

ระบบเฝ้าระวังอัจฉริยะ

INTELLIGENT SURVEILLANCE SYSTEM

ธนากร สำเภาทอง

ปวีศร มยานนท์

รุจรวี โกมลไสย

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา

วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

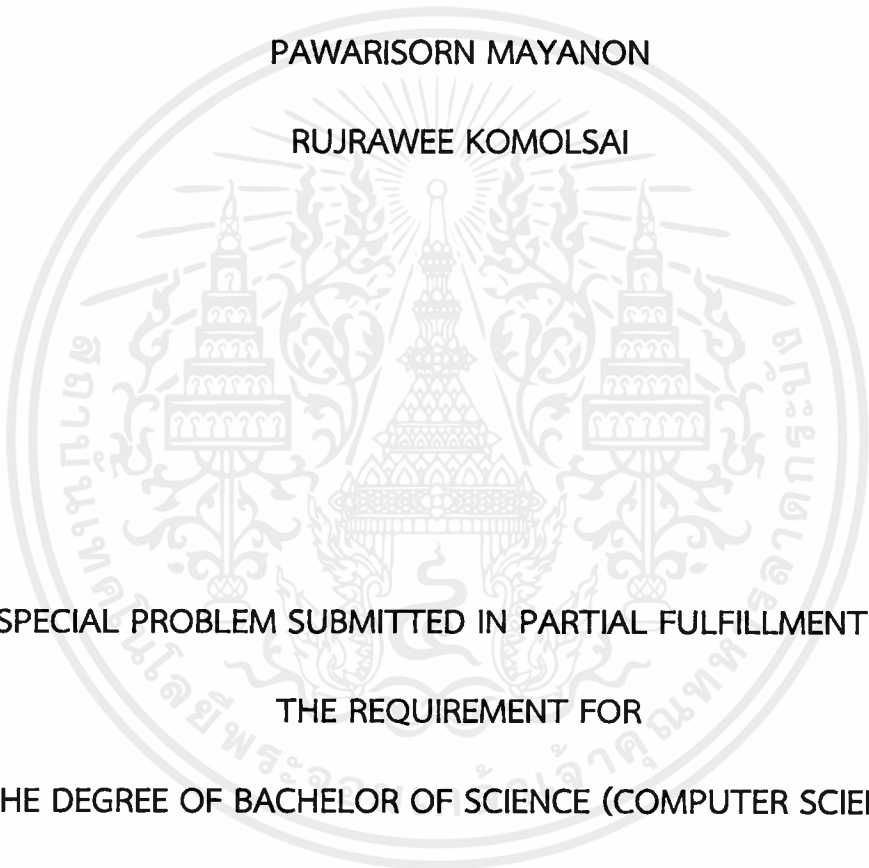
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INTELLIGENT SURVEILLANCE SYSTEM

THANAKORN SUMPAOTONG

PAWARISORN MAYANON

RUJRAWEE KOMOLSAI



SPECIAL PROBLEM SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR
THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)
DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2017

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	ระบบเฝ้าระวังอัจฉริยะ	
ชื่อนักศึกษา	นายธนากร สำเภาทอง	รหัสนักศึกษา 57050245
	นายปวีศร มยานนท์	รหัสนักศึกษา 57050275
	นายรุจรวี โกมลไสย	รหัสนักศึกษา 57050315
ปริญญา	วิทยาศาสตรบัณฑิต	
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์	
คณะ	วิทยาศาสตร์	
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)	
ปีการศึกษา	2560	
อาจารย์ที่ปรึกษา	ผศ.ดร.อนันตพร ทรราชคุณาฒย์	

บทคัดย่อ

ในปัจจุบันเทคโนโลยีทางการประมวลผลภาพ ถูกนำมาประยุกต์ใช้ในงานหลาย ๆ ด้าน เนื่องจากสามารถทำงานได้โดยอัตโนมัติและสามารถทดแทนแรงงานมนุษย์ได้ ทางผู้จัดทำจึงนำเทคโนโลยีนี้มาใช้พัฒนาเป็นระบบเฝ้าระวังอัจฉริยะ ที่สามารถตรวจจับท่าทางผิดปกติของมนุษย์ที่อาจจะก่อให้เกิดอันตราย เพื่อให้ผู้ใช้สามารถทำการประเมินสถานการณ์ต่อไป ระบบเฝ้าระวังอัจฉริยะสามารถตรวจจับสถานการณ์ที่ผิดปกติจากวิดีโอที่ต้องการโดยผู้ใช้สามารถเลือกไฟล์ที่ต้องการให้ตรวจจับหรือเลือกโดยใช้วิดีโอแบบเรียลไทม์จากกล้องวงจรปิด เมื่อระบบมีการตรวจจับเหตุการณ์ผิดปกติจะทำการแจ้งเตือนไปยังผู้ใช้ และทำการบันทึกการตรวจจับไว้ให้ผู้ใช้ดูรายละเอียดการตรวจจับ ระบบเฝ้าระวังอัจฉริยะเป็นการประยุกต์ใช้หลักการทางปัญญาประดิษฐ์ โดยการนำการเรียนรู้ของเครื่องมาใช้ในการพัฒนาโมเดลที่ใช้ในการตรวจจับสถานการณ์ผิดปกติ โดยโมเดลที่ใช้ถูกพัฒนาด้วยซอฟต์แวร์เวกเตอร์แมชชีนโดยเลือกใช้คอร์แนลแบบโคสควอร์ ประสิทธิภาพของโมเดลมีค่าความถูกต้องเท่ากับ 92.85% และมีค่า f-measure เท่ากับ 82.89%

คำสำคัญ : การตรวจจับการเคลื่อนไหว คอมพิวเตอร์วิทัศน์ ระบบอัจฉริยะ

Title	Intelligent surveillance system	
Students	Mr.Thanakorn Sumpaotong	Student ID 57050245
	Mr.Pawarison Mayanon	Student ID 57050275
	Mr.Rujrawee Komolsai	Student ID 57050315
Degree	Bachelor of Science	
Department	Computer Science	
Faculty	Science	
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)	
Academic Year	2017	
Advisor	Asst.Prof.Dr.Anantaporn Hanskunatai	

Abstract

In the present, image processing technology is used in many applications because it can work automatically and reduce human works. For that reason, this technology is used to develop the intelligent surveillance system. The system can detect unusual human motion that could be dangerous and notify user to analyze the situation. The intelligent surveillance system can detect an abnormal situation from selected video or video from a surveillance camera. When an unusual situation is detected, the system will notify user and record the history of detection information. The intelligent surveillance system is applied artificial intelligence and used machine learning for constructing the classification model to detect an unusual situation. For the model construction, SVM was used to build the model with the Chi-Square kernel. The predictive performance of the system was measured with accuracy and f-measure which are 92.85% and 82.89% respectively.

Keywords : Motion detection , Computer vision , Intelligent system

กิตติกรรมประกาศ

โครงการนี้สำเร็จลุล่วงได้ด้วยความกรุณาจากผศ.ดร.อนันตพร ทรราชคุณาฒย อาจารย์ที่ปรึกษาที่ได้ให้คำแนะนำ แนวคิด ตลอดจนแก้ไขข้อบกพร่องต่าง ๆ มาโดยตลอด จนปัญหาพิเศษเล่มนี้เสร็จสมบูรณ์รวมถึงได้ให้คำปรึกษาในด้านต่าง ๆ แม้จะไม่เกี่ยวข้องกับปัญหาพิเศษก็คอยให้คำปรึกษาแก่กลุ่มของผู้ศึกษาเป็นอย่างดี กลุ่มของผู้ศึกษาจึงขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณรศ.ดร.จีรพร วีระพันธุ์ และผศ.สิริลักษณ์ อนันต์สถิตย์สิน ที่ให้คำปรึกษา ข้อเสนอแนะ และแนวทางในการดำเนินปัญหาพิเศษตลอดจนติชมตัวผลงานปัญหาพิเศษในครั้งนี้จนเสร็จสมบูรณ์

ขอขอบคุณเจ้าหน้าที่ห้องวิจัยภาควิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง สำหรับการอำนวยความสะดวกเกี่ยวกับสวัสดิการนักศึกษา อาทิ กระจาดพิมพ์งาน

ขอขอบคุณภาควิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง สำหรับสิ่งอำนวยความสะดวกและอุปกรณ์เครื่องมือที่ใช้ในการทำปัญหาพิเศษ

ขอบคุณเพื่อน ๆ ในกลุ่มปัญหาพิเศษที่คอยให้ความช่วยเหลือและร่วมมือในการทำปัญหาพิเศษแม้จะพบกับปัญหาและความยากลำบาก แต่ก็ยังคงอยู่ร่วมกันทำปัญหาพิเศษต่อจนทำให้ปัญหาพิเศษในครั้งนี้สำเร็จลุล่วงไปได้ด้วยดี

ขอบคุณเพื่อน ๆ พี่ ๆ น้อง ๆ ที่ให้กำลังใจและคอยช่วยเหลือให้คำปรึกษาจนทำให้ปัญหาพิเศษในครั้งนี้สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบพระคุณครอบครัวที่คอยให้คำปรึกษาและให้กำลังใจในการทำปัญหาพิเศษในครั้งนี้ให้ดำเนินต่อไปได้ดีตลอดการทำปัญหาพิเศษ

ชนากร สำเภาทอง

ปวีศร มยานนท์

รุจรวี โกลมไสย

สารบัญ

	หน้า
บทคัดย่อ	ก
Abstract.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ช
สารบัญภาพ	ซ
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญและที่มาของปัญหา.....	1
1.2 วัตถุประสงค์ของการทำโครงการพิเศษ.....	1
1.3 ขอบเขตของโครงการพิเศษ	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 ขั้นตอนในการดำเนินงาน	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 การประมวลผลภาพดิจิทัล.....	3
2.1.1 จุดประสงค์ของการประมวลผลภาพดิจิทัล	4
2.1.2 ขั้นตอนพื้นฐานการประมวลผลภาพดิจิทัล	5
2.2 คอมพิวเตอร์วิทัศน์	10
2.2.1 การประมวลผลแบบเรียลไทม์ในคอมพิวเตอร์วิทัศน์.....	11
2.2.2 การลบภาพพื้นหลัง.....	12
2.2.3 คุณลักษณะของรูปภาพ.....	13
2.2.4 การตรวจจับมนุษย์.....	16
2.3 การเรียนรู้ของเครื่อง	25

2.3.1 ซัพพอร์ตเวคเตอร์แมชชีน	25
2.3.2 โครงข่ายประสาทเทียม	28
บทที่ 3 การวิเคราะห์และออกแบบระบบ	30
3.1 สถาปัตยกรรมระบบ	30
3.2 การวิเคราะห์ระบบ	31
3.2.1 Functional Requirement	31
3.2.2 Non - Functional Requirement	31
3.2.3 Use case Diagram	32
3.2.4 Activity Diagram	33
3.2.5 Sequence Diagram	36
3.3 การสร้างโมเดลตรวจจับท่าทางผิดปกติ	37
3.3.1 การจัดเตรียมข้อมูล	37
3.3.2 การสร้างโมเดล	40
3.3.3 การทดสอบโมเดล	42
3.3.4 การนำโมเดลไปใช้ในการจำแนกรูปภาพ	44
บทที่ 4 ผลการดำเนินงาน	45
4.1 ผลการทดลอง	45
4.1.1 การทดสอบโมเดลโดยใช้ชุดข้อมูลรูปภาพ	45
4.1.2 การทดสอบโมเดลโดยใช้ชุดข้อมูลวิดีโอ	50
4.2 หน้าจอแสดงผล	53
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ	58
5.1 บทสรุป	58
5.2 ข้อจำกัดของระบบ	58
5.3 ข้อเสนอแนะ	58
เอกสารอ้างอิง	59

ภาคผนวก.....	60
ภาคผนวก ก ขั้นตอนการติดตั้งและตั้งค่าเครื่องมือในระบบ	61
ภาคผนวก ข ผลการทดลองของการสร้างโมเดลด้วยเทคนิคการเรียนรู้ของเครื่อง.....	91



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
4.1 เปรียบเทียบค่า recall ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีน	46
4.2 เปรียบเทียบค่า precision ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีน	46
4.3 เปรียบเทียบค่า f-measure ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีน	47
4.4 เปรียบเทียบค่า accuracy ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีน	47
4.5 เปรียบเทียบค่า recall ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีนกับโครงข่ายประสาทเทียม	48
4.6 เปรียบเทียบค่า precision ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีนกับโครงข่ายประสาทเทียม	49
4.7 เปรียบเทียบค่า f-measure ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีนกับโครงข่ายประสาทเทียม ..	49
4.8 เปรียบเทียบค่าความแม่นยำของเทคนิคซ์พอร์ดเวคเตอร์แมชชีนกับโครงข่ายประสาทเทียม...	50
4.9 Confusion matrix ของวิดีโอเหตุการณ์ทะเลาะวิวาทที่สถานีรถไฟ.....	50
4.10 Confusion matrix ของวิดีโอเหตุการณ์ปล้นชิงทรัพย์บริเวณอาคาร.....	51
4.11 Confusion matrix ของวิดีโอเหตุการณ์ทะเลาะวิวาทที่ถนน	52
4.12 ค่า recall, precision, f-measure และ accuracy ของคลิปวิดีโอ.....	53
ข.1 ผลลัพธ์ที่ได้จากเคอร์เนล Linear ที่กำหนดให้ C-value = 0.1.....	91
ข.2 ผลลัพธ์ที่ได้จากเคอร์เนล Polynomial ที่กำหนดให้ gamma = 0.05, degree = 3, coef = 1, C-value = 0.1	91
ข.3 ผลลัพธ์ที่ได้จากเคอร์เนล Polynomial ที่กำหนดให้ gamma = 0.075, degree = 4, coef = 1, C-value = 0.1.....	92
ข.4 ผลลัพธ์ที่ได้จากเคอร์เนล RBF ที่กำหนดให้ gamma = 0.05, C-value = 0.1	92
ข.5 ผลลัพธ์ที่ได้จากเคอร์เนล RBF ที่กำหนดให้ gamma = 0.025, C-value = 0.1	92
ข.6 ผลลัพธ์ที่ได้จากเคอร์เนล RBF ที่กำหนดให้ gamma = 0.075, C-value = 0.1.....	93
ข.7 ผลลัพธ์ที่ได้จากเคอร์เนล Chi2 ที่กำหนดให้ gamma = 0.025, C-value = 2.....	93
ข.8 ผลลัพธ์ที่ได้จากเคอร์เนล Chi2 ที่กำหนดให้ gamma = 0.05, C-value = 2	93
ข.9 ผลลัพธ์ที่ได้จากเคอร์เนล Chi2 ที่กำหนดให้ gamma = 0.075, C-value = 0.1.....	94
ข.10 ผลลัพธ์ที่ได้จากเคอร์เนล Chi2 ที่กำหนดให้ gamma = 0.05, C-value = 0.1.....	94
ข.11 ผลลัพธ์ที่ได้จากโครงข่ายประสาทเทียม ที่กำหนดให้ Sigmoid = 0.1	94

สารบัญภาพ

รูปที่	หน้า
2.1 ขั้นตอนการประมวลผลภาพแบบดิจิทัล	4
2.2 ขั้นตอนพื้นฐานของการประมวลผลภาพแบบดิจิทัล	5
2.3 ตัวอย่างการปรับปรุงรูปภาพ	6
2.4 ตัวอย่างของการการฟื้นฟูภาพ	6
2.5 ตัวอย่างของการประมวลผลภาพสี	7
2.6 ตัวอย่างของการบีบอัดข้อมูล	8
2.7 ตัวอย่างของการประมวลผลด้านโครงสร้าง	8
2.8 ตัวอย่างของการแบ่งส่วนของภาพ	9
2.9 ตัวอย่างของการแทนที่และการอธิบาย	9
2.10 ตัวอย่างของการรู้จำรูปแบบ	10
2.11 ขั้นตอนการทำงานแบบเวลาจริงของการประมวลผลในทางการมองเห็นภาพของคอมพิวเตอร์ 11	11
2.12 เปรียบเทียบการประมวลผลระหว่างการใช้ CPU เพียงอย่างเดียวและเมื่อใช้ GPU เข้ามาช่วย 12	12
2.13 ขั้นตอนการทำงานของการแยกพื้นหลังออกจากพื้นหน้า	12
2.14 การสกัดเส้นขอบภาพ	13
2.15 การสกัดมุมของวัตถุในรูปภาพ	13
2.16 การสกัดคุณลักษณะในพื้นที่ที่สนใจ	14
2.17 คุณลักษณะของรูปภาพแบบภาพรวม (Global Feature)	15
2.18 คุณลักษณะภาพเฉพาะที่ (Local Feature).....	15
2.19 ขั้นตอนในการตรวจจับมนุษย์	16
2.20 ตัวอย่างการทำ Sliding Windows.....	17
2.21 ภาพแสดงการเคลื่อนไหวของลูกบอลในเฟรมภาพที่ต่อเนื่องกัน 4 เฟรม.....	18
2.22 ภาพแสดงการไล่ระดับเชิงมุม.....	20
2.23 การแบ่งรูปภาพออกเป็นเซลล์.....	21
2.24 การนำค่าใส่ภายในฮิสโตแกรม.....	21
2.25 ลักษณะฮิสโตแกรมหลังจากผ่านการคำนวณ.....	22
2.26 การทำ Normalization และแสดงแนวการไล่ระดับ.....	23
2.27 การเตรียมข้อมูลเข้าสู่กระบวนการเรียนรู้ด้วยเทคนิค SVM	24
2.28 ลำดับขั้นตอนการตรวจจับมนุษย์.....	24

2.29 ภาพแสดงการวัดมาร์จิ้นระหว่างไฮเพอร์เพลนและตำแหน่งข้อมูลใกล้เคียง.....	26
2.30 ชุดข้อมูลที่มีความซับซ้อนและไม่เป็นเชิงเส้น	26
2.31 การเปลี่ยนชุดข้อมูล 2 มิติไปเป็นรูปแบบ 3 มิติ.....	27
2.32 รูปภาพแสดงการแบ่งข้อมูลโดยใช้เคอร์เนล ลิเนียร์ โพลีโนเมียล และอาร์บีเอฟ ตามลำดับ ...	27
2.33 รูปภาพแสดงชิ้นการทำงานของโครงข่ายประสาทเทียม	28
2.34 รูปภาพแสดงโครงสร้างภายในของแต่ละโหนด	29
3.1 สถาปัตยกรรมของระบบ.....	30
3.2 Use case diagram ของระบบเฝ้าระวังอัจฉริยะ.....	32
3.3 Activity Diagram ของการ Input video.....	33
3.4 Activity Diagram ของ Detection	34
3.5 Activity Diagram ของฟังก์ชัน Detect Positive	35
3.6 Sequence Diagram ของฟังก์ชันตรวจจับจากวิดีโอแบบเรียลไทม์.....	36
3.7 Sequence Diagram ของฟังก์ชันการดูวิดีโอไฟล์	36
3.8 ตัวอย่างของชุดข้อมูลนำเข้าจากชุดข้อมูล Serre Lab.....	37
3.9 แผนภาพการจัดเตรียมรูปภาพจากวิดีโอ	38
3.10 ตัวอย่างรูปภาพที่ใช้ในการพัฒนาโมเดลตรวจจับ	39
3.11 รูปภาพหลังจากผ่านกระบวนการ Grey Scale.....	39
3.12 การแบ่งชุดข้อมูลฝึกสอนเพื่อนำไปใช้การสร้างและทดสอบประสิทธิภาพของโมเดล	40
3.13 โครงสร้างเครือข่ายประสาทเทียม.....	41
3.14 กระบวนการทดสอบประสิทธิภาพของโมเดล.....	42
3.15 กระบวนการทดสอบประสิทธิภาพของโมเดลกับชุดข้อมูลวิดีโอ	43
3.16 กระบวนการนำโมเดลมาใช้ในการตรวจจับสถานการณ์ผิดปกติในวิดีโอ	44
4.1 ตัวอย่างการตรวจจับภายในวิดีโอ 1.....	51
4.2 ตัวอย่างการตรวจจับภายในวิดีโอ 2.....	52
4.3 ตัวอย่างการตรวจจับภายในวิดีโอ 3.....	52
4.4 หน้าจอหลักของระบบเฝ้าระวังอัจฉริยะ.....	54
4.5 หน้าจอเลือกประเภทการตรวจจับ	55
4.6 หน้าจอแสดงข้อมูลการตรวจจับ (แบบไฟล์วิดีโอ).....	56
4.7 หน้าจอแสดงข้อมูลการตรวจจับ (แบบเรียลไทม์).....	57
ก.1 หน้าจอเว็บไซต์ของ OpenCV.....	62

ก.2 ไอคอนของโปรแกรมติดตั้ง OpenCV	62
ก.3 หน้าจอเลือก path	63
ก.4 หน้าจอในโฟลเดอร์หลังจากติดตั้ง OpenCV เสร็จสิ้น.....	63
ก.5 หน้าเว็บไซต์ของ Microsoft	64
ก.6 หน้า Visual Studio Downloads	64
ก.7 ไอคอนโปรแกรมติดตั้ง Visual studio	65
ก.8 หน้าจอเลือกประเภทของการติดตั้ง	65
ก.9 โฟลเดอร์ที่ติดตั้ง OpenCV.....	66
ก.10 หลังจาก search cmd ลงใน search menu.....	67
ก.11 ทำการค้นหา advance system settings.....	68
ก.12 หน้าจอเลือก Environment Variable	68
ก.13 หน้าจอเลือก System Variable	69
ก.14 การตั้งค่า System Path.....	69
ก.15 หน้าจอการสร้างโปรเจ็คใหม่.....	70
ก.16 ตั้งค่า Build สำหรับ Windows 32 และ 64.....	70
ก.17 หน้า Properties.....	71
ก.18 ส่วนของการ Configuration.....	72
ก.19 ส่วนของ Additional Include Directories.....	72
ก.20 หน้าจอใส่รายชื่อ Libraries ที่จำเป็น.....	73
ก.21 เว็บไซต์ของ QT.....	74
ก.22 หน้าจอดาวน์โหลดแบบออฟไลน์	75
ก.23 หน้าจอปรับค่า Proxy Settings.....	75
ก.24 หน้าจอ Log in.....	76
ก.25 เลือกโฟลเดอร์ติดตั้งโปรแกรม	76
ก.26 หน้าจอเลือก Package เพื่อติดตั้ง	77
ก.27 เวอร์ชันของ QT ที่ใช้ในการพัฒนาระบบ.....	77
ก.28 เลือก QT Creator	78
ก.29 ค้นหาโปรแกรม QT Creator.....	79
ก.30 หน้าจอโปรแกรม QT Creator.....	79
ก.31 หน้าจอเลือกประเภทของโปรเจ็คที่สร้างขึ้น	80

ก.32 เลือกไฟล์เดอร์ที่เก็บ Project	80
ก.33 เลือก Tool ที่ใช้ในการพัฒนา	81
ก.34 หน้าแสดงรายละเอียดของคลาส.....	82
ก.35 หน้าสรุปผลการสร้าง Project.....	82
ก.36 หน้าจอทาวนโฮลด์ตัวติดตั้ง CMake.....	85
ก.37 เลือก Option ก่อนติดตั้ง CMake.....	85
ก.38 เลือกไฟล์เดอร์ที่ต้องการติดตั้งโปรแกรม	86
ก.39 ค้นหาโปรแกรม Cmake ใน Windows Search.....	86
ก.40 เลือกไฟล์เดอร์ติดตั้ง	87
ก.41 หน้าจอให้เลือก generator	88
ก.42 หน้าจอหลังจากที่เปิด Visual studio ที่ตั้งค่าไว้.....	89
ก.43 ทำการ Build	90



บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของปัญหา

ปัจจุบันเทคโนโลยีการประมวลผลภาพ ได้เข้ามามีบทบาทในชีวิตประจำวันมากยิ่งขึ้น เนื่องจากเป็นเทคโนโลยีที่ช่วยอำนวยความสะดวกในการทำงานด้านต่าง ๆ ของมนุษย์ ตัวอย่างเช่น ในกรณีที่มีวิดีโอจำนวนมากซึ่งทำให้เกิดข้อมูลที่มีมหาศาล การทำการตรวจสอบข้อมูลเหล่านั้นทำได้ยากเนื่องจากมีความล่าช้าและความถูกต้องที่น้อย การใช้เทคโนโลยีการประมวลผลภาพมาช่วยในการตรวจสอบ จะทำให้งานที่ได้มีความรวดเร็วและแม่นยำมากกว่า ซึ่งการใช้เทคโนโลยีการประมวลผลภาพ เป็นการลดภาระงานให้กับมนุษย์ได้เป็นอย่างดี

เทคโนโลยีการประมวลผลภาพ ได้ถูกนำมาประยุกต์เพื่อใช้ในงานด้านต่าง ๆ เช่น ด้านความปลอดภัย ด้านการแพทย์ ด้านกฎหมาย เป็นต้น ซึ่งซอฟต์แวร์ที่จัดทำขึ้นมาเน้นไปที่การใช้งานด้านความปลอดภัย โดยจะมีการตรวจจับท่าทางที่ผิดปกติที่เกิดขึ้นในสถานการณ์ต่าง ๆ แล้วซอฟต์แวร์จะทำการวิเคราะห์ข้อมูลเพื่อทำการประเมินสถานการณ์ หากพบว่ามีความเป็นไปได้ว่าจะเกิดสถานการณ์ผิดปกติ ซอฟต์แวร์จะทำการแจ้งเตือนไปยังผู้ใช้งาน เพื่อให้ผู้ใช้ตัดสินใจและหาแนวทางการแก้ไขต่อไป แต่อย่างไรก็ตามการที่จะนิยามกิจกรรมใด ๆ ที่มีความผิดปกตินั้นนิยามได้ยาก เพราะขึ้นอยู่กับมุมมองของแต่ละบุคคล จึงมีการนำการเรียนรู้ของเครื่องมาใช้เพื่อให้ซอฟต์แวร์สามารถทำการนิยามได้ว่า สถานการณ์ใดเป็นสถานการณ์ที่ผิดปกติ

1.2 วัตถุประสงค์ของการทำโครงการพิเศษ

พัฒนาซอฟต์แวร์ที่สามารถทำการรับข้อมูลจากวิดีโอไฟล์ของผู้ใช้งานหรือจากกล้องแบบเรียลไทม์ มาประมวลผลทางคอมพิวเตอร์เพื่อตรวจจับท่าทางผิดปกติที่เกิดขึ้น และแสดงผลลัพธ์ของการตรวจจับให้กับผู้ใช้ เพื่อให้ผู้ใช้ตัดสินใจและหาแนวทางการแก้ไขต่อไป

1.3 ขอบเขตของโครงการพิเศษ

- 1) โปรแกรมสามารถทำงานได้บนเครื่องคอมพิวเตอร์แบบตั้งโต๊ะและแบบพกพา เท่านั้น
- 2) โปรแกรมสามารถรับข้อมูลไฟล์วิดีโอของผู้ใช้งาน และข้อมูลจากกล้องแบบเรียลไทม์

1.4 ประโยชน์ที่คาดว่าจะได้รับ

ระบบเฝ้าระวังอัจฉริยะช่วยให้ผู้ใช้มีตัวช่วยในการประเมินสถานการณ์ที่อาจจะเกิดขึ้น และยังสามารถทำงานได้แบบทันทีเพื่อลดความเสียหายที่เกิดขึ้นจากอันตรายต่าง ๆ

1.5 ขั้นตอนในการดำเนินงาน

- 1) ศึกษาเทคโนโลยีการประมวลผลภาพ คอมพิวเตอร์วิทัศน์ การตรวจจับการเคลื่อนไหว และการเรียนรู้ของเครื่อง
- 2) รวบรวมข้อมูลที่จำเป็นในการสร้างโปรแกรม
- 3) จัดทำข้อมูลฝึกสอนเพื่อใช้ในการสร้างโมเดล
- 4) ทำการออกแบบโครงสร้างต่าง ๆ ของโปรแกรม
- 5) เริ่มทำการเขียนโปรแกรม ไปพร้อมกับการทดสอบ
- 6) สรุปผลการทำงาน
- 7) จัดทำเอกสารคู่มือการใช้งาน

