที่มีค่าระดับเทา ซึ่งเมื่อรวมทุกค่าของนอร์มอลไลซ์ฮิสโตแกรมนี้จะได้เท่ากับ 1 โดยสามารถเขียนให้อยู่ในรูปสมการได้ดังสมการที่ (2.21) .

$$p(r_k) = \frac{n_k}{n} \tag{2.21}$$

เมื่อ $k = 0,1,...,L-1$

โดยที่เราสามารถสร้างกราฟฮิสโตแกรมได้โดยการกำหนดให้แกนนอนของกราฟเป็นค่าระดับเทา r_k และแกนแนวตั้งคือจำนวนค่าของฟังก์ชัน $h(r_k) = n_k$ หรือ $p(r_k) = n_k / n$ ก็ได้ในกรณีที่ทำนอร์มอลไลซ์ ดังแสดงในรูปที่ 2.12



(ก) (ข)

รูปที่ 2.12 การสร้างกราฟฮิสโตแกรมจากภาพระดับเทา (ก) ภาพต้นฉบับ (ข) กราฟฮิสโตแกรม

2.1.8.2 ฮิสโตแกรมของภาพสองระดับ

นอกจากฮิสโตแกรมของภาพระดับเทาแล้วยังมีฮิตโตแกรมอีกประเภทหนึ่งที่นำมาประยุกต์ใช้ในวิทยานิพนธ์ฉบับนี้คือ โปรเจคชันฮิสโตแกรม (Projection Histogram) ซึ่งเป็นการดำเนินการกับภาพสองระดับ โดยเป็นการแสดงถึงผลรวมของค่าไบนารี (Binary) ของพิกเซลในแต่ละแถวหรือหลัก สำหรับรวมกันในแนวแถวเรียกว่าโปรเจคชันฮิสโตแกรมแนวนอน (Horizontal Projection Histogram) และในกรณีรวมกันในแนวหลักเรียกว่าโปรเจคชันฮิสโตแกรมแนวตั้ง (Vertical Projection Histogram) ดังภาพที่ 2.13 โดยกำหนดให้พิกเซลสีดำมีค่าไบนารีเป็น 1 และพิกเซลสีขาวมีค่าไบนารีเป็น 0 ซึ่งสามารถคำนวณได้จากสมการที่ (2.22) และ (2.23)

$$H_H(x) = \sum_{y=1}^{n} B(x,y) \tag{2.22}$$

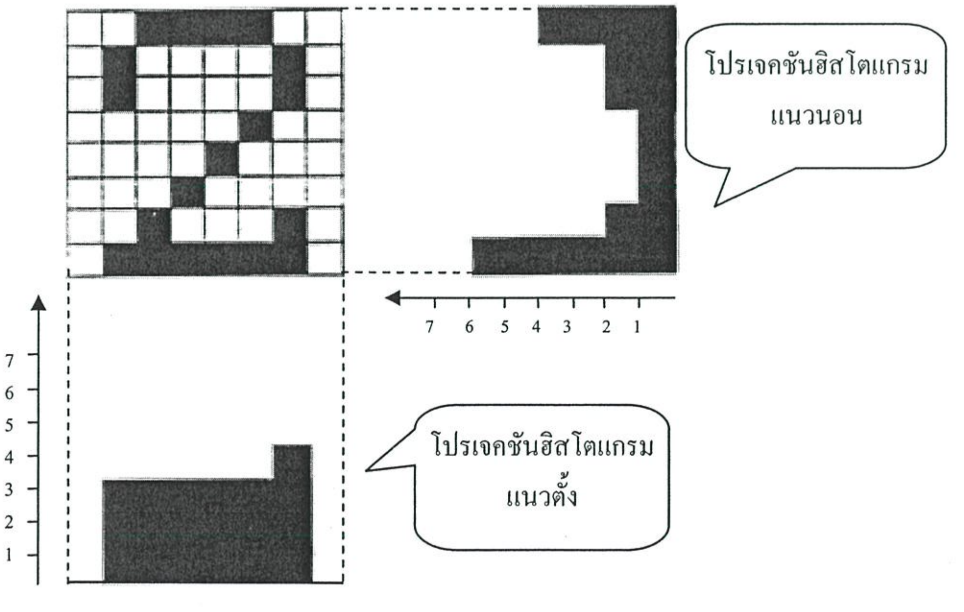
$$H_V(y) = \sum_{x=1}^{m} B(x,y) \tag{2.23}$$

โดยที่ $H_H(x)$ คือโปรเจคชันฮิสโตแกรมแนวนอนในแถว x

$H_V(y)$ คือโปรเจคชันฮิสโตแกรมแนวตั้งในหลัก y

$B(x,y)$ คือค่าไบนารีของพิกเซลที่ตำแหน่งแถว x และหลัก y

m และ n คือจำนวนแถวและหลักของภาพตามลำดับ



รูปที่ 2.13 ตัวอย่างการหาโปรเจคชันฮิสโตแกรมทั้งแนวนอนและแนวตั้งของภาพสองระดับ

2.1.9 การวัดประสิทธิภาพของการตรวจจับ

ตัวแปรสำคัญที่นำมาใช้สำหรับการวัดประสิทธิภาพของการตรวจจับเกี่ยวกับงานด้านคอมพิวเตอร์วิชัน (Computer Vision) จะมีอยู่ด้วยกัน 4 ตัวแปร โดยสมมติว่าเรากำลังตรวจหาวินโดว์ที่มีทะเบียนบรรจุอยู่

1. TP ย่อมาจาก "True Positive" หมายถึง จำนวนวินโดว์ที่คิดว่ามีป้ายทะเบียนและปรากฏว่ามีป้ายทะเบียนอยู่ในนั้นจริง
2. TN ย่อมาจาก "True Negative" หมายถึง จำนวนวินโดว์ที่ไม่คิดว่ามีป้ายทะเบียนและปรากฏว่าไม่มีป้ายทะเบียนอยู่จริง
3. FP ย่อมาจาก "False Positive" หมายถึง จำนวนวินโดว์ที่คิดว่ามีป้ายทะเบียนแต่กลับปรากฏว่าไม่มีป้ายทะเบียนอยู่จริง
4. FN ย่อมาจาก "False Negative" หมายถึง จำนวนวินโดว์ที่ไม่คิดว่ามีป้ายทะเบียนแต่กลับปรากฏว่าป้ายทะเบียนอยู่

ค่า False Positive Rate คืออัตราส่วนระหว่าง FP และ TN รวมกับ FP สามารถคำนวณได้จากสมการที่ (2.24)

$$False\,Positive\,Rate = \frac{FP}{TN + FP} \tag{2.24}$$

ค่า False Negative Rate คืออัตราส่วนระหว่าง FN และ FN รวมกับ TP สามารถคำนวณได้จากสมการที่ (2.25)

$$False\,Negative\,Rate = \frac{FN}{TP + FN} \tag{2.25}$$

ค่า DR ย่อมาจาก "Detection Rate" หรือบางครั้งเรียกว่า True Positive Rate คืออัตราความถูกต้องในการตรวจหา มีหน่วยเป็น (%) เปอร์เซ็นต์ ซึ่งแบบจำลองที่ให้ค่า DR มากกว่าแสดงถึงความถูกต้องในการตรวจหาได้ดีกว่า สามารถหาได้จากสมการที่ (2.26)

$$DR = \frac{TP}{TP + FN} \times 100\% \tag{2.26}$$

ค่า FAR ย่อมาจาก “False Alarm Rate” คือ อัตราเกิดความผิดพลาด มีหน่วยเป็น (%) เปอร์เซ็นต์ โดยแบบจำลองที่มีประสิทธิภาพมากกว่าจะมีค่า FAR ต่ำกว่า ซึ่งสามารถหาได้จากสมการที่ (2.27)

$$FAR = \frac{FP}{TP + FP} \times 100\% \tag{2.27}$$

2.2 การเคลื่อนที่ของวัตถุในแนวเส้นตรง

ในวิทยานิพนธ์ฉบับนี้นำหลักการของการเคลื่อนที่ของวัตถุในแนวเส้นตรงมาใช้ด้วยกันสามส่วนได้แก่ ตำแหน่ง ความเร็ว และความเร่ง ซึ่งเป็นปริมาณเวคเตอร์โดยเมื่อวัตถุเปลี่ยนตำแหน่งในช่วงเวลาหนึ่งจะทำให้เกิดความเร็วดังสมการที่ (2.28)

$$\mathrm{v} = \frac{\Delta \mathrm{d}}{\Delta t} = \frac{\mathrm{d}_2 - \mathrm{d}_1}{t_2 - t_1} \tag{2.28}$$

โดยที่ v คือความเร็ว

$\mathrm{d}_1, \mathrm{d}_2$ คือตำแหน่งจุดเริ่มต้นและจุดปลายตามลำดับ

t_1, t_2 คือเวลาที่จุดเริ่มต้นและจุดปลายตามลำดับ

และเมื่อวัตถุเปลี่ยนความเร็วในช่วงเวลาหนึ่งจะทำให้เกิดความเร่งดังสมการที่ (2.29)

$$\mathrm{a} = \frac{\Delta \mathrm{v}}{\Delta t} = \frac{\mathrm{v}_2 - \mathrm{v}_1}{t_2 - t_1} \tag{2.29}$$

โดยที่ a คือความเร่ง

$\mathrm{v}_1, \mathrm{v}_2$ คือความเร็วที่จุดเริ่มต้นและจุดปลายตามลำดับ

t_1, t_2 คือเวลาที่จุดเริ่มต้นและจุดปลายตามลำดับ

เราสามารถนำความรู้พื้นฐานเรื่องการเคลื่อนที่ในแนวเส้นตรงนี้ไปประยุกต์ใช้ในการตรวจสอบการเคลื่อนที่ของวัตถุว่าควรจะเป็นป้ายทะเบียนตามที่เราต้องการหรือไม่ โดยการหาตำแหน่งของวัตถุในแต่ละลำดับภาพ แล้วจึงนำตำแหน่งที่หาได้มาคำนวณหาความเร็วและความเร่งของวัตถุนั้นต่อไป

2.3 งานวิจัยที่เกี่ยวข้องกับการตรวจหาป้ายทะเบียน

Changping Liu [4] ได้เสนอขั้นตอนและวิธีการตรวจหาป้ายทะเบียนรถยนต์บนภาพพื้นหลังที่มีความซับซ้อน (Complex Background) โดยใช้หลักการการหาขอบในแนวตั้งของภาพ เนื่องจากตั้งสมมติฐานว่าป้ายทะเบียนนั้นจะมีขอบมากกว่าบริเวณอื่นๆของภาพ จากนั้นคำนวณหาความหนาแน่นของขอบโดยการใช้วินโดว์ขนาด 3×15 วิ่งไปทั่วทั้งภาพแล้วคำนวณหาประมาณของขอบในวินโดว์นั้นๆ ซึ่งจะได้ แมป (map) ของความหนาแน่นของขอบ แล้วนำไปผ่านกระบวนการ ไบนาไลเซชัน (Binarization) และไดเลชัน (Dilation) เพื่อรวมพื้นที่ส่วนเล็กๆให้รวมกัน จากนั้นนำไปผ่านกระบวนการวิเคราะห์หาส่วนที่ติดกัน (Connected Component Analysis) เพื่อแยกวัตถุแต่ละชิ้นบนภาพ แล้วนำมาวิเคราะห์ลักษณะทางกายภาพเช่นอัตราส่วนความกว้างต่อความยาว พื้นที่ และความหนาแน่นของวัตถุแต่ละชิ้นว่าชิ้นไหนควรเป็นป้ายทะเบียน

Chen Zhibin [5] ได้เสนอการหาตำแหน่งป้ายทะเบียนรถยนต์โดยการวิเคราะห์สี โดยเริ่มจากการหาขอบโดยใช้ตัวดำเนินการแบบโซเบล ใช้วินโดว์ขนาดใกล้เคียงกับป้ายทะเบียนในภาพวิ่งไปทั่วทั้งภาพเพื่อหาปริมาณของขอบในแต่ละวินโดว์โดยใช้สมการที่ (2.30)

$$R(i,j) = \sum_{m=0}^{H}\sum_{n=0}^{W} E(m+i, n+j) \tag{2.30}$$

โดยที่ $R(i,j)$ คือจำนวนของขอบที่อยู่ในวินโดว์

H,W คือความสูงและความกว้างของวินโดว์ที่ใช้ตรวจหา

หากวินโดว์ใดมีขอบมากเกินค่าจุดเริ่มเปลี่ยนจะตั้งข้อสงสัยว่าน่าจะมีป้ายทะเบียนอยู่ จากนั้นนำไปผ่านนิวรอลเน็ตเวิร์ค (Neural Network) เพื่อวิเคราะห์สีของวัตถุว่ามีความเป็นไปได้ที่จะเป็นป้ายทะเบียนได้มากน้อยแค่ไหน เช่นพื้นหลังต้องเป็นสีเหลือง ฟ้า หรือ ดำเท่านั้น และ ตัวอักษรต้องเป็นสีดำหรือขาวเท่านั้น เป็นต้น แล้วจึงตัดสินว่าวินโดว์ไหนควรจะเป็นวินโดว์ที่มีป้ายทะเบียน

วุฒิพงศ์ [6] ได้เสนอการหาตำแหน่งป้ายทะเบียนรถยนต์โดยการตรวจสอบขอบ โดยใช้วิธี Bound-Rebound ในการหาขอบ และใช้ค่าจุดเริ่มเปลี่ยนแบบเปลี่ยนแปลงได้มาคำนวณ จากนั้นหาโปรเจคชันฮิสโตแกรมทั้งแนวนอนและแนวตั้งเพื่อหาบริเวณของภาพที่มีปริมาณขอบอยู่มากทั้งแนวตั้งและแนวนอน จากนั้นวิเคราะห์จากนั้นตรวจสอบอัตราส่วนความกว้างและยาวของภาพส่วนที่ได้เพื่อขจัดส่วนที่ไม่เหมาะที่จะเป็นป้ายทะเบียนทิ้งไป แล้วนำภาพส่วนที่เหลือมาทำไบนาไลเซชัน แล้วกวาดไปตามแนวนอนเพื่อหาการเปลี่ยนแปลงจะขาวไปดำและดำไปขาวเพื่อ ทดสอบว่าเป็นป้ายทะเบียนหรือไม่เนื่องจากบริเวณที่เป็นป้ายทะเบียนนั้นขอบระหว่างตัวอักษรและพื้นหลังจะมีค่าอยู่ปริมาณช่วงหนึ่ง ซึ่งแตกต่างจะบริเวณอื่นๆ ของภาพ

Shen-Zheng Wang [7] ได้เสนอวิธีการหาป้ายทะเบียนจากรถจักรยานยนต์และรถยนต์บนถนนหลวง โดยใช้ภาพที่มีขนาดใหญ่ 2560×1920 pixels ทำให้สามารถหาป้ายทะเบียนได้ทีละหลายๆป้าย ซึ่งเริ่มจากการแบ่งภาพออกเป็นบล็อกเล็กๆ ขนาด 40×20 pixels แล้วตัดบล็อกที่มีความคมชัด (Contrast) ต่ำออก เนื่องจากผู้เขียนตั้งสมมติฐานว่าป้ายทะเบียนจะมีความคมชัดสูง จากนั้นตัดบล็อกที่มีความเปลี่ยนแปลงน้อยออก เพื่อคัดเอาเฉพาะส่วนที่มีการเคลื่อนไหวเก็บไว้เท่านั้น สาเหตุที่ทำดังนี้ก็เพื่อลดเวลาในการทำงานลงโดยตัดส่วนที่ไม่เกี่ยวข้องออกไปก่อน จากนั้นหาขอบโดยใช้ตัวดำเนินการโซเบล (Sobel Operator) ในแนวตั้ง แล้วทำโปรเจคชันฮิสโตแกรม (Projection Histogram) แนวนอนเพื่อตัดแถวแนวนอนที่มีเส้นขอบน้อยออก เนื่องจากผู้เขียนตั้งสมมติฐานว่าบริเวณที่เป็นป้ายทะเบียนจะมีขอบมาก จากนั้นทำโปรเจคชันฮิสโตแกรมในแนวตั้งเพื่อตัดแถวแนวตั้งที่มีเส้นขอบน้อยออกเช่นเดียวกัน ซึ่งจะได้ส่วนของภาพหลายๆส่วนที่คาดว่าจะเป็นป้ายะเบียน หลังจากพิจารณาภาพส่วนที่เหลือว่าเป็นป้ายทะเบียนหรือไม่จาก 1.ขนาดของภาพส่วนนั้น 2.อัตราส่วนของความกว้าง/สูง 3.ต้องไม่มีสีขาวหรือดำที่ต่อเป็นแนวยาวตลอดทั้งแถวในภาพนั้น 4.จำนวนจุดสีขาวสีดำที่ต่อเนื่องกันต้องไม่เกินกว่าที่กำหนดไว้เนื่องจากผู้เขียนตั้งสมมติฐานว่าป้ายทะเบียนนั้นจะต้องมีส่วนที่เป็นสีขาวและดำสลับกันไปตลอดทั้งภาพ ซึ่งภาพส่วนไหนที่ครบองค์ประกอบทั้ง 4 ดังกล่าวถือว่าภาพส่วนนั้นเป็นป้ายทะเบียน

Tran Le Hong Du [8] ได้เสนอการหาป้ายทะเบียนรถโดยใช้คอนทัวร์อัลกอริธึม (Contour Algorithm) ร่วมกับเฮ้าทรานส์ฟอร์ม (Hough Transform) โดยเริ่มจากการใช้ตัวดำเนินการโซเบลเพื่อหาขอบแล้วใช้คอนทัวร์อัลกอริธึมในการหารูปทรงปิด จากนั้นใช้เพื่อหาเส้นขนานสองคู่ที่ปลายทั้งสองชนกันแล้วเกิดเป็นรูปสี่เหลี่ยม โดยสาเหตุที่ต้องผ่านคอนทัวร์อัลกอริธึมก่อนก็เพื่อลดเวลาคำนวณในขั้นตอนทำเฮ้าทรานส์ฟอร์ม ซึ่งขั้นตอนนี้จะได้ส่วนที่คาดว่าจะเป็นป้ายทะเบียน จากนั้นนส่วนที่คาดว่าจะเป็นป้ายทะเบียนมาตรวจสอบว่าเป็นป้ายทะเบียนหรือไม่ โดยนำมาผ่านโรเทททรานส์ฟอร์มเพื่อหมุนภาพให้ตรง แล้วตรวจสอบอัตราส่วนของความกว้างและความสูงว่าอยู่ในช่วงที่กำหนดหรือไม่ และสุดท้าย ทำพิจารณาการตัดผ่านทางแนวนอน (Horizontal Crosscut) เพื่อทดสอบว่าจำนวนขอบที่ถูกตัดผ่านนั้นอยู่ในช่วงที่กำหนดหรือไม่ ถ้าผ่านขั้นตอนทั้งหมดก็จะถือว่าส่วนที่ได้นั้นเป็นป้ายทะเบียน

R. Khosal [9] ได้เสนอระบบการหาป้ายทะเบียนในวันที่มีฝนตก โดยใช้หลักการที่เรียกว่า Real Genetic Algorithm (RGA) ในการค่าจุดเริ่มเปลี่ยน (Threshold) ของค่ามากสุดและค่าน้อยสุดของแต่ละองค์ประกอบสีพื้นฐานแดงเขียวและน้ำเงินที่เหมาะสมที่จะเป็นป้ายทะเบียน แล้วนำค่าจุดเริ่มเปลี่ยนที่ได้ทั้งหมดมาสร้างกราฟเทียบกับความเข้มแสงเฉลี่ยของตัวรถ จากนั้นใช้วิธีรีเคอร์ซีฟลีสสแควร์ (Recursive Least Square) เพื่อนสร้างเป็นฟังก์ชัน แล้วใช้ฟังก์ชันนี้ในการคำนวณหาค่าจุดเริ่มเปลี่ยนที่เหมาะสมกับสภาพแสงของรถแต่ละคัน แล้วคัดเฉพาะพิกเซลที่อยู่ในช่วงค่าจุดเริ่มเปลี่ยนที่เหมาะสมออกมาจากภาพ แล้วจึงนำมาผ่านขั้นตอนการตัดแถวและหลักของภาพที่ไม่ใช่

ป้ายทะเบียนออกด้วยการใช้โปรเจคชันฮิสโตแกรมเพื่อตัดส่วนที่มีจำนวนพิกเซลน้อยออก เพื่อให้เหลือแต่ส่วนของภาพที่มีขนาดใหญ่ จากนั้นนำภาพส่วนที่คัดออกมาได้ มาพิจารณาความสมมาตรกัน ซึ่งผู้เขียนตั้งสมมติฐานว่าป้ายทะเบียนนั้นมีความสมมาตรกัน ซึ่งภาพส่วนไหนมีความสมมาตรมากที่สุดให้ภาพนั้นเป็นป้ายทะเบียน

Sei-Wan Chen [10] ได้เสนอการหาป้ายทะเบียนรถยนต์ด้วยการตรวจสอบสี โดยใช้โดยใช้โมโดลสี HSI มาสร้างแมปของสีเฉพาะสีที่เป็นสีของป้ายทะเบียนได้แก่สีแดง เขียว ขาวและดำ จากนั้นสร้างแมปของขอบสีเฉพาะสีที่เป็นป้ายทะเบียนเช่นขอบของสีขาว-ดำ ขาว-แดง และขาว-เขียว แล้วนำแมปทั้งหมดมารวมกันแล้วนำไปใช้กับภาพที่ต้องการหาป้ายทะเบียน ส่วนของภาพใดที่ตรงตามเงื่อนไขของแมปมากที่สุดก็ถูกกำหนดให้ส่วนนั้นเป็นป้ายทะเบียน

2.4 ระบบป้ายทะเบียนรถที่ใช้ในประเทศไทย

ในประเทศไทยนั้นรถยนต์จะถูกกำหนดให้ติดตั้งป้ายทะเบียนทั้งด้านหน้าและด้านหลังของรถทุกคัน ส่วนรถจักรยานยนต์นั้นให้ติดเฉพาะด้านหลังเท่านั้น โดยเกือบทั้งหมดจะถูกกำหนดและผลิตโดยกรมการขนส่งทางบก กระทรวงคมนาคม ยกเว้นรถที่ใช้ในราชการทหารและตำรวจ ที่ออกโดยกองพลาธิการทหารและตำรวจ รวมถึงรถในราชการสำนักพระราชวัง ที่ออกโดยกองพระราชพาหนะ สำนักพระราชวัง โดยป้ายทะเบียนที่พบเห็นส่วนใหญ่นั้นแบ่งเป็นสี่ประเภทคือ

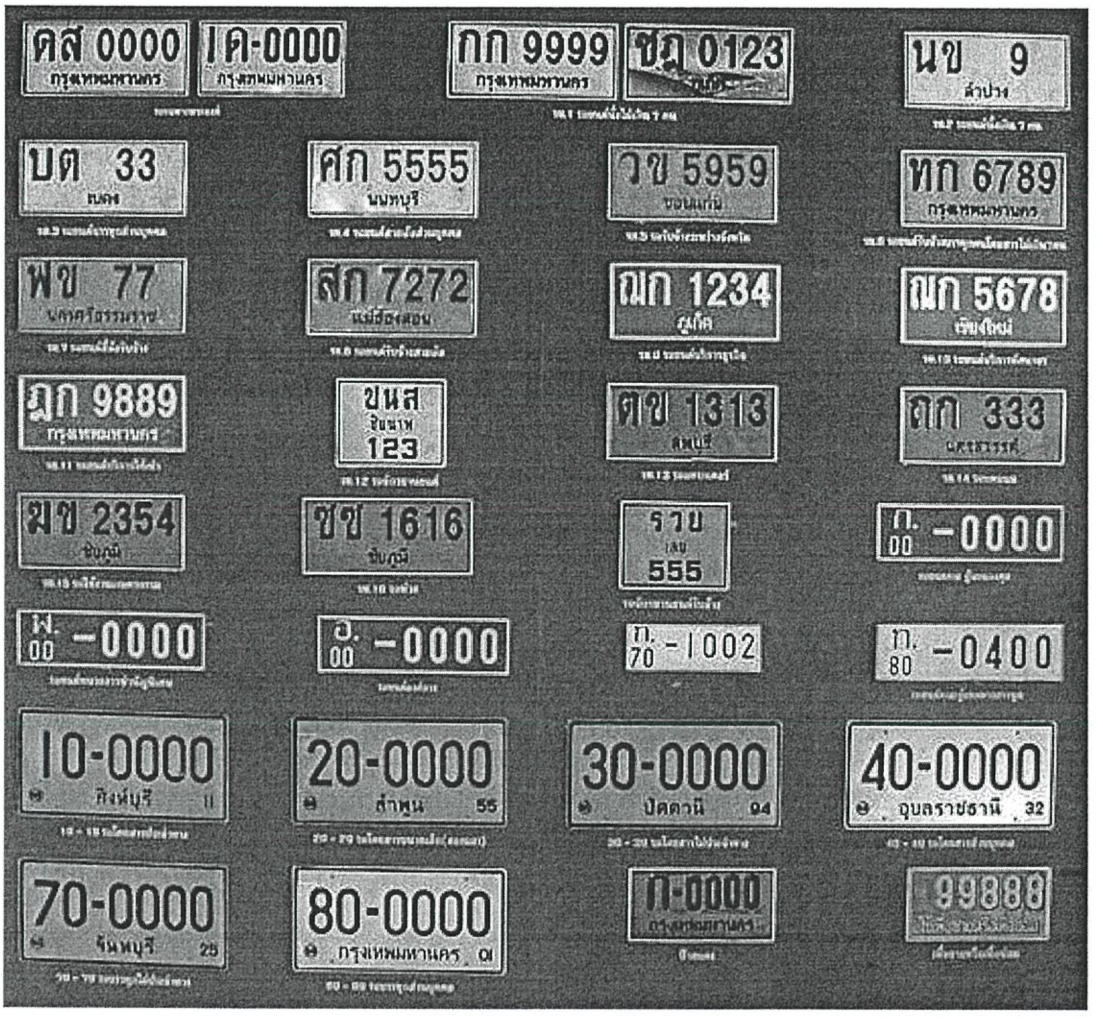
1. ป้ายรถยนต์รุ่นเก่า ในปัจจุบันนี้ยังมีป้ายชนิดนี้อยู่แต่ไม่มากนัก สำหรับซึ่งที่พบส่วนใหญ่เป็นป้ายทะเบียนพื้นขาวเลขทะเบียนสีดำ และป้ายแดงซึ่งมีพื้นสีแดงและเลขทะเบียนสีดำ โดยป้ายทะเบียนประกอบไปด้วยหมวดอักษรและหมายเลขทะเบียน ซึ่งหมวดอักษรสำหรับกรุงเทพมหานครนั้นจะมีตัวเลขหนึ่งหลักนำหน้าหมวดตัวอักษรด้วย ส่วนป้ายต่างจังหวัดและป้ายแดงจะมีเพียงตัวอักษรตัวเดียวเท่านั้น ตามด้วยเครื่องหมาย “-” และหมายเลขทะเบียน 4 หลัก ดังตัวอย่างในรูปที่ 2.14 โดยมีขนาดตามที่กฎหมายกำหนดในรูปที่ 2.15

2. ป้ายรถยนต์รุ่นใหม่ เป็นป้ายที่ใช้เป็นส่วนใหญ่ในปัจจุบันและมีการกำหนดสีแตกต่างกันไปตามประเภทของรถดังรูปที่ 2.14 แต่ที่พบส่วนใหญ่เป็นป้ายทะเบียนพื้นขาวเลขทะเบียนสีดำ ฟ้า และเขียว และป้ายทะเบียนพื้นเหลืองเลขทะเบียนสีดำในกรณีรถแท็กซี่ รวมถึงป้ายทะเบียนพิเศษที่ใช้วิธีประมูลที่มีพื้นหลังเป็นภาพสีแตกต่างกันไปตามแต่ละจังหวัด ป้ายทะเบียนรุ่นใหม่นี้ประกอบไปด้วยหมวดอักษรและหมายเลขทะเบียน โดยหมวดอักษรจะมีตัวอักษรสองตัวตามด้วยช่องว่างและเลขทะเบียนจำนวน 1, 2, 3 หรือ 4 หลักแล้วแต่หมายเลขทะเบียนมีขนาดตามที่กฎหมายกำหนดในรูปที่ 2.15

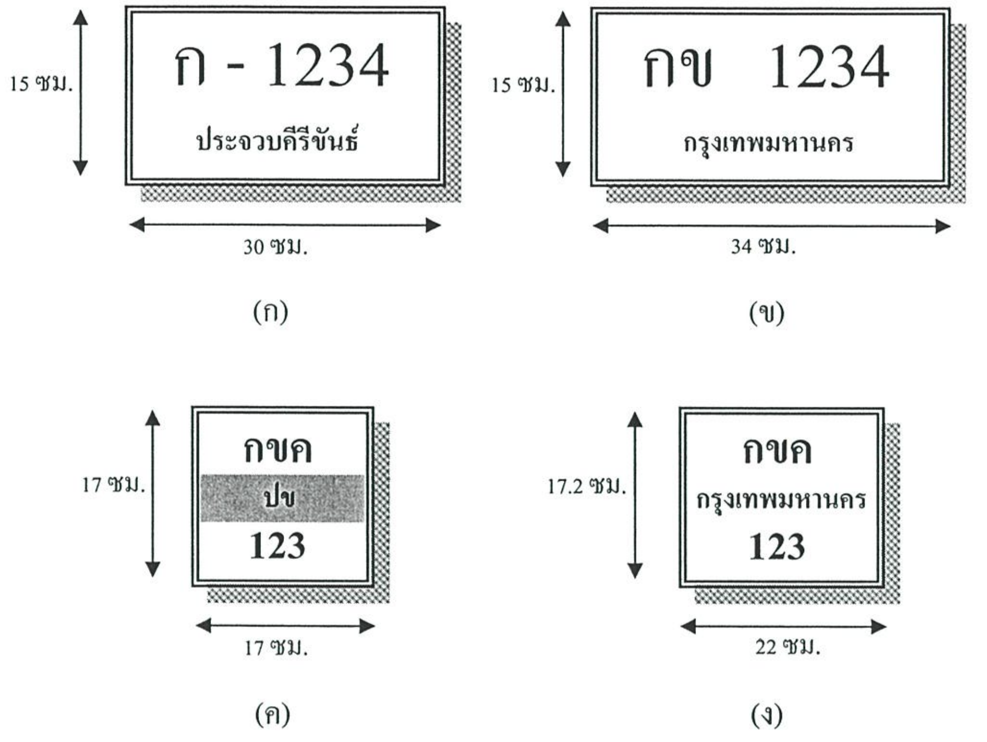
3. ป้ายรถจักรยานยนต์รุ่นเก่า ป้ายชนิดนี้ยังมีให้เห็นอยู่จำนวนมากในปัจจุบัน มีลักษณะเป็นป้ายพื้นหลังสีขาวยกเว้นบริเวณที่บอกชื่อย่อจังหวัดจะมีพื้นหลังสีแดงส่วนตัวอักษรจะเป็นสีดำ