บทที่ 2

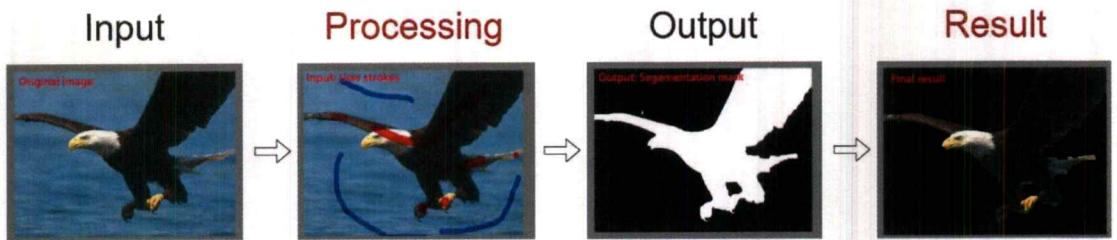
ทฤษฎีที่เกี่ยวข้อง

ระบบเฝ้าระวังอัจฉริยะได้แบ่งกระบวนการทำงานออกเป็นสามส่วนหลักๆ ซึ่งเป็นการประยุกต์ความรู้ทางด้านการประมวลผลภาพ, คอมพิวเตอร์วิทัศน์ และการเรียนรู้ของเครื่อง ในแต่ละกระบวนการจะมีหัวข้อการทำงานดังต่อไปนี้

2.1 การประมวลผลภาพดิจิทัล

เนื่องจากภาพที่นำมาใช้บางครั้งอาจมีความซับซ้อน คลุมเครือหรือไม่ชัดเจน วิธีการประมวลผลภาพ (Image Processing) จะช่วยทำให้ภาพมีความชัดเจนขึ้นหรือกำจัดสิ่งรบกวน (Noise) ที่ไม่จำเป็นออกไปก่อนที่จะนำรูปภาพนั้นไปใช้งาน การประมวลผลภาพดิจิทัล (Digital image processing) เป็นกระบวนการใช้อัลกอริทึมในการแปลงรูปภาพให้อยู่ในรูปของข้อมูลดิจิทัล โดยจะมีการดำเนินการกับภาพเพื่อปรับปรุงภาพให้ดีขึ้นหรือแยกส่วนบางข้อมูลที่มีประโยชน์มาใช้งาน การประมวลผลภาพเป็นประเภทหนึ่งของการจัดสรรสัญญาณที่มีตัวรับเข้าเป็นภาพ เช่น เพรมวิดีโอหรือภาพถ่าย และมีตัวนำออกเป็นภาพ ลักษณะที่เชื่อมโยงกับภาพเหล่านั้น หรือข้อมูลที่เกี่ยวข้องกับภาพนั้น ๆ โดยทั่วไปแล้วขั้นตอนการประมวลผลภาพแบ่งได้ 3 ขั้นตอนใหญ่ๆ ได้ดังนี้

- 1) การนำเข้าภาพ คือการนำภาพเข้าสู่ระบบ อาจจะต้องมีการปรับแต่งภาพ เช่น ปรับปรุงรายละเอียดหรือการลดสัญญาณรบกวน เพื่อให้ภาพที่จะนำเข้าสู่ระบบนั้นเป็นภาพที่มีคุณภาพที่ดีกว่าเดิม
- 2) ทำการวิเคราะห์และจัดการภาพ เช่น การบีบอัดภาพ การปรับปรุงภาพให้ดีขึ้น การประมวลผลภาพสี การรู้จำรูปร่าง หรือการทำแบบร่างเป็นจุดสำหรับกรณีที่เป็นภาพที่มนุษย์ไม่สามารถมองเห็นได้ด้วยตา เช่น ภาพถ่ายจากดาวเทียม
- 3) ตัวนำออกเป็นขั้นตอนสุดท้ายของกระบวนการประมวลผลภาพ การนำเสนอภาพที่มีผลลัพธ์ที่ถูกปรับเปลี่ยน หรือออกเป็นรายงานที่มีความเกี่ยวข้องกับภาพที่ถูกวิเคราะห์



รูปที่ 2.1 ขั้นตอนการประมวลผลภาพแบบดิจิทัล

จากรูปที่ 2.1 เป็นการอธิบายขั้นตอนของการแบ่งส่วนของภาพ โดยจะเริ่มจากการนำเข้าภาพ ต่อมาจะเป็นการดำเนินการเพื่อกำหนดส่วนที่เป็นวัตถุกับส่วนที่เป็นพื้นหลัง ผลลัพธ์ที่ได้จะเป็นส่วนพื้นหลังของภาพ และส่วนที่เป็นวัตถุในภาพ

ปัจจุบันการประมวลผลภาพนั้น นำมาประยุกต์ใช้กับงานในหลายๆด้าน เช่น เพื่อตรวจสอบคุณภาพของผลิตภัณฑ์ที่ผลิตออกมาจากโรงงาน เพื่อหาลักษณะแล้วนำไปเปรียบเทียบกับข้อมูลในฐานข้อมูล ใช้เพื่อช่วยการวินิจฉัยในทางการแพทย์

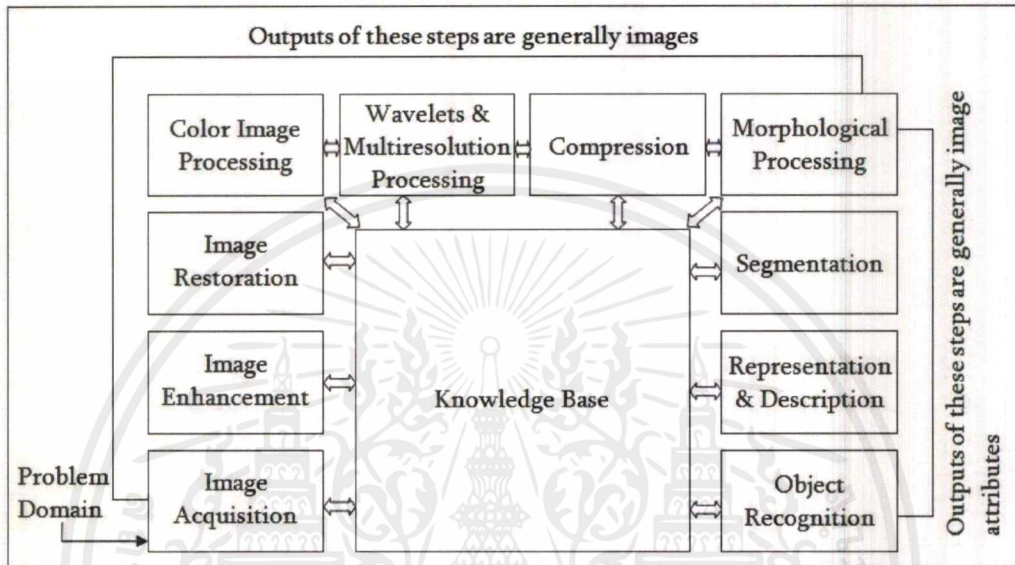
2.1.1 จุดประสงค์ของการประมวลผลภาพดิจิทัล

จุดประสงค์ของการประมวลผลภาพดิจิทัลแบ่งออกเป็น 5 กลุ่ม คือ

- 1) การทำให้มองเห็นวัตถุในภาพแบบชัดเจน เป็นการใช้อุปกรณ์ประมวลผลภาพดิจิทัลในการช่วยให้สังเกตเห็นวัตถุที่ไม่สามารถมองเห็นได้ในภาพนั้น ๆ
- 2) การทำให้ภาพให้คมชัดและการฟื้นฟูภาพ เป็นการใช้อุปกรณ์ประมวลผลภาพดิจิทัลปรับปรุงทำให้ภาพมีความคมชัดมากยิ่งขึ้น และสัญญาณรบกวนในภาพจะถูกขจัด
- 3) ช่วยในการสืบค้นภาพ เป็นการคัดแยกคุณลักษณะในภาพที่สนใจ แล้วนำไปเทียบเคียงกับคุณลักษณะของภาพแต่ละภาพที่เก็บไว้ในฐานข้อมูล
- 4) ช่วยในการรู้จำรูปแบบ เป็นการสกัดลักษณะในภาพที่สำคัญหลาย ๆ ภาพออกมา แล้วนำลักษณะที่ได้มาจัดเป็นหมวดหมู่

5) หาความแตกต่างระหว่างภาพ เป็นการจำแนกความแตกต่างของวัตถุในรูปภาพ จากการนำคุณลักษณะของภาพที่ได้จากการรู้จำรูปแบบของภาพแต่ละภาพมาเทียบ เพื่อหาความแตกต่าง

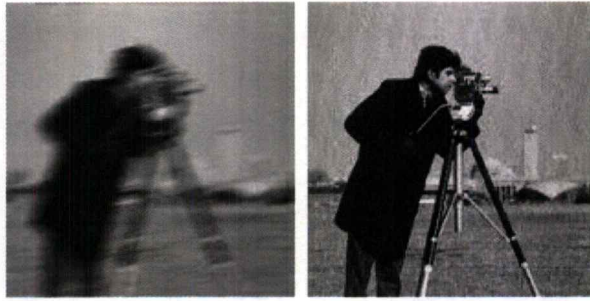
2.1.2 ขั้นตอนพื้นฐานการประมวลผลภาพดิจิทัล



รูปที่ 2.2 ขั้นตอนพื้นฐานของการประมวลผลภาพแบบดิจิทัล

จากรูปที่ 2.2 หลังจากที้นำเข้าภาพผ่านระบบต้องมีการปรับแต่งภาพ (Image Acquisition) ก่อน แล้วจึงนำภาพที่ได้จากการปรับแต่งนั้นไปเก็บไว้ในฐานความรู้ เพื่อที่จะนำภาพไปใช้ในขั้นตอนต่อไป สำหรับขั้นตอนต่าง ๆ ในการประมวลผลภาพสามารถแบ่งได้ดังนี้

- 1) การปรับปรุงภาพ (Image Enhancement) เป็นการปรับปรุงภาพที่ไม่มีคมชัดจนให้ได้ภาพที่มีความคมชัดมากยิ่งขึ้น หรือเพื่อให้ส่วนของลักษณะที่ต้องการจากภาพมีความเด่นชัดขึ้น



รูปที่ 2.3 ตัวอย่างการปรับปรุงรูปภาพ

จากรูปที่ 2.3 แสดงให้เห็นภาพทางซ้ายมือที่เป็นภาพที่ไม่มีความชัดเจน พอนำมาประมวลผลผ่านขั้นตอนการปรับปรุงภาพ จะทำให้ได้ภาพขวามือที่มีความชัดเจนมากยิ่งขึ้น

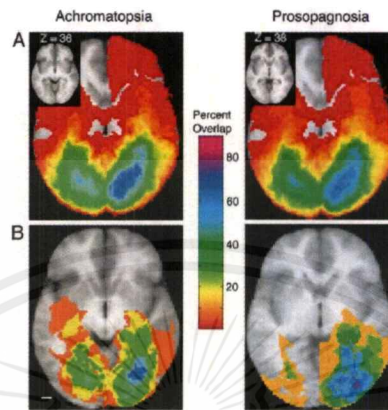
2) การฟื้นฟูภาพ (Image Restoration) เป็นการทำให้ภาพกลับสู่สภาพเดิม หรือปรับปรุงให้เหมาะสมแก่การมองเห็น ซึ่งจะเกี่ยวข้องกับความเสี่ยง



รูปที่ 2.4 ตัวอย่างของการการฟื้นฟูภาพ

จากรูปที่ 2.4 แสดงให้เห็นว่าภาพทางซ้ายมือคือภาพก่อนที่จะผ่านขั้นตอนการฟื้นฟูภาพ ซึ่งภาพทางซ้ายเป็นภาพที่มีความเสื่อมตามเวลา ส่วนภาพทางขวามือคือภาพที่ผ่านการฟื้นฟูภาพเรียบร้อยแล้วจะได้ภาพที่สามารถนำไปใช้งานได้เหมาะสม

3) การประมวลผลภาพสี (Color Image Processing) เป็นการประมวลผลโดยจะมีการใช้รูปแบบของสีแบบต่าง ๆ เช่น สี ความสว่างของสี เป็นตัวบอกการแยกคุณลักษณะที่สำคัญของภาพ เนื่องจากการใช้รูปแบบของสีมาระบุและแยกแยะวัตถุสามารถทำได้ง่าย



รูปที่ 2.5 ตัวอย่างของการประมวลผลภาพสี

จากรูปที่ 2.5 เป็นภาพที่แสดงภาวะการเสียการระสีรู้ทางตา โดยภาพที่แสดงสีแดงจะแสดงถึงการทับซ้อนกันของผลที่น้อยไล่ตามเฉดสีขึ้นไปจนถึงสีชมพู ซึ่งแสดงให้เห็นถึงบริเวณที่มีการทับซ้อนกันมากที่สุด

4) การแปลงแบบเวฟเล็ต (Wavelets & Multiresolution Processing) เนื่องจากขอบเขตของภาพเป็นจุดที่มีความถี่สูงสุดในภาพดิจิทัล การใช้ตัวกรองแบบเดิมจะช่วยจัดสิ่งรบกวนได้อย่างดี แต่จะทำให้ขอบภาพเบลอ ดังนั้นการแปลงแบบเวฟเล็ตจะเป็นการแปลงภาพ ที่จะเลือกแปลงสัญญาณตามขนาดของความถี่ทำให้ภาพที่ได้นั้นขอบภาพไม่เบลอ และการจัดสิ่งรบกวนก็ยังคงทำได้เป็นอย่างดี

5) การบีบอัดข้อมูล (Compression) เป็นการประมวลผลโดยการลดความละเอียดของภาพลง จากเดิมที่ภาพมีความละเอียดสูงหมายถึงภาพนั้นจะมีขนาดใหญ่เพราะมีการเก็บข้อมูลของสีแต่ละพิกเซลที่มากกว่าภาพที่มีความละเอียดต่ำ การบีบอัดข้อมูลจะเป็นการลดการเก็บข้อมูลของสีในแต่ละภาพลง แต่ยังคงสามารถมองภาพออกได้ว่าเป็นภาพอะไร ภาพที่ผ่านการบีบอัดมานั้น จะมีขนาดเล็ก ทำให้ลดการใช้พื้นที่หน่วยความจำลง



รูปที่ 2.6 ตัวอย่างของการบีบอัดข้อมูล

จากรูปที่ 2.6 ภาพทางซ้ายมือจะเป็นภาพที่ความละเอียดสูงสุด คือ เต็มหนึ่งร้อยเปอร์เซ็นต์ จะมีขนาดของภาพที่มากที่สุด ส่วนภาพถัดมาทางขวามือเป็นภาพที่ผ่านการบีบอัดเพื่อลดความละเอียดของภาพลงทำให้ขนาดของภาพเล็กลงเมื่อเทียบกับภาพที่มีความละเอียดสูงกว่า

6) การประมวลผลด้านโครงสร้าง (Morphological Processing) เป็นการประมวลผลเพื่อแยกเฉพาะรูปแบบหรือโครงสร้างของวัตถุในภาพออกมา โดยตัดส่วนที่ไม่ใช่โครงสร้างของวัตถุนั้นทิ้งไป



FIGURE 25-11 Binary skeletonization. The binary image of a fingerprint, (a), contains ridges that are many pixels wide. The skeletonized version, (b), contains ridges only a single pixel wide.

รูปที่ 2.7 ตัวอย่างของการประมวลผลด้านโครงสร้าง

จากรูปที่ 2.7 ภาพทางซ้ายมือคือภาพของลายนิ้วมือ ส่วนภาพทางขวาคือภาพที่ผ่านการประมวลผลโครงสร้างแล้ว ภาพที่ได้จะเป็นภาพโครงสร้างของลายนิ้วมือ

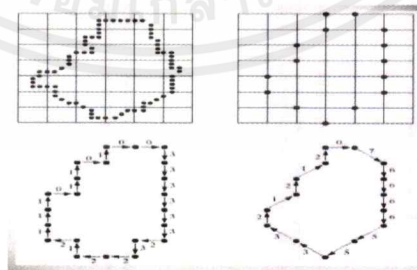
7) การแบ่งส่วนของภาพ (Segmentation) คือการแบ่งภาพออกเป็นหลาย ๆ ส่วนตามที่กำหนด เช่น แบ่งเป็น 2 หรือ 3 ส่วน จากนั้นจะทำการแยกส่วนทั้งหมดออกจากกัน แล้วจะเก็บข้อมูลไว้ในฐานข้อมูล หากต้องการนำส่วนไหนไปใช้ประมวลผลในขั้นตอนอื่น ๆ ก็สามารถนำไปใช้ได้โดยไม่ต้องนำภาพไปทั้งภาพ เนื่องจากใช้เวลามากกว่าการนำภาพไปส่วนเดียว



รูปที่ 2.8 ตัวอย่างของการแบ่งส่วนของภาพ

จากรูปที่ 2.8 ภาพทางซ้ายมือจะเป็นภาพเดิม ส่วนทางขวามือคือภาพที่แบ่งส่วนแต่ละส่วนไว้ โดยในภาพนี้จะแบ่งออกเป็น 2 ส่วน คือ ส่วนของพื้นหญ้า และส่วนของนก

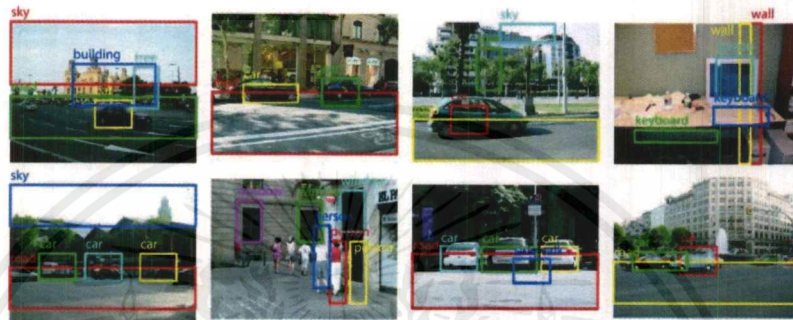
8) การแทนที่และการอธิบาย (Representation and Description) เป็นขั้นตอนหลังจากการแบ่งส่วนของภาพ หลังจากที่ได้ส่วนของภาพมาแล้วนั้น พื้นที่ที่เป็นส่วนนั้น ๆ จะถูกแทนที่และนำมาอธิบาย โดยการแทนที่จะมี 2 แบบ คือ การแทนที่ภายนอก คือจะสนใจที่ขอบเขตรูปร่างของพื้นที่ และการแทนที่ภายใน คือจะสนใจที่สีหรือเนื้อหาของส่วนของภาพนั้น



รูปที่ 2.9 ตัวอย่างของการแทนที่และการอธิบาย

จากรูปที่ 2.9 จะแสดงให้เห็นถึงการแทนที่ภายนอก ซึ่งสนใจที่ขอบเขตของภาพ ไม่ได้สนใจที่ตัวเนื้อหาของส่วนของภาพ

9) การรู้จำรูปร่าง (Object Recognition) เป็นการเก็บรูปแบบหรือโครงสร้างของภาพมา เพื่อให้การดำเนินการขั้นตอนหรือครั้งต่อไป สามารถนำรูปแบบหรือโครงสร้างที่เก็บไว้มาใช้ โดยไม่ต้องการประมวลผลภาพใหม่ ส่วนใหญ่จะใช้ในการตรวจจับวัตถุต่าง ๆ



รูปที่ 2.10 ตัวอย่างของการรู้จำรูปร่าง

จากรูปที่ 2.10 มีการเก็บรูปแบบของวัตถุต่าง ๆ เอาไว้ เช่น มีการเก็บรูปแบบของรถยนต์ อาคาร หรือบุคคล จะสังเกตได้ว่า วัตถุที่เป็นรถยนต์ในภาพจะถูกตรวจจับ เพราะได้มีการเก็บโครงสร้างของรถยนต์เอาไว้ในฐานข้อมูล ทำให้การตรวจจับนั้นเป็นไปได้อย่างรวดเร็ว เพราะไม่ต้องทำการประมวลผลใหม่ทุกครั้ง

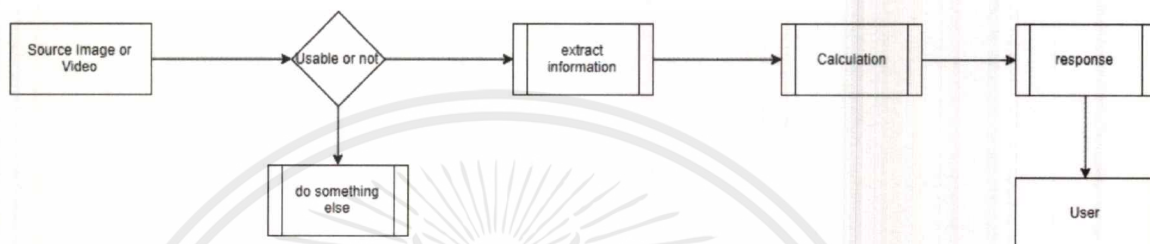
2.2 คอมพิวเตอร์วิทัศน์

คอมพิวเตอร์วิทัศน์ (Computer Vision) คือการนำวิธีการทางการประมวลผลภาพดิจิทัล มาประยุกต์ใช้เพื่อทำให้คอมพิวเตอร์สามารถประมวลผลรูปภาพให้ได้ผลลัพธ์ตามที่ต้องการ หรือทำให้คอมพิวเตอร์สามารถเรียนรู้และเข้าใจรูปภาพได้ โดยวิธีการทางคอมพิวเตอร์วิทัศน์ที่นำมาใช้มี 4 หัวข้อหลัก ได้แก่ 1. การประมวลผลแบบเรียลไทม์ในคอมพิวเตอร์วิทัศน์ (Real-Time Processing in Computer Vision) 2. การลบภาพพื้นหลัง (Background Subtraction) 3. คุณลักษณะของรูปภาพ (Image Features) 4. การตรวจจับมนุษย์ (Human Detection)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

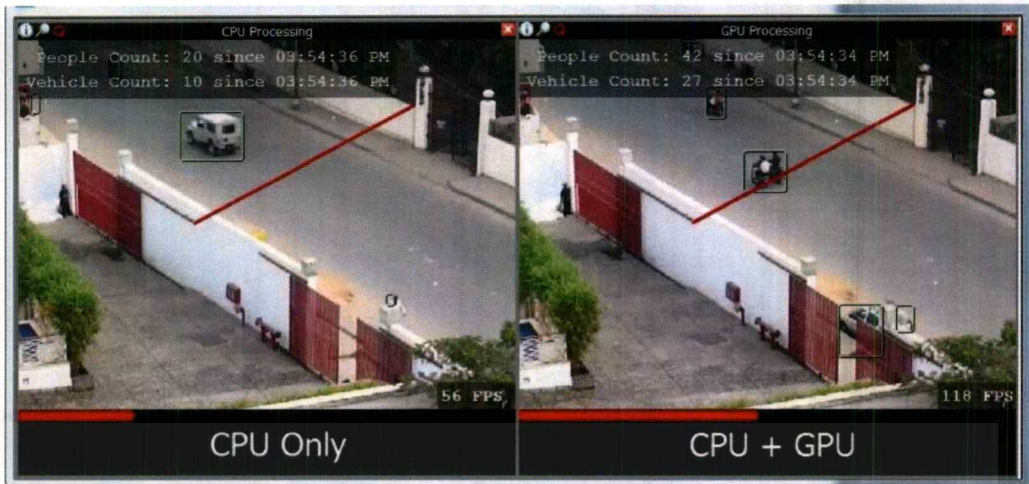
2.2.1 การประมวลผลแบบเรียลไทม์ในคอมพิวเตอร์วิทัศน์

การประมวลผลแบบเรียลไทม์ในคอมพิวเตอร์วิทัศน์นั้น เกี่ยวข้องกับการคำนวณทางคอมพิวเตอร์ให้สอดคล้องกับเวลาที่ต้องการหรือควรจะเป็น โดยในส่วนของมุมมองเห็นของคอมพิวเตอร์จะเป็นการสกัดข้อมูลจากรูปภาพหรือวิดีโอ ประมวลผลและตอบสนองกลับมายังผู้ใช้งานภายในเวลาที่กำหนดหรือที่ควรจะเป็นตามที่คาดการณ์ไว้



รูปที่ 2.11 ขั้นตอนการทำงานแบบเวลาจริงของการประมวลผลในทางการมองเห็นภาพของคอมพิวเตอร์

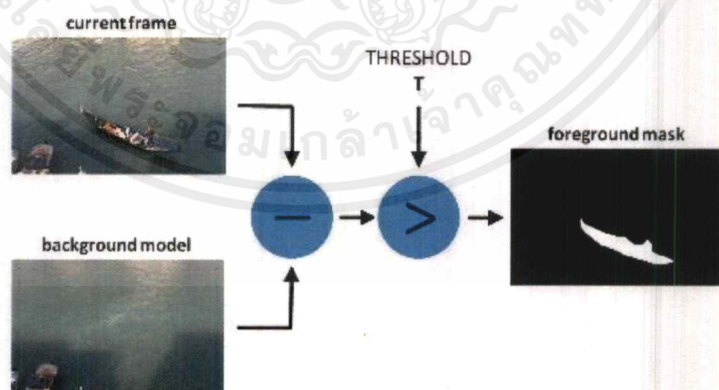
โดยที่วิดีโอทั่วไปจะถูกใช้งานอยู่ระหว่าง 24 เฟรมต่อวินาที ไปจนถึง 30 เฟรมต่อวินาที จึงจะถูกเรียกว่าสามดำเนินการได้ในแบบตามเวลาจริง ซึ่งก็ขึ้นอยู่กับข้อกำหนดของผู้ใช้งานหรือผู้ทดสอบการทำงาน เพื่อที่จะทำได้แบบเวลาจริงจึงต้องมีการนำ GPU เข้ามาช่วยประมวลผลแทนการใช้ CPU อย่างเดียวเพราะการทำเช่นนี้จะช่วยเพิ่มประสิทธิภาพการทำงานให้โปรแกรมเป็นอย่างมากกว่าการใช้เพียง CPU เพียงอย่างเดียว เนื่องจากหากใช้ CPU เพียงอย่างเดียวในการประมวลผลจะทำให้ไม่สามารถทำงานได้ในเวลาจริงซึ่งไม่ตรงตามจุดประสงค์ที่ได้ตั้งไว้ ดังภาพที่ 2.12



รูปที่ 2.12 เปรียบเทียบการประมวลผลระหว่างการใช้ CPU เพียงอย่างเดียวและเมื่อใช้ GPU เข้ามาช่วย

2.2.2 การลบภาพพื้นหลัง

การลบภาพพื้นหลัง เป็นขั้นตอนที่จำเป็นอย่างมากในการทำงานเกี่ยวกับการมองเห็นของคอมพิวเตอร์หรือแอปพลิเคชันที่เกี่ยวข้อง เป็นการจำแนกส่วนของภาพให้ชัดเจนยิ่งขึ้นเพื่อดำเนินขั้นต่อไป หรือกล่าวอีกนัยหนึ่งคือการแยกส่วนของเบื้องหน้าออกจากภาพเบื้องหลังที่คงที่ เพื่อแสดงข้อมูลว่ามีวัตถุใด ๆ อยู่ในภาพหรือเฟรมที่ต้องการ ดังรูปที่ 2.13



รูปที่ 2.13 ขั้นตอนการทำงานของ การแยกพื้นหลังออกจากพื้นหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 คุณลักษณะของรูปภาพ