โดยแบ่งออกเป็นสามบรรทัด บรรทัดแรกเป็นหมวดอักษรจำนวนสามตัวอักษร บรรทัดต่อมาเป็นตัวย่อแสดงจังหวัด ส่วนบรรทัดสุดท้ายเป็นหมวดตัวเลข 1,2 หรือ 3 หลักแล้วแต่หมายเลขทะเบียนนั้นๆ และมีขนาดตามที่กฎหมายกำหนดดังในรูปที่ 2.15

4. ป้ายรถจักรยานยนต์รุ่นใหม่ เป็นป้ายที่มีใช้อยู่ในปัจจุบันมากที่สุด โดยมีทั้งพื้นสีขาวตัวอักษรดำและพื้นสีเหลืองอักษรดำ ตามประเภทของรถดังรูปที่ 2.14 ซึ่งประกอบด้วยสามบรรทัด บรรทัดแรกเป็นหมวดอักษรสามตัวอักษร บรรทัดต่อมาเป็นชื่อเต็มบอกจังหวัด ส่วนบรรทัดสุดท้ายเป็นหมวดตัวเลขจำนวน 1,2 หรือ 3 หลักแล้วแต่หมายเลขทะเบียน โดยมีขนาดตามที่กฎหมายกำหนดดังรูปที่ 2.15



รูปที่ 2.14 ตัวอย่างป้ายทะเบียนรถที่ออกโดยกรมการขนส่งทางบก



รูปที่ 2.15 ขนาดของป้ายทะเบียนตามที่กฎหมายกำหนด (ก) ป้ายรถยนต์รุ่นเก่า (ข) ป้ายรถยนต์รุ่นใหม่ (ค) ป้ายรถจักรยานยนต์รุ่นเก่า (ง) ป้ายรถจักรยานยนต์รุ่นใหม่

บทที่ 3
การหาตำแหน่งของป้ายทะเบียนรถจากวีดิทัศน์

จากสมมุติฐานที่ว่าส่วนของภาพที่เป็นป้ายทะเบียนนั้น จะเป็นบริเวณที่มีความแตกต่างกันของสีในบริเวณใกล้เคียงกันชัดเจนมากกว่าส่วนอื่นๆ ของภาพ ซึ่งมีผลทำให้เกิดขอบจำนวนมากอันเนื่องมาจากขอบระหว่างตัวอักษรบนป้ายทะเบียนและพื้นหลังของป้ายทะเบียน วิธีการที่นำเสนอในวิทยานิพนธ์ฉบับนี้จึงเริ่มต้นด้วยการหาขอบแนวตั้ง (Vertical Edge) ที่เกิดขึ้นภายในภาพ จากนั้นใช้วินโดว์ (Window) ที่มีอัตราส่วนกว้างยาว(Aspect Ratio) ใกล้เคียงกับป้ายทะเบียนจำนวนสามขนาดสแกนทั่วทั้งภาพทั้งแนวนอนและแนวตั้ง แล้วคำนวณหาการกระจายของขอบที่ปรากฏอยู่ในแต่ละวินโดว์แล้วนำมาพิจารณาว่า การกระจายของขอบในวินโดว์ใดเหมาะสมกับการเป็นป้ายทะเบียนบ้าง และจะถูกเรียกว่าแคนดิเดทวินโดว์ (Candidate Windows) ซึ่งเป็นวินโดว์ที่คาดว่าจะมีป้ายทะเบียนบรรจุอยู่ จากนั้นคำนวณหาแคนดิเดทวินโดว์กับภาพจากเฟรม (Frame) ถัดไปด้วยวิธีการเดิม ต่อมาแคนดิเดทวินโดว์ทั้งหมดในแต่ละเฟรมจะถูกจัดกลุ่มกับแคนดิเดทวินโดว์จากเฟรมอื่นๆ เพื่อให้เป็นวินโดว์ที่มีความเกี่ยวข้องกัน (Corresponding Windows) จากนั้นพิจารณาตำแหน่งของวินโดว์ที่เกี่ยวข้องกันในแต่ละเฟรมเพื่อนำมาคำนวณหาความเร็วและความเร่งของแต่ละวินโดว์ เพื่อที่จะใช้ค่าที่คำนวณได้นี้มาจำแนกว่าวินโดว์นั้นควรจะเป็นป้ายทะเบียนหรือไม่จากสมมุติฐานที่ว่า ในขณะที่รถเคลื่อนที่ช่วงเวลาระหว่างเฟรมภาพนั้นสั้นมาก ดังนั้นในช่วงเวลาสั้นๆนี้ เราสามารถประมาณได้ว่าป้ายทะเบียนย่อมเคลื่อนที่เป็นเชิงเส้นด้วยความเร็วคงที่ กล่าวคือ ในช่วงเวลาระหว่างเฟรมภาพนั้นความเร็วของป้ายทะเบียนจะต้องไม่เป็นศูนย์เพราะว่ารถกำลังเคลื่อนที่ และความเร่งของป้ายทะเบียนจะต้องเป็นศูนย์หรือมีความเร็วคงที่นั่นเองซึ่งทั้งหมดนี้จะนำมากล่าวโดยละเอียดเป็นลำดับต่อไป

3.1 การตรวจหาขอบ (Edge Detection)

ดังที่กล่าวมาตอนต้นว่าการตรวจหาขอบเป็นขั้นตอนแรกของวิธีการที่นำเสนอในวิทยานิพนธ์ฉบับนี้ โดยการตรวจหาขอบจะใช้วิธีการตรวจหาขอบจากภาพระดับเทา ดังนั้นเราต้องแปลงภาพสีที่ได้จากลำดับภาพวีดิทัศน์ให้เป็นภาพระดับเทาเสียก่อน ดังสมการที่ (3.1)

$$GrayScale = \frac{R+G+B}{3} \tag{3.1}$$

โดยที่ *Gray Scale* คือค่าความสว่างของพิกเซลของภาพระดับเทาที่เราต้องการ และ *R, G, B* คือค่าความสว่างของสีแดง เขียว และน้ำเงินตามลำดับ

3.1.1 การตรวจหาขอบแบบโซเบล (Sobel Edge Detector)

เมื่อได้ภาพระดับเทาแล้วจะเข้าสู่กระบวนการหาขอบ ในวิทยานิพนธ์ฉบับนี้ใช้ การตรวจหาขอบแบบโซเบล ซึ่งสามารถตรวจหาขอบได้ทั้งแนวตั้งและแนวนอน ซึ่งจากการทดลองพบว่าในบริเวณที่เป็นป้ายทะเบียนนั้น จะมีขอบในแนวตั้งที่ชัดเจนกว่าบริเวณอื่นๆของภาพ ในขณะที่ขอบในแนวนอนนั้นไม่มีความแตกต่างกันอย่างชัดเจนมากนัก เนื่องจากบริเวณกระจังหน้ารถยนต์ส่วนใหญ่จะก่อให้เกิดขอบในแนวนอนซึ่งจะทำให้เกิดสิ่งรบกวน (Noise) ในการหาตำแหน่งป้ายทะเบียนได้ ดังนั้นวิทยานิพนธ์ฉบับนี้จึงเลือกใช้การตรวจหาขอบเฉพาะในแนวตั้งเท่านั้น โดยใช้มาสค์ (Mask) ดังแสดงในรูปที่ 3.1 และสมการที่ (3.2)

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

-1	0	1
-2	0	2
-1	0	1

รูปที่ 3.1 มาสค์สำหรับการตรวจหาขอบในแนวตั้งแบบโซเบล

$$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \tag{3.2}$$

โดยที่ *Gy* คือค่าความเป็นขอบในแนวแกนตั้งของพิกเซลตำแหน่ง z_5 ของรูปที่ 3.1

z_i คือค่าความสว่างของพิกเซลที่ตำแหน่งที่ i ของรูปที่ 3.1

เมื่อได้ค่า *Gy* ของทุกพิกเซลในภาพแล้ว ขั้นต่อมาต้องทำการพิจารณาว่าพิกเซลใดในภาพควรจะเป็นขอบหรือไม่โดยดูที่ค่า *Gy* ว่าถ้ามีขนาดมากกว่าหรือเท่ากับค่าจุดเริ่มเปลี่ยน (Threshold) พิกเซลนั้นก็จะเป็นขอบหรือถ้าน้อยกว่าก็ไม่เป็นขอบดังสมการที่ (3.3)

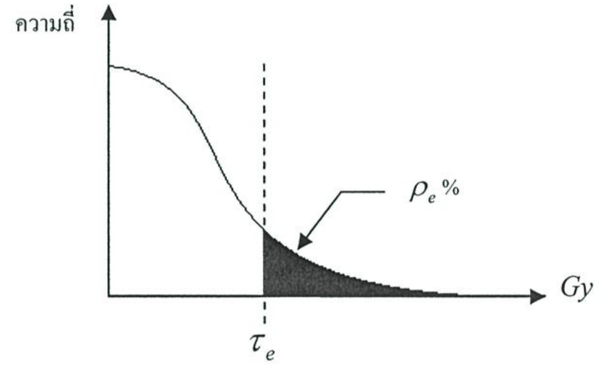
$$E = \begin{cases} 1 & \text{เมื่อ } G_y \geq \tau_e \\ 0 & \text{เมื่อ } G_y < \tau_e \end{cases} \tag{3.3}$$

เมื่อ E คือค่าไบนารี (Binary) ของพิกเซลที่แสดงถึงความเป็นขอบหรือไม่เป็นขอบ

τ_e คือค่าจุดเริ่มเปลี่ยนของการเป็นขอบหรือไม่เป็นขอบ

3.1.2 การหาค่าจุดเริ่มเปลี่ยนชนิดเปลี่ยนแปลงได้

การกำหนดค่าจุดเริ่มเปลี่ยนให้เป็นค่าคงที่นั้นจะทำให้ผลของการหาขอบมีความคลาดเคลื่อนได้เมื่อสภาพของแสงเปลี่ยนไป เช่นแสงในช่วงเวลาเที่ยงวันกับแสงในช่วงเวลากลางคืน ดังนั้นจึงต้องมีการหาค่าจุดเริ่มเปลี่ยนที่เหมาะสมในแต่ละสภาพแสง [9] โดยการสร้างฮิสโตแกรม (Histogram) แสดงค่าความถี่ของค่าความเป็นขอบ (Gy) ของทุกพิกเซลบนภาพดังรูปที่ 3.2 จากนั้นหาค่า τ_e จากค่าของตำแหน่งบนแกน Gy ที่ทำให้พื้นที่ใต้กราฟส่วนที่มีค่ามากกว่าหรือเท่ากับค่า τ_e(พื้นที่แรเงาในรูป 3.2) คิดเป็น ρ_e% ของพื้นที่ใต้กราฟทั้งหมด ซึ่งหมายถึงเปอร์เซ็นต์ของพิกเซลขอบเมื่อเทียบกับจำนวนพิกเซลทั้งหมดบนภาพ



รูปที่ 3.2 การหาค่า τ_e จากฮิสโตแกรม

ค่า ρ_e ที่เหมาะสมที่จะนำมาใช้หาค่า τ_e นั้น ขึ้นอยู่กับสภาพวะแวดล้อมในแต่ละสถานที่ซึ่งสามารถหามาได้จากการทดลอง หลังจากคำนวณหาค่า τ_e ได้แล้ว จึงนำไปแทนค่าในสมการที่ (3.3) ก็จะทำให้ได้ภาพขอบของภาพนั้นได้ดังแสดงในรูปที่ 3.3

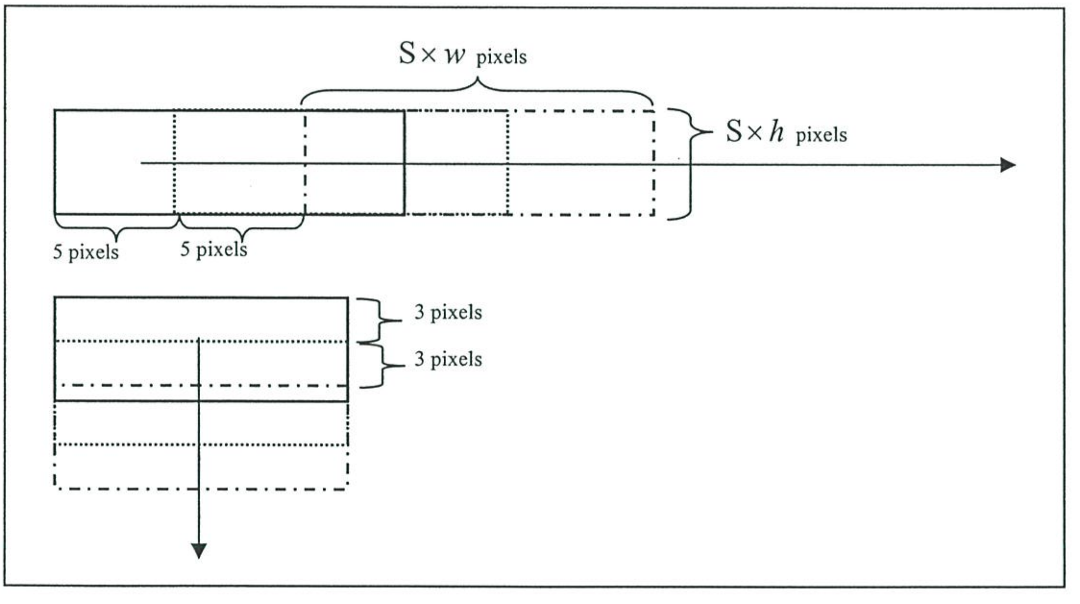
(ก)

(ข)

รูปที่ 3.3 ตัวอย่างที่ได้จากการตรวจหาขอบ (ก) ภาพต้นแบบ (ข) ภาพขอบ

3.2 การตรวจหาตำแหน่งของป้ายทะเบียน

จากสมมุติฐานที่ว่าบริเวณที่เป็นป้ายทะเบียนนั้นจะมีขอบอยู่ในปริมาณที่มากกว่าบริเวณอื่นของภาพ ดังนั้นขั้นตอนต่อไปนี้จะเป็นการหาการกระจายของขอบว่าบริเวณไหนที่มีขอบมากและเหมาะสมที่จะเป็นป้ายทะเบียน โดยการใช้วินโดว์ที่มีอัตราส่วนกว้างยาว(Aspect Ratio) ใกล้เคียงกับป้ายทะเบียนจำนวนสามขนาดสแกนไปตลอดทั้งภาพทั้งแนวตั้งและแนวนอนดังแสดงในรูปที่ 3.4 (อัตราส่วนของภาพอาจผิดไปจากความเป็นจริงเพื่อให้ง่ายต่อการทำความเข้าใจ) โดยใช้วินโดว์ขนาด $(Sw \times Sh)$ พิกเซล โดยที่ w คืออัตราส่วนความกว้างของวินโดว์ h คืออัตราส่วนความยาวของวินโดว์ และ S คือค่าสเกลลิง (Scaling Value) โดยการสแกนแต่ละครั้งจะขยับวินโดว์ครั้งละ 5 พิกเซลในแนวนอน และ 3 พิกเซลในแนวตั้ง แล้วทำการนับจำนวนของพิกเซลขอบที่อยู่ภายในวินโดว์แต่ละอัน เพื่อนำมาหาวินโดว์ที่มีพิกเซลขอบอยู่มากที่สุดสามอันดับแรก ซึ่งคาดว่าจะมีป้ายทะเบียนอยู่ภายในวินโดว์เหล่านั้น



รูปที่ 3.4 การสแกนภาพด้วยวินโดว์เพื่อคำนวณหาปริมาณของพิกเซลขอบที่อยู่ภายในวินโดว์แต่ละอัน

3.2.1 การหาค่าความเป็นไปได้ (Likelihood) ที่จะเป็นป้ายทะเบียน (L)

การตรวจหาตำแหน่งของป้ายทะเบียนในวิทยานิพนธ์ฉบับนี้ครอบคลุมป้ายทะเบียนของยานพาหนะส่วนใหญ่ที่ใช้ในปัจจุบัน ซึ่งสามารถแบ่งเป็นสองชนิดใหญ่ๆด้วยกัน ได้แก่ ป้ายทะเบียนชนิดบรรทัดเดียว และป้ายทะเบียนชนิดสองบรรทัด โดยป้ายทะเบียนชนิดบรรทัดเดียวคือป้ายทะเบียนที่มีตัวอักขระเพียงบรรทัดเดียว(ไม่รวมชื่อจังหวัด) เช่นป้ายทะเบียนรถยนต์ทั้งรุ่นเก่าและรุ่นใหม่ ป้ายทะเบียนรถบรรทุก ป้ายทะเบียนรถตู้ ป้ายทะเบียนรถโดยสารประจำทาง รวมถึง

ป้ายทะเบียนรถจักรยานยนต์รุ่นเก่าชนิดบรรทัดเดียวเป็นต้น ส่วนป้ายทะเบียนชนิดสองบรรทัด คือ ป้ายทะเบียนที่มีตัวอักขระแบ่งเป็นสองบรรทัด(ไม่รวมชื่อจังหวัด) ได้แก่ป้ายทะเบียนรถจักรยานยนต์รุ่นใหม่ และป้ายทะเบียนรถจักรยานยนต์รุ่นเก่าชนิดสองบรรทัด ซึ่งในการหาตำแหน่งของป้ายทะเบียนทั้งสองประเภทนี้ใช้หลักการเดียวกันแต่ต่างกันในรายละเอียด จึงแบ่งวิธีในการหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนออกเป็นสองกรณีด้วยกันคือ กรณีป้ายทะเบียนบรรทัดเดียวและกรณีป้ายทะเบียนสองบรรทัด

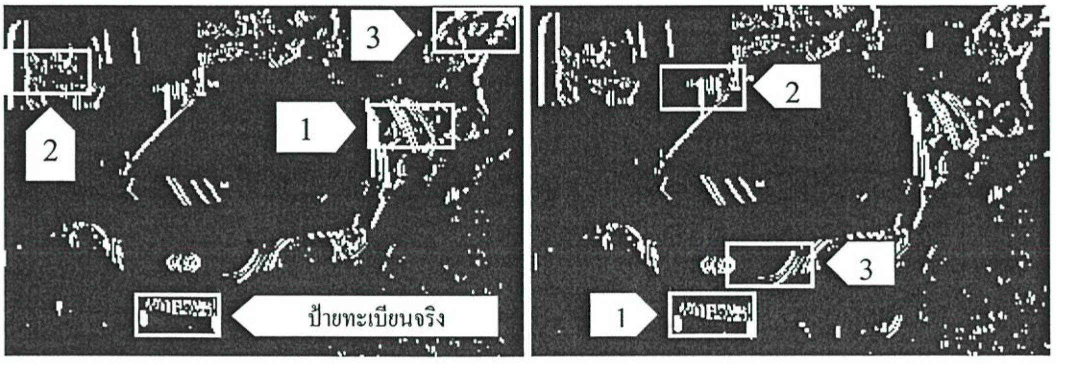
3.2.1.1 การหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนชนิดบรรทัดเดียว

จากการทดลองพบว่าปริมาณพิกเซลขอบที่มากเพียงอย่างเดียวไม่สามารถชี้ชัดได้ว่าเป็นป้ายทะเบียนเนื่องจากในกรณีที่ฉากหลัง (Background) หรือสภาพแวดล้อมของบริเวณที่ถ่ายภาพประกอบไปด้วยวัตถุหลากหลายชนิด รวมถึงสภาพของแสงขณะถ่ายภาพที่ไม่เหมาะสม อาจจะทำให้เกิดขอบจำนวนมากในภาพได้ ซึ่งจะทำให้การค้นหาตำแหน่งป้ายทะเบียนผิดพลาดได้ดังแสดงในรูปที่ 3.5 (ข) โดยจะสังเกต ได้ว่าวินโดว์ที่มีปริมาณพิกเซลขอบอยู่มากที่สุดสามอันดับแรกโดยเรียงตามลำดับหมายเลขในเครื่องหมายลูกศรนั้นไม่ใช่ป้ายทะเบียน และจะทำให้การค้นหาป้ายทำเบียนสำหรับภาพนี้ผิดพลาด

ดังนั้นวิทยานิพนธ์ฉบับนี้จึงเสนอวิธีในการแก้ปัญหานี้ โดยจากการสังเกตจะพบว่าพิกเซลขอบของป้ายทะเบียนจริงนั้นจะเรียงตัวกันอยู่ในบริเวณส่วนกลางของวินโดว์เป็นส่วนใหญ่ ขณะที่ส่วนบนและส่วนล่างจะมีพิกเซลขอบอยู่น้อย ในขณะที่ส่วนอื่นๆของภาพ พิกเซลขอบจะกระจายอยู่ทั่วไปทั้งวินโดว์ดังแสดงให้เห็นในรูปที่ 3.6 โดยในรูปที่ 3.6 (ก), (ข) และ (ค) นี้จะแสดงให้เห็นถึงโปรเจคชันฮิสโตแกรมในแนวนอน (Horizontal Projection Histogram) ของพิกเซลขอบในวินโดว์ของรูปที่ 3.5 (ข) ที่มีขอบมากที่สุดสามอันดับแรกซึ่งไม่ใช่ป้ายทะเบียนจริง เปรียบเทียบกับรูปที่ 3.6 (ง) ที่เป็นของป้ายทะเบียนจริงที่เราต้องการหา

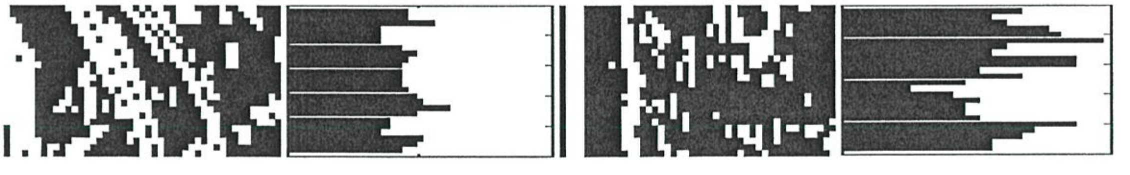
จากการสังเกตข้างต้นเราสามารถตั้งสมมติฐานได้ว่าวินโดว์ที่เป็นป้ายจะเบียนจะมีพิกเซลขอบน้อยในบริเวณส่วนบนและล่าง และมีพิกเซลขอบมากในบริเวณส่วนกลาง ดังนั้นเราสามารถหาแคนดิเดทวินโดว์ได้จากการหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน(L_C) จากสมการที่ (3.4) ซึ่งได้มาจากการแบ่งวินโดว์ออกเป็นสามส่วนโดยส่วนบน 25% ส่วนกลาง 50% และส่วนล่างที่เหลืออีก 25% ดังรูปที่ 3.7 แล้วคำนวณหาจำนวนพิกเซลขอบในส่วนกลางแล้วนำมาลบออกด้วยจำนวนพิกเซลขอบในส่วนบนและล่าง หลังจากคำนวณหาค่า L_C ได้แล้วให้เลือกวินโดว์ที่มีค่า L_C มาที่สุดสามอันดับแรกมาเป็นแคนดิเดทวินโดว์ และผลจากการหาแคนดิเดทวินโดว์ด้วยการหาค่า L_C นี้แสดงในรูปที่ 3.5 (ค) โดยเรียงลำดับค่ามากสุดไปน้อยสุดตามลำดับตัวเลขในเครื่องหมายลูกศร ซึ่งจะเห็นได้ว่าวิธีการนี้ทำให้ป้ายทะเบียนจริงถูกเลือกให้เป็นหนึ่งในสามของแคนดิเดทวินโดว์ได้อย่างถูกต้อง

(ก)

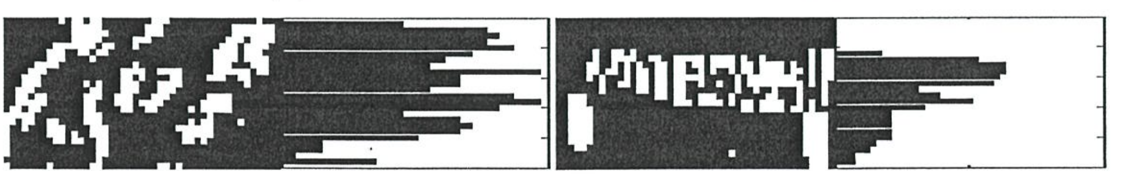


(ข) (ค)

รูปที่ 3.5 (ก) ภาพต้นแบบ (ข) แคนดิเดทวินโดว์ที่ได้จากการหาค่าขอบมากสุด (ค) แคนดิเดทวินโดว์ที่ได้จากการหาค่า L_C



(ก) (ข)



(ค) (ง)

รูปที่ 3.6 (ก), (ข), (ค) โปรเจคชันฮิสโตแกรมในแนวนอนของวินโดว์ที่มีขอบมากสุดสามอันดับซึ่งไม่ใช่ป้ายทะเบียนจริง (ง) โปรเจคชันฮิสโตแกรมในแนวนอนของวินโดว์ที่มีป้ายทะเบียน

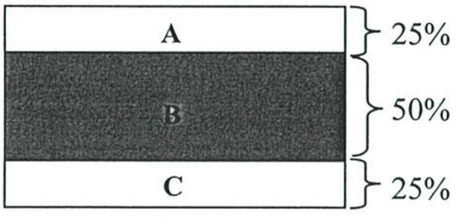
$$L_C = \sum_{c=1}^{n}\left[\sum_{r=\lfloor 0.25m\rfloor+1}^{\lceil 0.75m\rceil} E(r,c) - \sum_{r=1}^{\lfloor 0.25m\rfloor} E(r,c) - \sum_{r=\lceil 0.75m\rceil+1}^{m} E(r,c)\right] \tag{3.4}$$

โดยที่ L_C คือค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนชนิดบรรทัดเดียว

$E(r,c)$ คือค่าไบนารีของพิกเซลในตำแหน่งแถวที่ r หลักที่ c

m คือจำนวนแถวของวินโดว์ที่ใช้สแกน

n คือจำนวนหลักของวินโดว์ที่ใช้สแกน



รูปที่ 3.7 การแบ่งส่วนวินโดว์ที่ใช้ในการสแกนเพื่อหาค่า L_C

เพื่อความเข้าใจเราสามารถเขียนสมการที่ 3.4 ให้อยู่ในรูปแบบที่เข้าใจได้ง่ายดังสมการที่ 3.5 ดังนี้

$$L_C = \beta - \alpha - \chi \tag{3.5}$$

โดยที่ L_C คือค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนชนิดบรรทัดเดียว

β คือจำนวนพิกเซลขอบในพื้นที่ B

α คือจำนวนพิกเซลขอบในพื้นที่ A

χ คือจำนวนพิกเซลขอบในพื้นที่ C

3.2.1.2 การหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนชนิดสองบรรทัด

ในกรณีที่ต้องการตรวจหาป้ายทะเบียนรถจักรยานยนต์ซึ่งมีลักษณะและขนาดแตกต่างจากป้ายทะเบียนรถยนต์ก็สามารถใช้หลักการเดียวกันแต่มีรายละเอียดแตกต่างกัน เมื่อพิจารณาถึงป้ายทะเบียนรถจักรยานยนต์ในรูปที่ 3.8 (ก) และลักษณะการกระจายของพิกเซลขอบดังรูปที่ 3.8 (ข) แล้วจะพบว่า พิกเซลขอบจะมีอยู่มากอยู่บริเวณส่วนบนและล่างของวินโดว์ในขณะที่พิกเซลขอบจะมีอยู่น้อยในบริเวณส่วนกลางของวินโดว์ ซึ่งสามารถนำข้อสังเกตนี้มาสร้างเป็นสมการหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน (L_M) ได้โดยใช้หลักการเช่นเดียวกับกรณีของป้ายทะเบียนรถยนต์ แต่เนื่องจากในการทดลองพบว่าป้ายทะเบียนรถจักรยานยนต์มีขนาดเล็กกว่ารถยนต์ รวมถึงลักษณะการกระจายตัวของพิกเซลขอบของป้ายทะเบียนรถจักรยานยนต์มีความใกล้เคียงกับการกระจายตัว

ของพิกเซลขอบของวัตถุอื่นๆในสภาพแวดล้อมค่อนข้างมาก จึงจำเป็นต้องเพิ่มเอกลักษณ์อื่นๆ เพิ่มเติมเข้าไปในสมการในการหาค่า L_M ซึ่งจะทำให้สามารถแยกแยะป้ายทะเบียนรถจักรยานยนต์ออกจากสภาพแวดล้อมได้ จากการสังเกตเพิ่มเติมพบว่าพิกเซลขอบของป้ายทะเบียนรถจักรยานยนต์นั้นจะมีความสมมาตรในแนวแกนตั้งมากกว่าบริเวณอื่นของภาพ ซึ่งสามารถนำมาใช้ช่วยในการคำนวณหาค่า L_M ได้ ซึ่งทำให้สมการในการหาค่า L_M สำหรับรถจักรยานยนต์นั้นประกอบด้วยสองส่วนด้วยกันคือ ส่วนของค่าลักษณะการกระจายของพิกเซลขอบและส่วนของค่าความสมมาตรในแนวแกนตั้ง ดังแสดงในสมการที่ (3.6) โดยดูรูปที่ 3.8 (ค) ประกอบ

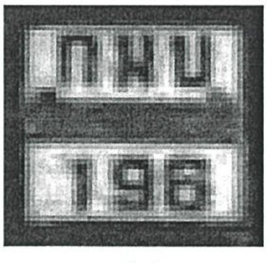
$$L_M = \underbrace{\left[(\alpha_1+\alpha_2)+(\chi_1+\chi_2)-(\beta_1+\beta_2)\right]}_{\text{ลักษณะการกระจาย}} - \underbrace{\left[|\alpha_1-\alpha_2|+|\beta_1-\beta_2|+|\chi_1-\chi_2|\right]}_{\text{ความสมมาตร}} \quad (3.6)$$

โดยที่ L_M คือค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนชนิดสองบรรทัด

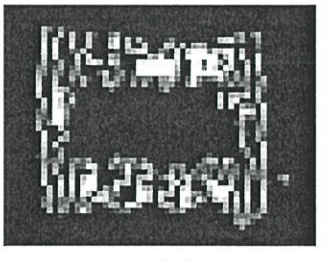
α_1, α_2 คือจำนวนพิกเซลขอบในพื้นที่ A1 และ A2 ตามลำดับ

β_1, β_2 คือจำนวนพิกเซลขอบในพื้นที่ B1 และ B2 ตามลำดับ

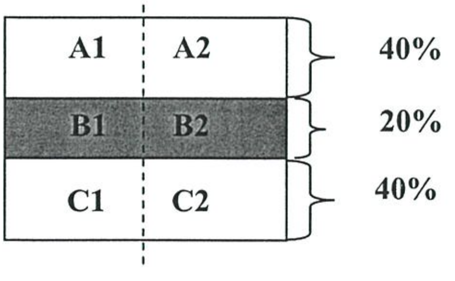
χ_1, χ_2 คือจำนวนพิกเซลขอบในพื้นที่ C1 และ C2 ตามลำดับ



(ก)



(ข)



(ค)

รูปที่ 3.8 (ก) ป้ายทะเบียนรถจักยานยนต์ (ข) ลักษณะการกระจายของพิกเซลขอบของป้ายทะเบียนรถจักรยานยนต์ (ค) การแบ่งพื้นที่ของวินโดว์ที่ใช้ในการคำนวณห่าค่า L_M

3.2.1.3 การนอร์มอลไลซ์ (Normalization) ค่า L_C และ L_M

เนื่องจากการหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนทั้งป้ายทะเบียนชนิดบรรทัดเดียวและป้ายทะเบียนชนิดสองบรรทัดนั้น กระทำบนวินโดว์ที่มีการสแกนไปตลอดทั้งภาพซึ่งมีขนาดแตกต่างกันตามค่าสเกลลิง (S) ดังนั้นค่า L_C และ L_M ที่ได้มานั้น จะต้องมีการทำนอร์มอลไลซ์ก่อนที่จะนำมาเปรียบเทียบกันเพื่อตัดผลกระทบที่เกิดขึ้นจากขนาดของวินโดว์ที่ใช้ในการสแกน โดยในการทำนอร์มอลไลซ์นี้จะทำโดยการนำค่า L_C และ L_M ที่คำนวนได้มาหารด้วยจำนวนพิกเซลทั้งหมดในวินโดว์ที่ใช้ในการสแกนหรือก็คือพื้นที่ของวินโดว์ที่ใช้ในการสแกนนั่นเองดังสมการที่ (3.7) หรือ (3.8) แล้วแต่กรณีว่าเป็นป้ายทะเบียนชนิดบรรทัดเดียวหรือสองบรรทัด

$$L = \frac{L_C}{W \times H} \tag{3.7}$$

หรือ

$$L = \frac{L_M}{W \times H} \tag{3.8}$$

โดยที่ L คือค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนที่นอร์มอลไลซ์แล้ว

L_C คือค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนชนิดบรรทัดเดียว

L_M คือค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนชนิดสองบรรทัด

W คือความกว้างของวินโดว์ที่ใช้สแกนในหน่วยพิกเซล

H คือความสูงของวินโดว์ที่ใช้สแกนในหน่วยพิกเซล

ซึ่งค่า L ที่ได้นี้จะนำไปเป็นส่วนประกอบหลักในการพิจารณาหาแคนดิเดทวินโดว์ในขั้นตอนต่อๆไป

3.2.2 การหาความหนาแน่นสัมพัทธ์ของพิกเซลขอบ

อย่างไรก็ตาม การหาแคนดิเดทวินโดว์จากค่า L เพียงอย่างเดียวนั้นยังไม่เพียงพอ เนื่องจากหากพิจารณารูปที่ 3.7 ประกอบ ในบางกรณีที่ถึงแม้ว่าบริเวณส่วนบนและล่าง (พื้นที่ A และ C) จะมีพิกเซลขอบเป็นจำนวนมากซึ่งน่าจะทำให้บริเวณนั้นไม่ควรเป็นป้ายทะเบียน แต่หากบริเวณตอนกลางของวินโดว์ (พื้นที่ B) มีพิกเซลขอบในปริมาณสูงมากเช่นเดียวกัน ก็อาจจะทำให้ค่า L ที่คำนวณได้จากสมการที่ (3.5) มีค่ามากด้วยเช่นกัน ซึ่งวินโดว์นี้อาจจะถูกนำไปพิจารณาเป็นแคนดิเดทวินโดว์และทำให้การตรวจหาตำแหน่งของป้ายทะเบียนผิดพลาดได้ ดังตัวอย่างในรูปที่ 3.9 (ข) ซึ่งแสดงถึงแคนดิเดทวินโดว์ที่ได้จากการหาวินโดว์ที่มีค่า L มากที่สุดสามอันดับแรกตามลำดับลูกศรในภาพ ซึ่งแสดงให้เห็นว่าแคนดิเดทวินโดว์ทั้งสามไม่ครอบคลุมถึงป้ายทะเบียนที่แท้จริง โดยเมื่อพิจารณารูปที่ 3.10 ซึ่งเป็นโปรเจคชันฮิสโตแกรมในแนวนอนของวินโดว์ลำดับที่ 1 และ 3 รวมถึง

ป้ายทะเบียนที่แท้จริงของรูปที่ 3.9 (ข) จะพบว่าความผิดพลาดที่เกิดขึ้นเกิดจากการที่วินโดว์มีพิกเซลขอบอยู่บริเวณตอนกลางเป็นจำนวนมากจนกระทั่ง จำนวนพิกเซลขอบที่บริเวณส่วนบนและล่างของวินโดว์ไม่มากพอที่จะทำให้ค่า L ลดลงได้มากเท่าที่ควร จนเป็นเหตุให้วินโดว์เหล่านี้ถูกคัดเลือกให้เป็นแคนดิเดทวินโดว์

เพื่อแก้ปัญหานี้ จึงใช้วิธีการหาความหนาแน่นสัมพัทธ์ของพิกเซลขอบเข้ามาพิจารณาร่วมกับค่า L เพื่อใช้ในการหาแคนดิเดทวินโดว์ ซึ่งการหาความหนาแน่นสัมพัทธ์ของพิกเซลขอบนั้นแบ่งได้สองกรณีเช่นเดียวกันกับการหาค่า L กล่าวคือกรณีของป้ายทะเบียนชนิดบรรทัดเดียวและกรณีป้ายทะเบียนชนิดสองบรรทัด

3.2.2.1 การหาการหาความหนาแน่นสัมพัทธ์ของพิกเซลขอบในกรณีของป้ายทะเบียนชนิดบรรทัดเดียว

เป็นการเปรียบเทียบจำนวนพิกเซลขอบในบริเวณส่วนบนและล่างของวินโดว์เทียบกับจำนวนพิกเซลขอบในบริเวณส่วนกลางของวินโดว์ดังสมการที่ (3.6) และ (3.7) เนื่องจากป้ายทะเบียนชนิดบรรทัดเดียวนั้นเราตั้งสมมติฐานว่าจำนวนพิกเซลขอบในบริเวณส่วนบนและล่างของวินโดว์จะน้อยกว่าบริเวณส่วนกลาง

$$R_{AB} = \frac{\alpha}{\beta} \tag{3.9}$$

$$R_{CB} = \frac{\chi}{\beta} \tag{3.10}$$

โดยที่ R_{AB} คือความหนาแน่นสัมพัทธ์ของพิกเซลขอบของพื้นที่ A เทียบกับ B

R_{CB} คือความหนาแน่นสัมพัทธ์ของพิกเซลขอบของพื้นที่ C เทียบกับ B

α คือจำนวนพิกเซลขอบในพื้นที่ A

β คือจำนวนพิกเซลขอบในพื้นที่ B

χ คือจำนวนพิกเซลขอบในพื้นที่ C

3.2.2.2 การหาการหาความหนาแน่นสัมพัทธ์ของพิกเซลขอบในกรณีของป้ายทะเบียนชนิดสองบรรทัด

สำหรับกรณีของป้ายทะเบียนรถจักรยานยนต์หรือป้ายทะเบียนชนิดสองบรรทัดจะเป็นการเปรียบเทียบจำนวนพิกเซลขอบในบริเวณกลางของวินโดว์เทียบกับจำนวนพิกเซลขอบในบริเวณส่วนส่วนบนและล่างของวินโดว์ดังสมการที่ (3.8) และ (3.9) รวมถึงการเปรียบเทียบระหว่างจำนวนพิกเซลขอบในบริเวณส่วนบนและล่างของวินโดว์ดังสมการที่ (3.10) เพิ่มเติมด้วย เนื่องจากป้ายทะเบียนชนิดสองบรรทัดนั้นเราตั้งสมมติฐานว่าจำนวนพิกเซลขอบในบริเวณส่วนกลางของ

วินโดว์จะน้อยกว่าบริเวณส่วนส่วนบนและล่าง รวมถึงพิกเซลขอบในบริเวณส่วนบนและส่วนล่างนั้นมีความใกล้เคียงกัน

$$R_{BA} = \frac{\beta}{\alpha} \tag{3.11}$$

$$R_{BC} = \frac{\beta}{\chi} \tag{3.12}$$

$$R_{CA} = \frac{\chi}{\alpha} \tag{3.13}$$

โดยที่ R_{BA} คือความหนาแน่นสัมพัทธ์ของพิกเซลขอบของพื้นที่ B เทียบกับ A

R_{BC} คือความหนาแน่นสัมพัทธ์ของพิกเซลขอบของพื้นที่ B เทียบกับ C

R_{CA} คือความหนาแน่นสัมพัทธ์ของพิกเซลขอบของพื้นที่ C เทียบกับ A

α คือจำนวนพิกเซลขอบในพื้นที่ A

β คือจำนวนพิกเซลขอบในพื้นที่ B

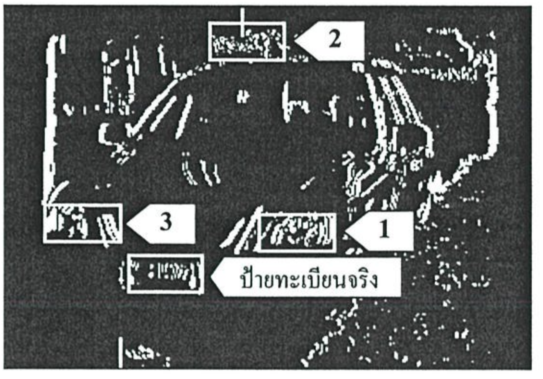
χ คือจำนวนพิกเซลขอบในพื้นที่ C

หลังจากที่แต่ละวินโดว์ถูกนำมาคำนวณหาค่าความหนาแน่นสัมพัทธ์ของพิกเซลขอบ R_{AB}, R_{CB} หรือ R_{BA}, R_{BC}, R_{CA} ตามแต่ละชนิดของป้ายทะเบียนแล้ว ค่าที่คำนวณได้จะถูกนำมาพิจารณาว่าอยู่ในช่วงของค่าเริ่มเปลี่ยน $\tau_{AB}, \tau_{CB}, \tau_{BA}, \tau_{BC}, \tau_{CA}$ ตามแต่ละประเภทของค่า R หรือไม่ หากอยู่ในช่วงค่าเริ่มเปลี่ยนที่กำหนด จึงจะนำวินโดว์นั้นมาคำนวณหาค่า L แต่หากไม่อยู่ในช่วงค่าเริ่มเปลี่ยนที่กำหนด วินโดว์นั้นจะไม่ถูกนำมาคำนวณหาค่า L ซึ่งจะมีผลทำให้วินโดว์นั้นไม่ถูกนำมาพิจารณาเป็นแคนดิเดทวินโดว์ จากนั้นวินโดว์ที่มีค่า L มากที่สุดสามอันดับแรกจะถูกคัดเลือกให้เป็นแคนดิเดทวินโดว์

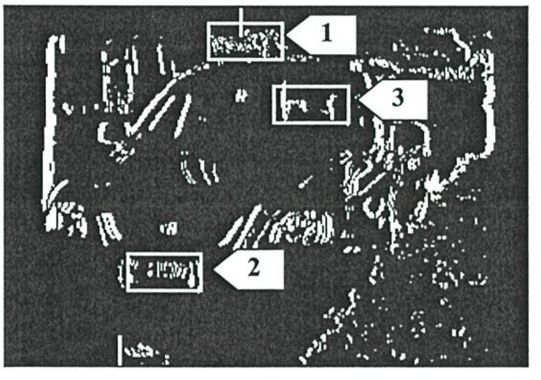
3.2.3 การแก้ไขการทับซ้อนกันของแคนดิเดทวินโดว์

หลังจากที่เราได้วินโดว์ที่มีค่า L สูงสุดสามอันดับแรกแล้ว โดยส่วนใหญ่วินโดว์ทั้งหมดที่ได้มักจะเกิดการซ้อนทับกัน ดังในรูปที่ 3.11 (ข) อันเนื่องมาจากแต่ละวินโดว์ที่ทำการสแกนไปนั้นจะขยับไปทีละน้อยซึ่งทำให้ลักษณะของพิกเซลขอบในแต่ละวินโดว์ที่อยู่ติดกันมีความใกล้เคียงกัน ดังนั้นค่า L ที่คำนวณได้ก็มักจะใกล้เคียงกันด้วย ดังนั้นในกรณีที่แคนดิเดทวินโดว์ที่เลือกมานั้นซ้อนทับกัน ให้เลือกเอาเฉพาะวินโดว์ที่มีค่า L มากที่สุดเป็นแคนดิเดทวินโดว์ ส่วนวินโดว์ที่ซ้อนกันที่เหลือให้ตัดทิ้งแล้วให้เลือกวินโดว์ที่มีค่า L รองลงให้เป็นแคนดิเดทวินโดว์มาจนกว่าจะครบสามลำดับ เนื่องจากเราถือว่าแคนดิเดทวินโดว์ที่ซ้อนทับกันนั้นคือวัตถุชิ้นเดียวกันนั่นเอง ดังแสดงในรูปที่ 3.11 (ค)

(ก)



(ข)

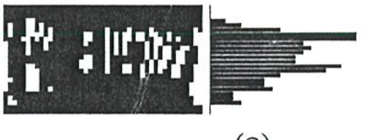


(ค)

รูปที่ 3.9 (ก) ภาพต้นแบบ (ข) แคนดิเดทวินโดว์ที่ได้จากการพิจารณาค่า L เพียงอย่างเดียว (ค) แคนดิเดทวินโดว์ที่ได้จากการพิจารณาค่า L ร่วมกับค่า R_{AB} และ R_{CB}

(ก)

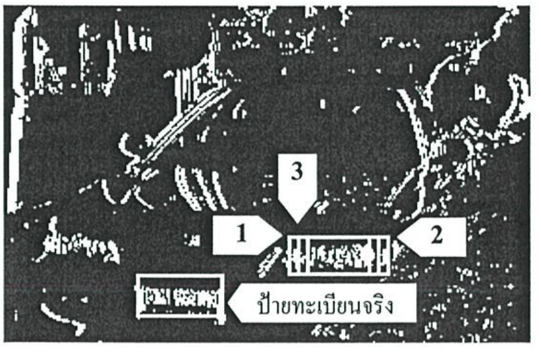
(ข)



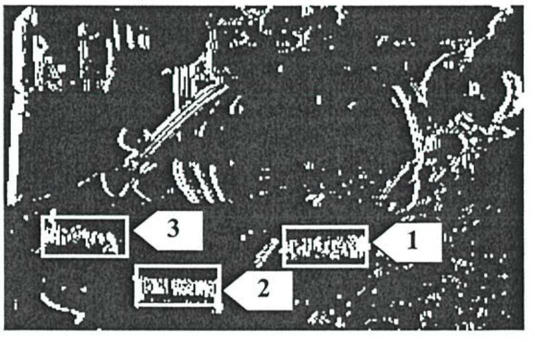
(ค)

รูปที่ 3.10 (ก), (ข) โปรเจคชันฮิสโตแกรมในแนวนอนของวินโดว์ที่ไม่ใช่ป้ายทะเบียน (ค) โปรเจคชันฮิสโตแกรมในแนวนอนของวินโดว์ที่เป็นป้ายทะเบียนจริง

(ก)



(ข)



(ค)

รูปที่ 3.11 (ก) ภาพต้นแบบ (ข) การซ้อนทับกันของวินโดว์ที่มีค่า L มากที่สุดสามลำดับแรก (ค) แคนดิเดทวินโดว์ที่ได้ หลังจากการแก้ไขการซ้อนทับกันเรียบร้อยแล้ว

3.3 การลดค่าฟอลส์โพสซิทีฟ (False Positive Suppression)

จากขั้นตอนที่แล้วเราจะได้แคนดิเดทวินโดว์จำนวนสามวินโดว์ ซึ่งจะมีทั้งวินโดว์ที่เป็นป้ายทะเบียนที่แท้จริง และวินโดว์ที่ไม่ใช่ป้ายทะเบียนที่แท้จริงซึ่งเรียกว่าฟอลส์โพสซิทีฟติดมาด้วย โดยฟอลส์โพสซิทีฟที่เกิดขึ้นนี้เป็นส่วนที่เราไม่ต้องการ ดังนั้นในขั้นตอนนี้จะเป็นส่วนของกระบวนการลดค่าฟอลส์โพสซิทีฟให้ลดลงด้วยการคัดกรองแคนดิเดทวินโดว์ที่ได้มาอีกรอบหนึ่ง โดยการแยกแคนดิเดทวินโดว์ที่ไม่ใช่ป้ายทะเบียนออกให้ได้มากที่สุดและพยายามคงเหลือเฉพาะแคนดิเดทวินโดว์ให้น้อยที่สุดที่ป้ายทะเบียนจริงเท่านั้น ในวิทยานิพนธ์ฉบับนี้ได้เสนอขั้นตอนวิธีการลดค่าฟอลส์โฟสซิทีฟนี้ไว้สองขั้นตอนด้วยกันได้แก่ ขั้นตอนการวิเคราะห์ข้อมูลเชิงเวลา

(Temporal Analysis) และ ขั้นตอนการวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน (L) โดยในการทดลองนั้นได้ทดลองทั้งแบบ ผ่านขั้นตอนการวิเคราะห์ข้อมูลเชิงเวลาเพียงอย่างเดียว และแบบผ่านทั้งขั้นตอนการวิเคราะห์ข้อมูลเชิงเวลาร่วมกับขั้นตอนการวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน ซึ่งจะมีการเรียบเทียบถึงผลที่ได้จากการทดลองทั้งสองแบบนี้โดยจะกล่าวถึงรายละเอียดไว้ในบทที่ 4

3.3.1 การวิเคราะห์ข้อมูลเชิงเวลา

วิธีนี้จะเป็นการหาข้อมูลเพิ่มเติมเพื่อคัดแยกแคนดิเดทวินโดว์ว่าควรเป็นป้ายทะเบียนหรือไม่ เนื่องจากภาพที่เราใช้ในการหาตำแหน่งป้ายทะเบียนป้ายทะเบียนนั้นมาจากลำดับภาพวีดิทัศน์ ซึ่งมีข้อมูลในเชิงเวลาที่เพิ่มขึ้นมาจากภาพนิ่ง ทำให้เราสามารถใช้ข้อมูลนี้ในการตัดสินใจว่าแคนดิเดทวินโดว์ใดควรหรือไม่ควรเป็นป้ายทะเบียนได้

3.3.1.1 การจับคู่วินโดว์ที่เกี่ยวข้องกัน

จากสมมติฐานที่ว่ารถยนต์ในช่วงเวลาที่ถูกบันทึกภาพนั้นย่อมเคลื่อนที่เป็นเส้นตรงและมีความเร็ว(ไม่อยู่นิ่ง) และความเร็วนั้นคงที่ ซึ่งหมายถึงมีความเร็ว (Velocity) ไม่เป็นศูนย์ ในขณะที่ความเร่ง (Acceleration) เป็นศูนย์ดังนั้นแคนดิเดทวินโดว์ใดที่มีความเร็วและความเร่งไม่ตรงตามเงื่อนไขดังกล่าวจะถือว่าไม่เป็นป้ายทะเบียนซึ่งต้องถูกคัดออกไป แต่ก่อนที่เราจะหาความเร็วและความเร่งของแต่ละแคนดิเดทวินโดว์ได้นั้นเราต้องทราบเส้นทางการเคลื่อนที่ของวินโดว์นั้นๆ เสียก่อน โดยการหาแคนดิเดทวินโดว์จากหลายๆลำดับภาพที่ติดกัน แล้วพิจารณาว่าแคนดิเดทวินโดว์ใดในแต่ละลำดับภาพควรเป็นวัตถุชิ้นเดียวกันหรือเกี่ยวข้องกันกับแคนดิเดทวินโดว์ใดในลำดับภาพอื่นๆ โดยพิจารณาจากความแตกต่างที่ประกอบด้วยสองส่วนคือ ความแตกต่างกันทางตำแหน่งและความแตกต่างกันทางการกระจายของพิกเซลขอบ กล่าวคือแคนดิเดทวินโดว์ที่เกี่ยวข้องกันนั้นควรจะอยู่ใกล้กันและมีการกระจายของพิกเซลขอบใกล้เคียงกัน

ขั้นตอนแรกเริ่มจากการหาแคนดิเดทวินโดว์จากทุกลำดับภาพที่ติดกันสำหรับในวิทยานิพนธ์ฉบับนี้ใช้จำนวน 4 ลำดับภาพโดยแต่ละลำดับภาพจะประกอบไปด้วย 3 แคนเดเดทวินโดว์ จากนั้นเริ่มต้นที่ลำดับภาพที่เป็นปัจจุบัน (Current Frame) ที่สุดแล้วจับคู่กับลำดับภาพที่ติดกันก่อนหน้า (Previous Frame) โดยคำนวณหาคะแนนความแตกต่างกัน ($score$) ของทุกคู่ของแคนดิเดทวินโดว์จากทั้งสองลำกับภาพดังสมการที่ (3.14) จากนั้นเลือกคู่ของแคนดิเดท 3 คู่ที่ให้ค่า $score$ ต่ำสุด ซึ่งจะเป็นคู่ของวินโดว์ที่เกี่ยวข้องกัน จากนั้นพิจารณาคู่ของแคนดิเดทวินโดว์จากลำดับภาพถัดไปจนกระทั่งครบทั้ง 4 ลำดับภาพ ซึ่งท้ายที่สุดจะได้กลุ่มของวินโดว์ที่เกี่ยวข้องกัน 3 กลุ่ม ซึ่งจะถูกใช้อ้างอิงในการหาความเร็วและความเร่งของแคนดิเดทวินโดว์ทั้ง 3 ในลำดับต่อไป

$$score = \left[\frac{\sum_{i=1}^{m}\left|H_i^C - H_i^P\right|}{m \times n}\right] + \left[\frac{(x_C - x_P)^2 + (y_C - y_P)^2}{(M-m)^2 + (N-n)^2}\right] \tag{3.14}$$

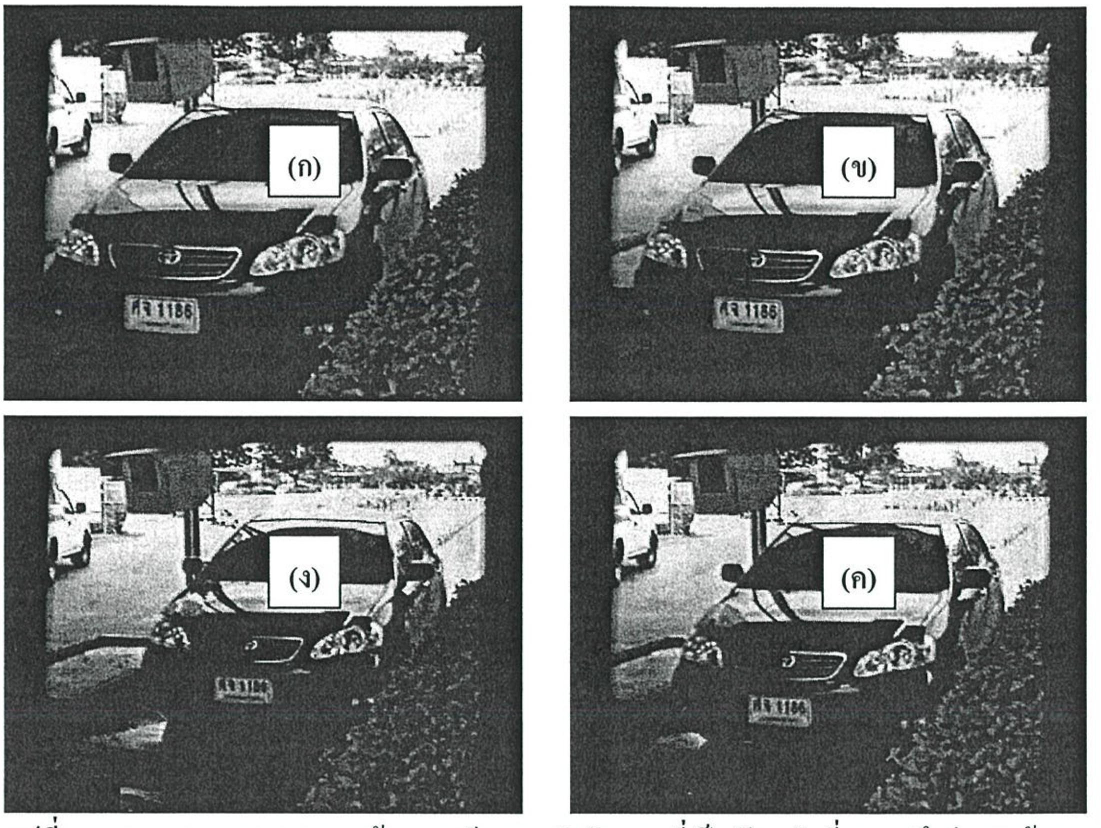
ซึ่ง

$$H_i^C = \sum_{j=1}^{n} E^C(i,j) \tag{3.15}$$

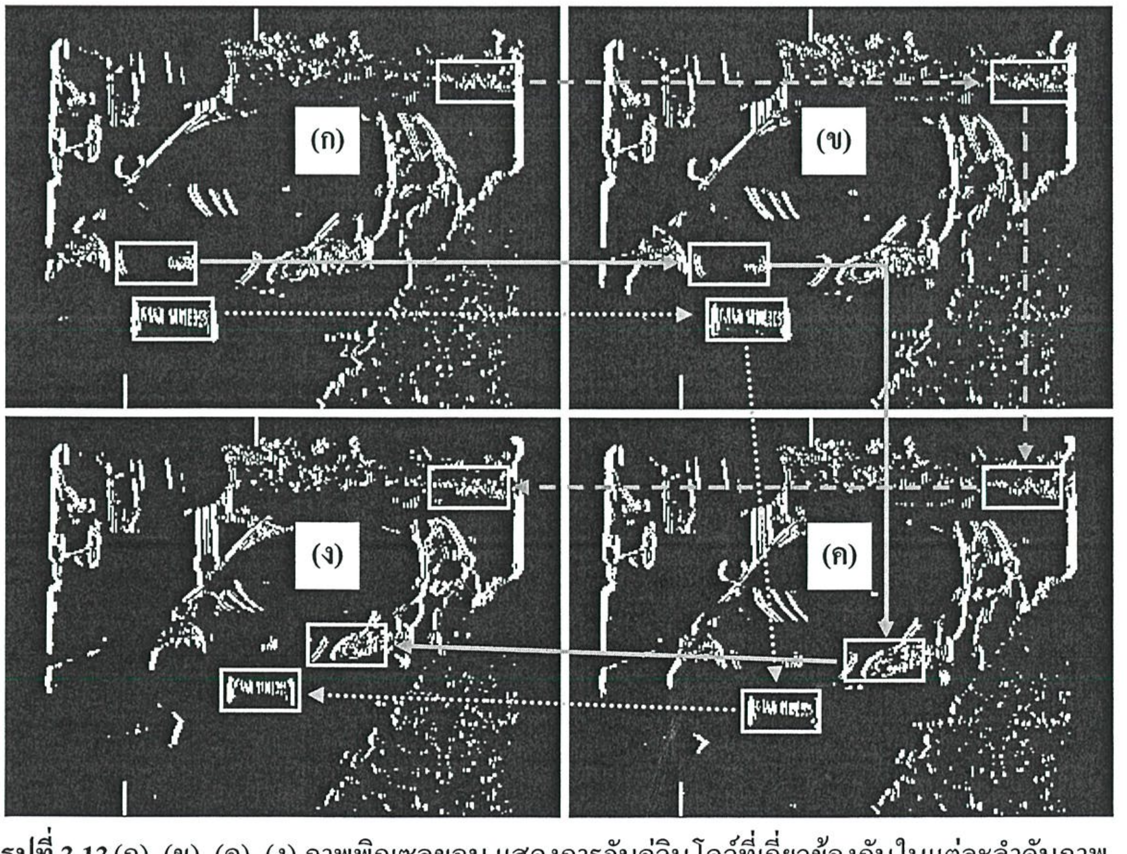
$$H_i^P = \sum_{j=1}^{n} E^P(i,j) \tag{3.16}$$