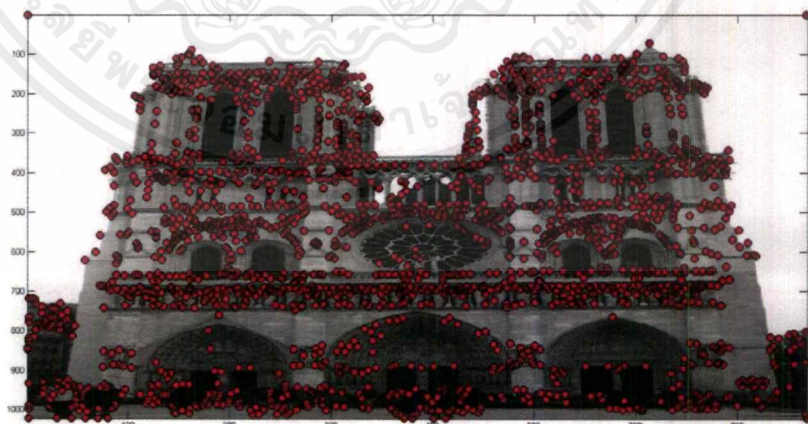
Image Feature คือคุณลักษณะเด่นซึ่งเป็นส่วนที่สำคัญที่ประกอบรวมกันจนเกิดเป็นรูปภาพ ในปัจจุบันเทคโนโลยีทางการประมวลผลภาพ และคอมพิวเตอร์วิทัศน์ได้ให้ความสำคัญในการจัดการกับคุณลักษณะของรูปภาพ คุณลักษณะของภาพถือเป็นองค์ประกอบหลักที่สำคัญในกระบวนการทำงานต่าง ๆ เช่น การรู้จำใบหน้าหรือตัวอักษร การตรวจวัตถุหรือมนุษย์ เป็นต้น ดังนั้นการดึงเอาคุณลักษณะของรูปภาพออกมานั้น ต้องใช้วิธีการที่เหมาะสมและมีประสิทธิภาพเพราะคุณลักษณะที่เหมาะสมจะทำให้การตรวจจับมีความแม่นยำและมีประสิทธิภาพขึ้น ในทางตรงกันข้ามหากค่าคุณลักษณะมีความไม่เหมาะสมยิ่งทำให้เกิดข้อผิดพลาดมากขึ้น คุณลักษณะของรูปภาพแบ่งออกเป็น 3 ลักษณะได้แก่

1) เส้นขอบ (Edges)



รูปที่ 2.14 การสกัดเส้นขอบภาพ

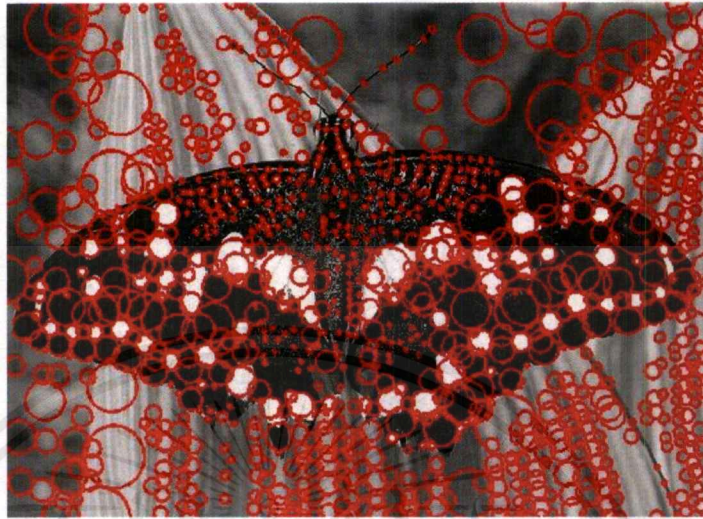
2) มุมหรือจุดที่สนใจ (Corners/Interest point)



รูปที่ 2.15 การสกัดมุมของวัตถุในรูปภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) Blobs หรือพื้นที่ของจุดที่สนใจ (Region of Interest point)



รูปที่ 2.16 การสกัดคุณลักษณะในพื้นที่ที่สนใจ

2.2.3.1 ลักษณะแบบภาพรวม

ลักษณะแบบภาพรวม (Global Feature) คือค่าคุณลักษณะของรูปภาพโดยการหาแบบภาพรวมของรูปภาพทั้งหมด โดยทำการดึงเอาคุณลักษณะจากด้านต่าง ๆ เช่น สี รูปร่าง หรือพื้นผิว ออกมาในรูปแบบของค่าตัวเลขเวกเตอร์ค่าหนึ่ง เช่น ถ้าต้องการแยกความแตกต่างระหว่างภาพทะเลทรายและภาพภูเขาหิมะ ค่าคุณลักษณะที่นำมาใช้ในการจำแนกคือค่าของสีในแต่ละภาพ ข้อดีของการเลือกใช้ลักษณะแบบภาพรวมคือสามารถประมวลผลได้รวดเร็วและใช้ทรัพยากรน้อย ข้อเสียคือไม่เหมาะสำหรับรูปภาพเคลื่อนไหวและภาพที่มีความยุ่งเหยิงและบิดเบี้ยว



รูปที่ 2.17 คุณลักษณะของรูปภาพแบบภาพรวม (Global Feature)

2.2.3.2 ลักษณะเฉพาะที่

ลักษณะเฉพาะที่ (Local Feature) คือค่าคุณลักษณะของรูปภาพโดยการหาจากพื้นที่บางส่วนของรูปภาพเรียกว่า “พื้นที่ส่วนที่สนใจ (Region of Interest)” ค่าคุณลักษณะได้ถูกคำนวณด้วยวิธีการเดียวกันกับลักษณะแบบภาพรวม ข้อดีของการคุณลักษณะแบบลักษณะเฉพาะที่ คือเหมาะสำหรับภาพเคลื่อนไหวและทำงานได้ดีในภาพที่มีลักษณะยุ่งยากและบิดเบี้ยว ข้อเสียคือค่อนข้างใช้เวลาในการประมวลผลและเปลืองทรัพยากร



รูปที่ 2.18 คุณลักษณะภาพเฉพาะที่ (Local Feature)

2.2.4 การตรวจจับมนุษย์

เทคโนโลยีการตรวจจับมนุษย์ได้รับความนิยมเป็นอย่างมากในปัจจุบันและถูกนำมาประยุกต์ใช้กับโปรแกรมมากมาย เช่น โปรแกรมเฝ้าระวัง ผู้ช่วยขับยานพาหนะ เป็นต้น อย่างไรก็ตาม การตรวจจับมนุษย์เป็นงานที่มีความท้าทายเนื่องจากลักษณะของมนุษย์มีความซับซ้อนมากกว่าวัตถุอื่น ๆ นอกจากนั้นการตรวจจับแบบเรียลไทม์ต้องอาศัยวิธีการที่มีประสิทธิภาพที่จะนำมาช่วยลดระยะเวลาในการคำนวณและมีความแม่นยำสูง ดังนั้นในการตรวจจับบุคคลภายในภาพเคลื่อนไหวประกอบด้วยสามส่วนหลัก ๆ ได้แก่

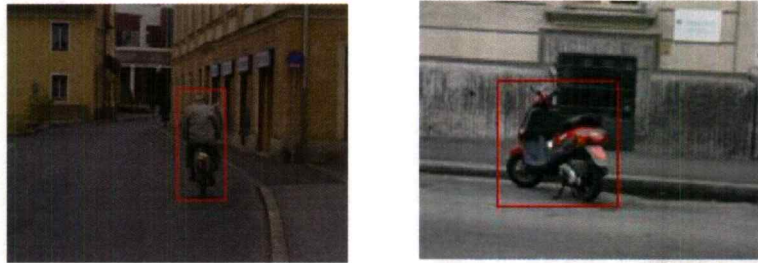
- 1) บริเวณของรูปภาพที่สนใจ (Region of Interest)
- 2) การสกัดคุณลักษณะจากรูปภาพ (Feature Extraction) ของรูปภาพโดยการใช้ Hog Algorithm (Histogram of Oriented Gradients)
- 3) การจำแนกโดยการใช้ SVM (Support Vector Machine)



รูปที่ 2.19 ขั้นตอนในการตรวจจับมนุษย์

2.2.4.1 บริเวณของรูปภาพที่สนใจ

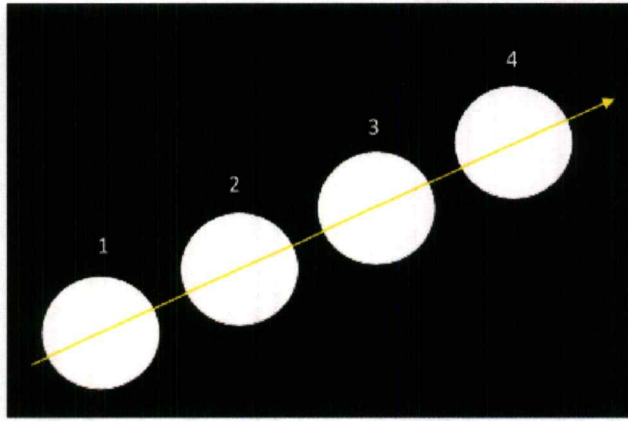
บริเวณของรูปภาพที่สนใจ (ROI) คือการเลือกพื้นที่ที่จะนำมาทำการตรวจจับ วิธีการที่นำมาใช้ในการทำ ROI คือวิธีการเลื่อนหน้าต่าง (Sliding Windows) และวิธีแบบออฟติคัลโฟลว (Optical flow) สำหรับวิธีการเลื่อนหน้าต่าง ในแต่ละหน้าต่างจะมีความกว้างและความยาวที่คงที่ โดยจะทำการเลื่อนหน้าต่างจากมุมบนซ้ายไปจนถึงมุมล่างขวาของรูปภาพ วิธีการนี้ถูกนำมาใช้ในการจำแนกวัตถุ การตรวจจับใบหน้ารวมทั้งการตรวจจับมนุษย์เพราะเป็นวิธีการที่มีความยืดหยุ่นและมีประสิทธิภาพ ในการเลื่อนหน้าต่างแต่ละครั้งภายในหน้าต่างนั้นจะประกอบไปด้วยคุณลักษณะต่าง ๆ อยู่เช่น พื้นผิว ข้อมูลรูปร่างและแนวการไล่ระดับของสีซึ่งจะถูกนำมาสกัดและนำค่าคุณลักษณะที่ได้ไปใช้ในการจำแนกวัตถุ



รูปที่ 2.20 ตัวอย่างการทำ Sliding Windows

การทำ Motion Detection คือการลดทอนพื้นที่ส่วนที่ไม่จำเป็นออกไป โดยจะเลือกตรวจจับเฉพาะพื้นที่ที่มีการเคลื่อนไหวระหว่างเฟรมรูปภาพปัจจุบันกับเฟรมรูปภาพก่อนหน้า วิธีการนี้จะช่วยลดระยะเวลาในการคำนวณส่งผลให้การตรวจจับมีความรวดเร็วมากยิ่งขึ้น เพราะจะทำการคำนวณเพียงแค่บริเวณที่เคลื่อนไหวและนำค่าคุณลักษณะที่ได้ไปทำการจำแนกว่าเป็นมนุษย์ที่มีท่าทางผิดปกติ มนุษย์ทั่วไปหรือวัตถุอื่น ๆ ในบางครั้ง ในส่วนพื้นที่ที่ไม่มีการเคลื่อนไหวจะไม่ถูกคำนวณ เพราะจะถือว่าพื้นที่ในส่วนดังกล่าวเป็นเพียงแค่พื้นหลังเท่านั้นบริเวณพื้นที่ที่เคลื่อนไหวอาจมีบุคคลอยู่หลายคนเนื่องจากระยะห่างระหว่างพวกเขาไม่น้อยมากหรือมีวัตถุอื่นอยู่ร่วมกับมนุษย์เพราะทั้งสองสิ่งมีการเคลื่อนไหวที่ซ้อนทับกัน วิธีการเลื่อนหน้าต่างและอัลกอริทึมของการตรวจจับมนุษย์สามารถแบ่งแยกสิ่งเหล่านี้ออกจากกันได้

วิธีแบบออปติคัลโฟลว เป็นการกำหนดบริเวณที่สนใจของภาพจากกรณีที่วัตถุมีการเคลื่อนไหว โดยการใช้หลักการทางคณิตศาสตร์แทนที่การเคลื่อนไหวของวัตถุในภาพ ระหว่างเฟรมภาพที่ต่อเนื่องกัน กำหนดให้แต่ละเฟรมภาพเป็นพื้นที่เวกเตอร์ 2 มิติ โดยที่การเคลื่อนไหวของวัตถุของเฟรมที่ต่อกันจะมีการเคลื่อนออก



รูปที่ 2.21 ภาพแสดงการเคลื่อนไหวของลูกบอลในเฟรมภาพที่ต่อเนื่องกัน 4 เฟรม

วิธีแบบออปติคัลโฟลวทำงานบนสมมติฐานที่ว่า ความหนาแน่นของจุดพิกเซลส์ในวัตถุจะไม่มีเปลี่ยนแปลง และจุดพิกเซลส์ที่อยู่ใกล้เคียงกันจะมีการเคลื่อนไหวคล้าย ๆ กัน

การคำนวณด้วยวิธีแบบออปติคัลโฟลว จะสมมติให้จุดพิกเซลส์ใด ๆ เป็นค่าหนึ่ง และให้กำหนดระยะทางที่จุดพิกเซลส์มีการเคลื่อนไหวในเฟรมถัดไป จะได้สมการ 2.1

$$I(x,y,t) = I(x+dx,y+dy,t+dt) \quad (2.1)$$

เมื่อ $I(x,y,t)$ คือ จุดพิกเซลส์ที่สนใจ

dx, dy คือ ระยะทางที่จุดพิกเซลส์เคลื่อนที่ในเฟรมถัดไป

dt คือ เวลาที่จุดพิกเซลส์เคลื่อนไปยังอีกจุดหนึ่งในเฟรมที่ต่อเนื่องกัน

หลังจากนั้นใช้อนุกรมเทเลอร์ในการคำนวณจนได้สมการที่ 2.2

$$f_x U + f_y V + f_t = 0 \quad (2.2)$$

$$\text{เมื่อ } f_x = \frac{\partial f}{\partial x}, f_y = \frac{\partial f}{\partial y}$$

$$U = \frac{dx}{dt}, V = \frac{dy}{dt}$$

จากสมการ 2.2 เรียกว่าสมการออฟดิคัลโฟลว เราสามารถหา f_x และ f_y ได้จากการหาความชันของภาพ ส่วน f_t คือความชันของเวลา แต่ U, V นั้นเราไม่ทราบค่า เนื่องจากสมการ 2.2 เป็นสมการสองตัวแปรแต่เรามีสมการเดียวทำให้ไม่สามารถแก้สมการได้ วิธีที่ใช้ในการแก้ปัญหาหน้าคือวิธีของลูคัสและคาเนต

วิธีของลูคัสและคาเนต จากสมมติฐานของออฟดิคัลโฟลว วิธีของลูคัสและคาเนตจะทำการนำจุดรอบ ๆ จุดพิกเซลที่ต้องการนั้น มาใช้ในการคำนวณและเนื่องจากจุดพิกเซลที่อยู่ใกล้เคียงกันจะมีการเคลื่อนไหวคล้าย ๆ กัน ทำให้เราจะได้สมการมาทั้งหมด 9 สมการ 2 ตัวแปรทำให้เราสามารถทราบค่า U และ V ได้

วิธีแบบออฟดิคัลโฟลวในกรณีที่คุณลักษณะในภาพมีหนาแน่น วิธีของลูคัสและคาเนตถูกใช้ในกรณีที่ภาพนั้นมีคุณลักษณะน้อย สำหรับปัญหาคุณลักษณะในภาพหนาแน่น จะใช้วิธีคำนวณทุกจุดในเฟรมภาพจากนั้นจะสร้างอาร์เรย์ 2 ช่องที่ใช้ในการเก็บค่า U และ V จากนั้นจะคำนวณค่าที่สำคัญและเส้นทางการเคลื่อนที่ของวัตถุ

2.2.4.2 การสกัดคุณลักษณะจากรูปภาพ

การสกัดคุณลักษณะคือการแสดงคุณสมบัติที่โดดเด่นออกมาจากรูปภาพโดยใช้กระบวนการคณิตศาสตร์เป็นค่าบ่งชี้คุณลักษณะดังกล่าว ในการดึงเอาค่าคุณลักษณะภายในรูปภาพนั้นต้องใช้อัลกอริทึมที่เหมาะสมเพราะค่าคุณลักษณะที่เหมาะสมจะนำไปสู่การตรวจจับที่มีประสิทธิภาพและมีความแม่นยำ หากค่าคุณลักษณะที่นำมาใช้ไม่มีความเหมาะสมก็จะทำให้เกิดข้อผิดพลาด ดังนั้นการเลือกใช้วิธีการในการดึงเอาค่าคุณลักษณะออกมาจากรูปภาพจึงเป็นสิ่งสำคัญหลักในการทำการตรวจจับมนุษย์ ฮิสโตแกรมของการไล่ระดับเชิงมุม (Histogram of Oriented Gradients เป็นอัลกอริทึมที่ถูกเลือกมาใช้ในการพัฒนาระบบเพราะเป็นอัลกอริทึมที่มีประสิทธิภาพ มีความแม่นยำและประยุกต์ใช้ได้จริง

ฮิสโตแกรมของการไล่ระดับเชิงมุม (Histogram of Oriented Gradients)

ฮิสโตแกรมของการไล่ระดับเชิงมุมคือการสกัดเอาคุณลักษณะเด่นของพื้นที่ที่สนใจออกมาเป็นค่าคุณลักษณะค่าหนึ่ง ซึ่งค่าคุณลักษณะค่านั้นจะได้มาจากการหาขนาดและทิศทางของการไล่ระดับสี ค่าคุณลักษณะที่ได้จากกระบวนการนี้จะถูกนำไปใช้ในกระบวนการจำแนกวัตถุอินพุตของอัลกอริทึมนี้คือรูปภาพที่จะทำการจำแนก ส่วนเอาท์พุต คือค่าเวกเตอร์ที่เป็น

ค่าคุณลักษณะของรูปภาพ กระบวนการทำงานของฮิตโทแกรมของการไล่ระดับเชิงมุมแบ่งออกเป็น 3 ขั้นตอนคือ

- 1) **Gradient Computation** เป็นกระบวนการในการจัดการกับรูปภาพโดยการปรับความเข้มหรือสีของภาพ ผลลัพธ์ที่ได้จากวิธีการดังกล่าวจะลดความซับซ้อนของรูปภาพซึ่งช่วยลดระยะเวลาในการคำนวณในขั้นตอนต่อไป รูปภาพที่ได้จะปรากฏขอบของวัตถุอย่างชัดเจนและพื้นหลังจะถูกเพิ่มความเข้มจนกลมกลืนกัน การปรับความเข้มหรือสีของภาพจะทำได้โดยการนำแม่แบบตัวอย่าง $[-1,0,1]$ เข้าไปคำนวณทุก ๆ พิกเซลในแนวตั้งและแนวนอนของรูปภาพ

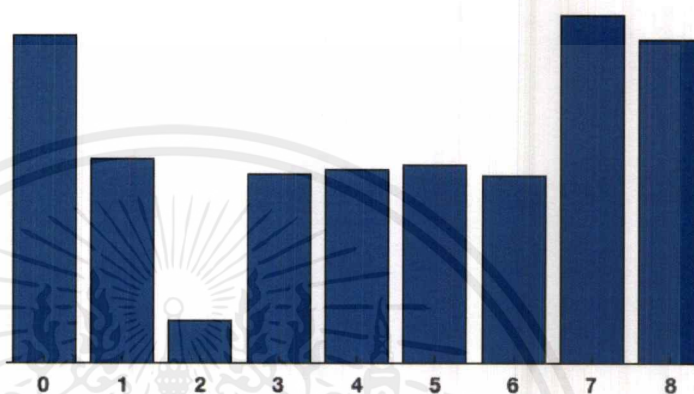


รูปที่ 2.22 ภาพแสดงการไล่ระดับเชิงมุม

ภาพซ้าย : ค่าสัมบูรณ์ของการไล่ระดับในแนวแกน x, ภาพกลาง : ค่าสัมบูรณ์ของการไล่ระดับสีในแนวแกน y, ภาพขวา : การไล่ระดับสีแบบเชิงมุม

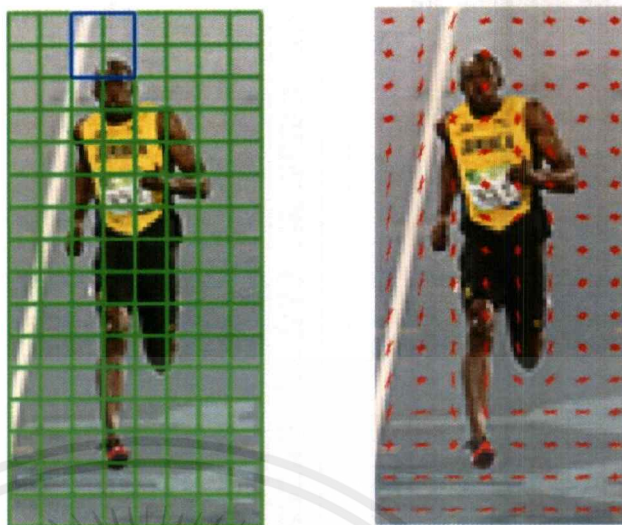
- 2) **Spatial / Orientation Binning** เป็นการคำนวณค่าในแต่ละพิกเซลของรูปภาพ โดยรูปภาพจะถูกแบ่งออกเป็นส่วนย่อย ๆ ที่เรียกว่า “เซลล์” ในแต่ละเซลล์จะมีขนาด 8×8 พิกเซล การคำนวณค่าในแต่ละพิกเซลจะแบ่งออกเป็นสองค่า คือ ค่าขนาดและองศาของการไล่ระดับสี ดังนั้นจำนวนครั้งในการคำนวณค่าพิกเซลจะเท่ากับ $8 \times 8 \times 2 = 128$ ค่าเวกเตอร์ที่ได้จะถูกนำไปใส่ภายในฮิตโทแกรม 9 ช่อง ฮิตโทแกรมจะไล่ระดับองศาตั้งแต่ $0 - 180$ แต่ละช่องจึงมีความกว้างเท่ากับ 20 องศา

ไล่ระดับมีค่าเท่ากับ 80 และในช่องวงกลมสีฟ้ามีขนาดเท่ากับ 2 จึงนำค่า 2 มาใส่ไว้ในอิทธิโทแกรมช่องที่ 5 แต่ถ้าหากค่าขนาดของการไล่ระดับอยู่ที่ขอบชั้นของอิทธิโทแกรม ตัวอย่างเช่น ในช่องวงกลมสีแดงของค่ามุมของการไล่ระดับมีค่าเท่ากับ 10 องศา ซึ่งอยู่ระหว่างช่วงที่ 1 และ 2 จะทำการแบ่งค่าขนาดของการไล่ระดับออกเป็นสองส่วนอย่างละครึ่งแล้วจึงนำไปใส่ไว้ในอิทธิโทแกรม



รูปที่ 2.25 ลักษณะอิทธิโทแกรมหลังจากผ่านการคำนวณ

3) **Normalization and Descriptor Blocks** ช่วงความเข้มของการไล่ระดับค่อนข้างแตกต่างกันมากเนื่องมาจากแสงเงาและความคมชัดของรูปภาพซึ่งส่งผลให้เกิดค่าที่ผิดเพี้ยน จึงต้องทำการปรับค่าความเข้มของการไล่ระดับสีให้อยู่ในค่าที่เป็นมาตรฐาน โดยทำการคำนวณครั้งละ 4 เซลล์เรียกว่า “บล็อก” ค่าพิกเซลที่อยู่ภายในเซลล์จะถูกปรับให้เป็นค่าที่เหมาะสม วิธีการเลื่อนของบล็อกจะทำการเลื่อนทีละครั้งบล็อกเพื่อให้พื้นที่ในเซลล์ของบล็อกก่อนหน้าได้ถูกคำนวณในบล็อกใหม่

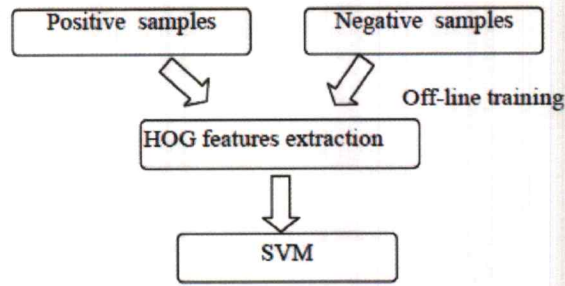


รูปที่ 2.26 การทำ Normalization และแสดงแนวการไล่ระดับ

2.2.4.3 การจำแนกหมวดหมู่โดยการใช้ซัพพอร์ตเวกเตอร์แมชชีน

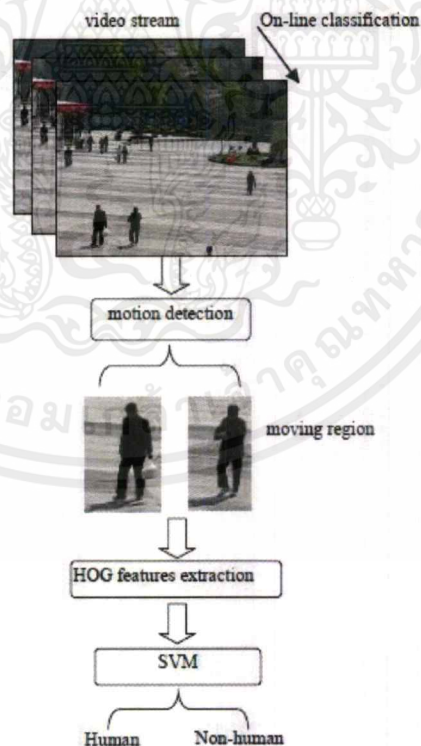
Support Vector Machine (SVM) เป็นส่วนหนึ่งของรูปแบบการเรียนรู้แบบมีผู้สอนในด้านการเรียนรู้ของเครื่อง (Machine Learning) ซึ่งช่วยในการจำแนกหมวดหมู่ของข้อมูลที่น่ามาฝึกสอน ในการตรวจจับมนุษย์โดยการใช้วิธีการ SVM จะแบ่งขั้นตอนออกเป็น 2 ขั้นตอนหลัก ดังนี้

- 1) การเรียนรู้ค่าคุณลักษณะ (Feature Learning) เป็นการนำข้อมูลมาทำการฝึกสอนได้แก่รูปภาพบุคคลในท่าทางและลักษณะต่าง ๆ เรียกว่า “Positive Image” และภาพพื้นหลังทั่วไปที่ไม่มีบุคคลอยู่เรียกว่า “Negative Image” รูปภาพทั้งหมดจะถูกคำนวณหาคุณลักษณะ (Feature) โดยใช้ฮิสโตแกรมของการวัดระดับเชิงมุม (Histogram of Oriented Gradients) ค่าคุณลักษณะที่ได้จะถูกจำแนกโดยวิธีการ Support Vector Machine และจะทำการจัดเก็บค่าไว้เพื่อนำมาใช้ในการจำแนกในขั้นตอนต่อไป



รูปที่ 2.27 การเตรียมข้อมูลเข้าสู่กระบวนการเรียนรู้ด้วยเทคนิค SVM

2) การตรวจจับมนุษย์ (Human Detection) คือการรับภาพเข้ามาในระบบ จากนั้นจะทำการสกัดคุณลักษณะ (Feature Extraction) จากรูปภาพดังกล่าว วิธีการในการสกัดคุณลักษณะจะใช้วิธีการเดียวกันกับการเรียนรู้คุณลักษณะ คือ ฮิตโทแกรมของการวัดระดับเชิงมุม โดยค่าคุณลักษณะที่ได้นั้นจะถูกนำมาเปรียบเทียบกับค่าคุณลักษณะจากข้อมูลที่ถูกฝึก และทำการจำแนกว่าภายในภาพดังกล่าวมีมนุษย์อยู่หรือไม่



รูปที่ 2.28 ลำดับขั้นตอนการตรวจจับมนุษย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