โดยที่	E	คือค่าไบนารีของพิเซลที่แสดงถึงความเป็นขอบจากสมการที่ (3.3)
	i	คือพิกัดแถวของพิกเซลในแคนดิเดทวินโดว์
	j	คือพิกัดหลักของพิกเซลในแคนดิเดทวินโดว์
	x	คือพิกัดของหลักของพิกเซลมุมบนซ้ายสุดของแคนดิเดทวินโดว์
	y	คือพิกัดของแถวของพิกเซลมุมบนซ้ายสุดของแคนดิเดทวินโดว์
	m	คือจำนวนแถวของแคนดิเดทวินโดว์
	n	คือจำนวนหลักของแคนดิเดทวินโดว์
	M	คือจำนวนแถวของลำดับภาพ
	N	คือจำนวนหลักของลำดับภาพ
	C	คือเครื่องหมายบอกว่าเป็นลำดับภาพปัจจุบัน (Current Frame)
	P	คือเครื่องหมายบอกว่าเป็นลำดับภาพก่อนหน้า (Previous Frame)

เมื่อพิจารณาที่สมการที่ (3.15) และ (3.16) แล้ว H_i^C และ H_i^P ก็คือผลรวมของพิกเซลขอบในแถวที่ i ของแคนดิเดทวินโดว์ในลำดับภาพปัจจุบันและก่อนหน้าตามลำดับ และสำหรับสมการที่ (3.14) นั้นสามารถแบ่งได้เป็นสองส่วนในแต่ละเครื่องหมายวงเล็บ โดยในส่วนแรกจะเป็นส่วนที่บ่งบอกถึงความแตกต่างกันของการกระจายของพิกเซลขอบ โดยการหาความแตกต่างกันของจำนวนพิกเซลขอบในแต่ละแถวแล้วจึงนำความแตกต่างของทุกแถวมารวมกัน และส่วนหลังจะเป็นส่วนที่บ่งบอกถึงความแตกต่างกันทางตำแหน่ง โดยการหาระยะทางระหว่างตำแหน่งมุมบนซ้ายสุดของทั้งสองวินโดว์ และจะสังเกตเห็นว่าในสมการที่ (3.14) จะตัดการคำนวณหารากที่สองออกไปเพื่อลดเวลาในการคำนวณ โดยที่เราต้องทำการนอร์มอลไลซ์ (Normalize) ทั้งสองส่วนของสมการก่อนจะนำมารวมกันด้วยการหารแต่ละส่วนด้วยค่ามากที่สุดที่สามารถเป็นไปได้ของส่วนนั้นๆ ผลที่ได้จากการกระทำในขั้นตอนนี้กับภาพต้นแบบในรูปที่ 3.12 แสดงดังรูปที่ 3.13



รูปที่ 3.12 (ก), (ข), (ค), (ง) ภาพต้นแบบเรียงจากลำดับภาพที่เป็นปัจจุบันที่สุดจนถึงก่อนหน้าสุด

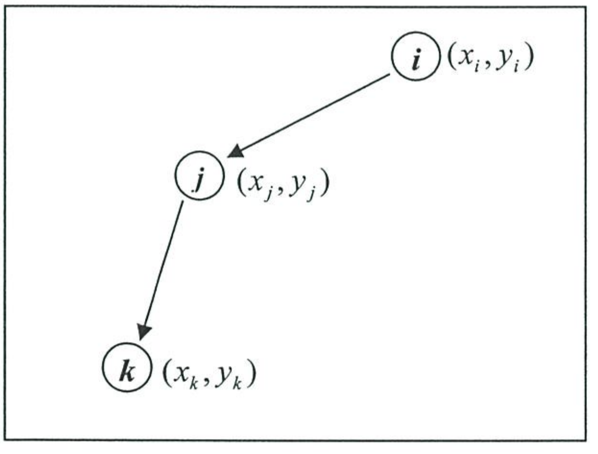


รูปที่ 3.13 (ก), (ข), (ค), (ง) ภาพพิกเซลขอบ แสดงการจับคู่วินโดว์ที่เกี่ยวข้องกันในแต่ละลำดับภาพเรียงจากลำดับภาพที่เป็นปัจจุบันที่สุดจนถึงก่อนหน้าสุด

ในรูปที่ 3.13 แสดงภาพขอบของลำดับภาพตั้งแต่ปัจจุบันที่สุดไปจนถึงก่อนหน้าสุด เรียงลำดับตั้งแต่ (ก), (ข), (ค) และ (ง) ตามลำดับ ซึ่งหลังจากผ่านกระบวนการจับคู่วินโดว์ที่เกี่ยวข้องกันระหว่างลำดับภาพปัจจุบันและลำดับภาพก่อนหน้าที่ได้แก่ คู่ของรูปที่ 3.13 (ก) และ (ข), คู่ของรูปที่ 13.13 (ข) และ (ค), คู่ของรูปที่ 13.13(ค) และ (ง) แล้ว เราจะได้กลุ่มของแคนดิเดทวินโดว์ที่เกี่ยวข้องกัน 3 กลุ่มแยกตามเครื่องหมายลูกศรลักษณะต่างๆ

3.3.1.2 การวิเคราะห์ความเร็วและความเร่งของแคนดิเดทวินโดว์

ในขั้นตอนนี้จะเป็นการคัดแยกแคนดิเดทวินโดว์ที่ไม่ใช่ป้ายทะเบียนออกไป ดังที่ได้กล่าวมาแล้วว่าเมื่อเวลาผ่านไปป้ายทะเบียนนั้นย่อมเคลื่อนที่ และเคลื่อนที่ด้วยความเร็วคงที่ซึ่งหมายความว่าป้ายทะเบียนต้องมีความเร็วไม่เป็นศูนย์และความเร่งเท่ากับศูนย์ ดังนั้นในขั้นตอนนี้เราจะใช้สมมติฐานนี้ในการคัดแยกแคนดิเดทวินโดว์ที่ไม่ใช่ป้ายทะเบียนออกไป โดยขั้นแรกเราต้องหาความเร็วและความเร่งของแคนดิเดทวินโดว์ให้ได้เสียก่อนโดยพิจารณารูปที่ 3.14



รูปที่ 3.14 การเคลื่อนที่แนวเส้นตรงของวัตถุ

เมื่อวัตถุเคลื่อนที่จากจุด i ไปยังจุด j และจุด k ซึ่งมีพิกัดตำแหน่งดังรูป ซึ่งความเร็วคือระยะทางต่อเวลา แต่เนื่องจากเวลาระหว่างลำดับภาพเท่ากันดังนั้นเราสามารถละทิ้งค่าเวลาในการคำนวณหาความเร็วได้ดังนั้นความเร็วจากจุด i ไปยังจุด j ก็คือระยะทางจากจุด i ไปยังจุด j และในกรณีความเร็วจากจุด j ไปยังจุด k ก็เช่นเดียวกัน ซึ่งได้เวคเตอร์ความเร็วดังนี้

$$\mathbf{v_{ij}} = (\mathbf{x_j} - \mathbf{x_i}) + (\mathbf{y_j} - \mathbf{y_i}) \tag{3.17}$$

$$\mathbf{v_{jk}} = (\mathbf{x_k} - \mathbf{x_j}) + (\mathbf{y_k} - \mathbf{y_j}) \tag{3.18}$$

หาขนาดของเวคเตอร์จะได้

$$v_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \tag{3.19}$$

$$v_{jk} = \sqrt{(x_k - x_j)^2 + (y_k - y_j)^2} \tag{3.20}$$

เพื่อลดเวลาในการคำนวณเราสามารถละทิ้งรากที่สองได้ดังนั้นจะได้

$$v_{ij} = (x_j - x_i)^2 + (y_j - y_i)^2 \tag{3.21}$$

$$v_{jk} = (x_k - x_j)^2 + (y_k - y_j)^2 \tag{3.22}$$

โดยที่ v_{ij} คือความเร็วจากจุด i ไปยังจุด j

v_{jk} คือความเร็วจากจุด j ไปยังจุด k

x คือพิกัดตำแหน่งของจุดในแนวแถว ในรูปที่ 3.14

y คือพิกัดตำแหน่งของจุดในแนวหลัก ในรูปที่ 3.14

ในขณะที่ความเร่งคืออัตราการเปลี่ยนแปลงความเร็วต่อเวลาซึ่งเวลาในแต่ละลำดับภาพมีค่าเท่ากันดังนั้นเราสามารถละทิ้งค่าเวลาในการคำนวณได้ ซึ่งจะได้เวคเตอร์ความเร่งดังนี้

$$\mathbf{a}_{\mathbf{ik}} = \mathbf{v}_{\mathbf{jk}} - \mathbf{v}_{\mathbf{ij}} \tag{3.23}$$

แทนค่าสมการที่ (3.17) และ (3.18) ลงในสมการที่ (3.23) จะได้

$$\mathbf{a}_{\mathbf{ik}} = [(\mathbf{x}_{\mathbf{k}} - \mathbf{x}_{\mathbf{j}}) - (\mathbf{x}_{\mathbf{j}} - \mathbf{x}_{\mathbf{i}})] + [(\mathbf{y}_{\mathbf{k}} - \mathbf{y}_{\mathbf{j}}) - (\mathbf{y}_{\mathbf{j}} - \mathbf{y}_{\mathbf{i}})] \tag{3.24}$$

$$\mathbf{a}_{\mathbf{ik}} = (\mathbf{x}_{\mathbf{k}} - 2\mathbf{x}_{\mathbf{j}} + \mathbf{x}_{\mathbf{i}}) + (\mathbf{y}_{\mathbf{k}} - 2\mathbf{y}_{\mathbf{j}} + \mathbf{y}_{\mathbf{i}}) \tag{3.25}$$

หาขนาดของเวคเตอร์จะได้

$$a_{ik} = \sqrt{(x_k - 2x_j + x_i)^2 + (y_k - 2y_j + y_i)^2} \tag{3.26}$$

และเพื่อลดเวลาในการคำนวณเราสามารถละทิ้งรากที่สองได้ดังนั้นจะได้

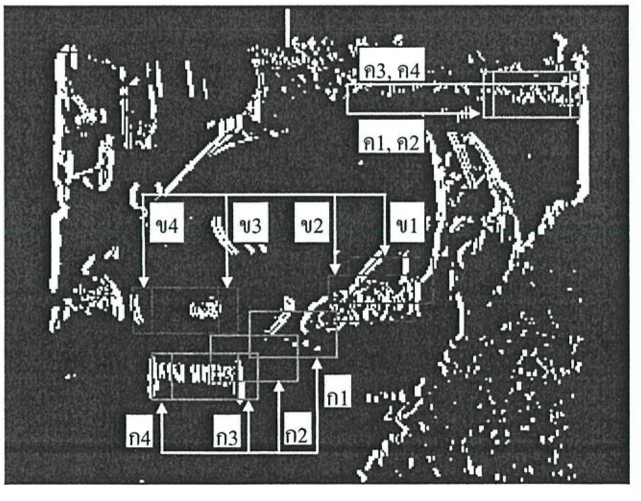
$$a_{ik} = (x_k - 2x_j + x_i)^2 + (y_k - 2y_j + y_i)^2 \tag{3.27}$$

โดยที่ a_{ik} คือความเร่งจากจุด i ไปยังจุด k

x คือพิกัดตำแหน่งของจุดในแนวแถว ในรูปที่ 3.14

y คือพิกัดตำแหน่งของจุดในแนวหลัก ในรูปที่ 3.14

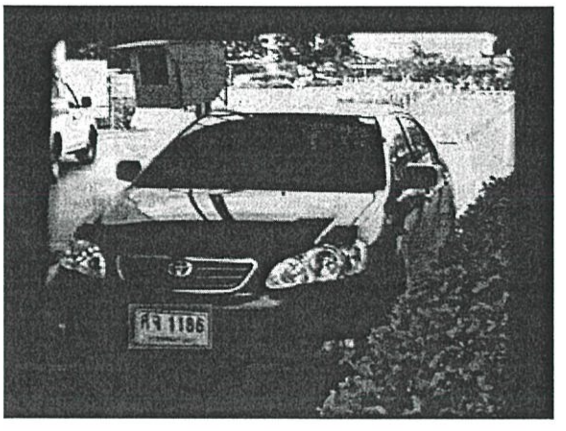
ในการหาความเร็วและความเร่งของแคนดิเดทวินโดว์ทำได้โดยการแทนค่าพิกัดตำแหน่งมุมบนซ้ายสุดของแคนดิเดทวินโดว์ลงในสมการที่ (3.21), (3.22) และ (3.27) โดยกำหนดให้ *i, j, k* คือลำดับภาพเรียงจากปัจจุบันสุดไปถึงก่อนหน้าสุด จากนั้นก็นำมาเข้าสู่กระบวนการคัดแยกโดยตรวจสอบเงื่อนไขที่ว่าแคนดิเดทวินโดว์จะเป็นป้ายทะเบียนต้องมีความเร็วไม่เป็นศูนย์และความเร่งเป็นศูนย์ซึ่งเป็นสมมติฐานในอุดมคติ แต่ในความเป็นจริงจะต้องกำหนดค่าจุดเริ่มเปลี่ยน (Threshold) ของความเร็ว (τ_v) และของความเร่ง (τ_a) ขึ้นเพื่อมาทดสอบเงื่อนไขแทนค่าศูนย์เพื่อลดความผิดพลาด ในการทดลองกำหนดค่า $\tau_v = 3^2$ และ $\tau_a = 45^2$ ซึ่งเป็นค่าที่เหมาะสมที่สุดที่ได้จากการทดลอง (สาเหตุที่ต้องยกกำลังสองเนื่องจากในขั้นตอนคำนวณหาความเร็วและความเร่งเราละทิ้งการคำนวณรากที่สอง) และใช้ลำดับภาพทั้งหมด 4 ลำดับภาพ ดังนั้นในแต่ละแคนดิเดทวินโดว์จะคำนวณหาได้ความเร็วทั้งหมดจำนวน 3 ค่า และความเร่งจำนวน 2 ค่า จากนั้นนำค่าทั้งหมดมาพิจารณาเงื่อนไขว่าแคนดิเดทวินโดว์นั้นจะเป็นป้ายทะเบียนได้ก็ต่อเมื่อ ความเร่งทั้ง 2 ค่าจะต้องไม่มากเกินกว่า τ_a และอย่างน้อย 2 ใน 3 ค่าของความเร็วจะต้องไม่ต่ำกว่า τ_v



รูปที่ 3.15 การเคลื่อนที่ของแคนดิเดทวินโดว์

พิจารณารูปที่ 3.15 ซึ่งเป็นภาพขอบจากรูปที่ 3.13(ก) และมีภาพต้นแบบจากรูปที่ 3.12(ก) ซึ่งเป็นภาพที่ปัจจุบันที่สุด และแสดงถึงแคนดิเดทวินโดว์จากทุกลำดับภาพ โดยที่วินโดว์ที่มีสีเดียวกันซึ่งได้แก่ กลุ่ม ก. (ก1, ก2, ก3, ก4) กลุ่ม ข. (ข1, ข2, ข3, ข4) และ กลุ่ม ค. (ค1, ค2, ค3, ค4) เป็นกลุ่มของวินโดว์ที่เกี่ยวข้องกันหรือเป็นวัตถุชิ้นเดียวกัน ซึ่งเรียงจากลำดับภาพปัจจุบันที่สุด

จนถึงก่อนหน้าที่สุด ทำให้เห็นถึงเส้นทางการเคลื่อนที่ของแต่ละแคนดิเดทวินโดว์ สังเกตที่แคนดิเดทวินโดว์สีเขียว (กลุ่ม ค.) ค1, ค2 และ ค3, ค4 ซ้อนทับกันสองคู่แสดงว่าไม่มีการเคลื่อนที่ซึ่งทำให้ความเร็วที่คำนวณได้สองในสามค่ามีค่าเป็นศูนย์ซึ่งน้อยกว่าค่าเริ่มเปลี่ยน (τ_v =3^2) ทำให้แคนดิเดทวินโดว์สีเขียว (กลุ่ม ค.)นี้ถูกคัดออก ต่อมาสังเกตที่แคนดิเดทวินโดว์สีน้ำเงิน (กลุ่ม ข.) ซึ่งระยะทางจาก ข2 ไป ข3 มีความแตกต่างกับระยะทางจาก ข1 ไป ข2 และ ข3 ไป ข4 อย่างเห็นได้ชัด แสดงว่าให้เห็นว่าแคนดิเดทวินโดว์สีน้ำเงิน (กลุ่ม ข.) นี้มีความเร่งมาก ซึ่งมากกว่าค่าเริ่มเปลี่ยน (τ_a =45^2) ดังนั้นวินโดว์สีน้ำเงิน (กลุ่ม ข.) จึงถูกคัดออกด้วยเช่นกัน คงเหลือแต่แคนดิเดทวินโดว์สีแดง (กลุ่ม ก.) เท่านั้น ซึ่งแสดงว่าเป็นป้ายทะเบียน ดังแสดงไว้ในรูปที่ 3.16



รูปที่ 3.16 ผลที่ได้จากการตรวจหาป้ายทะเบียน

3.3.2 การวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน

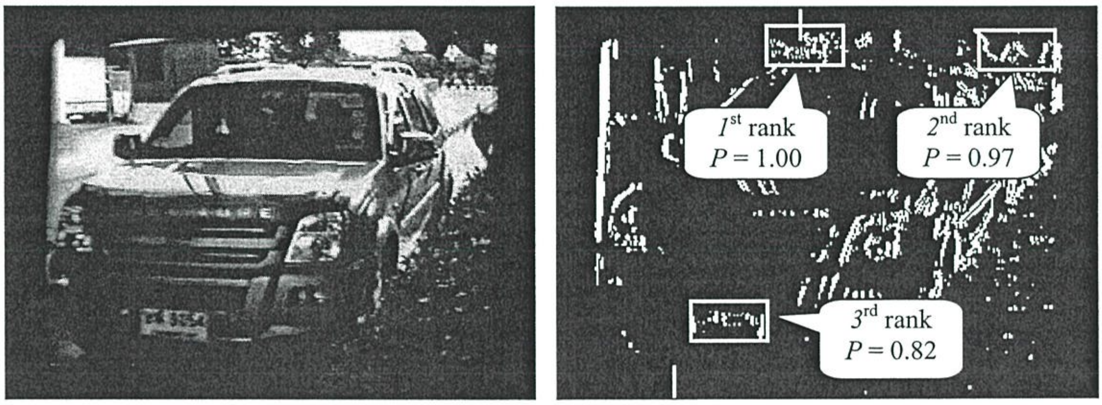
การลดค่าฟอลส์โพสซิทีฟด้วยวิธีนี้เป็นการใช้วิธีวิเคราะห์ค่าความเป็นไปที่ได้จะเป็นป้ายทะเบียน (L) ของแคนดิเดทวินโดว์ทั้งสาม ซึ่งจากการสังเกตในการทดลองแล้วพบว่าค่า L ของวินโดว์ที่เป็นป้ายทะเบียนที่แท้จริงนั้นมักจะมีค่ามากที่สุดหรือใกล้เคียงกับค่า L ที่มากที่สุดในจำนวนค่า L ทั้งหมดจากทั้งสามแคนดิเดทวินโดว์ ดังนั้นเราสามารถเขียนอัตราส่วนระหว่างค่า L ของแต่ละแคนดิเดทวินโดว์เปรียบเทียบกับค่า L ที่มากที่สุดจากทั้งสามแคนดิเดทวินโดว์ได้ดังสมการที่ (3.28)

$$P_i = \frac{L_i}{L_{\max}} \tag{3.28}$$

โดยที่ P_i คืออัตราส่วนของค่า L ของแคนดิเดทวินโดว์ลำดับที่ i กับค่า L ที่มากที่สุด ($L_{\max}$)จากทั้งสามแคนดิเดทวินโดว์ ซึ่งจากการสังเกตข้างต้นแสดงว่าวินโดว์ที่เป็นป้ายทะเบียนที่แท้จริงนั้นค่าของ P_i จะต้องเท่ากับ 1 หรือใกล้เคียงกับ 1 ดังนั้นถ้าหากค่า P_i ของแคนดิเดทวินโดว์ลำดับที่เท่าใดมีค่าน้อยกว่าค่าเริ่มเปลี่ยน P_{limit} แล้ว แคนดิเดทวินโดว์ลำดับนั้นจะถูกคัดออกจากการ

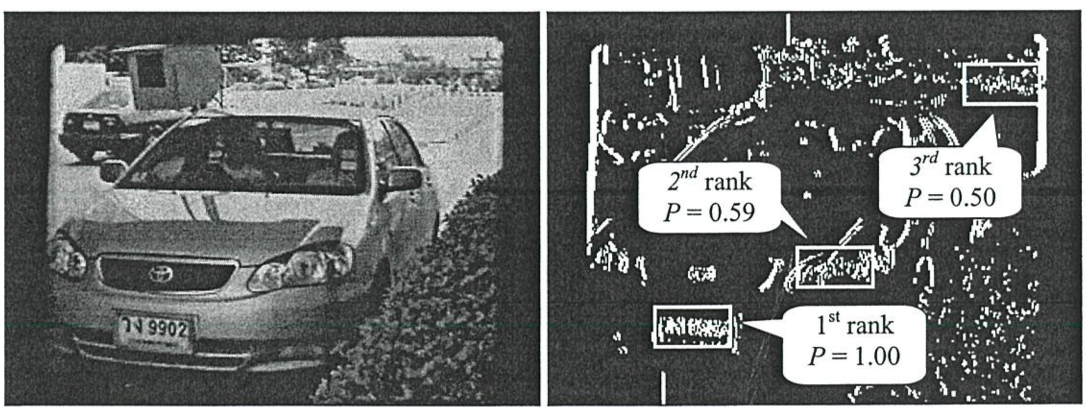
เป็นแคนดิเดทวินโดว์ โดยที่ค่าเริ่มเปลี่ยน P_{limit} นี้สามารถปรับให้ใกล้เคียงกับ 1 หรือห่างจาก 1 ได้ตามความเหมาะสมในแต่ละงานที่นำไปใช้ได้

รูปที่ 3.17 แสดงให้เห็นในกรณีที่ ถึงแม้ว่าป้ายทะเบียนจริงอยู่ในแคนดิเดทวินโดว์ที่มีค่า L มากเป็นลำดับสุดท้าย (ลำดับที่สาม) แต่ค่า P_3 ก็ยังคงค่อนข้างมาก (P_3=0.82) ซึ่งแคนดิเดทวินโดว์นี้จะถูกพิจารณาให้เป็นแคนดิเดทวินโดว์ที่คาดว่าน่าจะเป็นป้ายทะเบียนต่อไป ในขณะที่รูปที่ 3.18 แสดงให้เห็นถึงในอีกกรณีหนึ่ง ที่ซึ่งป้ายทะเบียนจริงอยู่ในแคนดิเดทวินโดว์ที่มีค่า L มากเป็นลำดับแรกสุด ส่วนแคนดิเดทวินโดว์ที่อยู่ในลำดับที่สองและสามไม่ใช่ป้ายทะเบียน เมื่อพิจารณาดูที่ค่า P_2, P_3 ซึ่งเป็นของแคนดิเดทวินโดว์ที่ลำดับสองและสามตามลำดับแล้ว จะพบว่ามีค่าค่อนข้างน้อย (P_2=0.59 , P_3=0.50) ซึ่งจะถูกคัดออกจากการเป็นแคนดิเดทวินโดว์ไปในที่สุด คงเหลือแต่แคนดิเดทวินโดว์ลำดับที่หนึ่งเท่านั้นที่จะถูกเลือกให้เป็นแคนดิเดทวินโดว์ที่คาดว่าจะเป็นป้ายทะเบียนที่แท้จริงเพียงวินโดว์เดียว



(ก) (ข)

รูปที่ 3.17 กรณีป้ายทะเบียนอยู่ในลำดับที่สาม (ก) รูปต้นฉบับ (ข) ภาพขอบแสดงค่า P_i ในแต่ละลำดับของแคนดิเดทวินโดว์

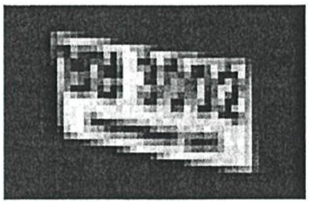


(ก) (ข)

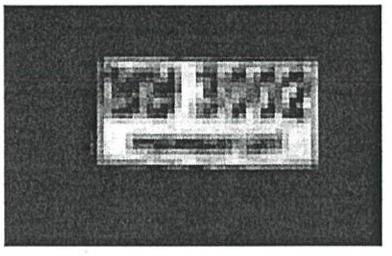
รูปที่ 3.18 กรณีป้ายทะเบียนอยู่ในลำดับที่หนึ่ง (ก) รูปต้นฉบับ (ข) ภาพขอบแสดงค่า P_i ในแต่ละลำดับของแคนดิเดทวินโดว์

3.4 การแก้ไขภาพเพอร์สเปคทีฟ(Perspective Correction)

ในกรณีที่มุมในการถ่ายภาพของกล้องกับระนาบของป้ายทะเบียนไม่ตั้งฉากกันจะทำให้ภาพที่ได้มีความบิดเบี้ยวไปจากความเป็นจริง ซึ่งเรียกว่าภาพเพอร์สเปคทีฟ และจะทำให้ภาพของตัวอักขระบนป้ายทะเบียนบิดเบี้ยวไปด้วย ทำให้การอ่านป้ายทะเบียนนั้นผิดพลาดได้ด้วยเช่นกัน ดังนั้นจึงต้องมีการแก้ไขภาพนั้นเสียก่อนที่จะส่งไปสู่กระบวนการการอ่านป้ายทะเบียน ดังแสดงในรูปที่ 3.19



(ก)



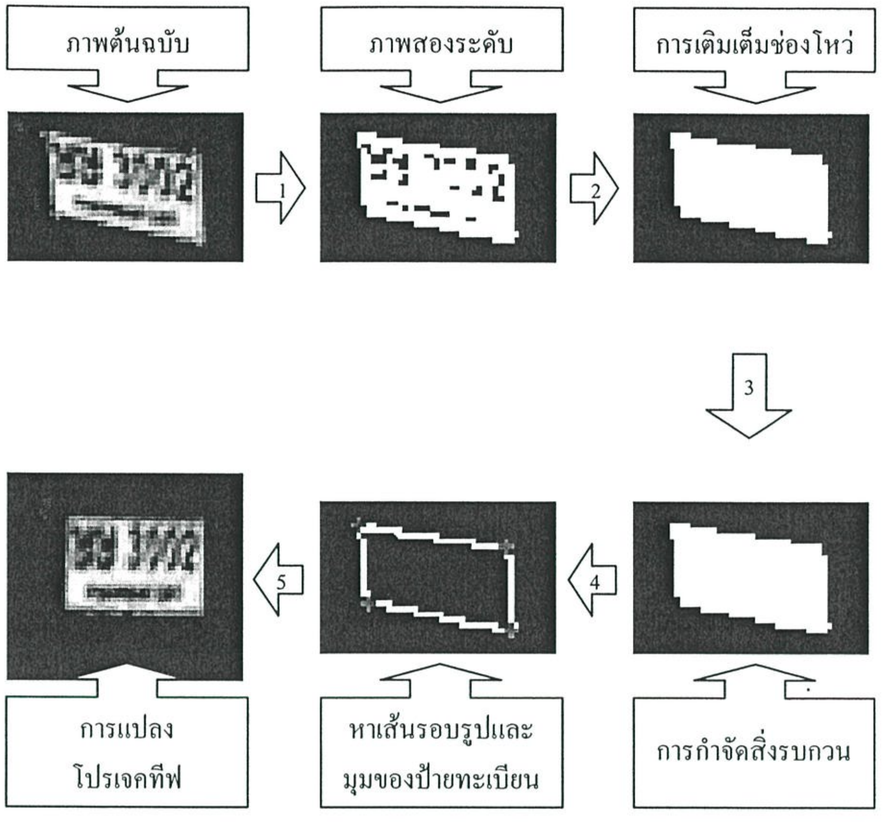
(ข)

รูปที่ 3.19 ภาพเพอร์สเปคทีฟ (ก) ก่อนการแก้ไข (ข) หลังการแก้ไข

3.4.1 ขั้นตอนในการแก้ไขภาพเพอร์สเปคทีฟ

เนื้อหาในส่วนนี้จะกล่าวถึงขั้นตอนวิธีการในการแก้ไขภาพเพอร์สเปคทีฟให้กลับมาตรงดังเดิม ก่อนที่จะส่งภาพส่วนนั้นไปสู่ขั้นตอนการอ่านป้ายทะเบียนต่อไป โดยการหามุมทั้งสี่ของป้ายทะเบียนจากภาพต้นฉบับ จากนั้นใช้การแปลงโปคเจคทีฟ (Projective Transformation) ในการแปลงภาพให้กลับมาตรงดังเดิมโดยใช้จุดมุมทั้งสี่ที่หามาได้นั้นเป็นจุดอ้างอิงในการดึงภาพ ดังขั้นตอนที่แสดงในรูปที่ 3.20 โดยประกอบด้วย 5 ขั้นตอนหลัก เรียงลำดับดังต่อไปนี้

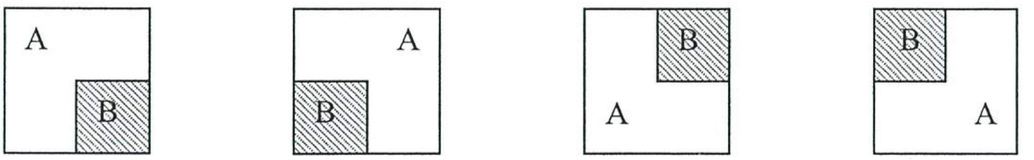
1. แปลงภาพต้นฉบับไปเป็นภาพสองระดับ (Binary) –เป็นการแปลงภาพต้นฉบับให้อยู่ในรูปของภาพสองระดับ
2. เติมเต็มช่องโหว่ (Fill Hole) – ทำการเติมจุดลงไปในบริเวณที่เป็นช่องโหว่จนเต็ม
3. กำจัดสิ่งรบกวน (Noise) – เป็นการกำจัดจุดเล็กๆ ที่เป็นสัญญาณรบกวนออกไป
4. หาเส้นรอบรูปและมุมของป้ายทะเบียน – เป็นการหาเส้นรอบรูปของป้ายทะเบียนเพื่อนำข้อมูลที่ได้ มาใช้ในการหามุมของป้ายทะเบียนต่อไป
5. ใช้การแปลงโปรเจคทีฟ (Projective Transformation) ในการแปลงภาพต้นฉบับให้กลับมาตรงดังเดิม



รูปที่ 3.20 ขั้นตอนในการแก้ไขภาพเพอร์สเปคทีฟ

3.4.2 การตรวจหามุมของป้ายทะเบียน

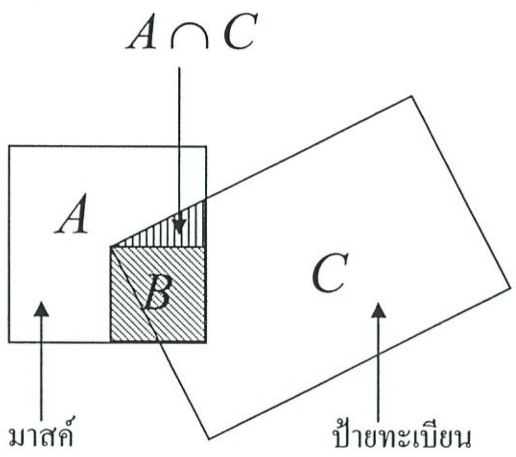
ในการตรวจหามุมของป้ายทะเบียนนั้น เมื่อได้เส้นรอบรูปของป้ายทะเบียนแล้วเราจะใช้มาสค์สี่เหลี่ยมที่มีจุดศูนย์กลางอยู่บนเส้นรอบรูปแล้วเคลื่อนที่ไปยังจุดทุกจุดบนเส้นรอบรูปนั้น โดยมาสค์ที่ใช้จะแตกต่างกันตามมุมของป้ายทะเบียนที่เราต้องการหา ได้แก่มาสค์ในรูปที่ 3.21(ก) ใช้สำหรับหาจุดมุมซ้ายบน มาสค์ในรูปที่ 3.21(ข) ใช้สำหรับหาจุดมุมขวาบน มาสค์ในรูปที่ 3.21(ค) ใช้สำหรับหาจุดมุมซ้ายล่าง มาสค์ในรูปที่ 3.21(ง) ใช้สำหรับหาจุดมุมขวาล่าง โดยจุดที่จะเป็นมุมของป้ายทะเบียนได้นั้นจะเป็นจุดที่ทำให้พื้นที่ที่ไม่ได้แรเงา (A) ทับกับพื้นที่ภายในป้ายทะเบียน(C) น้อยที่สุดดังแสดงในรูปที่ 3.22 หรือสามารถเขียนให้อยู่ในรูปสมการที่ (3.29)



(ก) (ข) (ค) (ง)

รูปที่ 3.21 มาสค์ที่ใช้ในการหามุมต่าง (ก)มุมซ้ายบน (ข)มุมขวาบน (ค)มุมซ้ายล่าง (ง)มุมขวาล่าง

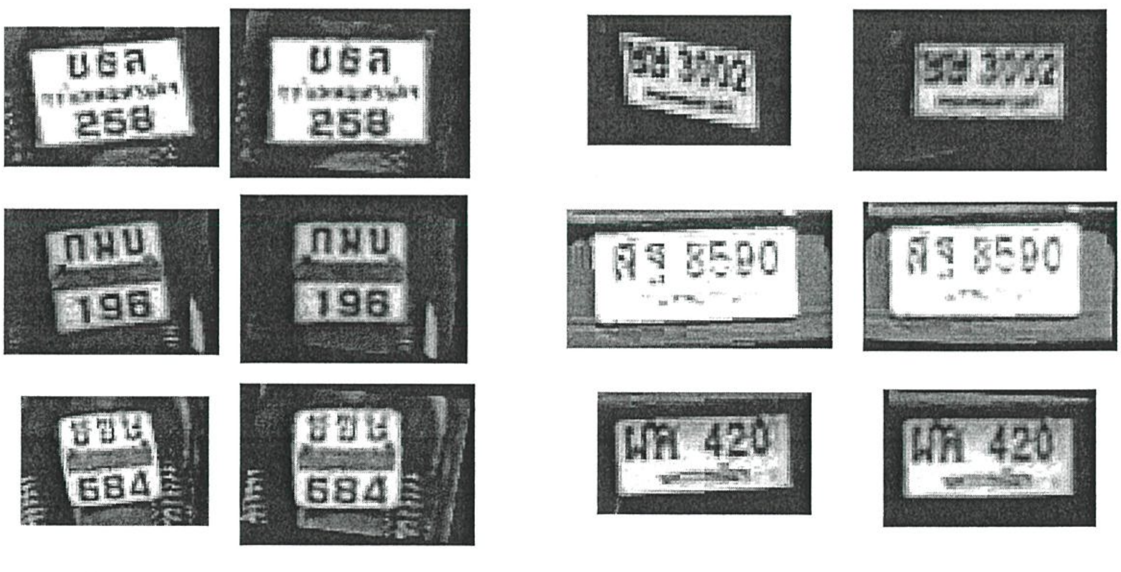
$$\text{จุดมุม} = \min \sum \left(A \cap C \right) \tag{3.29}$$



รูปที่ 3.22 ตัวอย่างการหาจุดมุมบนซ้ายโดยใช้มาสค์ในรูปที่ 3.21(ก) เคลื่อนที่ไปตลอดแนวเส้นรอบรูปของป้ายทะเบียน แล้วหาตำแหน่งที่ทำให้ $A \cap C$ มีค่าน้อยสุด

3.4.3 ผลที่ได้จากการแก้ไขภาพเพอร์สเปคทีฟ

หลังจากที่เราทำการแก้ไขภาพเพอร์สเปคทีฟแล้ว เราสามารถนำภาพที่ได้ส่งผ่านไปยังกระบวนการอ่านป้ายทะเบียนต่อไป โดยจะสังเกตได้ว่าภาพตัวอักขระบนป้ายทะเบียนที่ผ่านกระบวนการแล้วจะถูกดึงให้ตรงดังแสดงในรูปที่ 3.23 ซึ่งจะทำให้การอ่านป้ายทะเบียนนั้นทำได้อย่างถูกต้องแม่นยำมากยิ่งขึ้น



รูปที่ 3.23 ตัวอย่างภาพป้ายทะเบียนที่ผ่านกระบวนการแก้ไขภาพเพอร์สเปคทีฟเรียบร้อยแล้ว

บทที่ 4

วิธีการทดลองและผลการทดลอง

ในบทนี้จะกล่าวถึงขั้นตอนในการเก็บภาพตัวอย่างที่จะนำมาใช้ในการทดลอง วิธีที่ใช้ในการเลือกกลุ่มภาพตัวอย่างสำหรับการทดลอง ขั้นตอนและวิธีการทดลองการทำงานกับวิธีการต่างๆ ที่ได้เสนอไว้ในวิทยานิพนธ์ฉบับนี้ แสดงผลการทดลองทั้งหมดที่ได้ในแต่ละวิธีการทดลอง รวมถึงแสดงการนำผลการทดลองที่ได้จากแต่ละวิธีการมาเปรียบเทียบกัน เพื่อให้เห็นถึงประสิทธิภาพการทำงานของแต่ละวิธีการด้วย

4.1 แหล่งที่มาของภาพวีดิทัศน์ที่ใช้ในการทดลอง

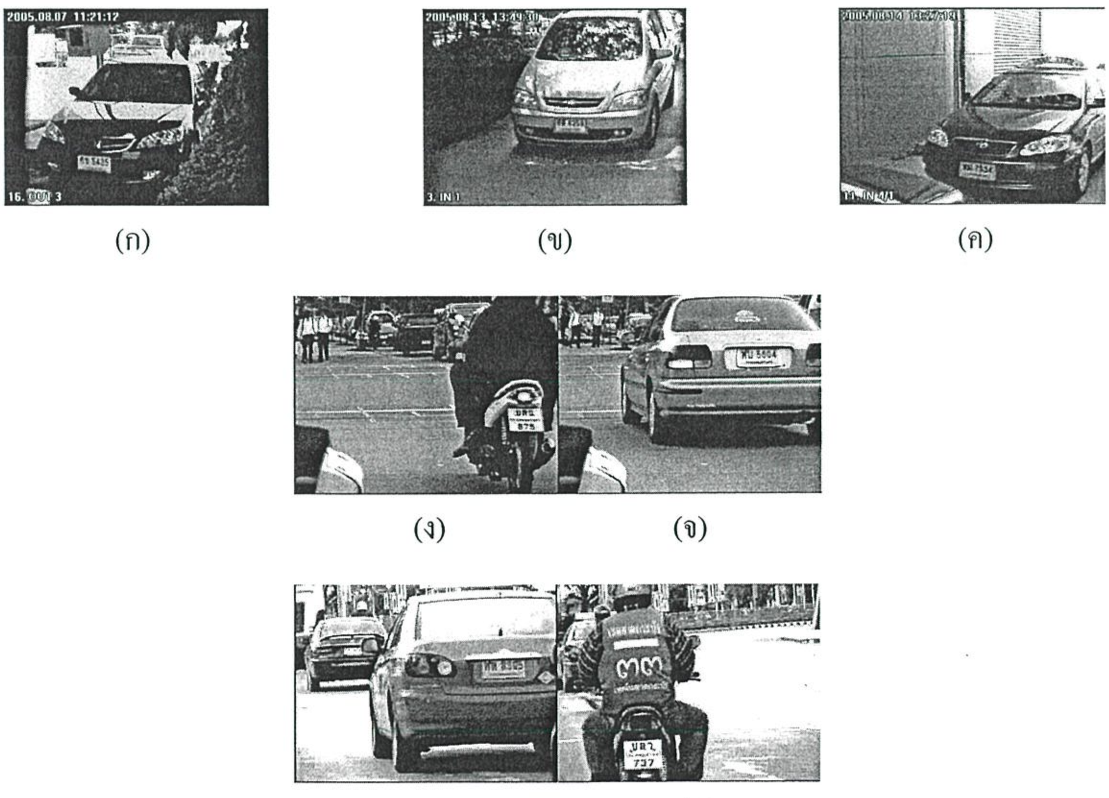
ขั้นตอนการทดลองในวิทยานิพนธ์ฉบับนี้เริ่มจากการเก็บภาพวีดีทัศน์ตัวอย่างจากแหล่งภาพวีดิทัศน์ โดยผู้ทำวิทยานิพนธ์แบ่งการเก็บภาพตัวอย่างเป็นสองกลุ่ม ได้แก่ กลุ่มภาพวีดิทัศน์ตัวอย่างจากทางเข้าออกอาคารและกลุ่มภาพวีดิทัศน์ตัวอย่างจากถนน

4.1.1 ภาพวีดิทัศน์ตัวอย่างจากทางเข้าออกอาคาร

ภาพวีดิทัศน์ตัวอย่างจากกลุ่มนี้ได้มาจากการใช้งานจริงในชีวิตประจำวัน โดยเลือกนำภาพจากฐานข้อมูลภาพวีดิทัศน์ ของกล้องวงจรปิดทางเข้าออกศูนย์การค้าแห่งหนึ่งในกรุงเทพมหานคร ซึ่งมีการติดตั้งกล้องวงจรปิดบริเวณทางเข้าออก และไม่มีเครื่องกั้นทางเข้าออก มีเพียงลูกระนาดเพื่อชะลอความเร็วรถเท่านั้น ทำให้รถที่ผ่านเข้าออกขณะทำการบันทึกภาพนั้นเคลื่อนที่อย่างช้าๆอยู่ตลอดเวลา และทำการบันทึกภาพการเข้าออกไว้ในฐานข้อมูล โดยมีการบีบอัดภาพให้มีขนาดของข้อมูลลดลงเพื่อความสะดวกและประหยัดพื้นที่ในการจัดเก็บ ทั้งนี้เนื่องจากผู้ทำวิทยานิพนธ์เล็งเห็นว่าศูนย์การค้าเป็นสถานที่ที่เหมาะสมในการรวบรวมข้อมูลเนื่องจากเป็นแหล่งรวมของรถหลากหลายประเภท หลายยี่ห้อ และหลายรุ่น โดยการเก็บภาพนี้ได้จัดเก็บภาพตั้งแต่เวลาเช้าจนถึงกลางคืนเพื่อให้ครอบคลุมทุกสภาพแสง และภาพวีดิทัศน์ที่ได้มานั้นมีความละเอียดอยู่ที่ 320×240 พิกเซล และเฟรมเรตอยู่ที่ 4 เฟรม/วินาที โดยภาพวีดิทัศน์ตัวอย่างกลุ่มนี้ประกอบไปด้วยรถเก๋งและรถกระบะเท่านั้น ไม่รวมถึงรถตู้ รถโดยสาร รถบรรทุกหรือรถจักรยานยนต์ ซึ่งผู้ทำวิจัยได้เลือกฉาก (Scene) ในการเก็บภาพวีดิทัศน์ของรถตัวอย่างจากด้านหน้ารถไว้สามฉากด้วยกัน แบ่งเป็นฉากที่หนึ่งจำนวน 70 คัน ฉากที่สองจำนวน 63 คัน และฉากที่สามจำนวน 64 คัน รวมเป็นภาพวีดิทัศน์ตัวอย่างจากทางเข้าออกอาคารทั้งหมด 197 คัน ดังแสดงไว้ในรูปที่ 4.1 (ก),(ข) และ (ค) โดยภาพวีดิทัศน์ตัวอย่างของรถแต่ละคันที่ได้มานั้นจะถูกนำมาตัด (Capture) ออกมาเป็นจำนวน 4 เฟรม ก่อนที่จะนำไปเป็นอินพุต (Input) ที่ใช้ในการทดลอง โดยจะกล่าวถึงรายละเอียดในหัวข้อ 4.2

4.1.2 ภาพวีดิทัศน์ตัวอย่างจากถนน

ภาพวีดิทัศน์ตัวอย่างจากกลุ่มนี้ได้มาจากการตั้งกล้องวีดิโอไว้บริเวณริมถนนแล้วบันทึกภาพรถขณะกำลังเคลื่อนที่ผ่านรางรถไฟ ซึ่งมีผลทำให้รถแต่ละคันชะลอความเร็วลงขณะกำลังเคลื่อนที่ โดยทำการบันทึกภาพลงบนม้วนวีดิโอเทป จากนั้นจึงถ่ายโอนลงในคอมพิวเตอร์โดยผ่านการ์ดวีดิโอแคปเจอร์(Video Capture Card) ซึ่งใช้การบีบอัดภาพแบบ MPEG1 ที่ความละเอียด 320×240 พิกเซล เฟรมเรตอยู่ที่ 30 เฟรม/วินาที โดยการเก็บภาพนี้ได้จัดเก็บภาพตั้งแต่ช่วงบ่ายจนถึงกลางคืนเพื่อให้ครอบคลุมทุกสภาพแสง ในขณะที่ภาพวีดิทัศน์ตัวอย่างกลุ่มนี้ประกอบไปด้วยรถยนต์ทุกประเภททั้งรถเก๋ง รถกระบะ รถตู้ รถโดยสาร รถบรรทุก รวมถึงรถจักรยานยนต์ โดยในการบันทึกภาพนั้นจะเป็นการบันทึกภาพจากด้านหลังของรถ เนื่องจากรถจักรยานยนต์มีป้ายทะเบียนติดไว้บริเวณหลังรถเท่านั้น ซึ่งฉาก (Scene) ในการเก็บภาพวีดิทัศน์ตัวอย่างกลุ่มนี้มีด้วยกันสองฉาก แบ่งเป็นฉากที่หนึ่ง รถยนต์จำนวน 69 คัน รถจักรยานยนต์จำนวน 62 คัน ฉากที่สอง รถยนต์จำนวน 64 คัน รถจักรยานยนต์จำนวน 48 คัน รวมภาพวีดิทัศน์ตัวอย่างของกลุ่มนี้จำนวนทั้งหมด 243 คันโดยแบ่งเป็นรถยนต์จำนวน 133 คัน และรถจักรยานยนต์จำนวน 110 ซึ่งภาพวีดิทัศน์ตัวอย่างกลุ่มนี้แสดงไว้ในรูปที่ 4.1 (ง),(จ),(ฉ) และ (ช)



(ก) (ข) (ค)

(ง) (จ)

(ฉ) (ช)

รูปที่ 4.1 ตัวอย่างภาพวีดิทัศน์จากทางเข้าออกอาคาร (ก) ฉากที่หนึ่ง (ข) ฉากที่สอง (ค) ฉากที่สาม และตัวอย่างภาพวีดิทัศน์จากจากถนน (ง),(จ) ฉากที่หนึ่ง (ฉ),(ช) ฉากที่สอง

4.2 การสร้างภาพอินพุต (Input) จากภาพวีดิทัศน์

หลังจากที่ได้แหล่งของตัวอย่างภาพวีดิทัศน์แล้ว ขั้นตอนต่อไปคือการสร้างภาพที่จะนำไปเป็นอินพุตสำหรับการทดลอง โดยภาพที่ใช้ในการทดลองเป็นภาพนิ่งที่ตัด (Capture) ด้วยมือโดยผู้ทำวิทยานิพนธ์ จากภาพวีดิทัศน์จำนวน 4 เฟรมติดต่อกัน โดยเลือกภาพขณะที่มีรถปรากฏอยู่ในเฟรมในระยะที่สามารถมองเห็นป้ายทะเบียนได้ชัดทั้ง 4 เฟรมดังตัวอย่างในรูปที่ 4.2

รูปที่ 4.2 ตัวอย่างของภาพนิ่งที่ใช้เป็นอินพุตจำนวน 4 เฟรมซึ่งตัด (Capture) มาจากภาพวีดิทัศน์

4.3 การกำหนดค่าเริ่มเปลี่ยน (Threshold) และค่าคงที่ที่ใช้ในการทดลอง

ในการทดลองนั้นจะต้องมีการกำหนดค่าเริ่มเปลี่ยนต่างๆ รวมถึงค่าคงที่ เพื่อใช้สำหรับการประมวลผลของระบบ ในหัวข้อนี้จะกล่าวถึงค่าเริ่มเปลี่ยนและค่าคงที่ต่างๆและวิธีการที่ได้มาถึงค่าเหล่านั้น โดยสามารถแบ่งออกเป็นสองส่วน โดยส่วนที่หนึ่งได้แก่ค่าคงที่และการกำหนดค่าคงที่ ส่วนที่สองคือค่าเริ่มเปลี่ยนและการกำหนดค่าเริ่มเปลี่ยน

4.3.1 ค่าคงที่และการกำหนดค่าคงที่

ค่าคงที่ที่ใช้ในการทดลองนี้ส่วนใหญ่เป็นค่าที่ใช้กำหนดคุณสมบัติของวินโดว์ที่ใช้ในการสแกนหาป้ายทะเบียน ประกอบด้วยค่า 3 ค่าได้แก่ ค่าอัตราส่วนความกว้างของวินโดว์ (w) ค่าอัตราส่วนความสูงของวินโดว์ (h) และค่าสเกลลิง (S) (Scaling Value) ที่ใช้สำหรับคูณเข้ากับค่า w และ h เพื่อใช้สำหรับกำหนดขนาดของวินโดว์ที่ใช้ในการสแกนซึ่งจะได้ขนาด (กว้าง × ยาว) ของวินโดว์ที่ใช้สแกนคือ $(Sw \times Sh)$ พิกเซล นอกจากนั้นแล้วยังมีค่าคงที่อีกค่าหนึ่งคือค่า ρ_e ซึ่ง

หมายถึงจำนวนเปอร์เซ็นต์ของพิกเซลขอบเมื่อเทียบกับจำนวนพิกเซลทั้งหมดบนภาพดังที่กล่าวไว้ในหัวข้อ 3.1.2 ซึ่งเป็นขั้นตอนหนึ่งในกระบวนการหาภาพขอบ (Edge Image)

โดยในการสแกนหาป้ายทะเบียนนั้น เราต้องทำการสแกนด้วยวินโดว์ที่มีอัตราส่วนกว้างยาวใกล้เคียงกับป้ายทะเบียนจริง สำหรับการทดลองนี้กำหนดค่า w = 45 พิกเซล h = 24 พิกเซล ในกรณีของการหาป้ายทะเบียนรถยนต์ และกำหนดค่า w = 35 พิกเซล h = 28 พิกเซล ในกรณีของการหาป้ายทะเบียนรถจักรยานยนต์ ในขณะเดียวกันก็จะทำการสเกลลิงขนาดของวินโดว์ด้วยการเปลี่ยนค่า S เพื่อให้ครอบคลุมป้ายทะเบียนที่ขนาดและระยะห่างต่างๆกัน ซึ่งในการทดลองนี้ทำการสแกนทั้งหมดสองรอบ ด้วยการเปลี่ยนค่า S ที่ใช้ในแต่ละรอบดังนี้ รอบที่หนึ่ง กำหนดค่า S = 1 และรอบที่สองกำหนดค่า S = 1.3 โดยใช้ค่าดังกล่าวทั้งการหาป้ายทะเบียนรถยนต์และรถจักรยานยนต์ ในส่วนของค่า ρ_e จะมีการกำหนดให้เหมาะสมกับฉาก(Scene) แต่ละฉากซึ่งขึ้นอยู่กับลักษณะของวัตถุต่างๆในสภาพแวดล้อมนั้น สำหรับภาพที่ได้จากทางเข้าออกอาคารฉากที่ 1 กำหนดค่า ρ_e = 0.08 ฉากที่ 2 กำหนดค่า ρ_e = 0.09 และฉากที่ 3 กำหนดค่า ρ_e = 0.09 ส่วนภาพตัวอย่างที่ได้จากถนนทุกฉากกำหนดค่า ρ_e = 0.06 เท่ากันทุกฉาก ซึ่งค่าคงที่ที่กำหนดไว้ทั้งหมดนี้ได้มาจากการทดลองซ้ำๆหลายครั้งจนได้ค่าที่เหมาะสมที่สุดและสามารถสรุปได้ดังตารางที่ 4.1

4.3.2 ค่าเริ่มเปลี่ยนและการกำหนดค่าเริ่มเปลี่ยน

การทดลองนี้มีการกำหนดค่าเริ่มเปลี่ยนอยู่ในสองขั้นตอนการทำงาน ซึ่งประกอบด้วย ค่า P_{limit} ที่ใช้ในขั้นตอนการลดค่าฟอลส์โพสซิทีฟด้วยการวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนในหัวข้อ 3.3.2 และค่าเริ่มเปลี่ยนความหนาแน่นสัมพัทธ์ของพิกเซลขอบที่ใช้ในขั้นตอนการหาแคนดิเดทวินโดว์ในหัวข้อ 3.2.2 โดยสามารถแบ่งเป็นค่า τ_{AB} , τ_{CB} สำหรับการหาป้ายทะเบียนรถยนต์และค่า τ_{BA} , τ_{BC} , τ_{CA} สำหรับการหาป้ายทะเบียนรถจักรยานยนต์

ปัจจัยที่มีผลกระทบต่อการกำหนดค่า P_{limit} คือความคล้ายคลึงกันของการกระจายของพิกเซลขอบของวัตถุอื่นกับป้ายทะเบียน ในกรณีที่มีความคล้ายคลึงกันมากค่าของ P_{limit} ก็จะสูงขึ้น ในทางตรงกันข้ามหากมีความคล้ายคลึงกันน้อยค่า P_{limit} ก็จะน้อยลงด้วย ซึ่งในกรณีของภาพตัวอย่างจากทางเข้าออกอาคารนั้นในฉากที่ 1 กำหนดค่า P_{limit} =0.75 ในฉากที่ 2 กำหนดค่า P_{limit} =0.85 ในฉากที่ 3 กำหนดค่า P_{limit} =0.75 ส่วนภาพตัวอย่างที่ได้จากถนนกำหนดค่า P_{limit} =0.75 เท่ากันทั้งหมดในทุกฉาก ในขณะที่ค่าเริ่มเปลี่ยนความหนาแน่นสัมพัทธ์ของพิกเซลขอบจะขึ้นอยู่กับลักษณะการกระจายของพิกเซลขอบของป้ายทะเบียนแต่ละชนิดว่ามีความหนาแน่นในบริเวณใดหรือมีความเบาบางในบริเวณใด ซึ่งในการทดลองนี้กำหนดค่า τ_{AB} = 0.2 และ τ_{CB} = 0.2 สำหรับการหาป้ายทะเบียนรถยนต์ในทุกฉาก และกำหนดค่า τ_{BA} = 0.2 τ_{BC} = 0.2 τ_{CA} = 0.4 สำหรับการหาป้ายทะเบียนรถจักรยานยนต์ในทุกฉาก ซึ่งค่าที่กำหนดนี้ ได้มาจากการทดลองแล้วหลายๆครั้งจนได้ค่าที่เหมาะสมที่สุด ซึ่งสามารถเขียนสรุปได้ดังตารางที่ 4.1

ตารางที่ 4.1 ค่าคงที่และค่าเริ่มเปลี่ยนที่ใช้ในการทดลองในแต่ละฉาก(Scene)

ค่าคงที่และค่าเริ่มเปลี่ยน	ภาพจากทางเข้า-ออกอาคาร			ภาพจากถนน			
	รถยนต์			รถยนต์		รถจักรยานยนต์	
	ฉากที่ 1	ฉากที่ 2	ฉากที่ 3	ฉากที่ 1	ฉากที่ 2	ฉากที่ 1	ฉากที่ 2
w	45					35	
h	24					28	
S	1 และ 1.3						
ρ_e	0.08	0.09	0.09	0.06			
P_{limit}	0.75	0.85	0.75	0.75			
τ_{AB}	0.2					-	
τ_{CB}	0.2					-	
τ_{BA}	-					0.2	
τ_{BC}	-					0.2	
τ_{CA}	-					0.4	

4.4 สมมติฐานที่ใช้ในการทดลอง

ในการทดลองนี้เป็นการทดสอบการทำงานของกรรมวิธีในการหาตำแหน่งป้ายทะเบียนรถที่ได้นำเสนอไว้ในวิทยานิพนธ์ฉบับนี้ โดยใช้ข้อมูลภาพของรถจำนวน 440 คัน ซึ่งภาพของรถแต่ละคันที่ใช้ในการทดสอบเป็นภาพนิ่งจำนวน 4 เฟรมที่ตัด (Capture) จากไฟล์ภาพวีดิทัศน์ ด้วยมือในขณะที่รถอยู่ห่างจากกล้องในระยะที่เหมาะสม ภายใต้เงื่อนไขที่รถจะต้องมีการเคลื่อนที่ในความเร็วที่เหมาะสม โดยการทดลองนี้เป็นการประมวลผลแบบออฟไลน์ (Offline Processing)

4.5 ผลการทดลอง

หลังจากทำการกำหนดค่าให้กับค่าคงที่และค่าเริ่มเปลี่ยนตามที่กำหนดไว้ในหัวข้อก่อนหน้านี้ รวมถึงเมื่อนำภาพอินพุตที่ได้จากขั้นตอนดังกล่าวไปผ่านกระบวนการหาตำแหน่งป้ายทะเบียนตามที่ได้นำเสนอไว้ในวิทยานิพนธ์ฉบับนี้ ผลการทดลองสามารถแยกออกเป็น 3 ส่วนด้วยกันดังนี้

1. ผลการทดลองของภาพวีดิทัศน์ที่บันทึกจากทางเข้า-ออกอาคาร ซึ่งมีลักษณะเป็นฉากคงที่ (Static Scene) เนื่องจากสภาพแวดล้อมของฉากหลังค่อนข้างคงที่ มีการเคลื่อนที่หรือเปลี่ยนแปลงของฉากหลังไม่มากนัก เพราะว่ารถจะต้องวิ่งเข้ามาตามช่องทางที่กำหนดทีละคัน ดังแสดงในตารางที่ 4.2

2. ผลการทดลองของภาพวีดิทัศน์ที่บันทึกจากถนน มีลักษณะเป็นฉากเคลื่อนที่ (Dynamic Scene) เนื่องจากสภาพแวดล้อมของฉากหลังมีการเคลื่อนที่อยู่ตลอดเวลาเพราะว่าบนถนนนั้นรถสามารถเคลื่อนได้อย่างอิสระ ดังแสดงในตารางที่ 4.3

3. ผลการทดลองรวมของแหล่งภาพทั้งสองแหล่งภาพ เป็นการรวมผลการทดลองจากทั้งสองเข้าด้วยกันเพื่อแสดงให้เห็นถึงการทำงานของระบบในภาพรวม ดังแสดงในตารางที่ 4.4

ตารางที่ 4.2 ผลการทดลองของภาพวีดิทัศน์ที่บันทึกจากทางเข้า-ออกอาคาร (ฉากคงที่)

วิธีการลดค่าฟอลส์โพสซิทีฟ	จำนวนป้ายทะเบียน	จำนวนแคนดิเดทวินโดว์	ตรวจพบป้ายทะเบียน (ทรูโพสซิทีฟ)	ตรวจไม่พบป้ายทะเบียน (ฟอลส์เนกกาทีฟ)	ตรวจพบแต่ไม่ใช่ป้ายทะเบียน (ฟอลส์โพสซิทีฟ)
ไม่ลดค่าฟอลส์โพสซิทีฟ	197	591	194(98%)	3(2%)	397
วิธีวิเคราะห์ข้อมูลเชิงเวลา	197	296	194(98%)	3(2%)	115
วิธีวิเคราะห์ข้อมูลเชิงเวลาร่วมกับวิธีวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน (L)	197	232	194(98%)	3(2%)	38

ตารางที่ 4.3 ผลการทดลองของภาพวีดิทัศน์ที่บันทึกจากถนน (ฉากเคลื่อนที่)