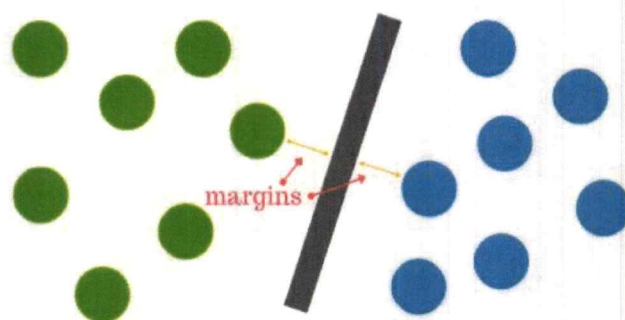
2.3 การเรียนรู้ของเครื่อง

Machine Learning หรือการเรียนรู้ของเครื่อง เป็นการทำให้ระบบคอมพิวเตอร์เรียนรู้ได้ด้วยตนเองโดยการใส่ข้อมูลเข้าไป จากนั้นคอมพิวเตอร์จะทำการเรียนรู้ด้วยวิธีต่าง ๆ เพื่อสร้างโมเดลที่คอมพิวเตอร์จะสามารถเรียนรู้และทำนายข้อมูลเองได้ การเรียนรู้ของเครื่องแบ่งออกได้เป็นสามประเภท คือ 1) การเรียนรู้แบบมีผู้สอน (Supervised learning) เป็นการเรียนรู้โดยที่ผู้สอนทำการใส่ข้อมูล และผลเฉลยให้คอมพิวเตอร์เพื่อทำการเรียนรู้และสร้างโมเดลในการทำนายออกมา 2) การเรียนรู้แบบไม่มีผู้สอน (Unsupervised learning) เป็นการเรียนรู้โดยการให้คอมพิวเตอร์จัดกลุ่มข้อมูลฝึกสอนเอง แล้วจากนั้นจึงทำการสร้างรูปแบบในการทำนายขึ้น 3) การเรียนรู้แบบเสริมกำลัง (Reinforcement learning) เป็นการเรียนรู้โดยใช้รางวัล (Reward) เป็นตัวกำหนดรูปแบบการเรียนรู้ขึ้นมา

2.3.1 ซัพพอร์ตเวกเตอร์แมชชีน

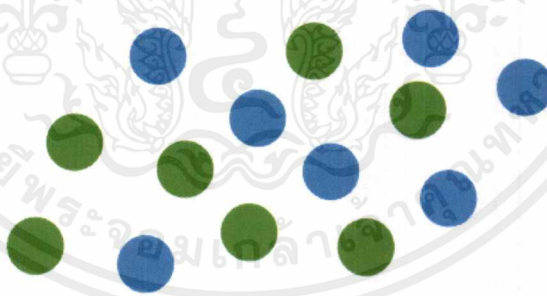
เป็นโมเดลการเรียนรู้ของเครื่องแบบมีผู้สอน ที่สามารถใช้ได้ในการจำแนกหมวดหมู่ อย่างไรก็ตามขั้นตอนวิธีนี้ถูกใช้ในปัญหาการแยกหมวดหมู่เป็นจำนวนมาก โดยการดำเนินการจะทำการเขียนข้อมูลแต่ละตัวลงไปเป็นจุดในกราฟรูปแบบจำนวนมิติหนึ่ง ๆ ซึ่งจำนวนมิติจะขึ้นอยู่กับจำนวนคุณลักษณะของข้อมูล โดยที่ค่าของคุณลักษณะแต่ละค่าจะมีพิกัดเฉพาะตัว จากนั้นก็จะทำการจำแนกหมวดหมู่ด้วยการทำไฮเปอร์เพลนที่จะแยกคลาสของคุณลักษณะที่จุดลงไปบนกราฟดังกล่าว

ไฮเปอร์เพลนคือตัวแบ่งแยกคลาสของชุดข้อมูลที่ต้องการจำแนกหมวดหมู่โดยที่ไฮเปอร์เพลนจะอยู่ห่างจากชุดของข้อมูลที่ต้องการแบ่งแยกให้มากที่สุดแต่ยังคงแบ่งแยกชุดข้อมูลออกมาได้อย่างถูกต้อง ดังรูปที่ 2.29 ระยะห่างระหว่างไฮเปอร์เพลน (แสดงเป็นเส้นตรงสีเทา) และจุดข้อมูลที่ใกล้เคียงที่สุด (แสดงเป็นวงกลมสีเขียวและสีฟ้า) จะถูกเรียกว่า มาร์จิน เป้าหมายคือการเลือกไฮเปอร์เพลนที่ค่ามาร์จินสูงสุดเท่าที่เป็นไปได้ระหว่างไฮเปอร์เพลนและตำแหน่งข้อมูลใด ๆ บนชุดข้อมูล ทำให้ข้อมูลใหม่ที่เข้ามามีโอกาสถูกจำแนกหมวดหมู่ได้ถูกต้องมากขึ้น

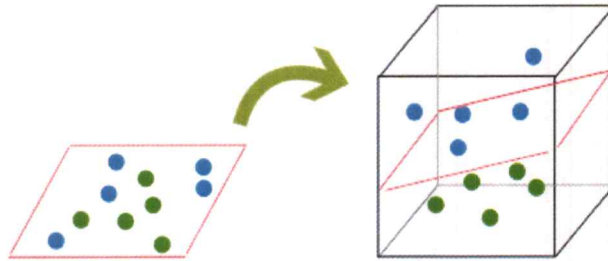


รูปที่ 2.29 ภาพแสดงการวัดมาร์จินระหว่างไฮเพอร์เพลนและตำแหน่งข้อมูลใกล้เคียง

ในกรณีที่ข้อมูลมีความซับซ้อนและไม่สามารถค้นหาไฮเพอร์เพลนที่แนบชิดได้เนื่องจากข้อมูลไม่สามารถคำนวณแบบเชิงเส้นได้ ดังรูปที่ 2.30 ทำให้ต้องทำการเปลี่ยนแปลงวิธีการคำนวณจากรูปแบบข้อมูล 2 มิติ ไปเป็นรูปแบบของข้อมูล 3 มิติ ดังรูปที่ 2.31 เพื่อให้ชุดข้อมูลทั้งสองชุดแสดงตำแหน่งที่ชัดเจนมากยิ่งขึ้นทำให้สามารถค้นหาไฮเพอร์เพลนได้ วิธีดังกล่าวนี้เรียกว่าการใช้ เคอร์เนล ดังนั้นจึงต้องมีการปรับเปลี่ยนเป็นรูปแบบเดียวกับข้อมูลคือ 3 มิติ (แสดงเป็นกรอบสี่เหลี่ยมสีแดง) และข้อมูลที่เข้ามาใหม่หรือหากยังไม่สามารถค้นหาไฮเพอร์เพลนที่ถูกต้องได้ ข้อมูลจะถูกจัดให้อยู่ในมิติที่สูงขึ้นจนกว่าจะสามารถค้นหาไฮเพอร์เพลนที่จำแนกชุดข้อมูลได้อย่างถูกต้องออกมา



รูปที่ 2.30 ชุดข้อมูลที่มีความซับซ้อนและไม่เป็นเชิงเส้น



รูปที่ 2.31 การเปลี่ยนชุดข้อมูล 2 มิติไปเป็นรูปแบบ 3 มิติ

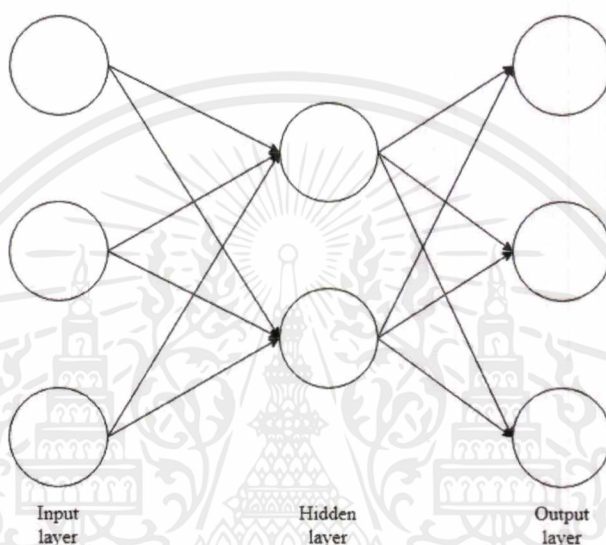
เคอร์เนลในเทคนิคซัพพอร์ตเวกเตอร์ที่นำมาใช้ในการพัฒนาระบบเฝ้าระวังอัจฉริยะ นำมาใช้ทั้งหมด 4 เคอร์เนล ได้แก่ ลิเนียร์ (Linear) เป็นการแบ่งข้อมูลโดยใช้เส้นตรง โพลีโนเมียล (Polynomial) เป็นการแบ่งข้อมูลโดยใช้เส้นโค้ง อาร์บีเอฟ (RBF) เป็นการแบ่งข้อมูลโดยการสร้างไฮเปอร์เพลนมากำหนดเขตในการแบ่งข้อมูล โคสแควร์ (CHI2) เป็นการแบ่งข้อมูลโดยการหาความสัมพันธ์ของตัวแปร 2 ตัวแล้วนำมาเปรียบเทียบกัน



รูปที่ 2.32 รูปภาพแสดงการแบ่งข้อมูลโดยใช้เคอร์เนล ลิเนียร์ โพลีโนเมียล และอาร์บีเอฟตามลำดับ

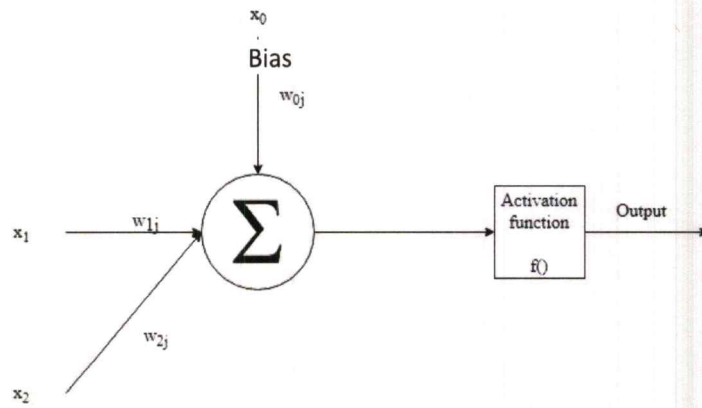
2.3.2 โครงข่ายประสาทเทียม

เป็นหนึ่งในวิธีการเรียนรู้แบบมีผู้สอน โครงข่ายประสาทเทียมเป็นการเลียนแบบการทำงานของสมองมนุษย์ โดยจะทำการแปลงข้อมูลที่ต้องการให้อยู่ในรูปตัวเลข และนำมาคำนวณ เพื่อทำการแบ่งกลุ่มข้อมูล โครงข่ายประสาทเทียมประกอบไปด้วยชั้นอินพุต (Input layer) 1 ชั้น ชั้นซ่อน (Hidden layer) และชั้นเอาต์พุต (Output layer) 1 ชั้น บางกรณีอาจไม่มีชั้นซ่อนและชั้นซ่อนอาจจะมีได้มากกว่า 1 ชั้น

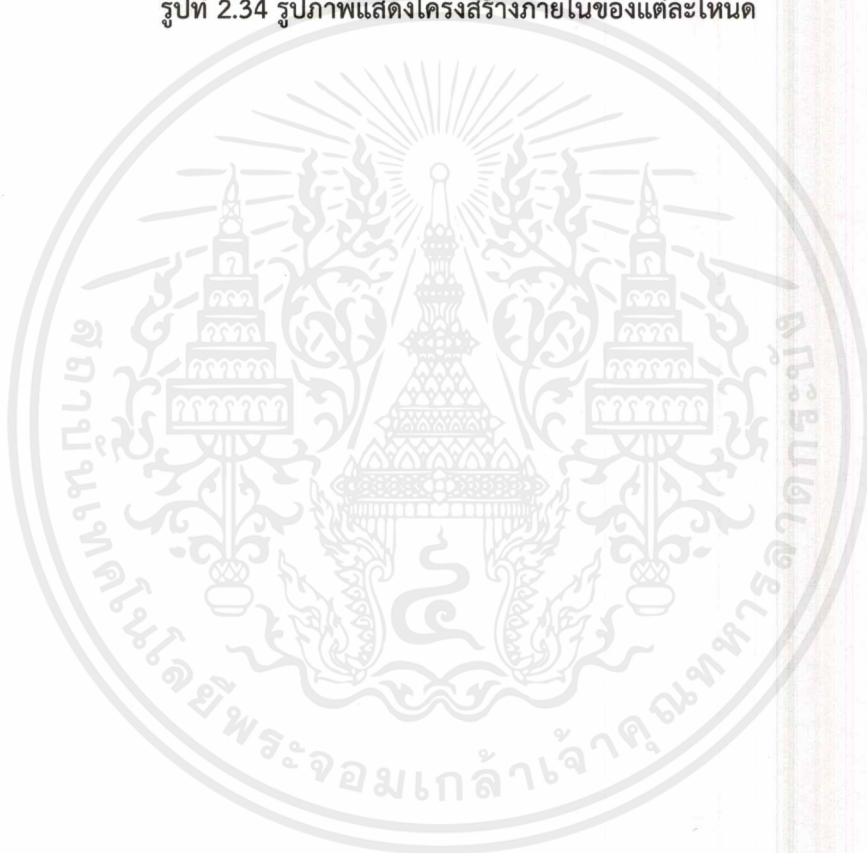


รูปที่ 2.33 รูปภาพแสดงชั้นการทำงานของโครงข่ายประสาทเทียม

ในแต่ละชั้นของโครงข่ายประสาทเทียม จะประกอบไปด้วยโหนด โดยในแต่ละโหนดจะมีโครงสร้างภายในประกอบไปด้วย ค่า x_i เป็นค่าที่นำเข้ามาในโหนด ค่า w_{ij} เป็นค่าน้ำหนัก ทุกค่าที่เข้ามาในโหนดจะต้องถูกคูณกับค่าน้ำหนัก จากนั้นนำค่าที่ได้มาผ่าน Transfer function (Σ) แล้วนำคำตอบที่ได้มาคำนวณใน Activation function ($f()$) คำตอบที่ได้จาก $f()$ เป็นการแบ่งประเภทของข้อมูลและจะถูกส่งต่อไปยังชั้นถัดไปเพื่อนำไปคำนวณต่อในชั้นถัดไป (ถ้ามี)



รูปที่ 2.34 รูปภาพแสดงโครงสร้างภายในของแต่ละโหนด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

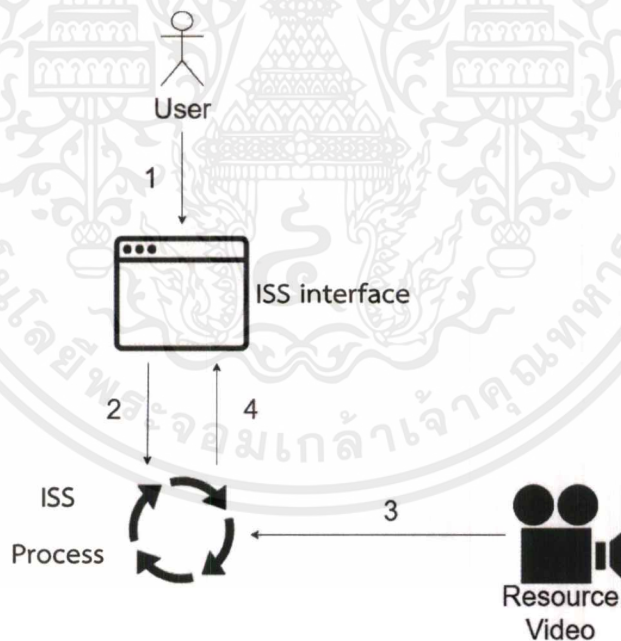
บทที่ 3

การวิเคราะห์และออกแบบระบบ

ในส่วนของการวิเคราะห์และออกแบบระบบ ระบบเฝ้าระวังอัจฉริยะได้แบ่งการทำงานในส่วนนี้ออกเป็นสามส่วน ได้แก่ สถาปัตยกรรมระบบ การวิเคราะห์ระบบ และการสร้างโมเดลตรวจจับท่าทางผิดปกติ

3.1 สถาปัตยกรรมระบบ

ระบบเฝ้าระวังอัจฉริยะ (Intelligent Surveillance System (ISS)) เป็นระบบที่ใช้ในการตรวจจับการเคลื่อนไหวที่ผิดปกติของมนุษย์ในวิดีโอ ซึ่งระบบจะแบ่งการทำงานหลักได้ดังนี้ ตรวจจับวิดีโอ สามารถตรวจจับการเคลื่อนไหวที่ผิดปกติของบุคคลจากไฟล์วิดีโอที่ใส่เข้าไปในระบบ หรือไฟล์จากกล้องแบบเรียลไทม์ได้ และแสดงผลลัพธ์ของการตรวจจับให้กับผู้ใช้ การแจ้งเตือน ระบบจะมีการแจ้งเตือนให้แก่ผู้ใช้กรณีที่มีการตรวจจับการเคลื่อนไหวที่ผิดปกติได้



รูปที่ 3.1 สถาปัตยกรรมของระบบ

จากรูปที่ 3.1 สามารถอธิบายขั้นตอนการทำงานตามลำดับดังต่อไปนี้

- 1) ผู้ใช้งานเปิดเข้าใช้งานระบบผ่านทางคอมพิวเตอร์
- 2) ระบบจะแสดงส่วนเมนูการใช้งานขึ้นมาให้ผู้ใช้งานเลือกการทำงาน
- 3) นำวิดีโอที่ต้องการตรวจจับเข้าสู่ระบบ โดยวิดีโอที่เข้ามานั้น อาจจะเป็นวิดีโอที่บันทึกไว้หรือวิดีโอจากกล้องแบบเรียลไทม์
- 4) ระบบทำการประมวลผลแล้วส่งผลลัพธ์ให้แก่ผู้ใช้งาน

3.2 การวิเคราะห์ระบบ

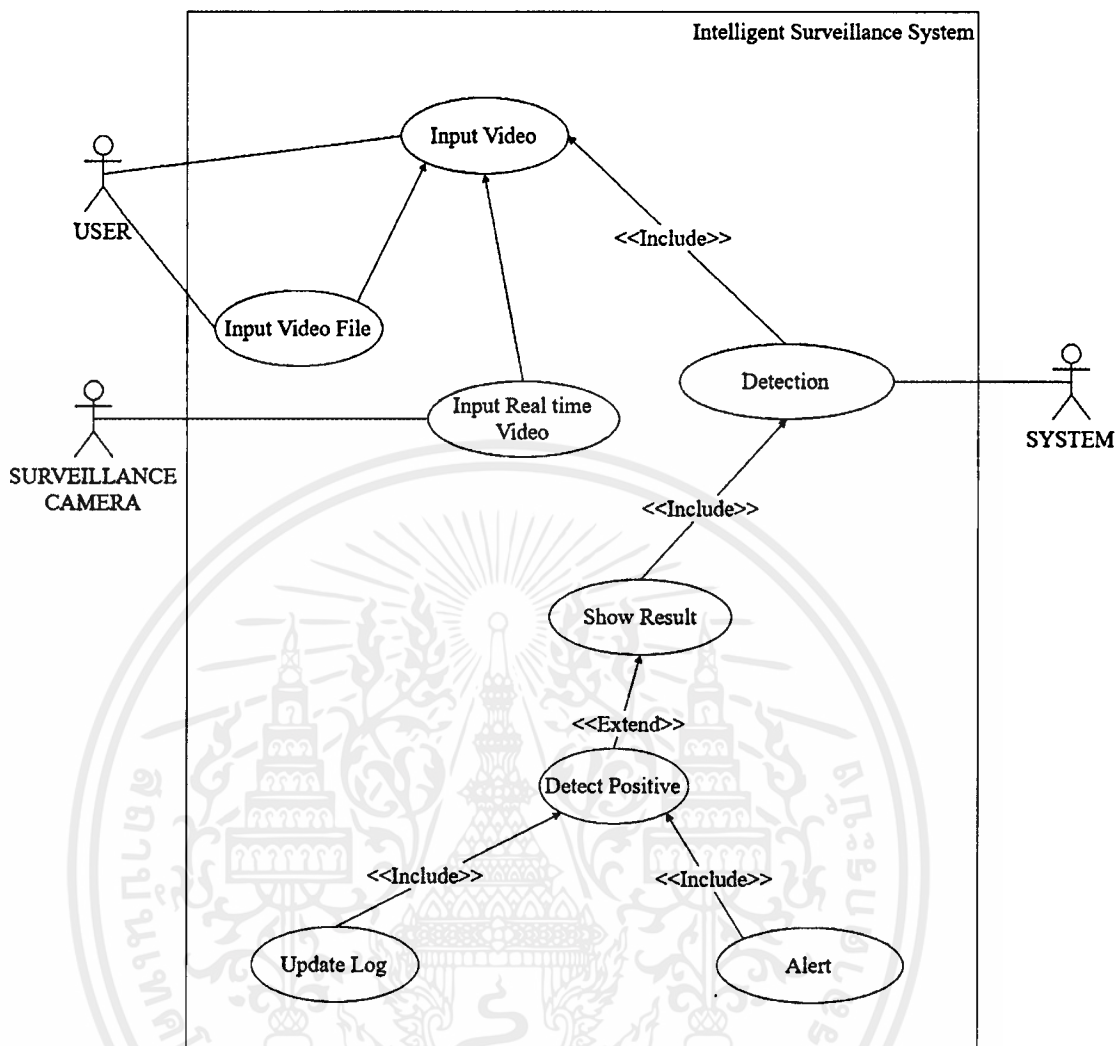
3.2.1 Functional Requirement

- 1) โปรแกรมเป็นโปรแกรมออฟไลน์บนระบบปฏิบัติการ Windows
- 2) Detection (Video File) เป็นฟังก์ชันที่ผู้ใช้งานสามารถเลือกไฟล์วิดีโอจากผู้ใช้ที่ต้องการตรวจจับลงไปที่ระบบ
- 3) Detection (Real time Video) เป็นฟังก์ชันที่ผู้ใช้งานเลือกตรวจจับวิดีโอแบบเรียลไทม์จากกล้อง
- 4) Info เป็นส่วนแสดงการแจ้งเตือนที่มีการแจ้งเตือนไปที่ผู้ใช้ เมื่อมีการตรวจจับการเคลื่อนไหวที่ผิดปกติได้

3.2.2 Non - Functional Requirement

- 1) Usability ระบบมีความซับซ้อนน้อย ทำให้ผู้ใช้งานสามารถใช้งานได้โดยง่าย
- 2) Reliability ระบบจะมีข้อผิดพลาดในระดับที่ผู้ใช้งานพอยอมรับได้
- 3) Efficiency ระบบสามารถทำงานแบบเรียลไทม์ได้อย่างมีประสิทธิภาพ
- 4) Robustness ระบบทำงานได้อย่างถูกต้องและแม่นยำ และสามารถกู้กลับคืนได้รวดเร็วในกรณีที่หยุดทำงานไป

3.2.3 Use case Diagram



รูปที่ 3.2 Use case diagram ของระบบเฝ้าระวังอัจฉริยะ

จากรูปที่ 3.2 สามารถอธิบายได้ว่า

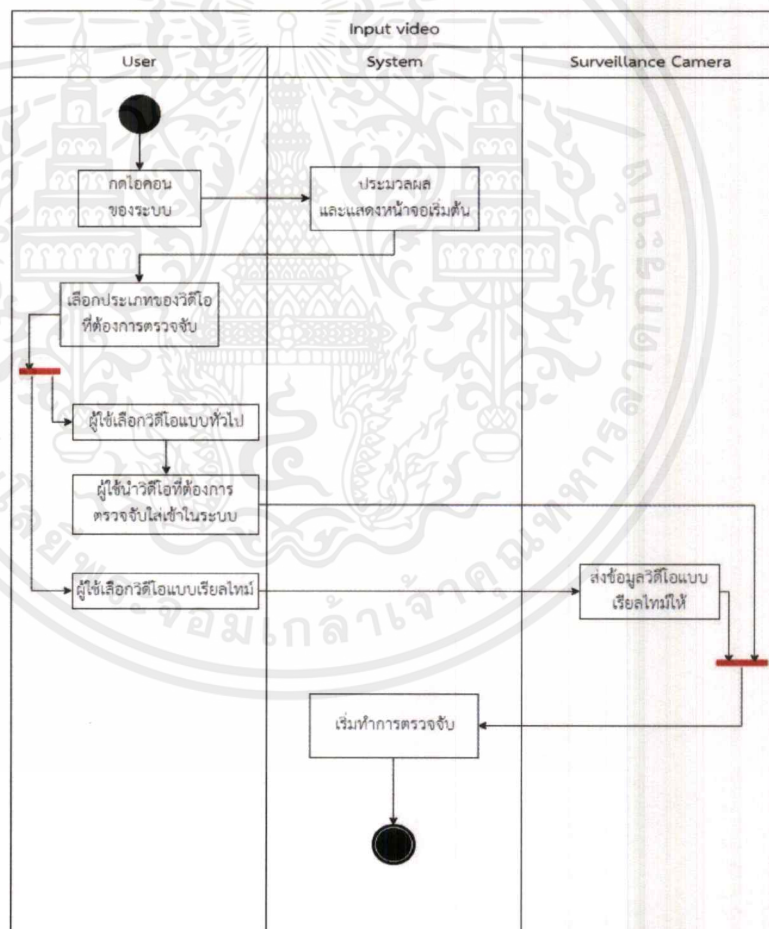
- 1) Input Video เป็นฟังก์ชันที่ผู้ใช้ทำการเลือกประเภทของวิดีโอที่ต้องการตรวจจับเข้าสู่ระบบ โดยแบ่งประเภทของวิดีโอเป็น 2 ประเภท คือ วิดีโอทั่วไปที่ผู้ใช้งานต้องการตรวจจับ (Input Video File) กับวิดีโอแบบเรียลไทม์ที่ได้จากกล้องวงจรปิด (Input Real time Video)

- 2) หลังจากที่ผู้ใช้ทำการเลือกประเภทของวิดีโอที่ทำการตรวจจับ ระบบจะทำการนำวิดีโอที่ได้รับมาทำการตรวจจับ (Detection) และแสดงผลลัพธ์ของการตรวจจับ (Show Result)
- 3) หากระบบตรวจจับเจอท่าทางผิดปกติ (Detect Positive) ระบบจะทำการแจ้งเตือนผู้ใช้ (Alert) และจะทำการบันทึกการตรวจจับไว้ให้ผู้ใช้ดูรายละเอียดการตรวจจับ (Update log)

3.2.4 Activity Diagram

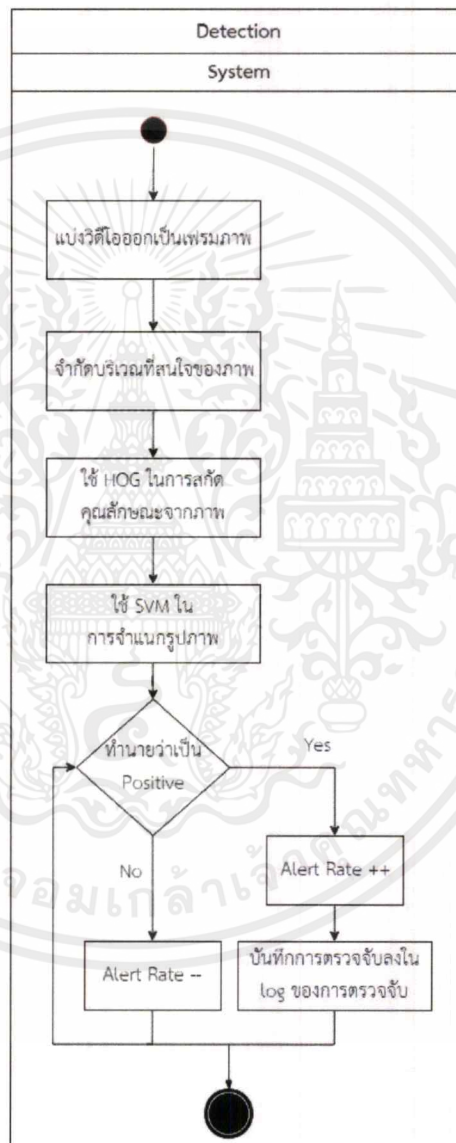
ภายในหน้าจอกำหนดการทำงานหลักจะพบกับฟังก์ชันการทำงานหลัก 3 ฟังก์ชัน ซึ่งสามารถแบ่ง Activity Diagram ได้ดังนี้

- 1) Activity Diagram ของการ Input video : ระบบจะให้ผู้ใช้เลือกประเภทของวิดีโอที่ต้องการตรวจจับเข้าสู่ระบบ



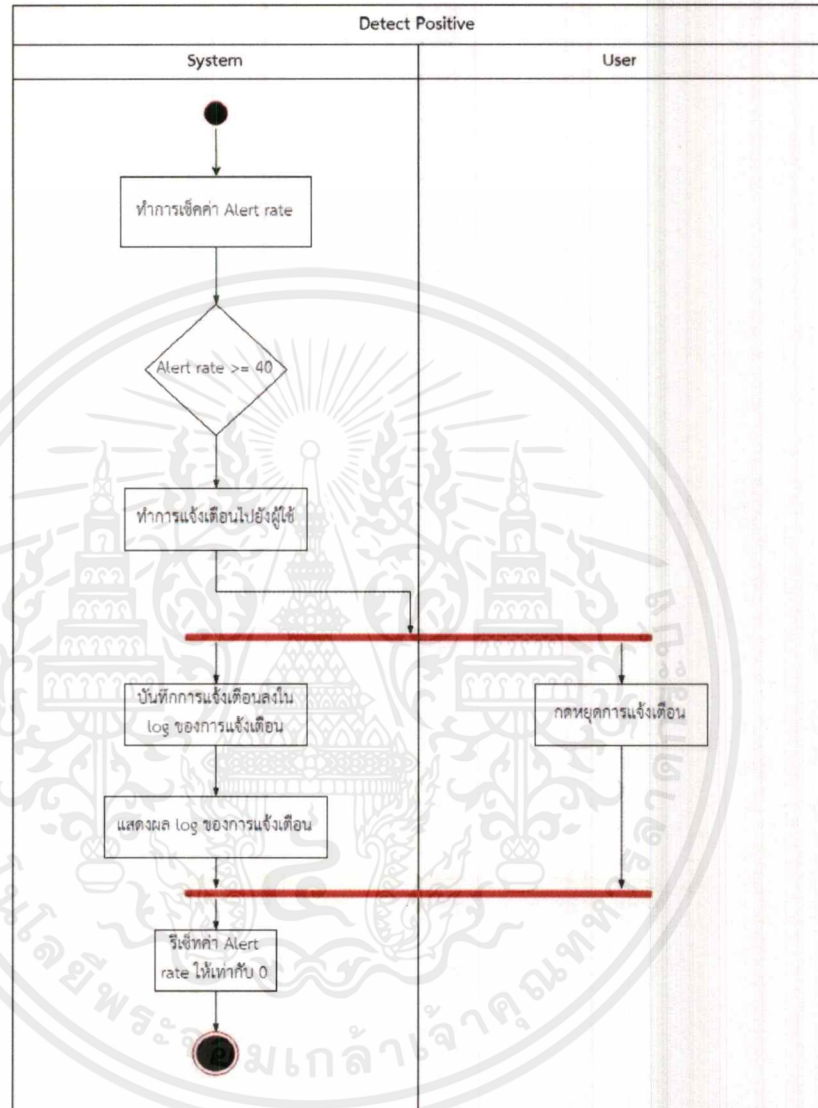
รูปที่ 3.3 Activity Diagram ของการ Input video

- 2) Activity Diagram ของ Detection : ขั้นตอนการตรวจจับหลังจากที่ระบบได้รับวิดีโอเข้ามาแล้ว กำหนดค่า Alert rate เป็นค่าของอัตราการเกิดเหตุการณ์ผิดปกติ ซึ่งจะเพิ่มขึ้นก็ต่อเมื่อมีการทำนายเฟรมรูปภาพเป็นสถานการณ์ผิดปกติและค่าจะลดลงเมื่อไม่มีการตรวจจับใด ๆ หรือทำนายเฟรมรูปภาพเป็นสถานการณ์ปกติ หากค่า Alert rate มีค่าถึงที่กำหนด (มากกว่าหรือเท่ากับ 40 ขึ้นไป) จะทำการแจ้งเตือนไปยังผู้ใช้



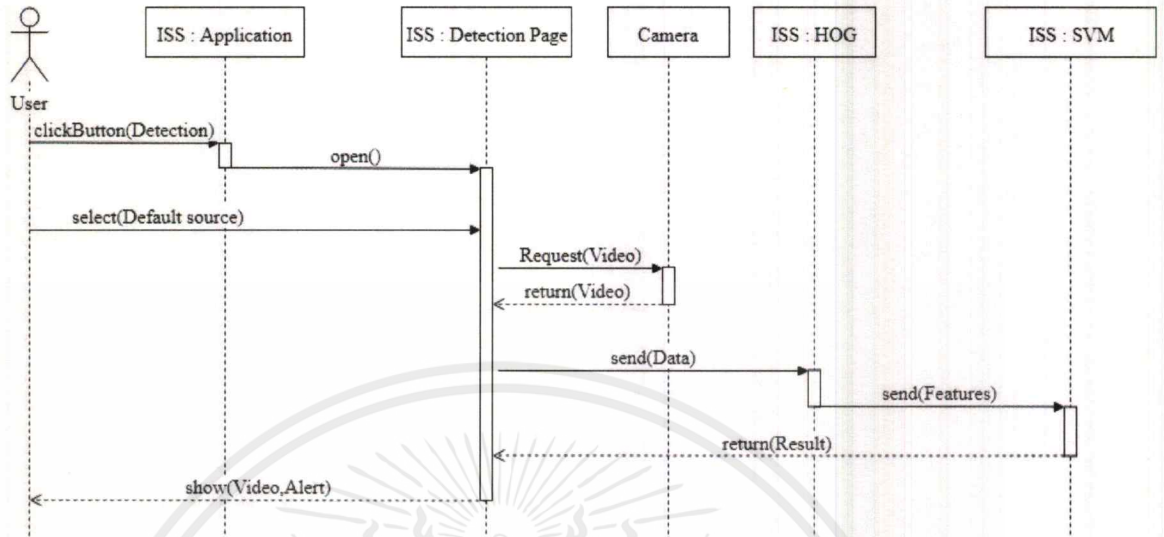
รูปที่ 3.4 Activity Diagram ของ Detection

- 3) Activity Diagram ของฟังก์ชัน Detect Positive : หากระบบตรวจจับทำทาง ผิดปกติได้ จะทำการเช็คค่า Alert rate หากตรงตามเงื่อนไขจะทำการแจ้งเตือนไปยังผู้ใช้งานและบันทึกการตรวจจับเอาไว้ แล้วจึงการรีเซ็ตค่า Alert rate ให้เท่ากับ 0



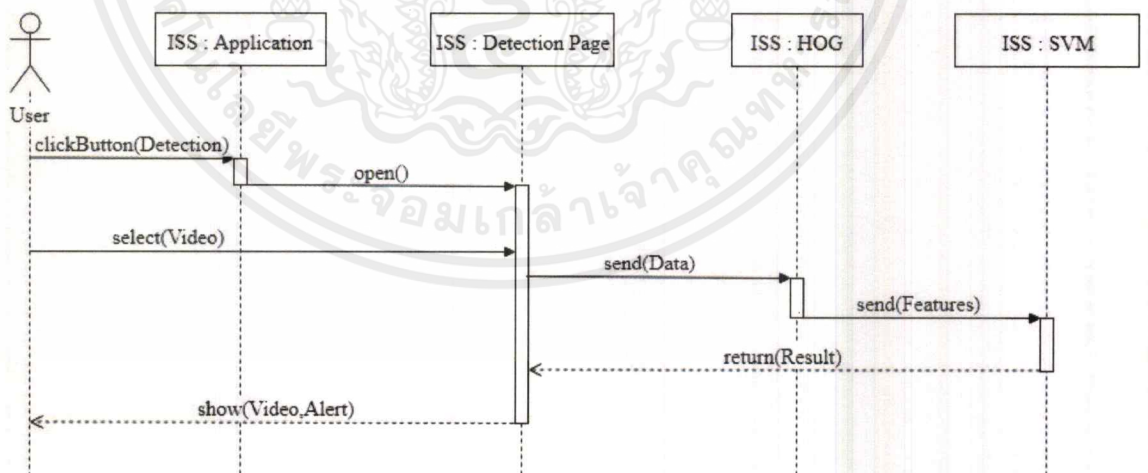
รูปที่ 3.5 Activity Diagram ของฟังก์ชัน Detect Positive

3.2.5 Sequence Diagram



รูปที่ 3.6 Sequence Diagram ของฟังก์ชันตรวจจับจากวิดีโอแบบเรียลไทม์

จากรูปที่ 3.6 อธิบายการทำงานได้ว่า เมื่อผู้ใช้คลิกที่ปุ่ม Detection เพื่อแสดงหน้าจอของการเลือกประเภทวิดีโอขึ้นมา ผู้ใช้ทำการเลือกวิดีโอที่ต้องการเป็น Default source ระบบจะไปทำการนำวิดีโอที่ได้มาจากกล้องวงจรปิดที่เชื่อมต่ออยู่ จากนั้นระบบจะทำการประมวลผลและแสดงให้ผู้ใช้ได้เห็นผลลัพธ์ของการตรวจจับและแจ้งเตือนไปยังผู้ใช้ในกรณีที่ตรวจจับได้



รูปที่ 3.7 Sequence Diagram ของฟังก์ชันการตรวจจับจากการเลือกไฟล์วิดีโอจากผู้ใช้

จากรูปที่ 3.7 อธิบายการทำงานได้ว่า เมื่อผู้ใช้คลิกที่ปุ่ม Detection เพื่อแสดงหน้าจอของการเลือกประเภทวิดีโอขึ้นมา ผู้ใช้ทำการเลือกไฟล์วิดีโอที่ต้องการตรวจจับลงไป จากนั้นระบบจะทำการประมวลผลและแสดงให้ผู้ผู้ใช้ได้เห็นผลลัพธ์ของการตรวจจับและแจ้งเตือนไปยังผู้ใช้ในกรณีที่ตรวจจับได้

3.3 การสร้างโมเดลตรวจจับท่าทางผิดปกติ

ขั้นตอนของการสร้างโมเดลเพื่อใช้ในการตรวจจับท่าทางผิดปกติประกอบไปด้วย 3 ส่วน ในส่วนแรกคือการจัดเตรียมข้อมูลโดยการเก็บรวบรวมข้อมูลรูปภาพและทำการปรับปรุงรูปภาพให้เหมาะสมสำหรับการนำไปใช้งานในส่วนต่อไป ส่วนที่สองคือสร้างโมเดลจำแนกรูปภาพโดยการใช้ข้อมูลฝึกสอนกับเทคนิคที่มีความเหมาะสม ส่วนสุดท้ายคือการทดสอบประสิทธิภาพของโมเดลที่ได้จากการฝึกสอนเพื่อวัดประสิทธิภาพและเลือกโมเดลที่ประสิทธิภาพที่สุดสำหรับการนำไปใช้งาน

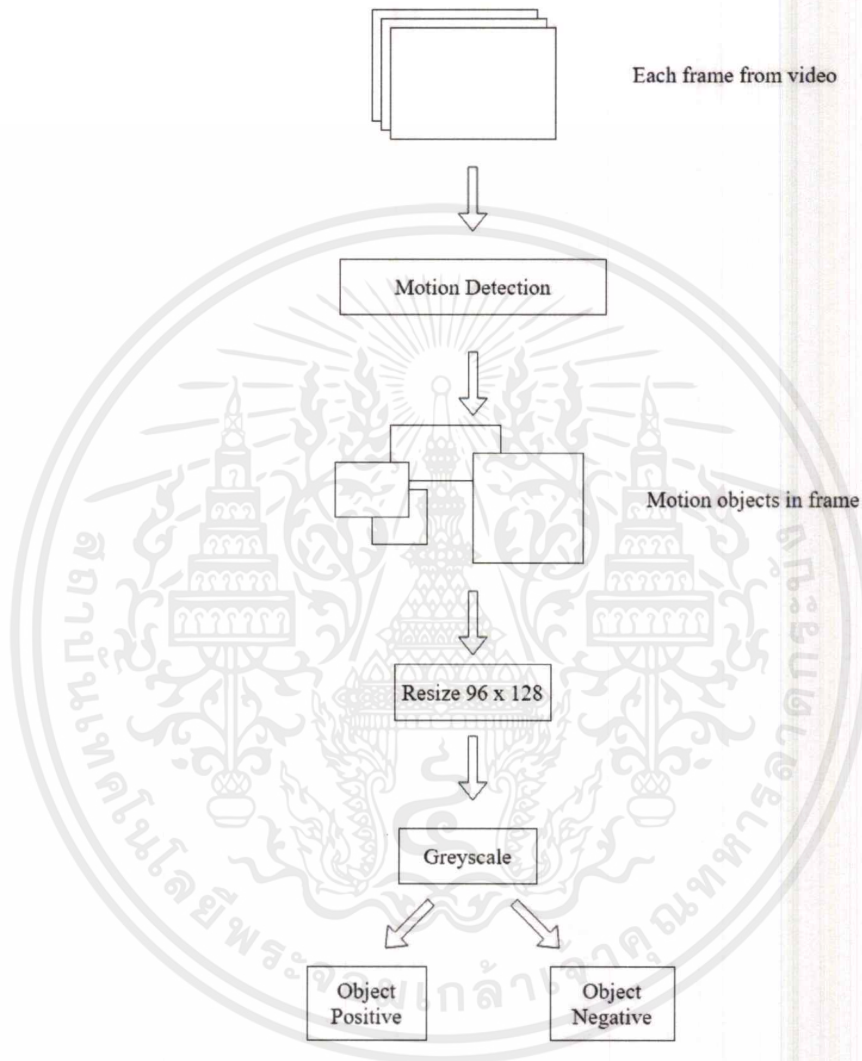
3.3.1 การจัดเตรียมข้อมูล

การเก็บข้อมูลที่จะใช้ฝึกสอนโมเดลในระบบที่พัฒนาขึ้นนำมาจากชุดข้อมูลฟรีจาก Serre Lab [4] ที่เป็นกลุ่มงานวิจัยของมหาวิทยาลัยบราวน์ ประเทศสหรัฐอเมริกา ซึ่งข้อมูลที่เป็นคลิปวิดีโอมีความยาวรวมมากกว่า 7000 ชั่วโมง และมีประเภทของท่าทางการเคลื่อนไหวของมนุษย์ที่ต่างกักันทั้งหมด 51 รูปแบบ ข้อมูลจาก INRIA [5] เป็นข้อมูลที่ถูกเก็บรวบรวมเพื่อใช้ในการวิจัยในการตรวจจับบุคคลในภาพและวิดีโอ และข้อมูลวิดีโอจากเว็บไซต์ Youtube [6-8]



รูปที่ 3.8 ตัวอย่างของชุดข้อมูลนำเข้าจากชุดข้อมูล Serre Lab

หลังจากการเก็บรวบรวมข้อมูลรูปภาพและวิดีโอ ในส่วนของข้อมูลที่เป็นวิดีโอจะทำการแปลงให้เป็นรูปภาพ จำนวนรูปภาพที่ได้มีจำนวนทั้งหมด 29,000 รูป ซึ่งประกอบด้วยข้อมูล 2 ส่วน คือ รูปภาพท่าทางปกติจำนวน 23,000 รูป และรูปภาพท่าทางผิดปกติจำนวน 6,000 รูป โดยที่ขนาดของแต่ละภาพ คือ 96×128 พิกเซล



รูปที่ 3.9 แผนภาพการจัดเตรียมรูปภาพจากวิดีโอ

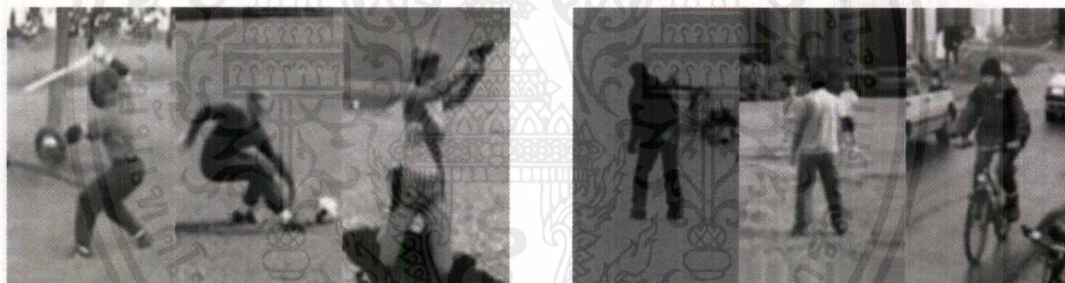
จากรูปที่ 3.9 เป็นการคัดเลือกรูปภาพเพื่อนำไปใช้เป็นชุดข้อมูลฝึกสอน โดยการเลือกพื้นที่เคลื่อนไหวในแต่ละเฟรมจากข้อมูลวิดีโอ จากนั้นทำการปรับขนาดรูปภาพและใช้เทคนิคเกรย์สเกล

เพื่อลดมิติสีของรูปภาพ จากนั้นทำการตัดแยกรูปภาพออกเป็นสองส่วนคือ ส่วนที่เป็นลักษณะท่าทางปกติและส่วนที่เป็นลักษณะท่าทางผิดปกติ



รูปที่ 3.10 ตัวอย่างรูปภาพที่ใช้ในการพัฒนาโมเดลตรวจจับ

ภาพซ้าย : รูปภาพผิดปกติ, ภาพขวา : รูปภาพปกติ

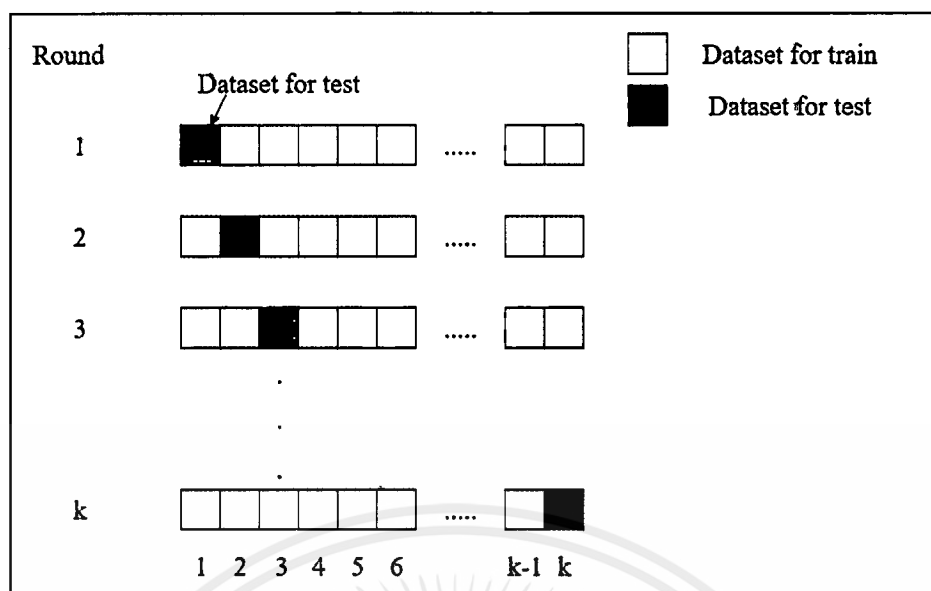


รูปที่ 3.11 รูปภาพหลังจากผ่านกระบวนการ Grey Scale

ภาพซ้าย : รูปภาพผิดปกติ, ภาพขวา : รูปภาพปกติ

เมื่อทำการจัดเตรียมชุดข้อมูลฝึกสอนเรียบร้อยแล้วจะนำรูปภาพไปคำนวณหาค่าคุณลักษณะที่เหมาะสม โดยเลือกใช้เทคนิคฮิสโตแกรมของการไล่ระดับเชิงมุม ผลลัพธ์ที่ได้จะอยู่ในรูปของค่าเวกเตอร์เพื่อที่จะสามารถนำไปใช้กับเทคนิคในการจำแนกรูปภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 การแบ่งชุดข้อมูลฝึกสอนเพื่อนำไปใช้ในการสร้างและทดสอบประสิทธิภาพของโมเดล

จากรูปที่ 3.12 คือการแบ่งชุดข้อมูลฝึกสอนเป็นส่วนย่อยจำนวน k ส่วนเพื่อนำไปใช้ในการฝึกสอนจำนวน $k - 1$ ส่วนและอีกหนึ่งส่วนจะใช้สำหรับการทดสอบโมเดล กระบวนการดังกล่าวเรียกว่า k - fold validation

3.3.2 การสร้างโมเดล

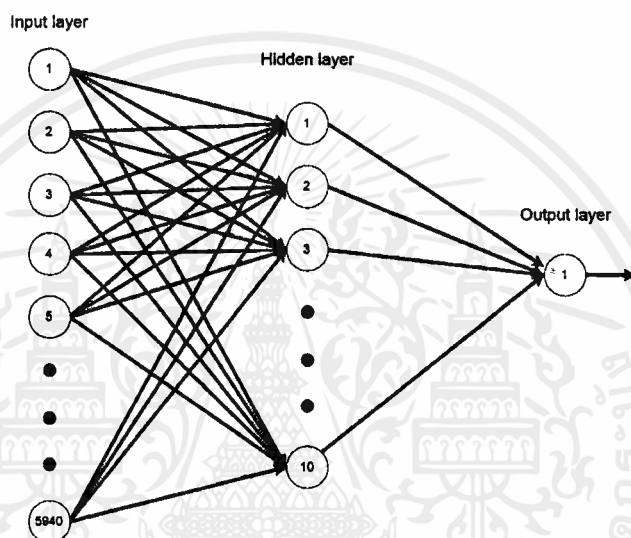
ขั้นตอนการสร้างโมเดลคือการนำเอารูปภาพที่ได้หลังจากขั้นตอนการเตรียมข้อมูลซึ่งจะถูกเรียกว่า ชุดข้อมูลฝึกสอน มาเข้ากระบวนการคำนวณทางคณิตศาสตร์โดยใช้เทคนิคทางการเรียนรู้ของเครื่อง ในการฝึกสอน ภายในโครงงานได้เลือกใช้เทคนิคซัพพอร์ตเวกเตอร์แมชชีนและโครงข่ายประสาทเทียม

- 1) ซัพพอร์ตเวกเตอร์แมชชีนจะทำการแบ่งข้อมูลออกเป็นสองลักษณะคือ ลักษณะที่ผิดปกติและไม่ผิดปกติ โดยการใช้ฟังก์ชันสมการในรูปแบบต่าง ๆ ที่เรียกว่าเคอร์เนล โดยเคอร์เนลที่นำมาใช้ได้แก่
 1. Linear การใช้เคอร์เนลนี้ไม่ต้องกำหนดค่าพารามิเตอร์
 2. Polynomial การใช้เคอร์เนลนี้จะต้องกำหนดค่าพารามิเตอร์สองตัวได้แก่ ค่าแกรมมา (γ) และค่าดีกรี (degree)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. RBF การใช้เคอร์เนลนี้จะต้องกำหนดค่าพารามิเตอร์คือ ค่าแกมมา
4. CHI2 การใช้เคอร์เนลนี้จะต้องกำหนดค่าพารามิเตอร์สองตัวได้แก่ ค่าแกมมา และค่า coef0

- 2) โครงข่ายประสาทเทียม จำนวนโหนดในชั้นอินพุตมีจำนวน 5940 โหนด จำนวนโหนดในชั้นซ่อน มีจำนวนเท่ากับ 10 โหนด และในชั้นเอาต์พุต มีเพียงโหนดเดียวดังรูป

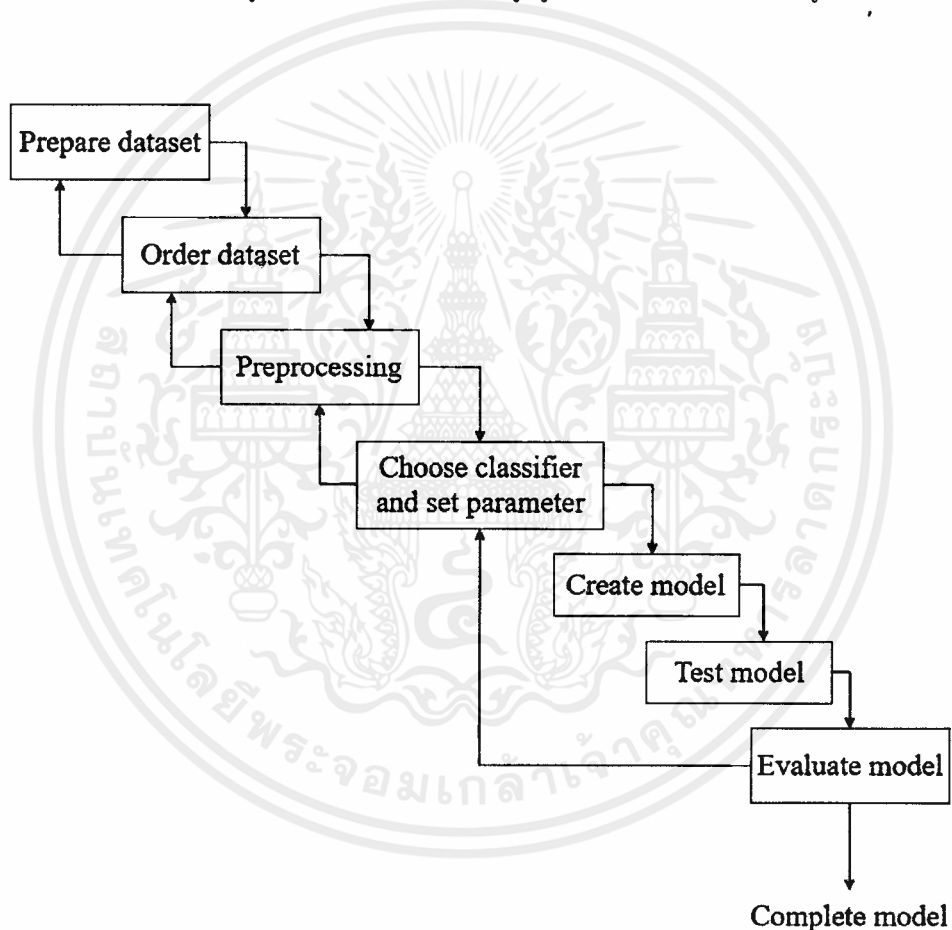


รูปที่ 3.13 โครงสร้างเครือข่ายประสาทเทียม

ในชั้นอินพุต และชั้นเอาต์พุต แต่ละโหนดจะใช้ฟังก์ชันซิกมอยด์ (sigmoid) เป็นฟังก์ชันกระตุ้น กำหนดจำนวนรอบในการทำงานไม่เกิน 1000 รอบ กำหนดค่าอัตราการเรียนรู้เท่ากับ 0.1 และกำหนดค่าอัตราความผิดพลาดที่ 0.0001 และใช้อัลกอริทึม back-propagation ในการปรับค่าน้ำหนัก (weight)