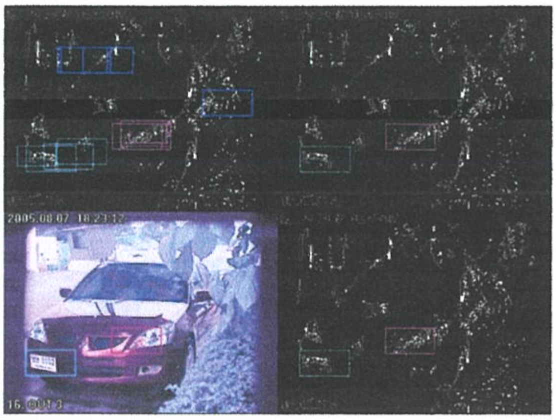
วิธีการลดค่าฟอลส์โพสซิทีฟ	จำนวนป้ายทะเบียน	จำนวนแคนดิเดทวินโดว์	ตรวจพบป้ายทะเบียน (ทรูโพสซิทีฟ)	ตรวจไม่พบป้ายทะเบียน (ฟอลส์เนกกาทีฟ)	ตรวจพบแต่ไม่ใช่ป้ายทะเบียน (ฟอลส์โพสซิทีฟ)
ไม่ลดค่าฟอลส์โพสซิทีฟ	243	729	239(98%)	4(2%)	490
วิธีวิเคราะห์ข้อมูลเชิงเวลา	243	438	239(98%)	4(2%)	199
วิธีวิเคราะห์ข้อมูลเชิงเวลาร่วมกับวิธีวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน (L)	243	275	239(98%)	4(2%)	36

ตารางที่ 4.4 ผลการทดลองรวมของแหล่งภาพทั้งสองแหล่งภาพ (ฉากคงที่ + ฉากเคลื่อนที่)

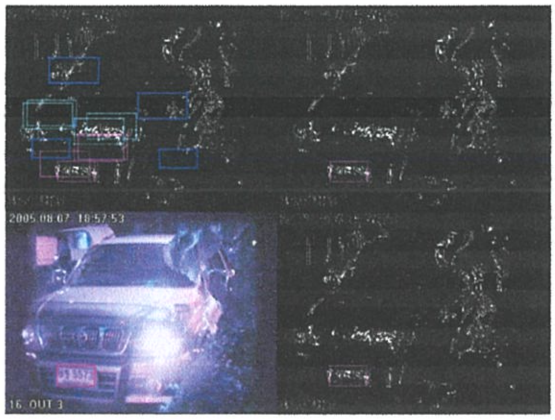
วิธีการลดค่าฟอลส์โพสซิทีฟ	จำนวนป้ายทะเบียน	จำนวนแคนดิเดทวินโดว์	ตรวจพบป้ายทะเบียน (ทรูโพสซิทีฟ)	ตรวจไม่พบป้ายทะเบียน (ฟอลส์เนกกาทีฟ)	ตรวจพบแต่ไม่ใช่ป้ายทะเบียน (ฟอลส์โพสซิทีฟ)
ไม่ลดค่าฟอลส์โพสซิทีฟ	440	1320	433(98%)	7(2%)	887
วิธีวิเคราะห์ข้อมูลเชิงเวลา	440	747	433(98%)	7(2%)	314
วิธีวิเคราะห์ข้อมูลเชิงเวลาร่วมกับวิธีวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน (L)	440	507	433(98%)	7(2%)	74

จากผลการทดลองในตารางที่ 4.4 สามารถเห็นได้ว่าจากจำนวนป้ายทะเบียนทั้งหมด 440 ป้าย ระบบสามารถตรวจพบทั้งหมด 433 ป้าย คิดเป็นความถูกต้องในการตรวจสอบถึง 98 % แต่ในขณะเดียวกันก็มีฟอลส์โพสซิทีฟสูงถึง 887 ป้าย แต่เมื่อผ่านกระบวนการวิเคราะห์ข้อมูลเชิงเวลาแล้วฟอลส์โพสซิทีฟลดลงเหลือ 314 ป้าย และเมื่อนำไปผ่านกระบวนการวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน (L) เพิ่มเติม ค่าฟอลส์โพสซิทีฟสามารถลดลงเหลือเพียง 74 ป้ายเท่านั้น และเมื่อแยกพิจารณาเป็นกรณีฉากคงที่ดังตารางที่ 4.2 จากจำนวนป้ายทะเบียน 197 ป้าย สามารถตรวจพบป้ายทะเบียน 194 ป้าย คิดเป็น 98% เกิดฟอลส์โพสซิทีฟจำนวน 397 ป้าย และเมื่อผ่านกระบวนการแล้วสามารถลดฟอลส์โพสซิทีฟลงได้เหลือเพียง 38 ป้ายเท่านั้น และในกรณีฉากเคลื่อนที่ดังตารางที่ 4.3 จากจำนวนป้ายทะเบียนทั้งหมด 243 ป้าย ตรวจพบป้ายทะเบียนทั้งหมด 239 ป้าย คิดเป็น 98% ในขณะที่เกิดฟอลส์โพสซิทีฟจำนวน 490 ป้าย ซึ่งสามารถทำให้ลดลงหลังจากผ่านกระบวนการลดค่าฟอลส์โพสซิทีฟแล้วเหลือเพียง 36 ป้ายเท่านั้น

จากผลการทดลองทำให้เห็นได้ว่า วิธีการที่นำเสนอในวิทยานิพนธ์นี้สามารถตรวจหาตำแหน่งป้ายทะเบียนได้อย่างถูกต้องและมีกระบวนการในการลดค่าฟอลส์โพสซิทีฟให้ต่ำลงอย่างเห็นได้ชัด โดยไม่ขึ้นอยู่กับลักษณะของฉากว่าจะเป็นลักษณะฉากคงที่หรือฉากเคลื่อนที่ อีกทั้งยังไม่ขึ้นอยู่กับประเภทของรถ ป้ายทะเบียน รวมถึงเวลาและสภาพแสงในขณะนั้น ไม่ว่าจะเป็นในที่ร่มหรือกลางแจ้ง ดังตัวอย่างผลการทดลองที่แสดงในรูปที่ 4.3 และ 4.4 ซึ่งเป็นตัวอย่างในกรณีที่สามารถตรวจหาตำแหน่งป้ายทะเบียนได้อย่างถูกต้องจากแหล่งภาพทางเข้าออกอาคาร และแหล่งภาพถนนตามลำดับ ส่วนรูปที่ 4.5 เป็นตัวอย่างภาพในกรณีที่การตรวจหาตำแหน่งป้ายทะเบียนมีความผิดพลาด



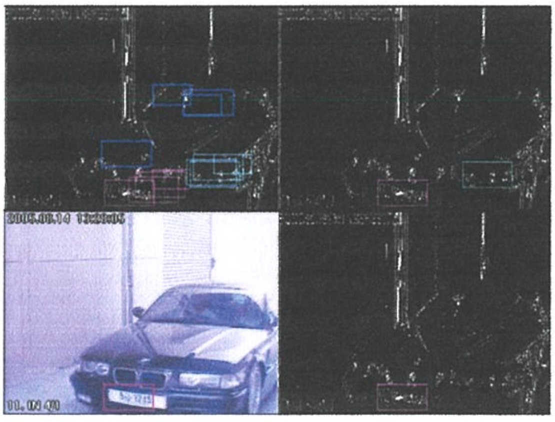
(ก)



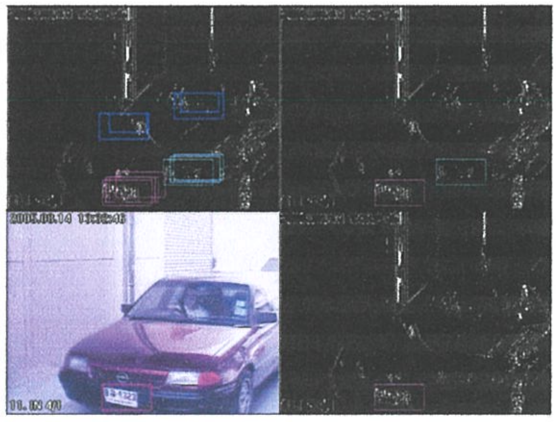
(ข)

(ค)

(ง)

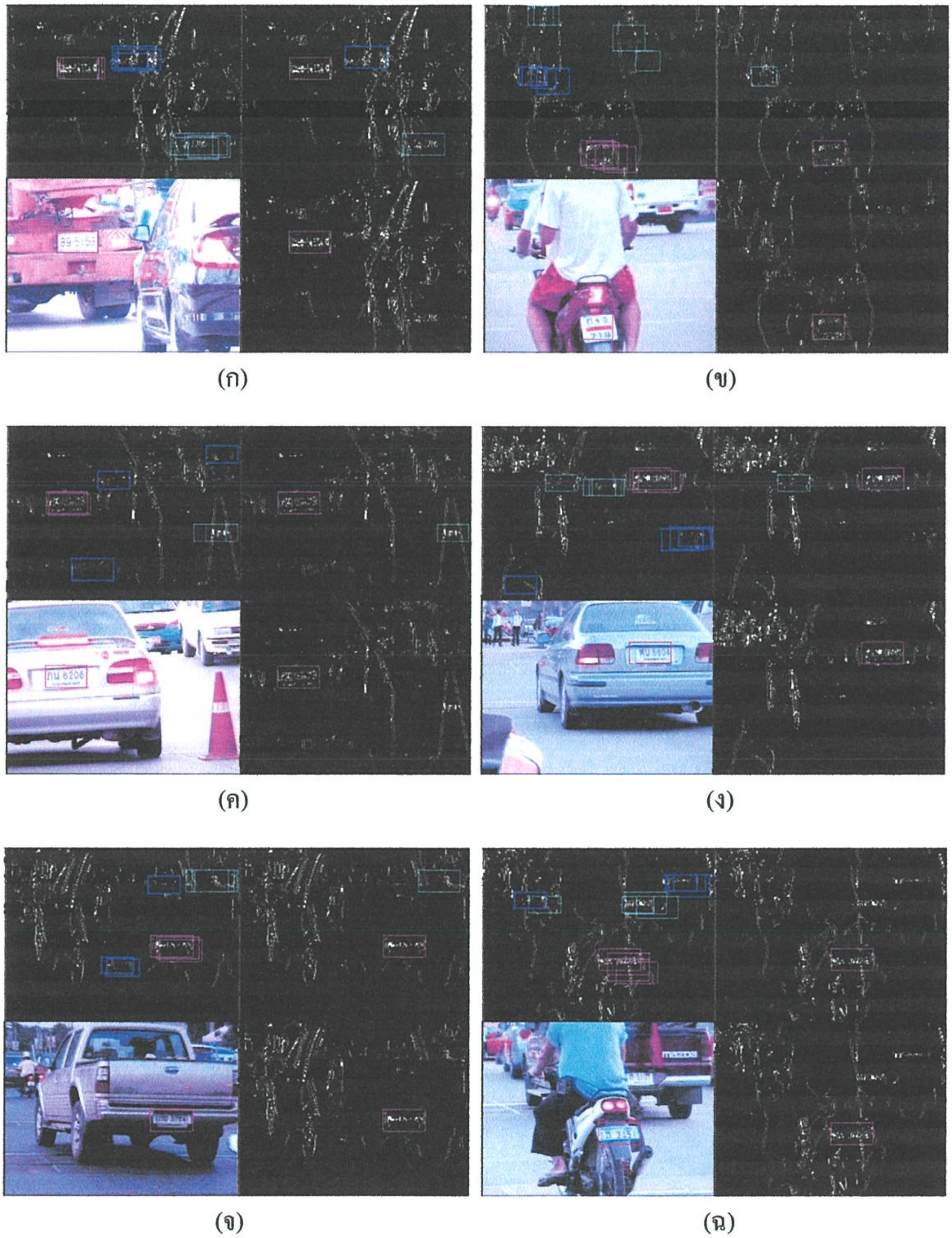


(จ)

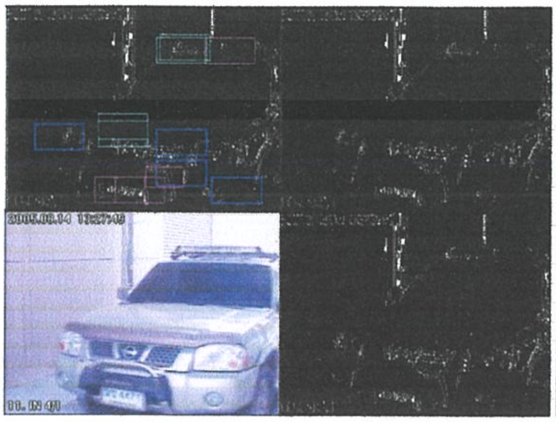


(ฉ)

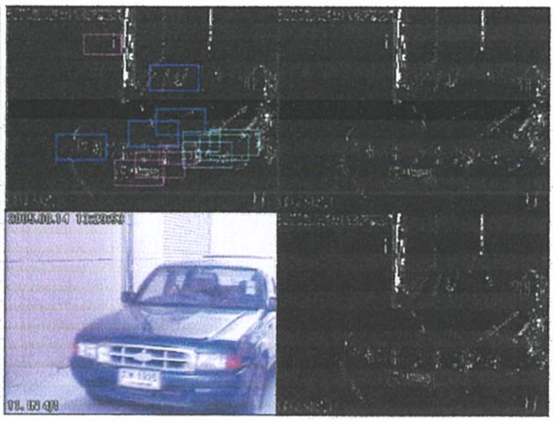
รูปที่ 4.3 ภาพตัวอย่างในกรณีที่สามารถตรวจหาตำแหน่งป้ายทะเบียนได้อย่างถูกต้องจากแหล่งภาพทางเข้าออกอาคาร (ก),(ข) ฉากที่หนึ่ง (ค),(ง) ฉากที่สอง (จ),(ฉ) ฉากที่สาม



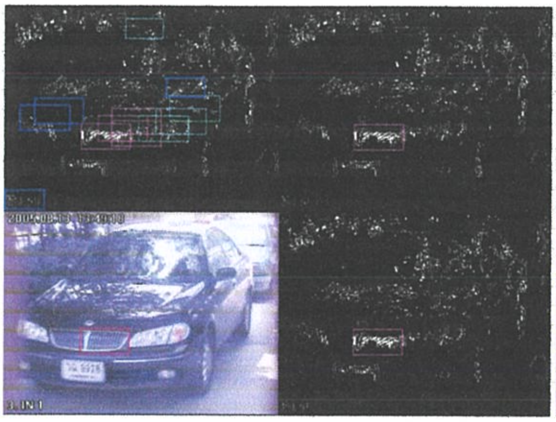
รูปที่ 4.4 ภาพตัวอย่างในกรณีที่สามารถตรวจหาตำแหน่งป้ายทะเบียนได้อย่างถูกต้องจากแหล่งภาพถนน (ก),(ข),(ค) ฉากที่หนึ่ง (ง),(จ),(ฉ) ฉากที่สอง



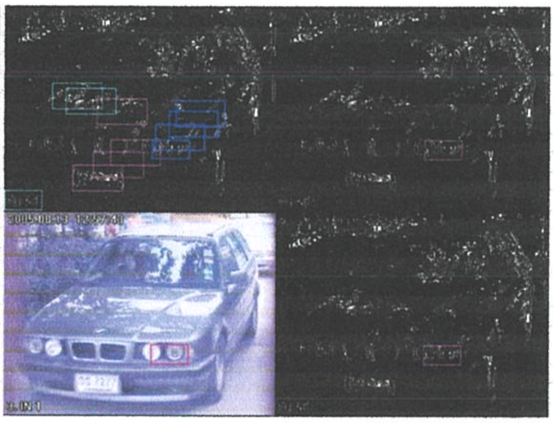
(ก)



(ข)



(ค)



(ง)

(จ)

(ฉ)

รูปที่ 4.5 ภาพตัวอย่างในกรณีที่การตรวจหาตำแหน่งป้ายทะเบียนมีความผิดพลาด

ในส่วนของความผิดพลาดในการตรวจหาตำแหน่งป้ายทะเบียนที่เกินขึ้นดังตัวอย่างในรูปที่ 4.5 นั้นเกิดจากหลายปัจจัยเช่น ในกรณีของรูปที่ 4.5 (ก)และ(ข) ในขณะที่รถมีการกระโดดอย่างเร็วขณะกำลังข้ามลูกระนาดเป็นเหตุทำให้ภาพที่บันทึกเบลอ (motion blur) จนกระทั่งไม่สามารถตรวจพบป้ายทะเบียนได้ เช่นเดียวกับในกรณีของรูปที่ 4.5 (ค)และ(ง) นั้นความเร็วของรถมีมากเกินไปจนกระทั่งภาพที่บันทึกได้เบลอมาก รวมถึงมีวัตถุอื่นที่มีลักษณะใกล้เคียงกับป้ายทะเบียนเช่นกระจังหน้าและไฟหน้าอยู่ด้วย ทำให้ระบบเข้าใจผิดคิดว่าวัตถุนั้นเป็นป้ายทะเบียน หรืออีกกรณีหนึ่งในรูปที่ 4.5 (จ)และ(ฉ) ที่รถเคลื่อนที่ออกห่างจากกล้องในทิศทางเดียวกับมุมกล้อง ทำให้ภาพที่บันทึกได้แทนที่ป้ายทะเบียนจะมีการเคลื่อนที่ไปยังตำแหน่งอื่นๆของภาพ แต่กลับไม่การเคลื่อนที่มีแต่ขนาดเล็กลงเท่านั้น ทำให้ระบบเข้าใจว่าวัตถุนั้นไม่มีการเคลื่อนที่ซึ่งจะทำให้ถูกคัดออกในขั้นตอนการวิเคราะห์ข้อมูลเชิงเวลาเนื่องจากในขั้นตอนนี้เราตั้งสมมติฐานไว้ว่าป้ายทะเบียนนั้นจะต้องมีการเคลื่อนที่แบบเชิงเส้นเมื่อเวลาผ่านไป

4.6 ผลการทดลองการปรับค่า P_{limit} เพื่อลดค่าฟอลส์โพสซิทีฟ

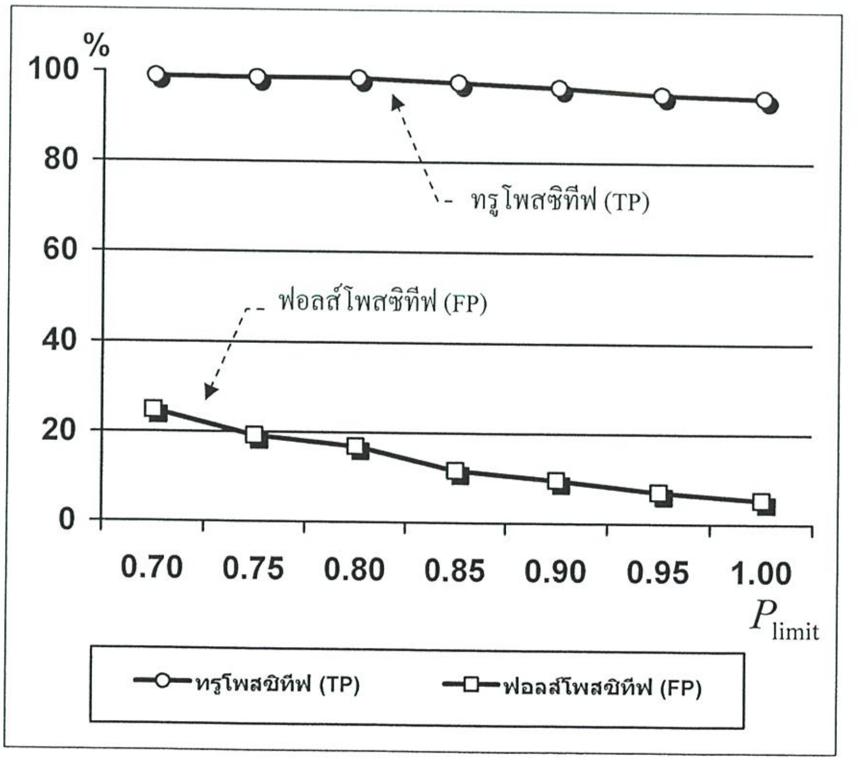
จากการทดลองข้างต้นผู้ทำวิทยานิพนธ์พบว่าเราสามารถปรับค่าฟอลส์โพสซิทีฟให้ลดลงได้แต่มีผลทำให้ค่าทรูโพสซิทีฟลดลงบ้างเล็กน้อย ซึ่งขึ้นอยู่กับงานที่จะนำไปใช้ว่าต้องการค่าฟอลส์โพสซิทีฟน้อยแค่ไหนและยอมรับให้ค่าทรูโพสซิทีฟลดลงได้เท่าใด โดยการปรับค่า P_{limit} ให้สูงขึ้น และเพื่อให้เห็นถึงผลของการปรับค่า P_{limit} ที่มีต่อค่าฟอลส์โพสซิทีฟได้อย่างชัดเจน ผู้ทำวิทยานิพนธ์ได้เพิ่มอินพุตเพื่อใช้ในการทดลองนี้ขึ้นมาอีกชุดหนึ่ง โดยใช้ภาพที่ได้จากภาพวีดิทัศน์ที่บันทึกจากทางเข้า-ออกอาคาร ในฉาก (Scene) ที่1 ซึ่งเป็นภาพรถตัวอย่างจำนวน 317 คัน มาผ่านกระบวนการหาตำแหน่งของป้ายทะเบียนเช่นเดิม แต่ใช้วิธีการวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนหรือค่า L ในการลดค่าฟอลส์โพสซิทีฟเพียงอย่างเดียว โดยทำการทดลองปรับเปลี่ยนค่า P_{limit} ตั้งแต่ 0.70 จนถึง 1.00 ซึ่งผลการทดลองออกมาดังตารางที่ 4.5 และสามารถเขียนเป็นกราฟได้ดังรูปที่ 4.6

เมื่อสังเกตจากตารางที่ 4.5 จะพบว่าวิธีการนี้เมื่อเราปรับค่า P_{limit} ให้เป็น 0.70 แล้วผลปรากฏว่าสามารถหาแคนดิเดทวินโดว์ได้ทั้งหมด 418 วินโดว์ โดยเป็นป้ายทะเบียนที่แท้จริงถึง 314 วินโดว์คิดเป็น 99% จากป้ายทะเบียนทั้งหมด และวินโดว์ที่เหลือ 104 วินโดว์ไม่ใช่ป้ายทะเบียนคิดเป็น 24.9% ของแคนดิเดทวินโดว์ทั้งหมด มีเพียง 3 ป้ายทะเบียนเท่านั้นที่ไม่สามารถตรวจพบได้หรือคิดเป็นเพียง 1% จากจำนวนป้ายทะเบียนทั้งหมด ด้วยวิธีการนี้เราสามารถลดค่าฟอลส์โพสซิทีฟเพื่อให้เหมาะสมกับงานที่นำไปใช้ได้อีก โดยการปรับค่า P_{limit} ซึ่งอาจจะลดความถูกต้องในการตรวจสอบลงบ้างเล็กน้อย ดังเช่นเมื่อเราปรับค่า P_{limit} ไปที่ 1.00 เราสามารถลดค่าฟอลส์โพสซิทีฟลงได้เหลือเพียง 5.4% ขณะที่ความถูกต้องในการตรวจสอบลดลงมาจากเดิมเล็กน้อยมาอยู่ที่ 94.6%

ซึ่งจากกราฟในรูปที่ 4.6 แสดงให้เห็นว่าเมื่อเราเพิ่มค่า P_{limit} จาก 0.70 จนถึง 1.00 ค่าฟอลส์โพสซิทีฟจะลดลงอย่างเห็นได้ชัดในขณะที่ค่าความถูกต้องในการตรวจสอบหรือค่าทรูโพสซิทีฟลดลงเพียงเล็กน้อยเท่านั้น

ตารางที่ 4.5 ผลการหาตำแหน่งของป้ายทะเบียนและลดค่าฟอลส์โพสซิทีฟด้วยวิธีการวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนเพียงอย่างเดียวที่ค่า P_{limit} ต่างๆ

P_{limit}	ตรวจพบป้ายทะเบียน (ทรูโพสซิทีฟ)	ตรวจไม่พบป้ายทะเบียน (ฟอลส์เนกกาทีฟ)	ตรวจพบแต่ไม่ใช่ป้ายทะเบียน (ฟอลส์โพสซิทีฟ)
0.70	314/317 (99.0%)	3/317 (1.0%)	104/418 (24.9%)
0.75	313/317 (98.7%)	4/317 (1.3%)	75/388 (19.3%)
0.80	313/317 (98.7%)	4/317 (1.3%)	64/377 (17.0%)
0.85	310/317 (97.8%)	7/317 (2.2%)	41/351 (11.7%)
0.90	307/317 (96.8%)	10/317 (3.2 %)	33/340 (9.7%)
0.95	302/317 (95.3%)	15/317 (4.7%)	23/325 (7.1%)
1.00	300/317 (94.6%)	17/317 (5.4%)	17/317 (5.4%)



รูปที่ 4.6 กราฟแสดงผลการทดลองการลดค่าฟอลส์โพสซิทีฟด้วยวิธีการวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนที่ค่า P_{limit} ต่างๆ ตั้งแต่ 0.70 จนถึง 1.00

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

จากจำนวนยานพาหนะที่เพิ่มขึ้นอย่างรวดเร็วในปัจจุบัน ทำให้จำเป็นต้องมีการควบคุม จัดระเบียบกิจกรรมต่างๆของยานพาหนะ เพื่อให้ไม่เกิดปัญหาทั้งความเสียหายที่เกิดจากอุบัติเหตุ และเวลาที่สูญเสียไปจากสภาพจราจรติดขัด ซึ่งการจะควบคุมกิจกรรมต่างๆนั้น ต้องอาศัยการระบุตัวตนของยานพาหนะแต่ละคันเสียก่อน และวิธีการที่ใช้ระบุตัวตนของยานพาหนะในปัจจุบันก็คือการกำหนดหมายเลขทะเบียนให้แก่ยานพาหนะแต่ละคัน โดยกำหนดให้มีป้ายทะเบียนติดอยู่ที่ด้านหลังและ/หรือด้านหน้ายานพาหนะเหล่านั้น

ปัจจุบันนี้กิจกรรมต่างๆที่ต้องอาศัยป้ายทะเบียนยานพาหนะมีมากมายไม่ว่าจะเป็น ระบบรักษาความปลอดภัยในการควบคุมการเข้าออกอาคาร การควบคุมการปฏิบัติตามการจราจร การควบคุมจัดระเบียบการจอดรถ เป็นต้น ซึ่งส่วนใหญ่จะต้องใช้มนุษย์ในการอ่านป้ายทะเบียนเหล่านั้น ประกอบกับการที่ระบบการรักษาความปลอดภัยในปัจจุบันเป็นสิ่งที่หลายหน่วยงานองค์กรให้ความสำคัญมากขึ้น จึงมีการติดตั้งกล้องวงจรปิดไว้บริเวณต่างๆ เพื่อใช้ในระบบรักษาความปลอดภัย ด้วยเหตุนี้เองทำให้เกิดความคิดว่าหากเราสามารถนำเอากล้องวงจรปิดที่มีติดตั้งอยู่ทั่วไป มาทำงานร่วมกับระบบการประมวลผลภาพดิจิตอลเพื่อเพิ่มความสามารถในการระบุตัวตนหรือการอ่านป้ายทะเบียนของยานพาหนะแต่ละคันได้ ก็จะเป็นประโยชน์ไม่น้อย เนื่องจากไม่จำเป็นต้องลงทุนติดตั้งอุปกรณ์อะไรเพิ่มเติมมากนัก เพราะทั้งกล้องวงจรปิดรวมถึงป้ายทะเบียนยานพาหนะนั้น ล้วนแต่เป็นสิ่งที่มีใช้กันอยู่ทั่วไปอยู่แล้ว เพียงแต่เพิ่มระบบคอมพิวเตอร์และซอฟท์แวร์ที่ใช้ในการประมวลผลเท่านั้น

ระบบการอ่านป้ายทะเบียนยานพาหนะนั้น สามารถแบ่งออกเป็นสองระบบงานย่อยด้วยกันได้แก่ ระบบตรวจหาตำแหน่งป้ายทะเบียน ซึ่งทำหน้าที่ในการแยกส่วนของภาพที่เป็นตำแหน่งของป้ายทะเบียนออกจากส่วนอื่นของภาพที่ได้รับเข้ามา จากนั้นจึงส่งต่อส่วนของภาพบริเวณที่เป็นป้ายทะเบียนนั้นให้แก่ระบบงานย่อยที่สองก็คือระบบการอ่านป้านทะเบียนที่จะทำหน้าที่ในการแปลความหมายจากภาพป้ายทะเบียนว่ายานพาหนะคันนั้นมีหมายเลขทะเบียนอะไร

ในวิทยานิพนธ์ฉบับนี้ได้นำเสนอระบบตรวจหาตำแหน่งของป้ายทะเบียนยานพาหนะ ซึ่งเป็นส่วนหนึ่งของระบบการอ่านป้ายทะเบียน โดยใช้วิธีการตรวจหาขอบ ที่ใช้จุดเริ่มเปลี่ยนชนิดเปลี่ยนแปลงได้ซึ่งเป็นวิธีที่ทนทานต่อสภาพการเปลี่ยนแปลงของแสงทั้งเวลากลางวันและกลางคืน โดยตั้งสมมุติฐานว่าบริเวณส่วนที่เป็นป้ายทะเบียนนั้นจะเกิดขอบเป็นจำนวนมาก ซึ่งเกิดจากความ