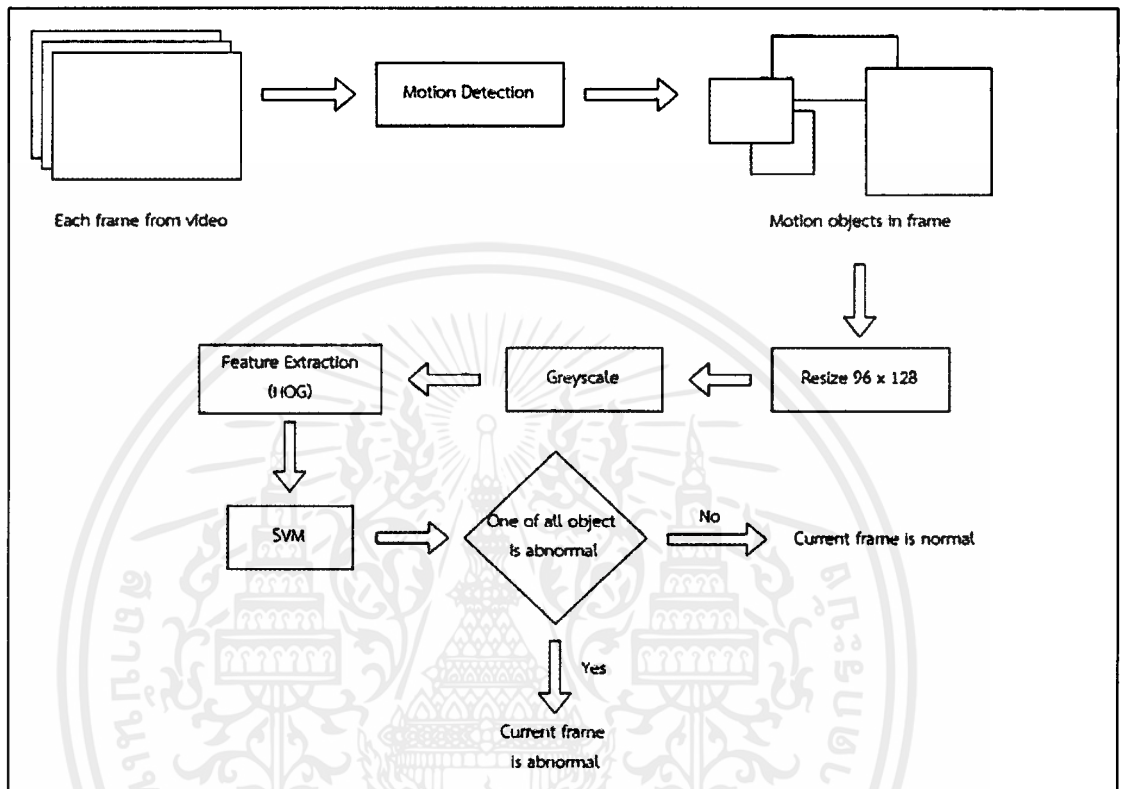
3.3.3 การทดสอบโมเดล

ข้อมูลที่ถูกแบ่งไว้อีกหนึ่งส่วนจะถูกนำมาทดสอบกับโมเดลเพื่อวัดประสิทธิภาพของชุดข้อมูลกับโมเดลว่ามีความแม่นยำมากน้อยเพียงใด โดยจะทำการทดสอบวนรอบจนครบ k ครั้ง กระบวนการนี้จะทำวนรอบระหว่างการสร้างโมเดลและทดสอบโมเดล เมื่อครบรอบจะทำการหาผลเฉลี่ยของความแม่นยำเพื่อประเมินว่าโมเดลที่สร้างมีประสิทธิภาพเพียงพอหรือไม่ หากพบว่าโมเดลมีประสิทธิภาพไม่เพียงพอจำเป็นต้องกลับไปทำการสร้างโมเดลใหม่โดยการเปลี่ยนเทคนิคและค่าพารามิเตอร์หรือทำการเปลี่ยนเทคนิคในการคำนวณค่าคุณลักษณะของรูปภาพที่เหมาะสม ถ้ายังพบว่าโมเดลที่ได้มีประสิทธิภาพไม่ดีพออาจจะต้องกลับที่กระบวนการจัดระเบียบชุดข้อมูลเนื่องจากการแยกข้อมูลยังไม่ถูกต้องหรือเริ่มกระบวนการเตรียมข้อมูลใหม่อีกครั้งเนื่องจากข้อมูลรูปภาพที่นำเข้ามายังไม่สมบูรณ์



รูปที่ 3.14 กระบวนการทดสอบประสิทธิภาพของโมเดล

การทดสอบประสิทธิภาพของโมเดลจะแบ่งการทดสอบออกเป็นสองส่วนคือ การทดสอบโมเดลกับชุดข้อมูลรูปภาพและการทดสอบโมเดลกับชุดข้อมูลวิดีโอ โดยจะทำการทดสอบโมเดลกับชุดข้อมูลรูปภาพก่อน จากนั้นจึงนำโมเดลไปทดสอบกับชุดข้อมูลวิดีโอ

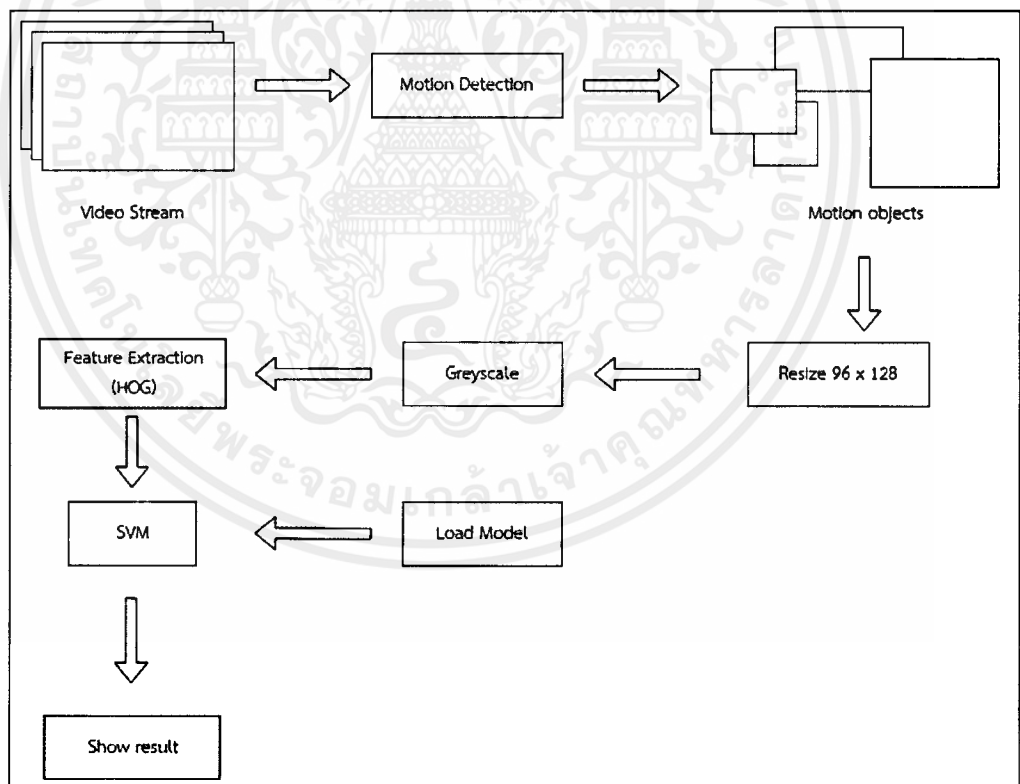


รูปที่ 3.15 กระบวนการทดสอบประสิทธิภาพของโมเดลกับชุดข้อมูลวิดีโอ

3.3.4 การนำโมเดลไปใช้ในการจำแนกรูปภาพ

ขั้นตอนของการนำโมเดลไปใช้งานจะอยู่ในส่วนที่ติดต่อกับผู้ใช้งาน ข้อมูลรูปภาพจะได้อามาจากการที่ผู้ใช้ได้ทำการนำข้อมูลเข้าระบบ แบ่งออกเป็นสามขั้นตอนได้แก่ การเลือกพื้นที่ที่สนใจ การคำนวณค่าคุณลักษณะ และการจำแนกรูปภาพโดยการใช้โมเดล

- 1) การเลือกพื้นที่ที่สนใจ คือการเลือกบริเวณที่สนใจจากกรณีที่วัตถุในรูปมีการเคลื่อนไหวจากเฟรมรูปภาพในแต่ละเฟรมโดยจะทำการเปรียบเทียบในเฟรมก่อนกับเฟรมปัจจุบันแล้วใช้หลักการทางคณิตศาสตร์แทนที่การเคลื่อนไหวของวัตถุ
- 2) การคำนวณค่าคุณลักษณะของรูปภาพ เป็นการคำนวณในรูปแบบเดียวกันกับการปรับปรุงรูปภาพและคำนวณค่าคุณลักษณะในการสร้างโมเดล โดยการใช้เทคนิคฮิสโตแกรมของการไล่ระดับเชิงมุม
- 3) การจำแนกรูปภาพ นำค่าคุณลักษณะที่อยู่ในรูปของค่าเวกเตอร์มาทำการจำแนกโดยใช้โมเดลที่ได้เลือกไว้ในการจำแนก ผลลัพธ์จะถูกแสดงไปยังหน้าจอของผู้ใช้งาน



รูปที่ 3.16 กระบวนการนำโมเดลมาใช้ในการตรวจจับสถานการณ์ผิดปกติในวิดีโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงาน

ในบทนี้จะแสดงผลการทดลองของชุดข้อมูลที่นำมาฝึกสอนเพื่อเลือกโมเดลที่มีประสิทธิภาพสูงสุดนำมาใช้พัฒนาระบบเฝ้าระวังอัจฉริยะ โดยใช้เทคนิคของการเรียนรู้ของเครื่องแบบมีผู้สอน 2 เทคนิค คือ เทคนิคซัพพอร์ตเวกเตอร์แมชชีน และโครงข่ายประสาทเทียม ในส่วนสุดท้ายจะอธิบายการทำงานของระบบเฝ้าระวังอัจฉริยะ

4.1 ผลการทดลอง

ในการสร้างโมเดลด้วยซัพพอร์ตเวกเตอร์แมชชีน ได้มีการใช้เคอร์เนลทั้งหมด 4 เคอร์เนล คือ Linear Polynomial RBF และ CHI2 และวัดประสิทธิภาพของโมเดลด้วย ค่า recall ค่า precision ค่า f-measure และค่า accuracy วิธีการทดสอบแบ่งออกเป็น 2 วิธีคือ การทดสอบโดยใช้ชุดข้อมูลรูปภาพและการทดสอบโดยใช้ชุดข้อมูลวิดีโอ กำหนดให้ค่าพารามิเตอร์ของแต่ละเคอร์เนลเป็นดังนี้ Polynomial มีค่า γ เท่ากับ 0.075, degree เท่ากับ 4, coef0 เท่ากับ 1 RBF มีค่า γ เท่ากับ 0.075 CHI2 มีค่า γ เท่ากับ 0.05

4.1.1 การทดสอบโมเดลโดยใช้ชุดข้อมูลรูปภาพ

เป็นการนำชุดข้อมูลฝึกสอนที่เตรียมไว้ มาทดสอบด้วยวิธี k - fold validation โดยกำหนดให้ k เท่ากับ 10 โดยใช้เทคนิคซัพพอร์ตเวกเตอร์แมชชีน หลังจากนั้นนำโมเดลที่พัฒนาขึ้นมาเปรียบเทียบกับโมเดลที่สร้างด้วยวิธีโครงข่ายประสาทเทียมเพื่อทำการเลือกโมเดลที่ให้ค่าความแม่นยำมากที่สุดมาใช้ในการพัฒนาระบบเฝ้าระวังอัจฉริยะต่อไป

ตารางที่ 4.1 เปรียบเทียบค่า recall ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีน

Fold	Recall (%)			
	Linear	Polynomial	RBF	CHI2
1	57.00	79.17	83.00	83.17
2	57.83	79.00	80.00	80.17
3	53.33	76.50	79.83	84.17
4	54.83	81.33	96.33	82.83
5	53.83	78.00	83.67	83.33
6	52.17	79.17	89.17	80.33
7	51.17	76.17	84.00	83.50
8	54.83	79.83	81.83	83.00
9	56.83	75.83	85.00	86.17
10	55.83	74.83	85.33	83.83
ค่าเฉลี่ย	54.77	77.98	84.82	83.05

ตารางที่ 4.2 เปรียบเทียบค่า precision ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีน

Fold	Precision (%)			
	Linear	Polynomial	RBF	CHI2
1	36.38	62.17	81.51	85.45
2	25.46	60.77	85.41	88.42
3	34.86	61.36	84.48	82.38
4	33.13	60.40	49.23	83.67
5	27.37	63.16	83.39	83.20
6	30.87	57.37	75.57	86.38
7	35.25	63.10	84.56	84.49
8	29.72	60.33	83.79	84.84
9	28.11	61.07	84.58	82.72
10	28.63	60.68	82.05	68.07
ค่าเฉลี่ย	30.98	61.04	79.46	82.96

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 เปรียบเทียบค่า f-measure ของเทคนิคซ์พอร์ตเวคเตอร์แมชชีน

Fold	f-measure (%)			
	Linear	Polynomial	RBF	CHI2
1	44.42	69.65	82.25	84.29
2	35.35	68.70	82.61	84.09
3	42.16	68.10	82.09	83.27
4	41.31	69.32	65.16	83.25
5	36.29	69.80	83.53	83.27
6	38.79	66.53	81.80	83.25
7	41.74	69.61	84.28	83.99
8	38.55	68.72	82.80	83.91
9	37.62	67.66	84.79	84.41
10	37.85	67.02	83.66	75.13
ค่าเฉลี่ย	39.41	68.51	81.30	82.89

ตารางที่ 4.4 เปรียบเทียบค่า accuracy ของเทคนิคซ์พอร์ตเวคเตอร์แมชชีน

Fold	Accuracy (%)			
	Linear	Polynomial	RBF	CHI2
1	70.48	85.72	92.59	93.59
2	56.24	85.10	93.03	93.72
3	69.72	85.17	92.79	93.00
4	67.76	85.10	78.69	93.10
5	60.90	86.03	93.17	93.07
6	65.93	83.52	91.79	93.31
7	70.45	86.24	93.52	93.41
8	63.83	84.97	92.97	93.41
9	61.00	85.00	93.69	93.41
10	62.07	84.76	93.10	88.52
ค่าเฉลี่ย	64.84	85.16	91.53	92.85

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการเปรียบเทียบแต่ละคอร์เนลของเทคนิคซ์พอร์ดเวคเตอร์แมชชีน แสดงให้เห็นว่าคอร์เนล CHI2 เป็นคอร์เนลที่มีประสิทธิภาพในการทำนายผลมากที่สุด ซึ่งมีค่าความแม่นยำเฉลี่ยสูงที่สุดเท่ากับ 92.85% จากนั้นนำไปเปรียบเทียบกับโครงข่ายประสาทเทียม เพื่อหาวิธีที่เหมาะสมในการนำไปพัฒนาโมเดลของระบบเฝ้าระวังอัจฉริยะ

ตารางที่ 4.5 เปรียบเทียบค่า recall ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีนกับโครงข่ายประสาทเทียม

Fold	SVM (%)	ANN (%)
1	83.17	51.50
2	80.17	48.33
3	84.17	50.83
4	82.83	50.17
5	83.33	53.67
6	80.33	55.00
7	83.50	48.00
8	83.00	46.00
9	86.17	47.17
10	83.83	50.83
ค่าเฉลี่ย	83.05	50.15

ตารางที่ 4.6 เปรียบเทียบค่า precision ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีนกับโครงข่ายประสาทเทียม

Fold	SVM (%)	ANN (%)
1	85.45	69.91
2	88.42	59.06
3	82.38	64.76
4	83.67	68.72
5	83.20	70.77
6	86.38	69.92
7	84.49	68.90
8	84.84	67.65
9	82.72	73.13
10	68.07	66.89
ค่าเฉลี่ย	82.96	67.97

ตารางที่ 4.7 เปรียบเทียบค่า f-measure ของเทคนิคซ์พอร์ดเวคเตอร์แมชชีนกับโครงข่ายประสาทเทียม

Fold	SVM (%)	ANN (%)
1	84.29	59.31
2	84.09	53.16
3	83.27	56.96
4	83.25	58.00
5	83.27	61.04
6	83.25	61.57
7	83.99	56.58
8	83.91	54.76
9	84.41	57.35
10	75.13	57.77
ค่าเฉลี่ย	82.89	57.65

ตารางที่ 4.8 เปรียบเทียบค่าความแม่นยำของเทคนิคซัพพอร์ตเวกเตอร์แมชชีนกับโครงข่ายประสาทเทียม

Fold	SVM (%)	ANN (%)
1	93.59	85.38
2	93.72	82.38
3	93.00	84.10
4	93.10	84.97
5	93.07	85.83
6	93.31	85.79
7	93.41	84.76
8	93.41	84.28
9	93.41	85.48
10	88.52	84.62
ค่าเฉลี่ย	92.85	84.76

จากการเปรียบเทียบข้างต้นส่วนของโมเดลที่ใช้ในระบบเฝ้าระวังอัจฉริยะ จะใช้เทคนิคซัพพอร์ตเวกเตอร์แมชชีนโดยเลือกใช้คอร์เนลแบบ Chi2 ซึ่งให้ค่า accuracy เท่ากับ 92.85% และมีค่า f-measure เท่ากับ 82.89%

4.1.2 การทดสอบโมเดลโดยใช้ชุดข้อมูลวิดีโอ

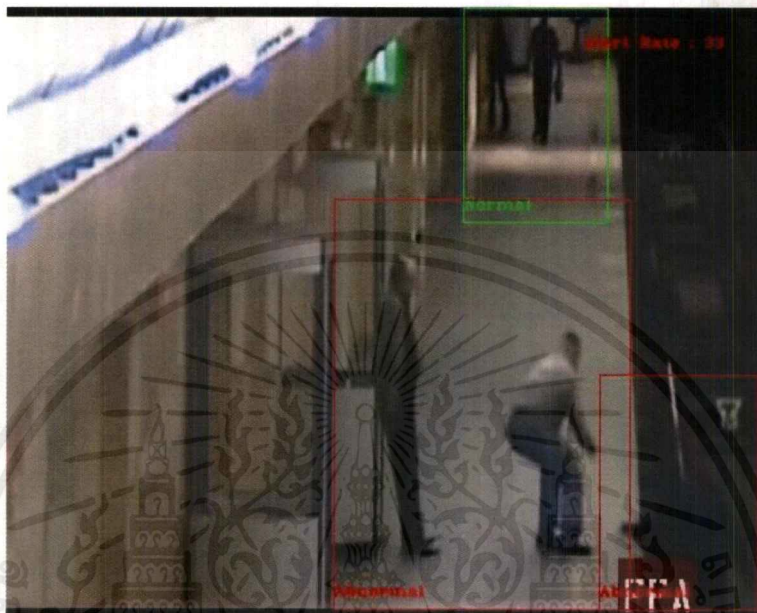
เป็นการทดสอบโมเดลเพื่อหาประสิทธิภาพของโมเดล โดยชุดข้อมูลที่ใช้ในการทดสอบเป็นวิดีโอที่แสดงถึง 3 เหตุการณ์ที่สภาพแวดล้อมแตกต่างกัน ได้แก่ การตรวจจับภายในอาคาร เป็นวิดีโอแสดงเหตุการณ์ทะเลาะวิวาทที่สถานีรถไฟ การตรวจจับบริเวณอาคาร เป็นวิดีโอแสดงเหตุการณ์ปล้นชิงทรัพย์บริเวณอาคาร และการตรวจจับภายนอกอาคาร เป็นการทะเลาะวิวาทที่ถนน

ตารางที่ 4.9 Confusion matrix ของวิดีโอเหตุการณ์ทะเลาะวิวาทที่สถานีรถไฟ

Video 1		Actual	
		True	False
Predict	True	109	90
	False	40	211

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คลิปวีดิโอนี้นำมาใช้เพื่อประเมินประสิทธิภาพในการตรวจจับภายในอาคาร เนื้อหาของวิดีโอ คือ มีผู้ชายคนหนึ่งผลักหญิงสาวตกลงไปในรางรถไฟทำให้ผู้ชายอีกคนหนึ่งที่อยู่ในเหตุการณ์ทำร้ายผู้ชายคนดังกล่าวและช่วยผู้หญิงผู้เคราะห์ร้ายขึ้นจากรางรถไฟ



รูปที่ 4.1 ตัวอย่างการตรวจจับภายในวิดีโอ 1

ตารางที่ 4.10 Confusion matrix ของวิดีโอเหตุการณ์ปล้นชิงทรัพย์บริเวณอาคาร

Video 2		Actual	
		True	False
Predict	True	145	37
	False	88	792

คลิปวีดิโอนี้นำมาใช้เพื่อประเมินประสิทธิภาพในการตรวจจับบริเวณอาคาร เนื้อหาของวิดีโอ คือ มีคนร้ายสองคนขับรถมอเตอร์ไซด์มาในบริเวณดังกล่าวจากนั้นได้ทำการแย่งชิงกระเป๋าจากชายคนหนึ่งที่ได้ยืนอยู่บริเวณนั้นแล้วหลบหนีไป



รูปที่ 4.2 ตัวอย่างการตรวจจับภายในวิดีโอ 2

ตารางที่ 4.11 Confusion matrix ของวิดีโอเหตุการณ์ทะเลาะวิวาทที่ถนน

Video 3		Actual	
		True	False
Predict	True	211	230
	False	45	577

คลิปวิดีโอนี้นำมาใช้เพื่อประเมินประสิทธิภาพในการตรวจจับภายนอกบริเวณอาคาร เนื้อหาของวิดีโอคือ มีผู้ชายสองคนยืนคุยกันอยู่กลางถนนจากนั้นชายคนหนึ่งได้ทำการชกไปที่ใบหน้าของชายอีกคนจนล้มลงไปที่ยื่น



รูปที่ 4.3 ตัวอย่างการตรวจจับภายในวิดีโอ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.12 ค่า recall, precision, f-measure และ accuracy ของคลิปวิดีโอ

	Recall (%)	Precision (%)	f-measure (%)	Accuracy (%)
Video 1	54.77	73.15	62.64	71.11
Video 2	79.67	62.23	69.88	88.23
Video 3	47.85	82.42	60.55	74.13

จากการทดสอบโดยชุดข้อมูลวิดีโอมีความแม่นยำน้อยกว่าชุดข้อมูลรูปภาพ เนื่องจากข้อมูลวิดีโอ มีสิ่งรบกวนอยู่มาก ทำให้การตรวจจับมีความคลาดเคลื่อนมากกว่าการทดสอบโดยใช้ชุดข้อมูลรูปภาพ

4.2 หน้าจอแสดงผล

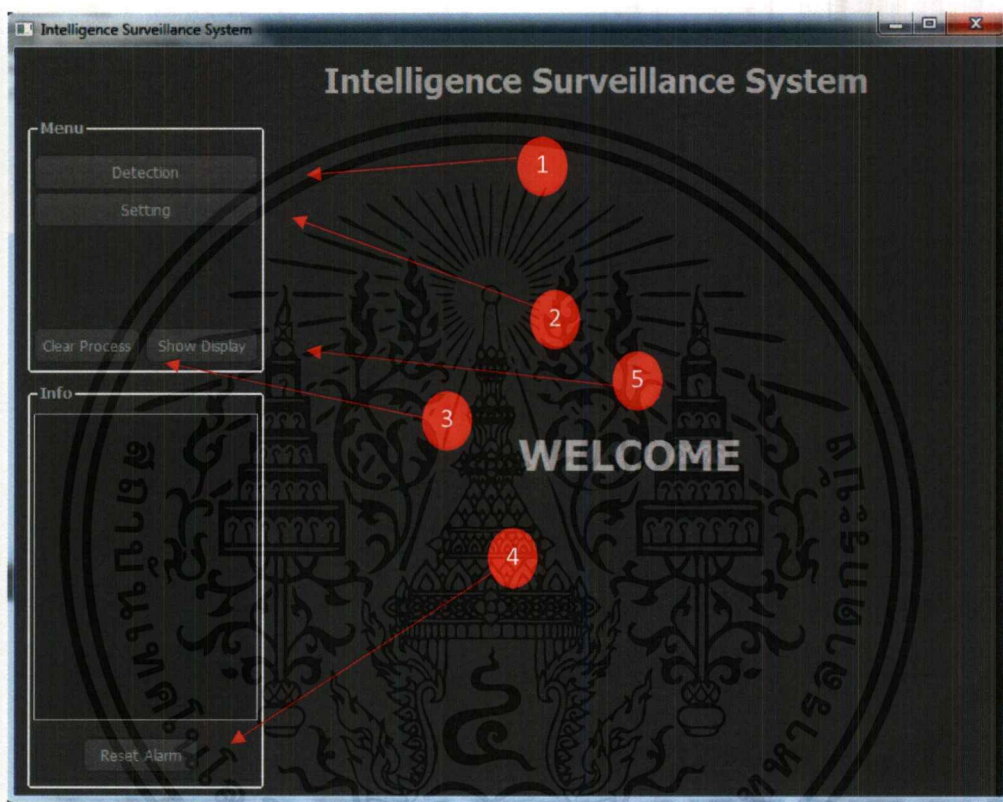
ในการพัฒนาระบบเฝ้าระวังอัจฉริยะ ได้มีการแบ่งหน้าจอของการทำงานออกมาดังต่อไปนี้

- 1) หน้าจอหลัก
- 2) หน้าจอเลือกประเภทการตรวจจับ
- 3) หน้าจอปรับแต่งทั่วไปของระบบ
- 4) ส่วนแสดงผลข้อมูลการตรวจจับ

ระบบเฝ้าระวังอัจฉริยะ เป็นระบบที่ประกอบด้วยความสามารถดังต่อไปนี้

- ผู้ใช้สามารถเลือกนำวิดีโอที่ต้องการทั้งวิดีโอแบบทั่วไปที่ได้รับการบันทึกไว้และวิดีโอแบบเรียลไทม์หรือการถ่ายทอดสดมาใช้ในการตรวจจับท่าทางผิดปกติภายในระบบ
- ภายในส่วนปรับแต่งผู้ใช้สามารถปรับแต่งระบบในส่วนต่าง ๆ ได้ภายในระดับที่ระบบมีการอนุญาตให้ปรับแต่งผ่านหน้าจอของระบบ
- การแสดงผลภายในหน้าจอการตรวจจับวิดีโอจะแสดงผลเป็นกรอบสี่เหลี่ยมที่ประกอบไปด้วยสีสองสีคือ สีเหลี่ยมสีแดงและสีเหลี่ยมสีเขียว ซึ่งสีแดงหมายถึงมีความผิดปกติและสีเขียวคือความปกติ
- ทันทีที่ระบบทำการตรวจจับท่าทางผิดปกติภายในวิดีโอที่กำลังเล่นในระบบได้ ระบบจะทำการจับตาดูบริเวณดังกล่าวบนวิดีโอ หากพบว่ามีความผิดปกติเกิดขึ้นถึงจนเกินไปภายในช่วงเวลาหนึ่ง ๆ ระบบจะทำการแจ้งเตือนในรูปแบบของเสียงและข้อความมายังผู้ใช้งานบนหน้าจอ

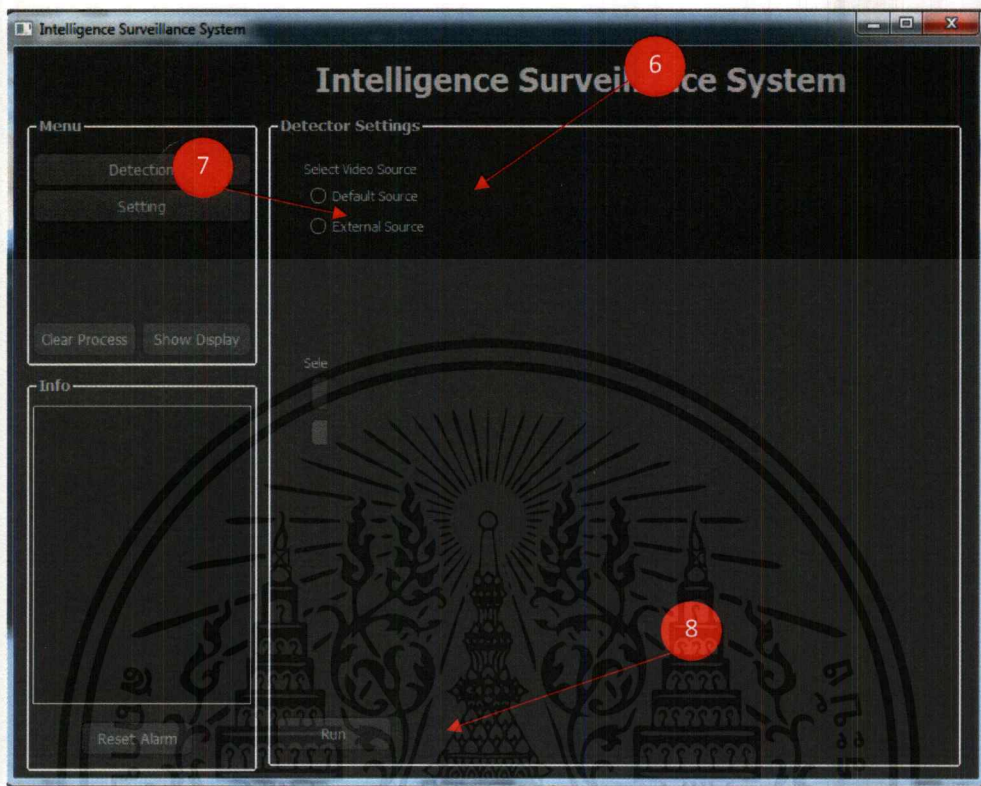
- การแจ้งเตือนในรูปแบบของเสียงจะไม่หยุดจนกว่าผู้ใช้จะรับรู้และกดปิดการแจ้งเตือนด้วยตนเอง
- ผู้ใช้สามารถเรียกดูผลข้อมูลการตรวจจับท่าทางผิดปกติได้ โดยผ่านทางหน้าจอแสดงผลข้อมูลการตรวจจับ
- การกระทำต่าง ๆ หรือการกระทำใด ๆ บนระบบเฝ้าระวังอัจฉริยะจะมีการบันทึกข้อมูลการทำงานลงในไฟล์พิเศษซึ่งผู้ใช้งานสามารถเรียกดูเหตุการณ์ใด ๆ บนระบบย้อนหลังได้



รูปที่ 4.4 หน้าจอหลักของระบบเฝ้าระวังอัจฉริยะ

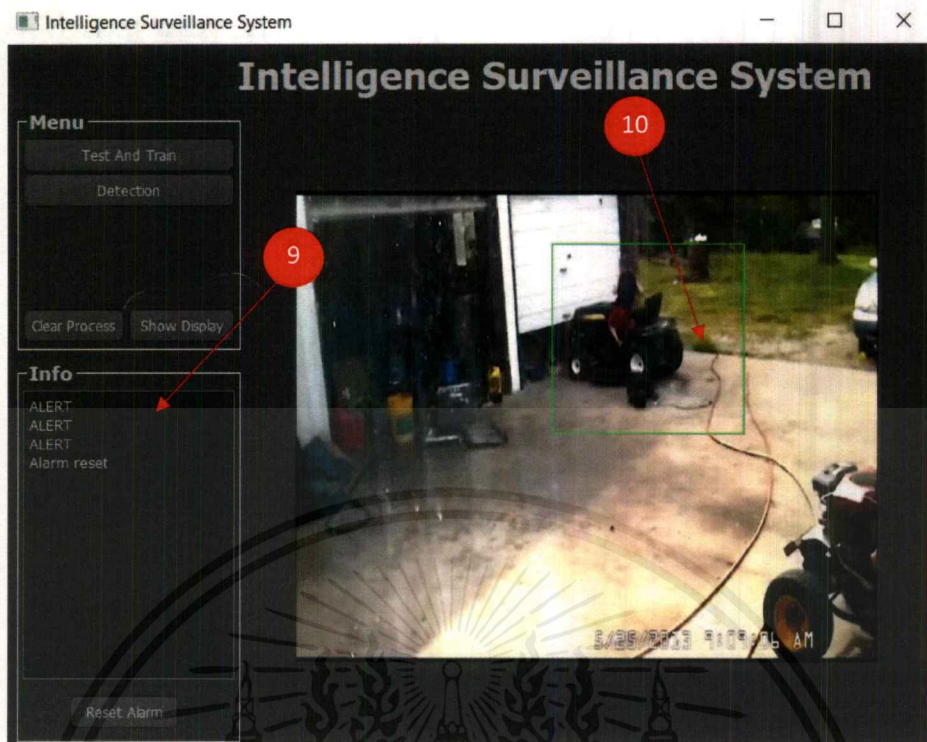
จากรูปที่ 4.4 หลังจากผู้ใช้เริ่มการใช้งานระบบ ภายในหน้าจอจะประกอบไปด้วย หมายเลข 1 เป็นปุ่มสำหรับย้ายไปยังส่วนหน้าจอการตรวจจับ (Detection) ซึ่งเป็นหน้าจอการปรับแต่งในส่วนเกี่ยวกับกการตรวจจับสิ่งผิดปกติภายในวิดีโอของระบบ หมายเลข 2 เป็นปุ่มสำหรับย้ายไปหน้าจอการตั้งค่า (Setting) มีไว้สำหรับให้ผู้ใช้แก้ไขธีม หรือเสียงแจ้งเตือน รวมถึงตั้งค่าทั่วไปในระบบ หมายเลข 3 เป็นการรีเซ็ตการทำงานทั้งหมดที่กำลังดำเนินการอยู่และกลับมาที่หน้าจอหลัก และหมายเลข 4 ปุ่มสำหรับให้ผู้ใช้ล้างการแจ้งเตือนที่กำลังแจ้งเตือนในขณะนั้นหากกดเมื่อขณะยังไม่

มีการแจ้งเตือนจะไม่มีผลอะไร สำหรับหมายเลข 5 เป็นปุ่มสำหรับแสดงหน้าจอแสดงผลการทำงานหลักหรือหน้าจอการตรวจจับวิดีโอแสดงผลเมื่อเริ่มทำการตรวจจับวิดีโอใด ๆ



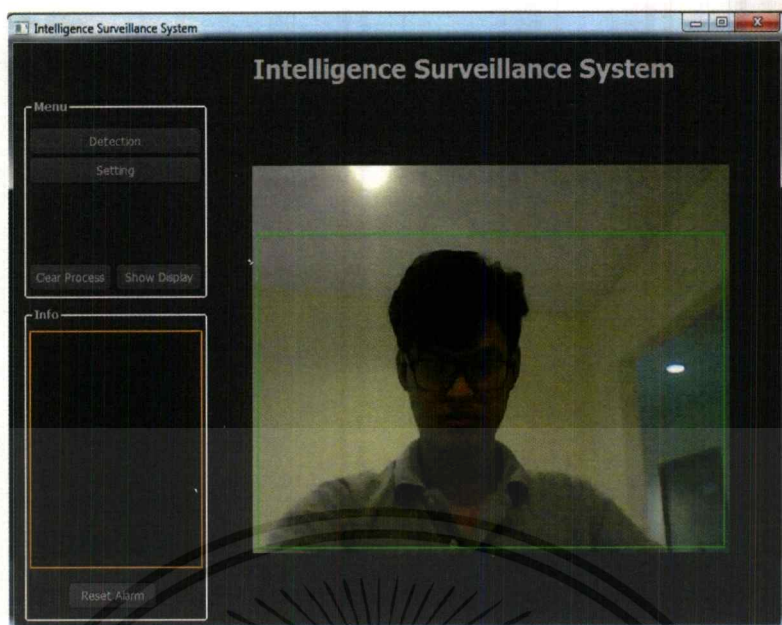
รูปที่ 4.5 หน้าจอเลือกประเภทการตรวจจับ

จากรูปที่ 4.5 ในส่วนของหน้าจอการตรวจจับ หมายเลข 6 เป็นการเลือกวิดีโอจากกล้องเพื่อทำการตรวจจับแบบเรียลไทม์ หมายเลข 7 เป็นส่วนของการเลือกวิดีโอจากแหล่งอื่นที่ไม่ใช่แบบเรียลไทม์มาใช้ในการตรวจจับ หมายเลข 8 หลังจากเลือกประเภทของการตรวจจับแล้ว ก็จะเปลี่ยนไปยังหน้าจอแสดงผลหลักที่ใช้ในการตรวจจับความผิดปกติในวิดีโอและเริ่มทำงานแต่หากกดในขณะที่ไม่ได้เลือกตัวเลือกใด ๆ การทำงานดังกล่าวจะไม่เกิด



รูปที่ 4.6 หน้าจอแสดงข้อมูลการตรวจจับ (แบบไฟล์วิดีโอ)

จากรูปที่ 4.6 เป็นหน้าจอหลังจากที่ผู้ใช้เลือกประเภทการตรวจจับแล้ว ทันทีที่เริ่มการทำงาน หากมีสิ่งเคลื่อนไหวภายในวิดีโอระบบจะทำการตรวจสอบและจับตามวัตถุดังกล่าวหากพบว่ามี การตรวจจับเจอท่าทางผิดปกติ ระบบจะแสดงผลเป็นกรอบสีแดงซึ่งแสดงถึงความผิดปกติ ถ้าหากพบ ความผิดปกติในบริเวณดังกล่าวถึงจนเกินไประบบจะแจ้งเตือนตรงบริเวณหมายเลข 9 ซึ่งผู้ใช้สามารถ เช็ตการแจ้งเตือนด้วยปุ่ม Reset Alarm บริเวณข้างล่าง ถ้าหากตรวจจับไม่เจอท่าทางผิดปกติ ก็ จะแสดงกรอบสีเขียวล้อมวัตถุเอาไว้ด้วยหมายเลข 10



รูปที่ 4.7 หน้าจอแสดงข้อมูลการตรวจจับ (แบบเรียลไทม์)

จากรูปที่ 4.7 เป็นหน้าจอหลังจากที่ผู้ใช้เลือกการตรวจจับแบบเรียลไทม์ (Default source) จะเป็นหน้าจอที่แสดงวิดีโอที่ได้รับจากกล้องหลักที่ติดอยู่กับเครื่องในขณะนั้นเพื่อใช้แสดงผลการตรวจจับของระบบเช่นเดียวกับแบบวิดีโอจากแหล่งอื่น

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 บทสรุป

ระบบเฝ้าระวังอัจฉริยะ เป็นระบบที่ถูกพัฒนาขึ้นเพื่อช่วยแก้ปัญหาที่เกี่ยวข้องกับงานด้านความปลอดภัย เป็นการตรวจจับท่าทางผิดปกติที่เกิดขึ้นในสถานการณ์ต่าง ๆ โดยจะมีการนำเข้าข้อมูลจากวิดีโอที่ต้องการตรวจจับ ซึ่งสามารถเป็นได้ทั้งวิดีโอแบบทั่วไปหรือวิดีโอแบบเรียลไทม์ จากนั้นระบบจะนำวิดีโอที่ต้องการมาทำการแบ่งข้อมูลวิดีโอออกมาเป็นเฟรมภาพ แล้วจึงทำการคัดแยกคุณลักษณะของภาพ แล้วนำคุณลักษณะภาพที่ได้มาทำการวิเคราะห์ว่าเป็นสถานการณ์อะไร โดยใช้โมเดลที่พัฒนาขึ้นด้วยเทคนิคซัพพอร์ตเวกเตอร์แมชชีนโดยเลือกใช้เคอร์เนลแบบ Chi2 และกำหนดให้ค่า γ เท่ากับ 0.05, c-value เท่ากับ 0.1 ซึ่งให้ค่าความแม่นยำอยู่ที่ 92.85% และมีค่า f-measure อยู่ที่ 82.89% หากระบบตรวจจับได้ว่าเป็นสถานการณ์ที่ผิดปกติจะทำการแจ้งเตือนไปยังผู้ใช้งานเพื่อให้ทำการตัดสินใจและหาแนวทางแก้ไขต่อไป

5.2 ข้อจำกัดของระบบ

- 1) ระบบไม่สามารถตรวจจับวัตถุในเฟรมภาพที่มีขนาดใหญ่ เนื่องจากเวลาที่ใช้ในการประมวลผลที่มาก ทำให้ไม่สามารถทำงานแบบเรียลไทม์ได้
- 2) สิ่งรบกวนในภาพ เช่น เงา แสง คุณภาพของภาพที่ได้มาจากกล้อง มีผลต่อการตรวจจับมากยิ่งขึ้นสิ่งรบกวนในภาพมาก ประสิทธิภาพของการตรวจจับก็จะลดน้อยลงไป
- 3) การตรวจจับในกรณีที่วัตถุอยู่ใกล้กันเกินไปจะไม่สามารถแยกวัตถุออกจากกันได้ ทำให้การตรวจจับผิดพลาด
- 4) ใช้เครื่องมือในการพัฒนาระบบหลายอย่างในการพัฒนา
- 5) การตั้งค่าระบบก่อนการเริ่มทำงานค่อนข้างมีความซับซ้อน เนื่องจากโปรแกรมที่ใช้ในการพัฒนากับไลบรารีไม่รองรับซึ่งกันและกัน ทำให้ต้องใช้โปรแกรมในการพัฒนาเพิ่มเติม

5.3 ข้อเสนอแนะ

- 1) การจะนำเข้าวิดีโอเพื่อใช้ในการตรวจจับ ควรจะใช้ภาพที่มีคุณภาพสูง เพื่อให้ประสิทธิภาพสูงสุดในการตรวจจับ
- 2) อาจจะต้องมีการสร้างข้อมูลฝึกสอนขึ้นมาเองเพื่อให้ได้ตรงตามที่ต้องการ
- 3) ต้องใช้ข้อมูลฝึกสอนมากขึ้นในการสร้างโมเดล เพื่อให้โมเดลที่ได้มีความแม่นยำที่มากขึ้น

เอกสารอ้างอิง

- [1] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: A Large Video Database for Human Motion Recognition. ICCV, 2011.
- [2] Hou Beiping and Zhu Wen. Fast Human Detection Using Motion Detection and Histogram of oriented. JOURNAL OF COMPUTER, Vol.6, No.8, August 2011.
- [3] Oren Boiman and Michal Irani. Detectig Irregularities in image and video. ICCV, 2005.
- [4] A Brown University Research Group. HMDB: a large human motion database. [Online]. Available : <http://serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/#about>.
- [5] Navneet DALAL. INRIA Person Dataset. [Online]. Available : <http://pascal.inrialpes.fr/data/human>.
- [6] Instant Karma Videos. Robbery Fails INSTANT KARMA Compilation & Instant Justice 2017 [Ep. #3]. [Online]. Available : <https://www.youtube.com/watch?v=t673PJShGc>
- [7] buddi Kit. NEW GANG FIGHT AT CAMPUS WITH KNIFES. [Online]. Available : https://www.youtube.com/watch?v=KqC10De-B7Q&has_verified=1
- [8] 5wh tv. Student Gang war live video...viral @ Trichandra collage, Kathmandu. [Online]. Available : https://www.youtube.com/watch?v=lkn5cHRPJil&has_verified=1
- [9] Fahimeh Rezazadegan, Sareh Shirazi, Ben Upcroft and Michael Milford. Action Recognition: From Static Datasets to Moving Robots. ICRA, 2017.
- [10]Rupali S.Rakibe and Bharati D.Patil. Background Subtraction Algorithm Based Human Motion Detection. International Journal of Scientific and Research Publications, Volume 3, Issue 5, May 2013.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

ขั้นตอนการติดตั้งและตั้งค่าเครื่องมือในระบบ

เครื่องมือที่จำเป็นต้องใช้ภายในระบบมีดังต่อไปนี้

1. OpenCV หรือ Opensource Computer Vision

OpenCV เป็นไลบรารีที่รวบรวมฟังก์ชันต่าง ๆ สำหรับการประมวลผลภาพและคอมพิวเตอร์วิชัน สามารถนำมาพัฒนารวมกับภาษาโปรแกรมต่าง ๆ เช่น C/C++, Python, Java เป็นต้น

2. Microsoft Visual Studio

Microsoft Visual Studio เป็นเครื่องมือที่ช่วยในการพัฒนาซอฟต์แวร์และระบบต่าง ๆ ช่วยลดเวลาในการทำงานและข้อผิดพลาดได้เป็นอย่างมาก

<https://www.visualstudio.com/downloads/>

3. CMake

CMake คือซอฟต์แวร์ที่นำมาสร้างสคริปต์ไฟล์ที่ประกอบไปด้วย ไฟล์ที่มีนิยามวิธีการคอมไพล์ ลิงก์โปรแกรมและไลบรารีที่เขียนด้วยภาษา C++ เพื่อที่จะนำไปใช้ร่วมกับเครื่องมือพัฒนาโปรแกรมต่าง ๆ อาทิ Microsoft Visual C++, Eclipse, Qt เป็นต้น

<https://cmake.org/download/>

4. QT Creator

QT Creator เป็นเครื่องมือในการสร้างแอปพลิเคชันและ GUI สามารถทำงานได้หลายระบบปฏิบัติการ และรองรับไลบรารีของ OpenCV

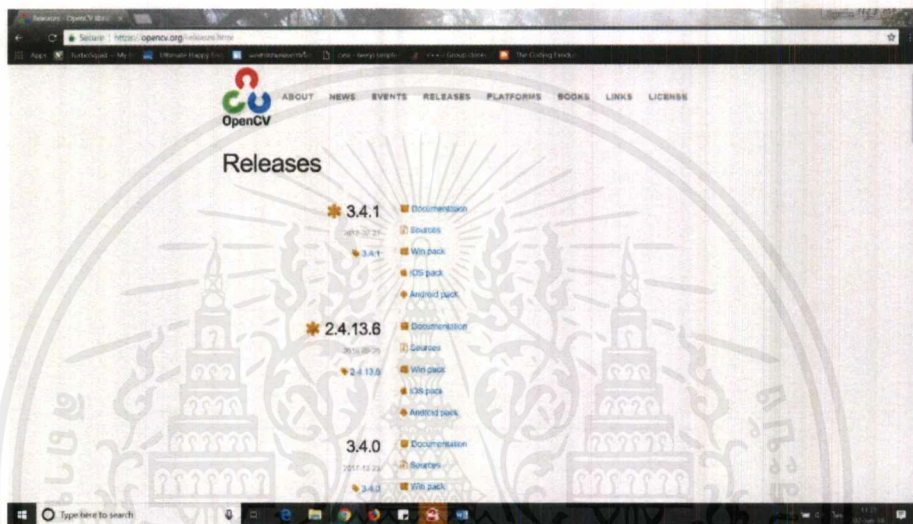
<https://www.qt.io/download-qt-installer?hsCtaTracking=9f6a2170-a938-42df-a8e2-a9f0b1d6cdce%7C6cb0de4f-9bb5-4778-ab02-bfb62735f3e5>

5. C++ runtime Libraries

C++ runtime Libraries คือไลบรารีที่เขียนด้วยภาษา C++ สำหรับนำมาใช้การพัฒนาโปรแกรมที่เขียนด้วยภาษา C++ สำหรับการพัฒนาปัญหาพิเศษนี้จำเป็นต้องใช้ version C++11 ขึ้นไป

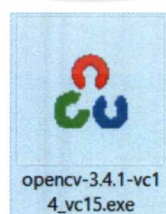
ก.1 ขั้นตอนการติดตั้ง OpenCV

1. สำหรับ C++ runtime Libraries สามารถทำการดาวน์โหลดได้จากเว็บไซต์ของ Microsoft ได้โดยตรง โดย version ที่ใช้ควรเป็น version ล่าสุด หรือ C++11 ขึ้นไปเท่านั้น
2. สำหรับ Opensource Computer Vision ให้ทำการดาวน์โหลด OpenCV จาก <https://opencv.org/releases.html> โดยเลือก Version ตามระบบปฏิบัติการที่ใช้งานอยู่



รูปที่ ก.1 หน้าจอเว็บไซต์ของ OpenCV

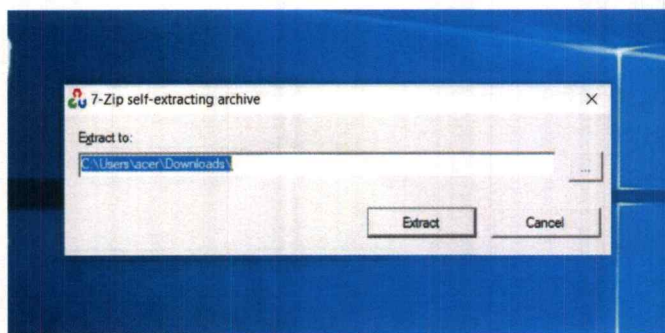
3. เมื่อดาวน์โหลดสำเร็จจะได้ไฟล์นามสกุล exe ชื่อว่า Opencv ตามหลังด้วย version (ในที่นี้คือ 3.4.1) และ version ของ Visual Studio ที่รองรับในที่นี้คือ VC14 และ VC 15



รูปที่ ก.2 ไอคอนของโปรแกรมติดตั้ง OpenCV

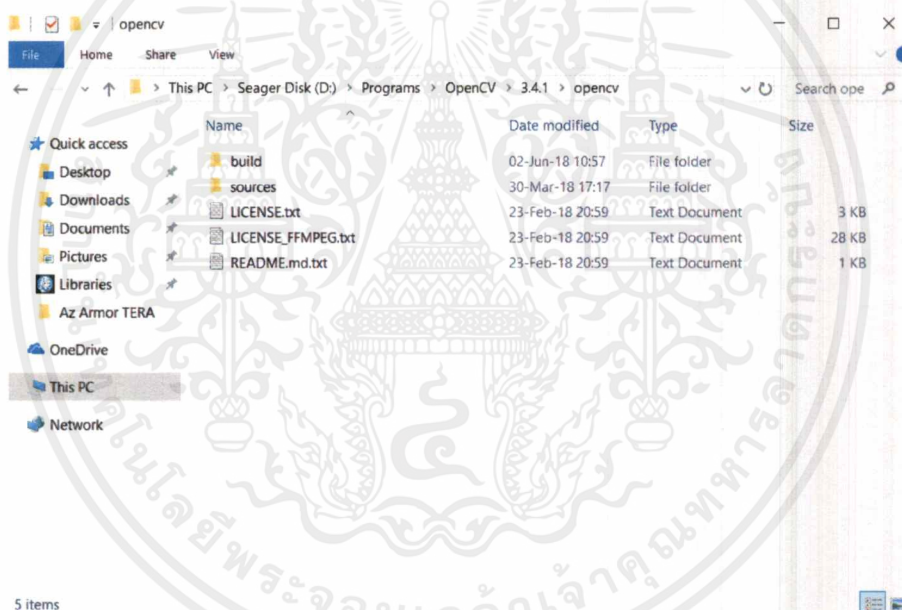
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ให้ทำการรันโปรแกรมขึ้นมาและเลือก path ของไฟล์ที่จะติดตั้งออกมาไว้ที่ใด



รูปที่ ก.3 หน้าจอเลือก path

5. หลังจากติดตั้งเสร็จสิ้นจะได้ไฟล์เตอร์ดังภาพ

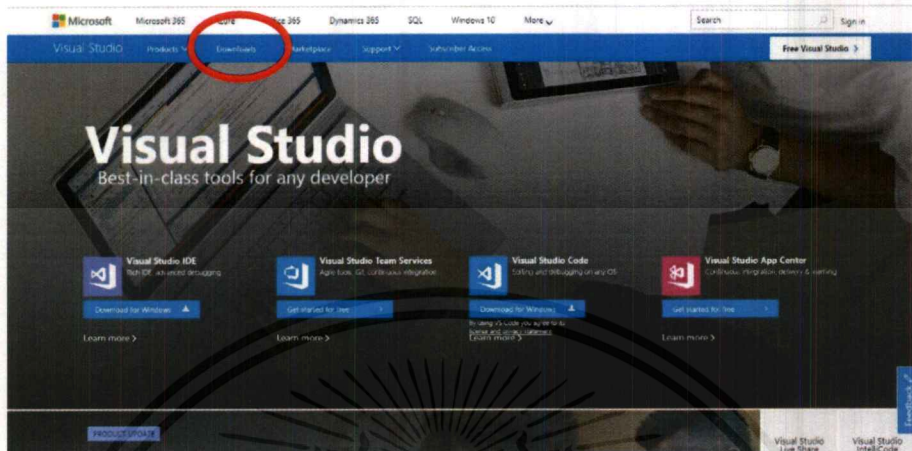


รูปที่ ก.4 หน้าจอในไฟล์เตอร์หลังจากติดตั้ง OpenCV เสร็จสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

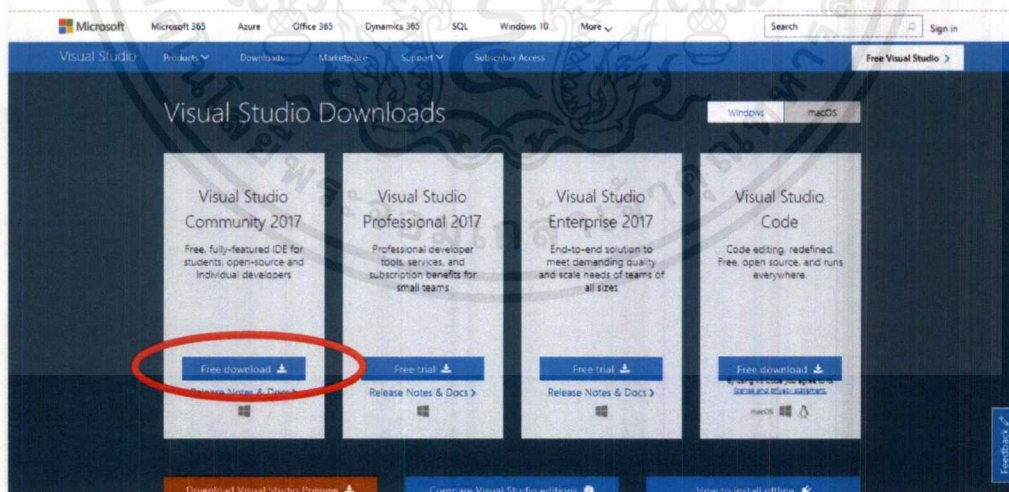
ก.2 ขั้นตอนการติดตั้ง Visual studio

1. เข้าเว็บไซต์ <https://www.visualstudio.com/> เพื่อทำการโหลดโปรแกรมติดตั้ง Visual studio



รูปที่ ก.5 หน้าเว็บไซต์ของ Microsoft


2. กดเลือก Download ที่เมนูจากนั้น เว็บไซต์จะเข้าไปสู่หน้า Visual Studio Downloads เพื่อให้ผู้ใช้เลือกประเภทของโปรแกรมที่ต้องการ ในที่นี้เลือก Visual Studio Community 2017 จากนั้นทำการดาวน์โหลด