แตกต่างกันของความเข้มของสีระหว่างพื้นหลังและตัวอักษรบนป้ายทะเบียน ในขณะเดียวกันขอบที่เกิดขึ้นนั้นจะมีการกระจายตัวในรูปแบบที่แตกต่างจากสภาพแวดล้อมอื่นๆของภาพ ทำให้สามารถใช้ข้อสมมติฐานนี้ ในการแยกแยะส่วนที่เป็นป้ายทะเบียนออกจากส่วนอื่นๆ ของภาพได้ โดยการสแกนภาพด้วยวินโดว์ที่มีขนาดใกล้เคียงกับป้ายทะเบียนตลอดทั้งภาพ แล้วทำการออกแบบวิธีการหาค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนของวินโดว์ เพื่อใช้ในการประเมินว่าลักษณะการกระจายตัวของขอบภายในวินโดว์นั้นมีความใกล้เคียงกับลักษณะของป้ายทะเบียนมากน้อยเพียงใด ซึ่งวินโดว์ที่มีค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนมากที่สุดสามอันดับแรกจะถูกคัดเลือกให้เป็นแคนดิเดทวินโดว์หรือวินโดว์ที่ถูกคาดหมายว่าจะเป็นป้ายทะเบียนนั่นเอง อย่างไรก็ตามวิธีการนี้ถึงแม้จะให้ผลการตรวจหาตำแหน่งป้ายทะเบียนได้ดี แต่ก็ยังมีข้อเสียคือ จะเกิดฟอลส์โพสซิทีฟจำนวนมาก ซึ่งก็คือวินโดว์ที่ไม่ใช่ป้ายทะเบียนที่แท้จริงแต่ถูกคัดเลือกให้เป็นแคนดิเดทวินโดว์หรือถูกคาดหมายว่าจะเป็นป้ายทะเบียนติดมาด้วย ในวิทยานิพนธ์ฉบับนี้จึงได้เสนอวิธีการในการลดค่าฟอลส์โพสซิทีฟที่เกิดขึ้น ด้วยการวิเคราะห์ข้อมูลเชิงเวลา เพื่อคำนวณหาความเร็วและความเร่งของแต่ละแคนดิเดทวินโดว์ โดยมีสมมติฐานว่าในช่วงเวลาสั้นๆระหว่างแต่ละเฟรมของภาพวีดิทัศน์ป้ายทะเบียนจะต้องมีการเคลื่อนที่แบบเชิงเส้นด้วยความเร็วคงที่ กล่าวคือมีความเร็วไม่เป็นศูนย์ในขณะที่มีความเร่งเป็นศูนย์ ดังนั้นแคนดิเดทวินโดว์ใดที่มีคุณสมบัติในการเคลื่อนที่ผิดจากคุณสมบัติดังกล่าวจะถูกคัดทิ้งออกไป แต่อย่างไรก็ตามจากการทดลองพบว่าด้วยวิธีการนี้จำนวนฟอลส์โพสซิทีฟที่เหลือยังคงสูงอยู่ ดังนั้นในวิทยานิพนธ์ฉบับนี้จึงได้เพิ่มกระบวนการที่ใช้ในการคัดกรองแคนดิเดทวินโดว์เพิ่มอีกชั้นหนึ่งคือ ขั้นตอนการวิเคราะห์ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียน โดยจากการสังเกตพบว่า ค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนของแคนดิเดทวินโดว์ที่เป็นป้ายทะเบียนที่แท้จริงนั้น มักจะมีค่ามากที่สุดหรือใกล้เคียงกับค่ามากที่สุดจากแคนดิเดทวินโดว์ทั้งหมดของภาพนั้น ดังนั้นถ้าแคนดิเดทวินโดว์ใดที่มีค่าความเป็นไปได้ที่จะเป็นป้ายทะเบียนห่างกับค่าที่มากที่สุดเกินกว่าขอบเขตที่กำหนด แคนดิเดทวินโดว์นั้นก็จะถูกคัดทิ้งออกไป ด้วยวิธีการนี้ทำให้จำนวนฟอลส์โพสซิทีฟที่เหลือสามารถลดลงได้จนเป็นที่น่าพอใจ

ผลการทดลองของวิธีการที่นำเสนอในวิทยานิพนธ์ฉบับนี้แสดงให้เห็นว่า วิธีการนี้สามารถหาตำแหน่งของป้ายทะเบียนยานพาหนะได้อย่างถูกต้องสูงถึง 98% จากจำนวนป้ายทะเบียนทั้งหมด 440 ป้าย ในขณะที่ค่าฟอลส์โพสซิทีฟที่เคยสูงถึง 887 ป้าย ก็สามารถถูกทำให้ลดลงเหลือเพียง 74 ป้ายเท่านั้น หรือคิดเป็น 8.3 % จากเดิม นอกจากนี้ยังสามารถปรับค่าคงที่บางตัวเพื่อลดหรือเพิ่มค่าความถูกต้องและค่าฟอลส์โพสซิทีฟได้อีกด้วย โดยสามารถลดค่าฟอลส์โพสซิทีฟด้วยการยอมให้ค่าความถูกต้องลดลงบ้างเล็กน้อย ทั้งนี้ขึ้นอยู่กับงานที่นำไปใช้ว่าต้องการค่าฟอลส์โพสซิทีฟเท่าใดและสามารถยอมรับให้ค่าความถูกต้องนั้นลดลงได้เท่าไร

อย่างไรก็ตามยังคงมีความผิดพลาดในการตรวจสอบเกิดขึ้น ทั้งนี้เกิดจากหลายสาเหตุด้วยกันเช่น การเบลอจากการเคลื่อนไหว ความไม่ชัดเจนที่เกิดจากสภาพอากาศเช่นฝนตกหนัก

ความสกปรกหรือการฉีกขาดของป้ายทะเบียน รวมถึงมีวัตถุอื่นที่มีลักษณะใกล้เคียงกับป้ายทะเบียนมากอยู่ในภาพนั้นด้วยเป็นต้น ทำให้วัตถุอื่นถูกมองว่าเป็นป้ายทะเบียน อีกทั้งยังมีข้อจำกัดในกรณีที่รถเคลื่อนที่ออกห่างจากกล้องในแนวเดียวกับมุมกล้อง ทำให้ดูคล้ายกับว่ารถไม่มีการเคลื่อนที่ และกรณีที่รถหยุดนิ่งหรือการจราจรติดขัดทำให้ป้ายทะเบียนไม่มีการเคลื่อนที่ การตรวจสอบในส่วนของการวิเคราะห์ข้อมูลเชิงเวลาอาจทำให้ป้ายทะเบียนนั้นถูกคัดออกจากการเป็นแคนดิเดทวินโดว์

5.2 ข้อเสนอแนะ

ในกรณีฟอลส์โพสซิทีฟที่เกิดขึ้นและหลงเหลืออยู่ในขั้นตอนสุดท้ายหลังผ่านกระบวนการลดค่าฟอลส์โพสซิทีฟแล้ว หากนำผลที่ได้ไปผ่านขั้นตอนของการอ่านป้ายทะเบียนก็จะทำให้ฟอลส์โพสซิทีฟส่วนนี้ถูกตัดออกไปได้อีกทางหนึ่ง

ส่วนในการพัฒนาขั้นต่อไปอาจจะมีการปรับแก้การเบลอจากการเคลื่อนไหวของภาพต้นฉบับเสียก่อนที่จะนำมาผ่านกระบวนการวิเคราะห์ รวมถึงการปรับแก้ไขภาพในกรณีที่สภาพอากาศไม่ดีก่อนด้วยเช่นกัน นอกจากนี้ควรมีการปรับปรุงเงื่อนไขในการวิเคราะห์ข้อมูลเชิงเวลาเพื่อให้สามารถลดค่าฟอลส์โพสซิทีฟได้อย่างมีประสิทธิภาพมากยิ่งขึ้นเพื่อให้ครอบคลุมถึงในกรณีที่รถไม่มีการเคลื่อนที่หรือดูคล้ายกับไม่มีการเคลื่อนที่เป็นต้น

เอกสารอ้างอิง

[1] Rafael C. Gonzalez and Richard E. Woods and Steven L. Eddins. **Digital Image Processing Using MATLAB®.** New Jersey : Pearson Prentice Hall. 2004

[2] ยุทธนา ลีลาศวัฒนกุล. **คู่มือการเขียนโปรแกรมวินโดว์ขั้นสูงด้วย Visual C++ .NET Episode One.** กรุงเทพฯ : หจก.ไทยเจริญการพิมพ์. 2546

[3] Rafael C. Gonzalez and Richard E. Woods. **Digital Image Processing.** 2nd ED. New Jersey: Pearson Prentice Hall. 2002

[4] Hongliang Bai, Junmin Zhu, and Changping Liu. "A Fast License Plate Extraction Method on Complex Background." **Proceeding of IEEE Intelligent Transportation Systems.** vol.2, 2003. pp. 985–987

[5] Xu Jianfeng, Li Shaofa and Chen Zhibin. "Color Analysis For Chinese Car Plate Recognition." **Proceeding of IEEE International Conference on Robotics, Intelligent Systems and Signal Processing.** vol.2, 2003. pp. 1312-1316

[6] วุฒิพงศ์ อารีกุล, ชัยสิทธิ์ ศิริเจริญไชย. "การหาตำแหน่งป้ายทะเบียนรถยนต์โดยการตรวจสอบขอบชัน" **การประชุมวิชาการทางวิศวกรรมไฟฟ้าครั้งที่ 24.** เล่มที่ 2, 2544, หน้า 1176-1181

[7] Hsi-Jian Lee, Si-Yuan Chen and Shen-Zheng Wang. "Extraction and recognition of license plates of motorcycles and vehicles on highways." **Proceeding of 17th International Conference on Pattern Recognition.** vol.4, 2004 pp. 356-359

[8] Tran Duc Duan, Duong Anh Duc and Tran Le Hong Du. "Combining Hough transform and contour algorithm for detecting vehicles' license-plates." **Proceeding of International Symposium on Intelligent Multimedia, Video and Speech Processing.** 2004. pp. 747–750

[9] Yoshimori S., Mitsukura Y., Fukumi M., Akamatsu N. and Khosal R. "License plate detection system in rainy days." **Proceeding of IEEE Computational Intelligence in Robotics and Automation.** vol.2, 2003. pp. 972-976

[10] Shyang-Lih Chang, Li-Shien Chen, Yun-Chung Chung and Sei-Wan Chen. "Automatic license plate recognition." **IEEE Trans. Intelligent Transportation Systems.** vol.5, Issue 1. 2004. pp. 42-53

ภาคผนวก

ภาคผนวก

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

1. Preemon Rattanathammawat, Thanarat H. Chalidabhongse. "A Car Plate Detector using Edge Information." **Proceeding of IEEE International Symposium on Communication and Information Technologies .**, 2006.
2. Thanarat H. Chalidabhongse, Preemon Rattanathammawat. "License Plate Localization of Moving Vehicles in Complex Scene." **Proceeding of 8th IEEE Asia Pacific Conference on Circuits and Systems.**, 2006.


ISCIT 2006
October 18-20, 2006
Grand Mercure Fortune Hotel, Bangkok, Thailand
ABSTRACTS
International Symposium on Communications
And Information Technologies 2006
ISCIT 2006
IEEE
IEEE CAS Society
NECTEC
ECTI

	important issues which need to be taken into consideration. The complexity of detection can be reduced with S-Rake and P-Rake. Especially, P-Rake can reduce complexity greatly because no channel estimation is needed. Simulation results show that the TH-BPSK system achieves better performance than the TH-BPPM system.

F3A Video/Image Processing III

F3A-1	**Novel Adaptive Temporal Filter Based On Motion Compensation for Video Noise Reduction**	
	Yan Li	(*Chinese Academy of Sciences*, China, liyan790701@eyou.com)
	YanFeng Qiao	(*Chinese Academy of Sciences*, China, doctorli1979@163.com)
	Noise in video signals will degrades the image quality. In this paper, a novel adaptive temporal filter based on motion compensation is proposed for noise reduction. The noise in video sequence is tracked by motion compensation in temporal domain. The filtering strength can be adaptive changed on object motion situation. The experiment results show that the proposed filter is better than the filter in spatial domain. The noise in video signals can be effectively reduced and the image will not be blurry as the result of spatial filter.	
F3A-2	**Acclerate a Dlt Motion Capture System With Quad-Tree Searching Scheme**	
	Thanapong Chaichana	(*King Mongkut's Institute of Technology Ladkrabang*, Thailand, venom25@gmail.com)
	Manas Sangworasil	(*King Mongkut's Institute of Technology Ladkrabang*, Thailand, venom25@gmail.com)
	Chuchart Pintavirooj	(*King Mongkut's Institute of Technology Ladkrabang*, Thailand, venom25@gmail.com)
	Sorapong Aootaphao	(*King Mongkut's Institute of Technology Ladkrabang*, Thailand, venom25@gmail.com)
	This paper describes a 3D motion capture system to generate a skeleton model representing the human body. The system consists of a number of digital cameras placed at arbitrary position around the subject. The subject's joints to be measured are attached with illuminated landmarks. The 3D dynamic information of these landmarks is then detected using the Direct Linear Transform (DLT) technique. In order to establish corresponding point of the landmarks across individual camera as required in the DLT procedure, the quad-tree searching scheme has been adopted which can speed up the dynamic 3D modelling process of up to 70 percent. The 3D motion capture system has been performed to visualize the motion of various subjects. The result is very promising	
F3A-3	**A Car Plate Detector using Edge Information**	
	Preemon Rattanathammawat	(*King Mongkut's Institute of Technology Ladkrabang*, Thailand, s4067431@kmitl.ac.th)
	Thanarat H.	(*King Mongkut's Institute of Technology*

0-7803-9740-1/06/$20.00 c2006 IEEE

	Chalidabhongse	*Ladkrabang*, Thailand, thanarat@it.kmitl.ac.th)
	This paper describes a new edge-based car plate detection technique to localize and extract car license plates in complex scene. The system works for both moving and stationary vehicles in compressed low resolution video streams obtained from real working environments. To cope with the varied illumination condition, which typically occurs in outdoor environment, the method works on the edge information of the image. We employs vertical Sobel edge detector with automatic variable thresholding. A moving window is then scanned through the edge image. For each location of moving window, we consider the distribution of edge pixels to determine whether it is likely to be a license plate. In the last step, we employ the temporal analysis to suppress false detections. In experiments, we tested our method on several compressed video streams containing 317 moving vehicles in varied illumination conditions from morning to night time. The license plate localization has 94% precision rate.	
F3A-4	**A Lorentzian Bayesian Approach for Robust Iterative Multiframe Super-Resolution Reconstruction with Lorentzian-Tikhonov Regularization**	
	Vorapoj Patanavijit	(*Assumption University*, Thailand, Patanavijit@yahoo.com)
	Somchai Jitapunkul	(*Chulalongkorn University*, Thailand, Somchai.j@chula.ac.th)
	Recently, it has seen a great deal of work in the development of algorithms addressing the problem of super-resolution. Although many such algorithms have been proposed, the almost SRR (Super-Resolution Reconstruction) estimations are based on L1 or L2 statistical norm estimation thus these SRR methods are usually very sensitive to their assumed model of data and noise that limits their utility. This paper reviews some of these SRR estimator methods and addresses their shortcomings. We propose a novel SRR approach based on the stochastic regularization technique of Bayesian MAP estimation by minimizing a cost function. The Lorentzian norm [10] is used for measuring the difference between the projected estimate of the high-resolution image and each low resolution image, removing outliers in the data and Tikhonov regularization and Lorentzian Tikhonov regularization are used to remove artifacts from the final answer and improve the rate of convergence. The experimental results confirm the effectiveness of our method and demonstrate its superiority to other super-resolution methods based on L1 and L2 norm for a several noise models such as noiseless, Additive White Gaussian Noise (AWGN) and Salt & Pepper Noise.	

F3B Computer System and Applications

F3B-1	**Self-Management Considerations in Designing a Pervasive Multimodal Multimedia Computing System**	
	Manolo Dulva Hina	(*École de Technologie Supérieure*, Canada, chbdz1@yahoo.ca)
	Chakib Tadj	(*École de Technologie Supérieure*, Canada, ctadj@ele.etsmtl.ca)
	Amar Ramdane Cherif	(*Université de Versailles-Saint-Quentin-en-Yvelines*, France , acherif@hotmail.com)
	The ever-growing complexity of information system demands that the system or its	

A Car Plate Detector using Edge Information

Preemon Rattanathammawat and Thanarat H. Chalidabhongse*
Faculty of Information Technology
King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand
Tel: +66-2-737-2551, Fax: +66-2-326-4332
*E-mail: thanarat@it.kmitl.ac.th

***Abstract*— This paper describes a new edge-based car plate detection technique to localize and extract car license plates in complex scene. The system works for both moving and stationary vehicles in compressed low resolution video streams obtained from real working environments. To cope with the varied illumination condition, which typically occurs in outdoor environment, the method works on the edge information of the image. We employs vertical Sobel edge detector with automatic variable thresholding. A moving window is then scanned through the edge image. For each location of moving window, we consider the distribution of edge pixels to determine whether it is likely to be a license plate. In the last step, we employ the temporal analysis to suppress false detections. In experiments, we tested our method on several compressed video streams containing 317 moving vehicles in varied illumination conditions from morning to night time. The license plate localization has 94% precision rate.**

I. INTRODUCTION

Automatic license plate recognition (LPR) has a number of applications such as vehicle access control, automatic toll collection, traffic law enforcement, and parking area security and management. A number of methods have been proposed [1-10]. Sirithinaphong and Chamnongthai [2] proposed a license plate recognition system for automatic parking using four layer backpropagation neural networks. Naito et al. [3] proposed a new sensing system with wide dynamic range by using two CCDs and a prism to split an incident ray into two lights with different intensities. Jianfeng, Shaofa, and Zhibin [5] proposed a color analysis method for Chinese car license plate recognition. The neural network is used as a classifier to identify pixel color. Yoshimori et al. [6] proposed a threshold determination method in various backgrounds using the real coded genetic algorithm. Chang et al. [7] proposed a system that can locate license plates and identify the license numbers. They employed soft computing techniques rooted in fuzzy and neural disciplines to compensate for uncertainties caused by noises and measurement errors. Duan, Duc, and Du [8] proposed a LPR system for Vietnamese license plates using Hough transform and contour algorithm. Lee, Chen, and Wang [9] recently proposed a system that can extract and recognize license plates of motorcycles and vehicles on highways. They employ block different method for moving object detection, and extract the license plate using a projection-based method. The license recognition is done using three statistical features: direction counts, crossing counts, and peripheral background areas.

Many previous works tested on images captured under restricted conditions, such as parking vehicles, fixed illumination, fine resolution, designated environment, and clean background. In this research, we propose a method that works on video streams obtained from real working environments with a few restrictions. The illumination conditions can be varied which typically occurs in outdoor environment. The vehicles might be moving or stationary, and the background scene can be complex. Our method is based on edge information that is robust to illumination changes. It is also low complexity which is suitable for many real-time applications. The method consists of three main steps. First, vertical Sobel edge detector with variable thresholding is applied to the input image. Then, the edge image is scanned by a predefined size window. For each window location, the edge pixel distribution inside the window is analyzed to determine whether it is likely to be a license plate. In the last step, we employ the temporal analysis to remove the false positives.

This paper is organized as follows. In Section II, the proposed method is described step by step. Experimental results are presented in Section III. Concluding remarks and ideas for future work are given in Section IV.

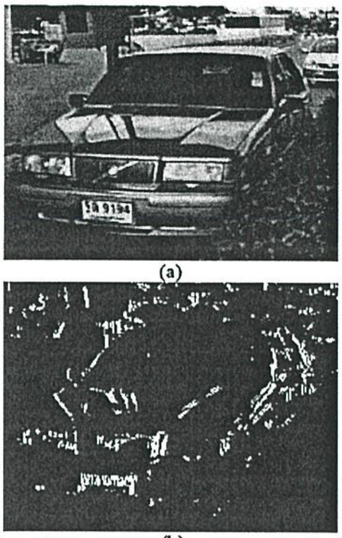
II. THE PROPOSED ALGORITHM

As mentioned, our proposed method consists of three main steps: a special edge detection, license plate localization based on distribution of edge pixels, and false detection suppression using temporal analysis

A. Edge Detection

To cope with various illumination conditions and various license plate colors, we work on edge domain. Vertical Sobel edge detector is used in this work since we noticed that license plate area is likely to contain more vertical edges than other areas due to the high contrast between alphabets and plate background color.

To make the system robust to illumination changes, we employ variable thresholding for edge detection using threshold calculation method appeared in [4]. Fig. 1 shows the result of the described edge detection.



(a)

(b)

Fig. 1. Vertical edge detection. (a) Original image, and (b) Edge image with variable threshold

B. *License plate localization*

To determine location of license plates in the images, we consider edge pixel distribution inside a moving window. The window size is defined based on the aspect ratio of the license plate, and approximate distance between camera and the capture area. We notice that the license plate areas usually contain dense edges and have a certain pattern of distribution. Thus, we move the window over the image and, at each location; we consider the number of edge pixels inside and its distribution. Fig.2 illustrates how the scanning window works. In the figure, the moving window size is 45x24 pixels, and it skips every 5 pixels horizontally and 3 pixel vertically. At each location, we count the number of edge pixels inside the moving window and also consider its distribution.

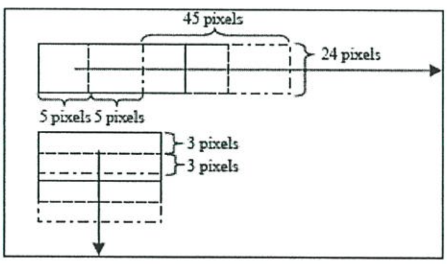
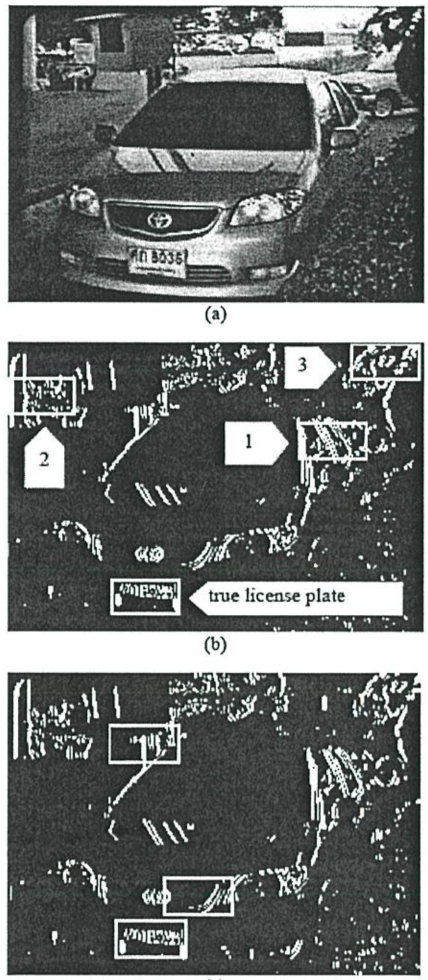


Fig. 2. Scanning image with window size 45x24, and skipping every 5 pixels horizontally and 3 pixels vertically.

Fig.3a shows an image of a car with noisy background scene. In its edge image shown in Fig.3b, there are many dense edge areas. The first three maximum dense edge areas are shown and ordered by its rank. The true license plate area, which is in the bottom of the image, contains less number of edges. Thus, using only the dense edges condition is not enough, we also consider its distribution. As we can notice, edges in license plate area are likely to cluster in the middle of plate. Fig.4 shows the vertical histograms of the windows in Fig.3b.



(a)

(b)

(c)

Fig. 3. (a) Original image, (b) Edge image with three maximum dense edge windows (c) Candidate license windows after considering the vertical distribution of edge pixels.

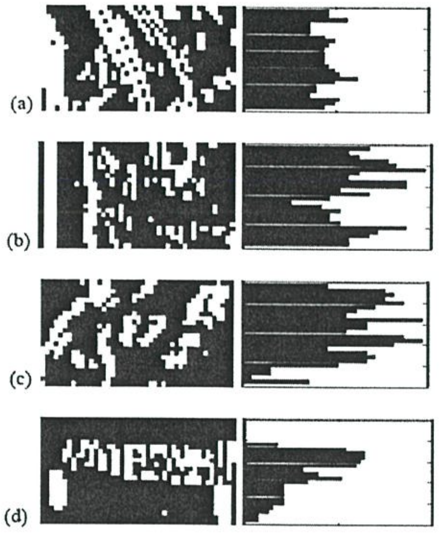


Fig. 4. (a)(b)(c) Nois edge window and its horizontal projection , (d) Edge of true license plate area and its horizontal projection

In considering the edge distribution, we divide the window vertically into three parts shown in Fig.5. We then count the number of edge pixels in the middle area (B) and subtract it with the sum of numbers of edge pixels in the peripheral areas (A and C). The result is used to specify the likelihood of being a license plate area. Let define this likelihood as L and it is given as

$$L = \beta - \alpha - \chi \qquad (1)$$

where α, β, and χ are numbers of edge pixels in areas A, B, and C of the moving window respectively. Then, we rank the locations of moving window based on this L values. Fig3c shows the first three window locations with highest L. They are selected as license plate candidates.

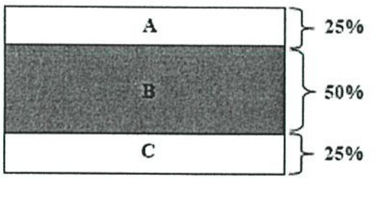


Fig. 5. The moving window is divided into three parts and used in computing the pattern of edge distribution.

C. *False detection suppression using temporal analysis*

To suppress false license plate detections from the previous step, we utilize temporal information. Base on the assumption that the car is moving, the detected license plate would move accordingly. Then, the real license plate would have a certain motion pattern. In this research, we use four consecutive frames for analyzing the license plate. We employ steps explained above for each frame to obtain candidate license plate locations. Then, each candidate window between two adjacent frames will be matched to find the corresponding windows. All four frames are performed to obtain the motion pattern of all candidates. The window matching score (S) is defined as

$$S = \left[\frac{\sum_{i=1}^{m} \left| H_i^C - H_i^P \right|}{m \times n} \right] + \left[\frac{(x_C - x_P)^2 + (y_C - y_P)^2}{(N-n)^2 + (M-m)^2} \right] \qquad (2)$$

where H_i^C and H_i^P are numbers of edge pixels in row i of candidate windows in current and previous frames orderly, x_C, y_C is coordinate of top-left corner of candidate window in current frame, x_P, y_P is coordinate of top-left corner position of candidate window in previous frame, M and N are numbers of row and column of image frame, m and n are numbers of row and column of candidate window. In our experiment, M=240, N=320, m=24 and n=45.

As we can see from (2), the window matching score is based on the similarity of horizontal projection of the candidate windows, and the proximity between them. Fig.6 shows a sequence of four image frames and matching result. Once matching is done, we then compute motion of each candidate. The velocity and acceleration are given as

$$v_{ij} = (x_j - x_i)^2 + (y_j - y_i)^2 \qquad (3)$$

$$a_{ik} = [(x_k - x_j) - (x_j - x_i)]^2 + [(y_k - y_j) - (y_j - y_i)]^2 \qquad (4)$$

where i, j, and k are the frame number ordered from most recent to former frames respectively, v_{ij} is velocity of motion from frame j to frame i, a_{ik} is acceleration of motion from frame k to frame i, (x,y) is position of top-left corner of the window. We ignore square root to reduce calculation time.

We have assumption that the car move linearly and constantly in short period of time (within four consecutive frames). By using this temporal analysis we can discriminate false alarms from the real license plates. Fig.7 shows locations of corresponding candidate windows laid out on the most current frame image. As shown in the figure, the motion of candidate "a" (see a1-a4) is linearly consistent while the motion of "b" (see b1-b4) is not. Thus "b" is false alarm. For "c", the motion is almost zero which is contradict to our knowledge that the car is moving, thus it is classified as a false alarm. Only "a" is detected as license plate for this sample image sequence.

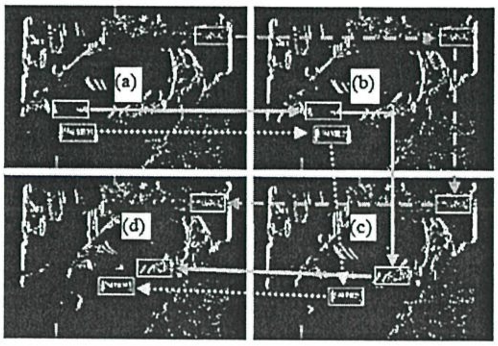


Fig. 6. Corresponding window matching of candidate license plate windows among four consecutive frames (a-d).

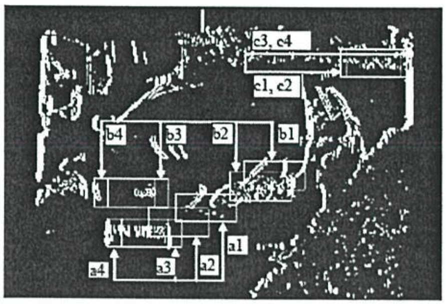


Fig. 7. Candidate windows and their motion paths

III. EXPERIMENTAL RESULTS

To evaluate the proposed method, we tested the system on compressed video streams recorded from real working environment of a shopping center in Bangkok. The shopping center installed CCTV cameras near entrance areas. There are no obstacles at the entrances, thus, the vehicles passing entrance are moving. The videos are compressed and recorded into media for future uses such as use in investigation if there is any incident occurs. The resolution of the video is 320x240 with 4 fps frame rate.

For this experiment, the streams contain 317 moving vehicles in varied illumination conditions from morning to night time. Fig.8 shows some sample images of the input and detection results on edge images.

TABLE I
LICENSE PLATE DETECTION RESULTS

True Positive (Correctly detected)	298/317 (94%)
False Negative (Miss)	19/317 (6%)
False Positive (False Alarm)	206/504 (41%)

Table 1 shows the detection results. There were 317 license plates in the tested videos. Our system detected 504 candidate areas for license plates. Among these 504 areas, there were 298 true license plates, while the other 206 areas were false alarms. We also missed 19 license plates. Fig.9 shows some of them.

Fig. 8. Examples of correctly locateded license plates.

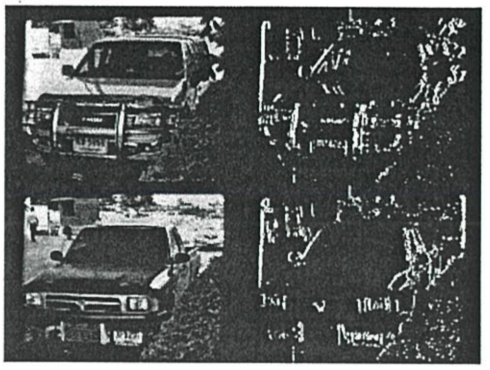


Fig. 9. Examples of failure to find license plates.

IV. CONCLUSION AND FUTURE WORKS

We have proposed a method for license plate localization of moving vehicles in complex scene using vertical edge distributions and temporal analysis. The result was satisfied with true detection rate 94% from 317 license plates. However, there are still some missing results due to various reasons such as motion blur, and license plate frame. For false alarm problem, we think moving toward license plate recognition will then help to eliminate some false alarms.

REFERENCES

[1] Hans A. Hegt, Ron J. De la Haye and Nadeem A. Khan, "A High Performance License Plate Recognition System," Proceeding of IEEE International Conference on Systems, Man, and Cybernetics, vol.5, pp. 4357 - 4362, 1998.

[2] Sirithinaphong T. and Chamnongthai K., "The recognition of car license plate for automatic parking system," Proceeding of Fifth International Symposium on Signal Processing and Its Applications, vol.1, pp. 455 - 457, 1999.

[3] Naito T., Tsukada T., Yamada K., Kozuka K. and Yamamoto S., "Robust license-plate recognition method for passing vehicles under outside environment" IEEE Trans. VTC, vol.49, Issue 6, pp. 2309 - 2319, 2000.

[4] C. Siricharoenchai and V. Areekul, "Car License Plate Extraction using Sharp Rising Edge Detection," Proceedings of 24th Electrical Engineering Conference, vol.2, pp. 1176-1181, 2001.

[5] Xu Jianfeng, Li Shaofa and Chen Zhibin, "Color Analysis For Chinese Car Plate Recognition," Proceeding of IEEE International Conference on Robotics, Intelligent Systems and Signal Processing, vol.2, pp. 1312 - 1316, 2003.

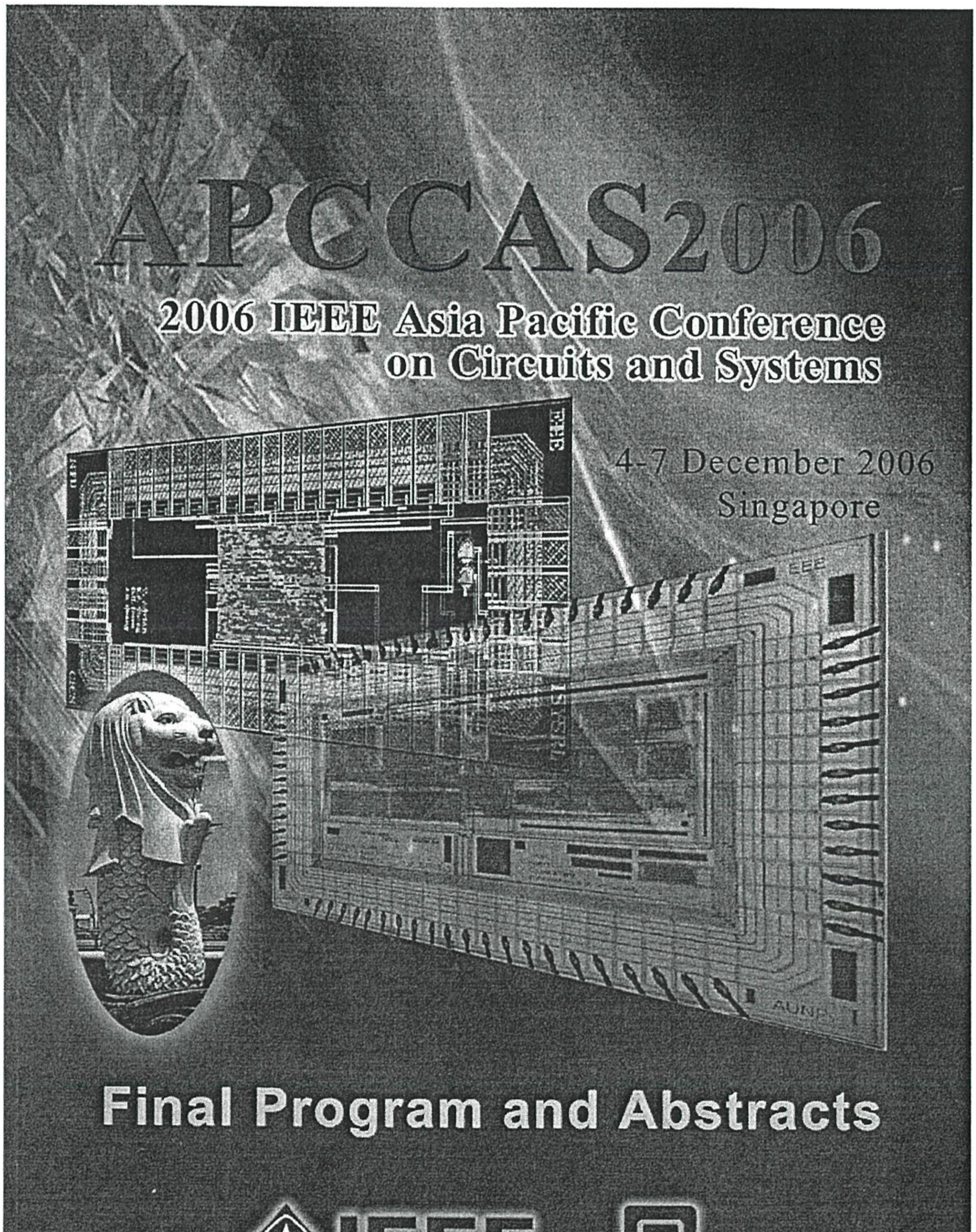
[6] Yoshimori S., Mitsukura Y., Fukumi M., Akamatsu N. and Khosal R., "License plate detection system in rainy days," Proceeding of IEEE Computational Intelligence in Robotics and Automation, vol.2, pp. 972 - 976, 2003.

[7] Shyang-Lih Chang, Li-Shien Chen, Yun-Chung Chung and Sei-Wan Chen, "Automatic license plate recognition," IEEE Trans. Intelligent Transportation Systems, vol.5, Issue 1, pp. 42 - 53, 2004.

[8] Tran Duc Duan, Duong Anh Duc and Tran Le Hong Du, "Combining Hough transform and contour algorithm for detecting vehicles' license-plates," Proceeding of International Symposium on Intelligent Multimedia, Video and Speech Processing, pp. 747 – 750, 2004.

[9] Hsi-Jian Lee, Si-Yuan Chen and Shen-Zheng Wang, "Extraction and recognition of license plates of motorcycles and vehicles on highways," Proceeding of 17th International Conference on Pattern Recognition, vol.4, pp. 356 - 359, 2004.

[10] Hongliang Bai, Junmin Zhu, and Changping Liu, "A Fast License Plate Extraction Method on Complex Background," Proceeding of IEEE Intelligent Transportation Systems, vol.2, pp. 985 - 987,2003.


APCCAS2006
2006 IEEE Asia Pacific Conference
on Circuits and Systems
4-7 December 2006
Singapore
Final Program and Abstracts
IEEE
CIRCUITS AND
SYSTEMS SOCIETY

K C Chang, G E Sobelman
University of Minnesota, UNITED STATES

An enhancement to the MB-OFDM system, known as Pulsed-OFDM, has been proposed to reduce the complexity and power consumption of the transceiver without sacrificing performance. In this paper, we describe the detailed FPGA implementation of a complete Pulsed-OFDM transceiver. The resource requirements are given for each of the major blocks for an implementation using a Xilinx VirtexTM-4 device. The entire system can be mapped onto a single FPGA chip.

Regular Session D2-PM2-RM7

Pattern Recognition II

1545 - 1700 12/6/2006 Wednesday

Room : **Falcon**

Chair(s) : **Pei-Yung Hsiao**

D2-PM2-RM7.1

Edge Detection on the Bayer Pattern

C H Chen, S J Chen, *P Y Hsiao
National Taiwan University, TAIWAN
*Chang Gung University, TAIWAN

This paper describes a novel investigation of edge detection on Bayer-patterned images. Inspired by the Color Filter Array (CFA) interpolation method, we design kernels to perform Gaussian-like smoothing and Laplacian-like edge detection on a Bayer-patterned image. Benefits of using our proposed kernels to process such image source include the computation saving during the interpolation and/or the color space transformation to a full color or grayscale image, and also lower memory requirement on the same time.

D2-PM2-RM7.2

License Plate Localization of Moving Vehicles in Complex Scene

T H Chalidabhongse, P Rattanathammawat
King Mongkut's Institute of Technology, Ladkrabang, THAILAND

In this paper, a license plate detection system is proposed to localize and extract license plates of moving vehicles in complex scene. The system works on compressed low resolution video streams obtained from real working environments. The illumination conditions are varied which typically occurs in outdoor environment. The vehicles are moving and the background scene is quite complex. Our method consists of three main steps. First, input images from video are passed through Sobel edge detector with automatic variable thresholding to cope with varied illuminations. Then, a predefined size moving window is passed through the image by skipping with a certain number of pixels. For each location of moving window, we consider the distribution of edge pixels to determine whether it is likely to be a license plate. The license plate likelihood scores of each window are then ranked and analyzed to suppress some false detection. The experimental results showed our method achieves up to 99% detection rate on compressed video streams containing 317 moving vehicles in varied illumination conditions from morning to night time.

D2-PM2-RM7.3

Study of Real-Time Detecting System for Driver's Safety

Q Zhang, Z Zhou, D Yin, J Yang
Wuhan University of Technology, CHINA

This paper presents a real-time detecting system based on the DSP processor. It uses real-time operating system DSP/BIOS to build tasks and interruptions, and will give out an incurrence of phonetic warning when the driver is tired. We use a comprehensive application of the complexion detection and differential frame arithmetic to make a correct orientation of the driver's face, then introduce a simple but credible method to judge the state of driver's mouth and a combination of horizontal integral projection and Hough transform to judge the state of driver's eyes.

D2-PM2-RM7.4

Feature Point Tracking for Car Speed Measurement

S Zhu, T Koga
Yamagata University, JAPAN

Moving car information in traffic video signals contains plenty of important information on road safety. In this paper, we propose an algorithm that extracts and tracks Feature Points of cars. The information is indispensable for measuring moving car speed, counting the number of cars, monitoring distance between two cars, classifying running directions, and monitoring congestion situation. Utilizing Harris operator for detecting edges and corners, Feature Points are extracted, followed by block-matching to track the Feature Points in successive video frames. Many cars can be tracked at the same time automatically since the information is obtained

License Plate Localization of Moving Vehicles in Complex Scene

Thanarat H. Chalidabhongs
Faculty of Information Technology
King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand
thanarat@it.kmitl.ac.th

Preemon Rattanathammawat
Faculty of Information Technology
King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand

***Abstract*—In this paper, a license plate detection system is proposed to localize and extract license plates of moving vehicles in complex scene. The system works on compressed low resolution video streams obtained from real working environments. The illumination conditions are varied which typically occurs in outdoor environment. The vehicles are moving and the background scene is quite complex. Our method consists of three main steps. First, input images from video are passed through Sobel edge detector with automatic variable thresholding to cope with varied illuminations. Then, a predefined size moving window is passed through the image by skipping with a certain number of pixels. For each location of moving window, we consider the distribution of edge pixels to determine whether it is likely to be a license plate. The license plate likelihood scores of each window are then ranked and analyzed to suppress some false detection. The experimental results showed our method achieves up to 99% detection rate on compressed video streams containing 317 moving vehicles in varied illumination conditions from morning to night time.**

I. INTRODUCTION

Automatic license plate recognition, also known as LPR, has been an interesting research topic for more than a decade [1,2]. There are still many systems being developed to improve performances and/or overcome the limitations. LPR can be used in many vision-based applications such as vehicles surveillance, automatic toll collection, car parking control, and so on. There are many approaches have been presented [3-14]. Hongliang Bai et al. [3] used edge density map and feature extraction technique to extract license plate from complex background. Lee, Chen, and Wang [6] developed cars and motorcycles license plate extraction in highways using projection-based method. Duan, Duc, and Du [7] proposed a LPR system for Vietnamese license plates using Hough transform and contour algorithm. Jianfeng, Shaofa, and Zhibin [8] employed color analysis with two-layer feed forward neural networks to extract car license plate. Thanongsak and Kosin [9, 10] extract car license plate for parking system using neural network and motor vehicle regulation. Seiki Yoshimori et al [11] use real code genetic algorithm to detect license in rainy days. Chang et al.[13] developed LPR base on fussy and neural disciplines.

Many previous works tested on images captured under restricted conditions, such as parking vehicles, fixed illumination, fine resolution, designated environment, and clean background. In this research, we consider working on video streams obtained from real working environments with a few restrictions. The illumination conditions can be varied which typically occurs in outdoor environment. The vehicles might be moving and the background scene can be complex. Our method consists of three steps. First, input images from video are passed through Sobel edge detector with automatic variable thresholding to cope with varied illuminations. Then, a predefined size moving window is passed through the image by skipping with a certain number of pixels. For each location of moving window, we consider the distribution of edge pixels to determine the license plate likelihood. To nominate the candidate windows, the proximity of the candidates are also considered to eliminate overlapped windows. Further, false alarms are reduced by considering the relative likelihood values of the candidates.

This paper is organized as follows. In Section II, the proposed method is described step by step. Experimental results are presented in Section III. Concluding remarks and ideas for future work are given in Section IV.

II. LICENSE PLATE DETECTION

Our proposed method consists of three main steps: horizontal edge detection, license plate localization based on distribution of edge pixels, and false detection suppression using likelihood score analysis.

A. Edge Detection

To cope with the varied illumination condition, which typically occurs in outdoor environment, the method works on the edge information of the image. We employs vertical Sobel edge detector with automatic variable thresholding [5] since we noticed that license plate area is likely to contain more vertical edges than other areas due to the high contrast between alphabets and plate background color. Fig. 1 show the comparison results of the vertical Sobel edge detections with fixed and variable thresholds.

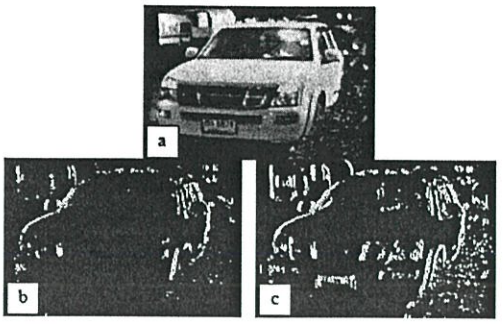


Figure 1. Vertical edge detection. (a) Original image (b) Edge image with fixed threshold (c) Edge image with variable threshold

B. License plate localization

To determine location of license plates in the images, we scan a predefined-size moving window through the entire image. For each location of moving window, we consider the distribution of edge pixels to determine whether it is likely to be a license plate. As described in [15], in considering the edge distribution, we divide the window vertically into three parts shown in Fig.2. We then count the number of edge pixels in the middle area (B) and subtract it with the sum of numbers of edge pixels in the peripheral areas (A and C). The result is used to specify the likelihood of being a license plate area. Let define this likelihood as L and it is given as

$$L=\beta-\alpha-\chi \tag{1}$$

where α, β, and χ are numbers of edge pixels in areas A, B, and C of the moving window respectively.

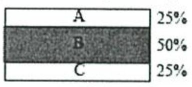


Figure 2. The moving window is divided into three parts and used in computing the pattern of edge distribution.

However, we found that this pixel counting is not a reliable metric especially when facing the case as shown in Fig. 3. The first three windows which have highest L are not license plate areas (Fig. 3b). Although they have high edge density in top and bottom areas, they also have very high edge density in middle. This makes L value high. Fig.4 shows the vertical histograms of the 1^{st} and 3^{rd} ranked windows as well as the true license plate one. As we can notice, the count is not a good metric in this case, we need to consider the relative edge pixel density of the peripheral areas comparing to the middle area, as shown in Eq. (2) and (3).

$$R_A=\frac{\alpha}{\beta} \tag{2}$$

$$R_C=\frac{\chi}{\beta} \tag{3}$$

Thus the license plate candidates will be the moving windows that have R_A and R_C within a threshold τ_R and are the top three windows with highest L values, as shown in Fig3c.

After selecting the top three windows with highest L values, those windows are often overlapped (See Fig.5b). This dues to our scanning step is small and edge distribution in neighboring windows are alike. To overcome with the overlapping candidates, we eliminate those overlapped windows that have lower L scores. The improved result is shown in Fig.5c.

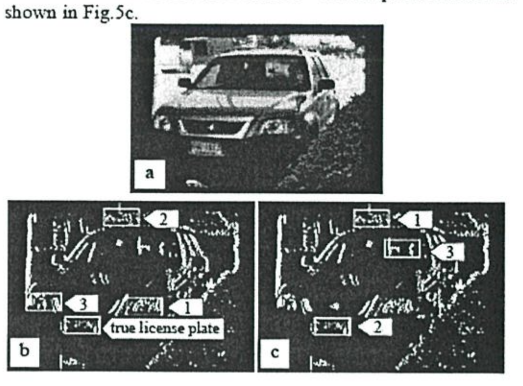


Figure 3. (a) Original image (b) Candidate windows considering the top three of L (c) Candidate windows after considering both pixel count L and relative density R_A and R_C.

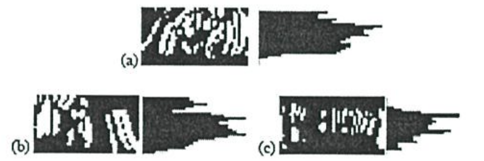


Figure 4. Horizontal projection of edge pixels for (a) and (b) Not license plate window (c) True license plate window

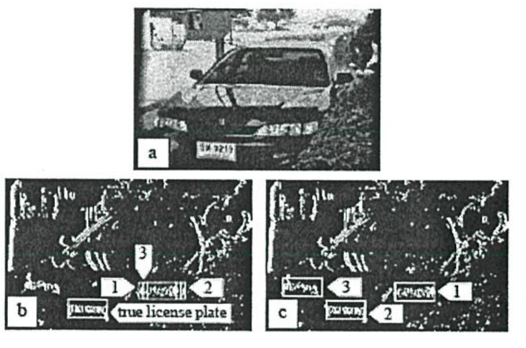


Figure 5. (a) Original image (b) Overlapping of first three highest L windows (c) Candidate windows after elimating overlaps.

C. *False alarm suppression(FAS) using likelihood score analysis*

To suppress false positive from the previous steps, we analyze the likelihood scores, L, among the top three windows. From our empirical analysis, we found that L value of true license plate window is usually highest or nearly to the highest, while L value of noise window is likely far from the highest. Thus, we compute the ratio between L value of each candidate windows and the highest L value as given in Eq.(4).

$$P_i = \frac{L_i}{L_{max}} \tag{4}$$

where P_i is ratio of L value of the rank i^{th} window and the maximum L (L_{max}). The candidate window i will be eliminated if its P_i is less than a threshold P_{limit}.

Fig.6 shows an example of a case that the true license plate window is of the lowest rank among those three candidates in term of L value, but its P value is relatively high. Thus, it should be kept as a license plate candidate window. Fig.7 shows another example. In this case, the true license plate window is of the 1st rank and noise windows are the 2nd and 3rd in term of L values. Considering the P values of those noise windows, they are relatively low. Thus, they are eliminated from the candidate list. With this technique, we reduce the false alarm significantly.

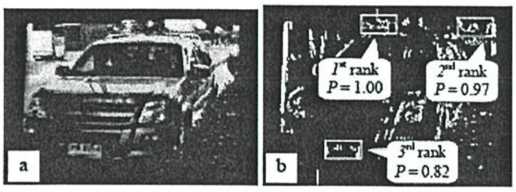


Figure 6. (a) Original image (b) Edge image with ranking candidate windows and its P value.

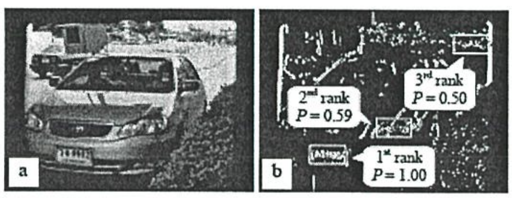


Figure 7. (a) Original image (b) Edge image with ranking candidate windows and its P value.

III. EXPERIMENTAL RESULTS

To evaluate the proposed method, we tested the system on compressed video streams recorded from real working environment of a shopping center in Bangkok. The shopping center installed CCTV cameras near entrance areas. There are no obstacles at the entrances, thus the cars passing entrance are moving. The videos were compressed and recorded into media for future uses such as using in investigation if there is any incident occurs. The resolution of the recorded video is 320x240 with 4 fps frame rate.

For this research, we extracted 317 images of moving cars in varied illumination conditions from morning to night time. The detection results are shown in Table I. When setting P_{limit} at 0.70, our proposed method achieved 99% detection rate, while having about 25% of false alarm rate. These false alarms can be easily reduced in recognition phase. However, we can adjust the P_{limit} to suppress false alarms by paying a small price of reducing detection rate as shown in Table I. When setting P_{limit} at 1.00, we can reduce the false alarm to 5.4%, while the detection rate is also reduced to 94.6%. Table II shows the comparative results of the method proposed here and the one proposed in [15]. In [15], we suppressed the false alarms using temporal analysis. From the results, we found the likelihood score analysis yields better result. Fig.8 shows the results in another form that can be easily visualized.

Fig.9 and Fig.10 show some sample images of original input images and detection results for successful cases and failure cases respectively.

TABLE I. LICENSE PLATE DETECTION RESULTS VARYING BY P_{limit} VALUES

P_{limit}	True Positive (Correctly detected)	False Positive (False Alarm)	False Negative (Miss)
0.70	314 (99.0%)	104/418 (24.9%)	3 (1.0%)
0.75	313 (98.7%)	75/388 (19.3%)	4 (1.3%)
0.80	313 (98.7%)	64/377 (17.0%)	4 (1.3%)
0.85	310 (97.8%)	41/351 (11.7%)	7 (2.2%)
0.90	307 (96.8%)	33/340 (9.7%)	10 (3.2 %)
0.95	302 (95.3%)	23/325 (7.1%)	15 (4.7%)
1.00	300 (94.6%)	17/317 (5.4%)	17 (5.4%)

TABLE II. COMPARATIVE RESULTS OF THE PROPOSED METHOD AND PREVIOUS ONE [15]

Methods	True Positive	False Positive	False Negative
No FAS	315 (99.4%)	636/951 (66.9%)	2 (0.6%)
FAS using temporal analysis [15]	298 (94.0%)	206/504 (40.9%)	19 (6.0%)
FAS using likelihood score analysis (P_{limit} = 0.7)	314 (99.0%)	104/418 (24.9%)	3 (1.0%)

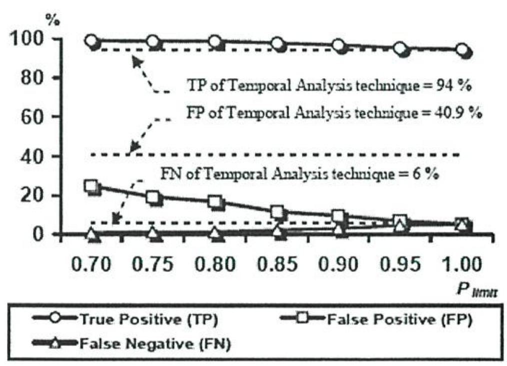


Figure 8. The results plotted at various P_{limit} values

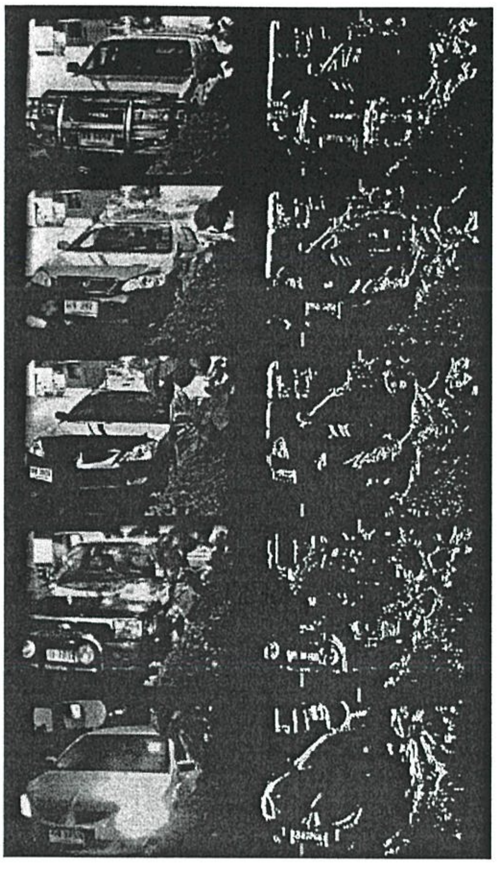


Figure 9. Examples of correctly locateded license plates.

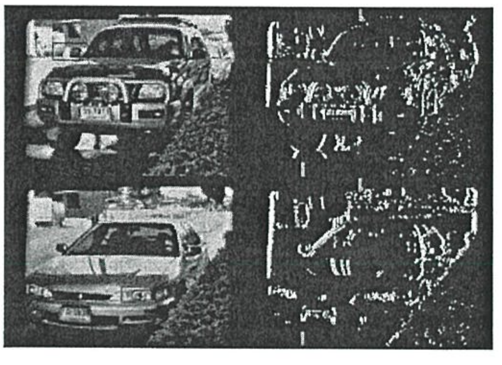


Figure 10. Examples of failure to find license plates.

IV. CONCLUSIONS AND FUTURE WORKS

We have proposed a new method for car plate detection that localizes and extracts car license plates in complex scene using vertical edge distributions and likelihood score analysis. The result was satisfied with 99% detection rate. However, there are still some missing results due to various reasons such as motion blur, and license plate frame.

REFERENCES

[1] Shapiro V., Gluhchev G., "Multinational license plate recognition system: segmentation and classification," Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004, vol.4, pp. 352 – 355, 2004.

[2] Rahman C.A., Badawy W., Radmanesh A., "A real time vehicle's license plate recognition system," Proceedings. IEEE Conference on Advanced Video and Signal Based Surveillance, pp. 163 – 166,2003.

[3] Hongliang Bai., Junmin Zhu., and Changping Liu, "A Fast License Plate Extraction Method on Complex Background," Proceeding of IEEE Intelligent Transportation Systems, vol.2, pp. 985 - 987, 200

[4] M. Sangwattanachai, C. Siriburaparat, and V. Areekul, "Real-time Automatic Thai License Plate Reading," Proceedings of 25th Electrical Engineering Conference (EECON-25), pp.20-24., 2002.

[5] C. Siricharoenchai and V. Areekul, "Car License Plate Extraction using Sharp Rising Edge Detection," Proceedings of 24th Electrical Engineering Conference, vol.2, pp. 1176-1181, 2001.

[6] Hsi-Jian Lee, Si-Yuan Chen and Shen-Zheng Wang, "Extraction and recognition of license plates of motorcycles and vehicles on highways," Proceeding of 17th International Conference on Pattern Recognition, vol.4, pp. 356 - 359, 2004.

[7] Tran Duc Duan, Duong Anh Duc and Tran Le Hong Du, "Combining Hough transform and contour algorithm for detecting vehicles' license-plates," Proceeding of International Symposium on Intelligent Multimedia, Video and Speech Processing, pp. 747 – 750, 2004.

[8] Xu Jianfeng, Li Shaofa and Chen Zhibin, "Color Analysis For Chinese Car Plate Recognition," Proceeding of IEEE International Conference on Robotics, Intelligent Systems and Signal Processing, vol.2, pp. 1312 - 1316, 2003.

[9] Sirithinaphong T. and Chamnongthai K., "The recognition of car license plate for automatic parking system," Proceeding of Fifth International Symposium on Signal Processing and Its Applications, vol.1, pp. 455 - 457, 1999.

[10] Sirithinaphong, T. and Chamnongthai, K., 1998, "Extracting of Car License Plate using Motor Vehicle Regulation and Character Pattern Recognition", The 1998 IEEE Asia-Pacific Conference on Circuits and Systems, pp. 559-562, 1998

[11] Yoshimori S., Mitsukura Y., Fukumi M., Akamatsu N. and Khosal R., "License plate detection system in rainy days," Proceeding of IEEE Computational Intelligence in Robotics and Automation, vol.2, pp. 972 - 976, 2003.

[12] Hans A. Hegt, Ron J. De la Haye and Nadeem A. Khan, "A High Performance License Plate Recognition System," Proceeding of IEEE International Conference on Systems, Man, and Cybernetics, vol.5, pp. 4357 - 4362, 1998.

[13] Shyang-Lih Chang, Li-Shien Chen, Yun-Chung Chung and Sei-Wan Chen, "Automatic license plate recognition," IEEE Trans. Intelligent Transportation Systems, vol.5, Issue 1, pp. 42 - 53, 2004.

[14] Naito T., Tsukada T., Yamada K., Kozuka K. and Yamamoto S., "Robust license-plate recognition method for passing vehicles under outside environment" IEEE Trans. VTC, vol.49, Issue 6, pp. 2309 - 2319, 2000.

[15] T. Chalidabhongse and P. Rattanathammawat, "A Car Plate Detector using Edge Information", To be appeared in Proc. IEEE International Symposium on Communications and Information Technologies, 2006.

ประวัติผู้เขียน

ชื่อ	นายปรีมนต์ รัตนาธรรมวัฒน์
วัน เดือน ปีเกิด	9 พฤษภาคม 2521 จังหวัดประจวบคีรีขันธ์
ที่อยู่	เลขที่ 513/133 ซ.จรัญสนิทวงศ์37 ถ.จรัญสนิทวงศ์ แขวงบางขุนศรี เขตบางกอกน้อย กรุงเทพมหานคร 10700
ประวัติการศึกษา	2542 วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