รูปที่ ก.6 หน้า Visual Studio Downloads

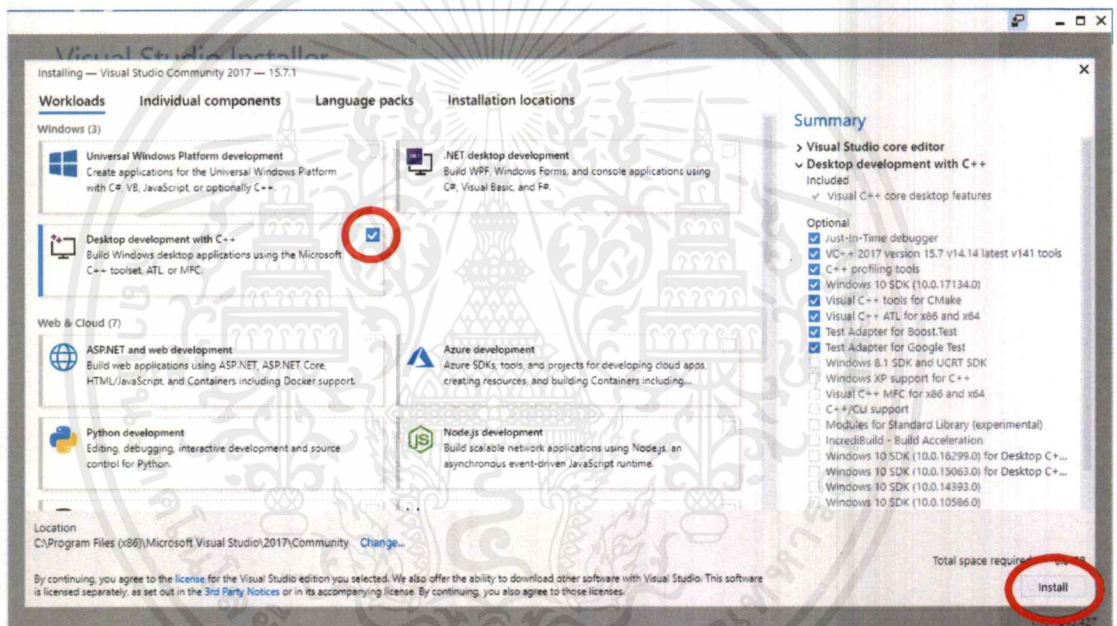
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. หลังจากดาวน์โหลดเสร็จสิ้น เลือกโปรแกรมติดตั้ง Visual studio ที่เก็บไว้ตามที่ผู้ใช้งานบันทึกไว้

<input type="checkbox"/> Name	Date modified	Type	Size
 vs_community_1764510563.15268313...	20/5/2561 23:45	Application	1,178 KB

รูปที่ ก.7 ไอคอนโปรแกรมติดตั้ง Visual studio

4. ระบบจะให้ผู้ใช้งานเลือกประเภทการติดตั้ง ให้ผู้ใช้เลือก Desktop development with C++ จากนั้นกดปุ่ม Install



รูปที่ ก.8 หน้าจอเลือกประเภทของการติดตั้ง

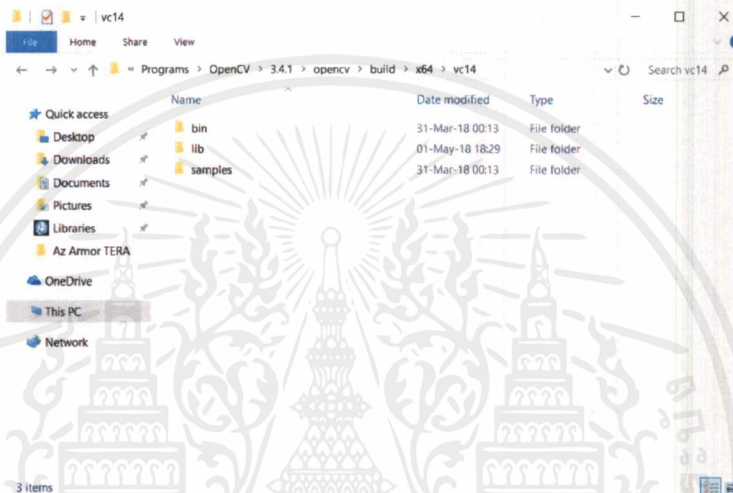
5. รอจนระบบดาวน์โหลด และติดตั้งเรียบร้อยแล้ว จากนั้นกดปุ่ม Restart เป็นอันเสร็จสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.3 การตั้งค่า Variable ของ OpenCV

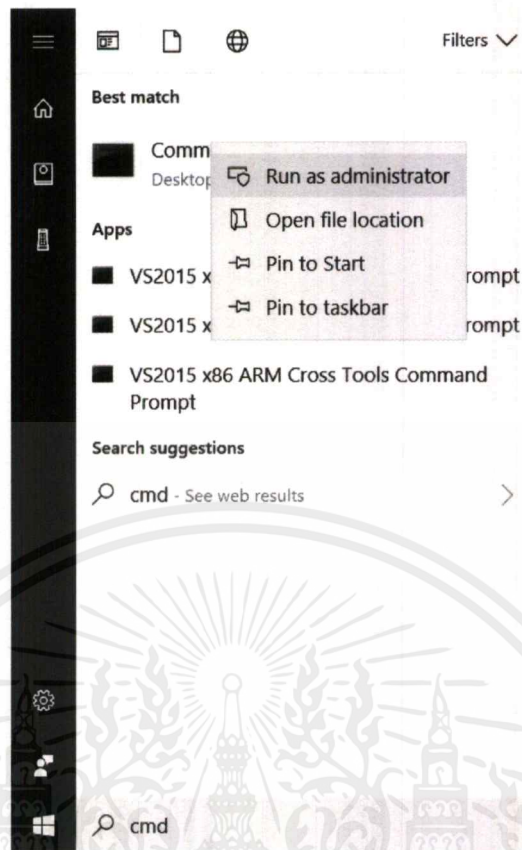
1. เข้าไปยังโฟลเดอร์ที่ทำการติดตั้ง OpenCV เอาไว้ และไปยัง 3.4.1\opencv\build\x64\vc14 โดยที่ x64 และ vc14 จะแตกต่างกันไปตามแต่ละเครื่องและประเภทของระบบปฏิบัติการ (หากใช้ VS2017 ให้เข้าไปที่ vc15 แทน และหากให้ระบบปฏิบัติการ 32 บิต ให้ใช้ x86 แทน เมื่อเข้าไปยังโฟลเดอร์ดังกล่าว จะพบไฟล์ดังภาพ)

หมายเหตุ : สำหรับ OpenCV 3.4.1 รองรับเพียง VS 2015 และ VS 2017 เท่านั้น



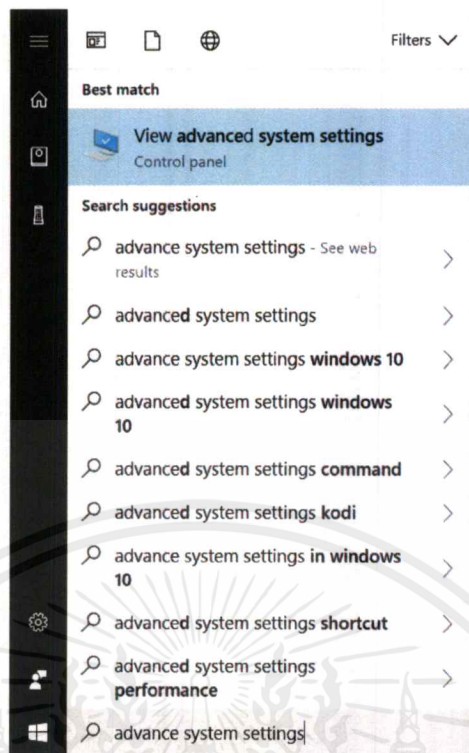
รูปที่ ก.9 โฟลเดอร์ที่ติดตั้ง OpenCV

2. จากนั้นทำการเปิด Command Prompt ขึ้นมาโดยการพิมพ์ cmd ใน search menu ของ Windows และคลิกขวา กด Run as Administrator



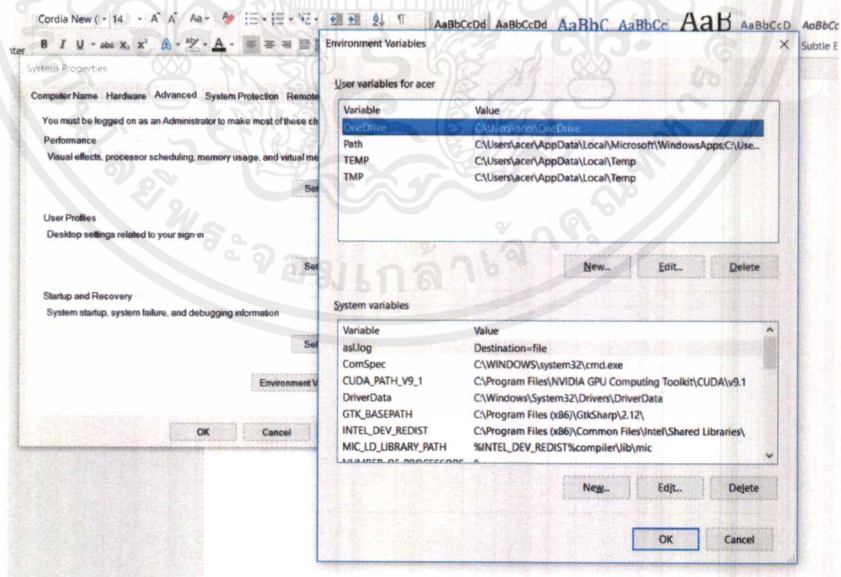
รูปที่ ก.10 หลังจาก search cmd ลงใน search menu

3. เมื่อเปิดหน้าจอ cmd ขึ้นมาแล้ว ให้พิมพ์คำสั่งดังนี้ `setx -m OPENCV_DIR *full path` ของ OPENCV ที่เราทำการเปิดไว้ตอนแรก * โดยไม่ต้องใส่เครื่องหมาย * ตัวอย่าง `setx -m OPENCV_DIR D:\OpenCV\Build\x64\vc14` จากนั้นกด enter
4. ทำการตั้งค่า path ภายใน environment variable โดยการพิมพ์ใน search menu ของ windows ด้วยคำว่า advance system settings และเข้าไปยัง view advanced system settings



รูปที่ ก.11 ทำการค้นหา advance system settings

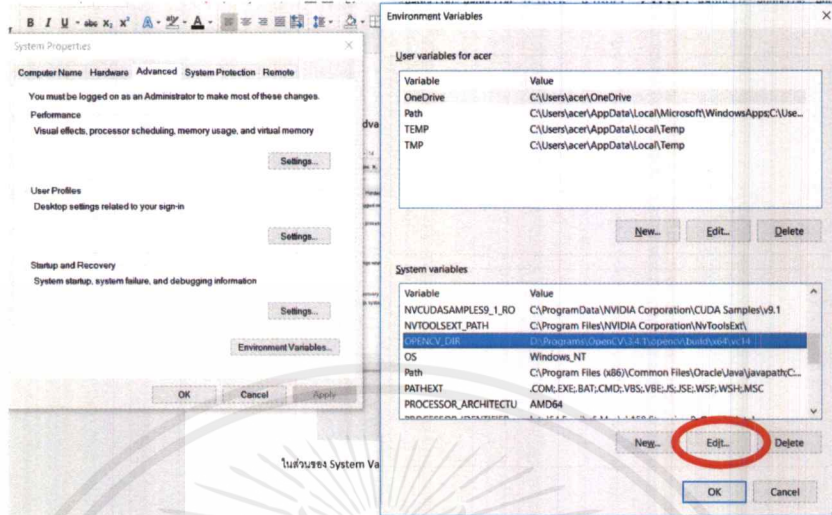
5. ไปยัง tab Advanced และเลือก Environment Variable



รูปที่ ก.12 หน้าจอเลือก Environment Variable

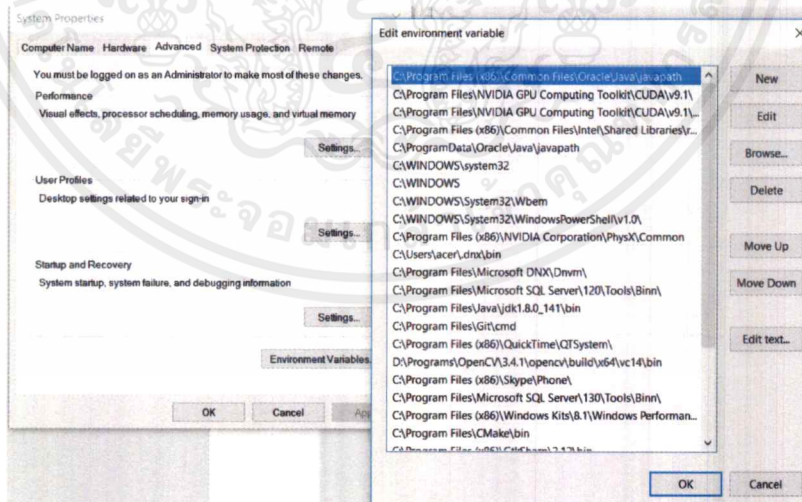
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ในส่วนของ System Variables ให้ค้นหา Variable ที่ชื่อว่า OPENCV_DIR ดังรูปที่ ก.13 จากนั้นไปที่ Path ที่อยู่ในส่วนเดียวกันและกด Edit



รูปที่ ก.13 หน้าจอเลือก System Variable

7. เมื่อกด edit แล้วจะปรากฏหน้าต่าง System path ออกมา ให้ทำการกด New และ ใส่ค่า Full Path ของ OPENCV Directory เข้าไป และลงท้ายด้วย \bin ตัวอย่าง D:\OpenCV\Build\x64\vc14\bin และกด Ok ทุกหน้าต่าง เป็นอันเสร็จสิ้นการตั้งค่า System Path



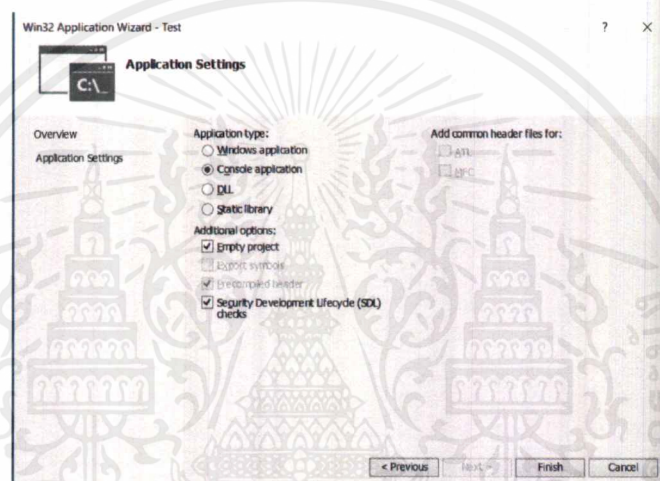
รูปที่ ก.14 การตั้งค่า System Path

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หากต้องการตรวจสอบ path ของ OPENCV ให้ทำการเปิด cmd และพิมพ์คำสั่ง %OPENCV_DIR%\bin และกด enter

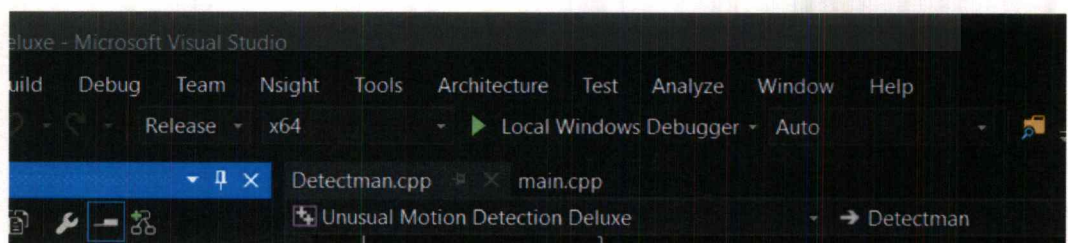
ก.4 การตั้งค่าของ Microsoft Visual Studio เพื่อใช้งานกับ Opencv libraries

- ทำการเปิดโปรแกรม Visual Studio เมื่อโปรแกรมพร้อมใช้งานให้ทำการกด New Project เลือก Template เป็น Visual C++ และ Win32 Project เลือกตั้งชื่อและ Directory ตามต้องการ และกด OK เมื่อกด สร้างเสร็จสิ้นจะมีหน้าต่างค่าเพิ่มเติมแสดงขึ้นมา ให้ทำการตั้งค่าดังภาพนี้ โดยที่เลือกเป็น Console Application และเช็คที่ Empty Project และกด Finish



รูปที่ ก.15 หน้าจอการสร้างโปรเจ็คใหม่

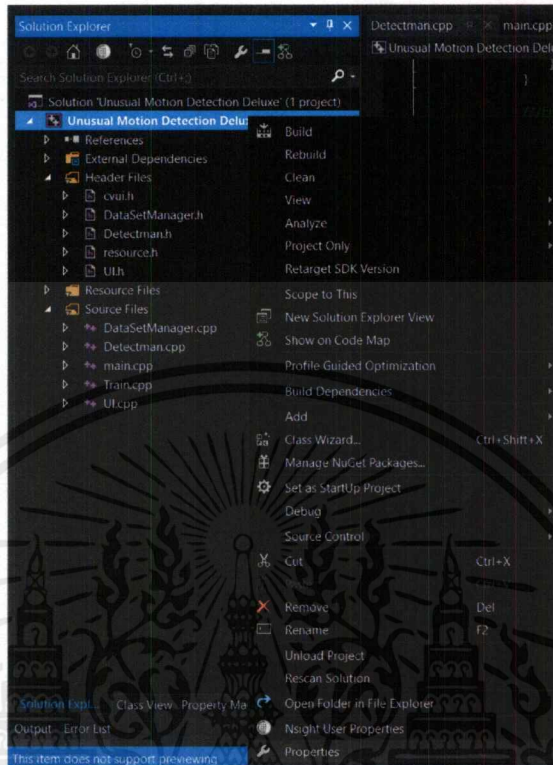
- จากนั้นเมื่อสร้างโปรเจคเสร็จสิ้น ในบริเวณ Build Type ที่อยู่บน tools bar จะแสดง Debug หรือ Release บริเวณด้านข้างจะมีแถบแสดงว่า x64 หรือ x86 ให้ปรับเป็น x64 สำหรับ Windows 64 บิต หรือ x86 สำหรับ Windows 32 บิต



รูปที่ ก.16 ตั้งค่า Build สำหรับ Windows 32 และ 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

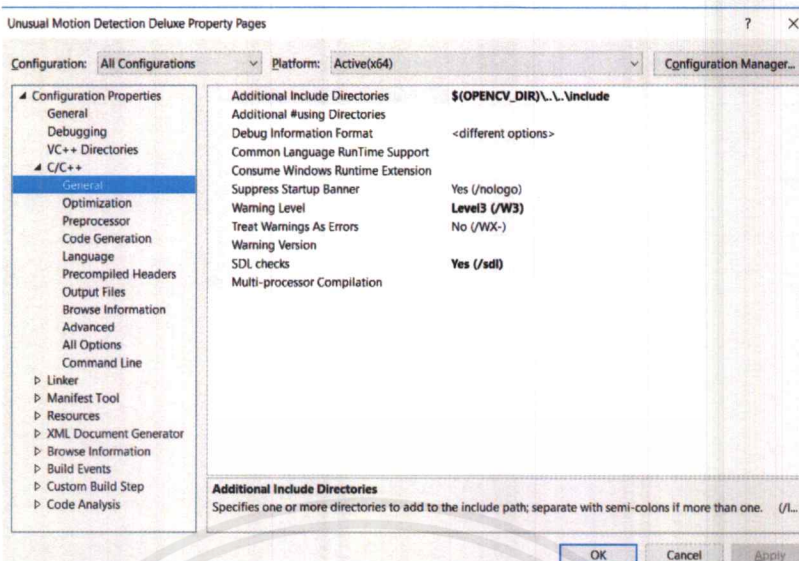
3. บริเวณ Solution Explorer บริเวณด้านซ้ายของหน้าจอให้คลิกขวาที่ชื่อของโปรเจกต์ที่ได้สร้างขึ้น (ไม่ใช่ Solution) และคลิกเลือกที่ Properties



รูปที่ ก.17 หน้า Properties

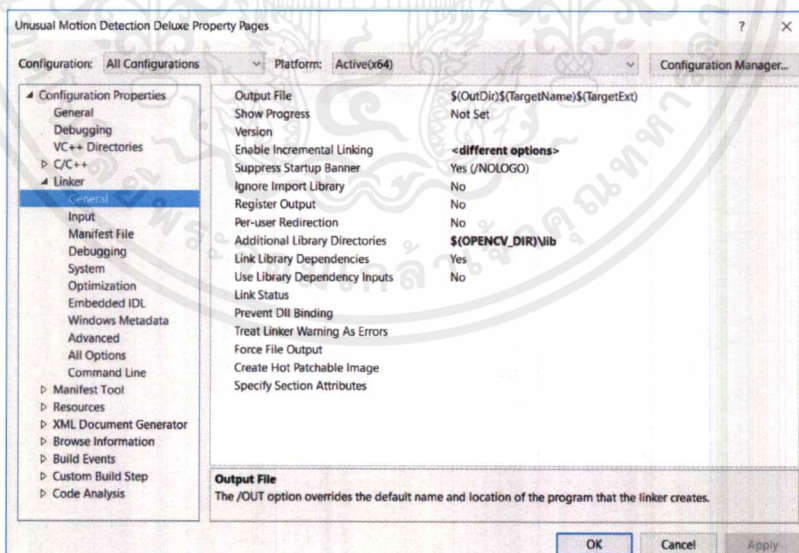
4. เมื่อเปิดหน้าจอ Properties ขึ้นมาแล้ว ในส่วนของ Configuration ในแถบด้านบนของหน้าจอให้ปรับเป็น All Configurations เพื่อให้การปรับถูกใช้งานทั้งแบบ Debug และ Release จากนั้นให้กดที่ C/C++ และเลือก General ดังภาพ

หมายเหตุ : ในกรณีนี้สามารถปรับเป็น Debug หรือ Release ได้โดยไม่จำเป็นต้องปรับเป็น All Configurations แต่หากปรับเป็น Debug หรือ Release การตั้งค่าจากตรงนี้เป็นต้นไปจะต้องทำสองครั้ง คือ ทำการตั้งค่าในโหมด Debug และ ในโหมด Release เพื่อให้โปรแกรมทำงานได้ตามปกติ แต่หากปรับให้เป็น All Configurations จะทำเพียงครั้งเดียว และการปรับทั้งสองแบบจะแตกต่างกันซึ่งจะกล่าวในขั้นตอนต่อไป



รูปที่ ก.18 ส่วนของการ Configuration

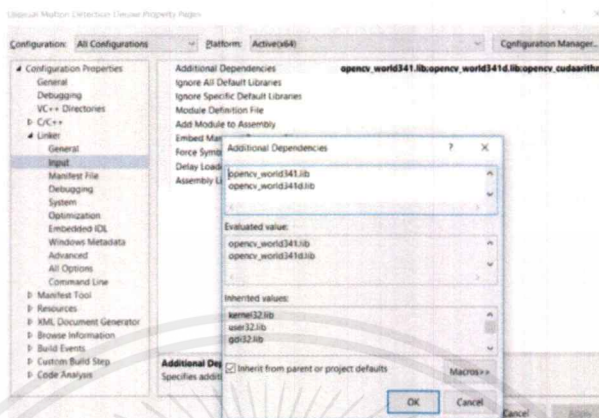
5. ในส่วนของ Additional Include Directories ให้ใส่ค่าดังนี้
 \$(OPENCV_DIR)\..\include จากนั้นไปที่ Linker > General ในส่วนของ
 Additional Library Directories ใส่คำสั่งดังนี้ \$(OPENCV_DIR)\lib



รูปที่ ก.19 ส่วนของ Additional Include Directories

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ไปยัง Linker > Input ในส่วน Additional Dependencies ให้ใส่รายชื่อ libraries ที่จำเป็นต้องใช้



รูปที่ ก.20 หน้าจอใส่รายชื่อ Libraries ที่จำเป็น

ในการพัฒนาระบบเฟิร์มแวร์อัจฉริยะ Libraries ที่ใช้จะมีดังนี้

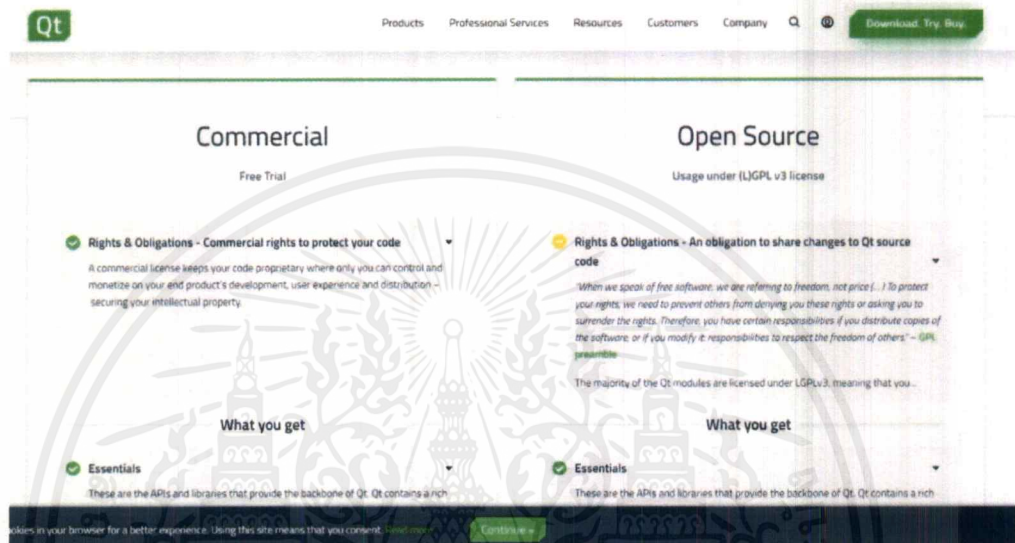
หมายเหตุ : ในส่วนนี้หากใช้ All Configuration ให้ทำการใส่ Libraries ทั้งแบบ 341.dll ธรรมดาและแบบ 341d.dll ซึ่ง dll ที่มี d หมายถึง libraries สำหรับการ debug และที่ไม่มีคือ libraries ของ Release mode ซึ่งการเลือกใช้มีผลต่อขนาดและประสิทธิภาพตอนทำ package สำหรับ Release ในอนาคต ฉะนั้นจะต้องเลือกให้ดีและใช้เท่าที่จำเป็นเท่านั้น แต่ถ้าหากใช้ config แบบ debug และ release แยกกันที่ไม่ใช่ All Configuration ภายใน Debug mode ให้ใส่เฉพาะ dll ที่ลงท้ายด้วย d เท่านั้นสำหรับ debug และสำหรับ release mode ให้ใส่เฉพาะ dll ปกติที่ไม่มี d สำหรับ libraries world เป็น libraries พิเศษที่เป็น libraries รวมฟังก์ชันหรือ libraries ย่อยเอาไว้ทั้งหมดแล้ว ฉะนั้นใน pre-built libraries หรือ libraries ที่ทำการดาวน์โหลดมาและไม่มีการปรับแต่งใช้เพียงแค่ world ก็พร้อมทำงานแล้ว

ในส่วนของ CUDA libraries ของ Nvidia ที่ใช้ GPU ในการช่วยประมวลต้อง include เพิ่มจาก world หากมีการปรับแต่งตัว libraries ที่ pre-built มาแล้ว ซึ่งจะกล่าวในขั้นตอนของ CMAKE ต่อไป

7. เมื่อใส่ libraries ที่จำเป็นเรียบร้อยแล้ว กด OK เป็นอันเสร็จการตั้งค่าใน Visual Studio การเรียกใช้ Libraries เหล่านี้เพียงใช้คำสั่ง `#include <>` ใน source ไฟล์ ตามปกติก็สามารถเรียกใช้งาน OpenCV libraries ได้

ก.5 การติดตั้งและตั้งค่าโปรแกรม QT

1. ไปยังเว็บไซต์ <https://www.qt.io/download> จากนั้นให้เลือก Open Source



รูปที่ ก.21 เว็บไซต์ของ QT

2. เมื่อกดปุ่ม Go open source แล้วจะลิงค์ไปยังอีกหน้าหนึ่ง ให้เลื่อนลงมาบริเวณด้านล่างสุดของหน้าจอ จะมีข้อความ “We do recommend you use the online installer for the best installation experience, however if you are looking for offline packages you can get them here.” ให้กดที่ offline packages you can get them here เมื่อมายังหน้าดาวน์โหลดแบบออฟไลน์ให้เลือกที่ Windows Host (QT version ที่ใช้ในการพัฒนาระบบคือ 5.10.1 64 bits)

Offline Installers

Qt offline installer is a stand-alone binary package including Qt libraries and Qt Creator.

Linux Host

› Qt 5.11.0 for Linux 64-bit (1.2 GB) [\(info\)](#)

macOS Host

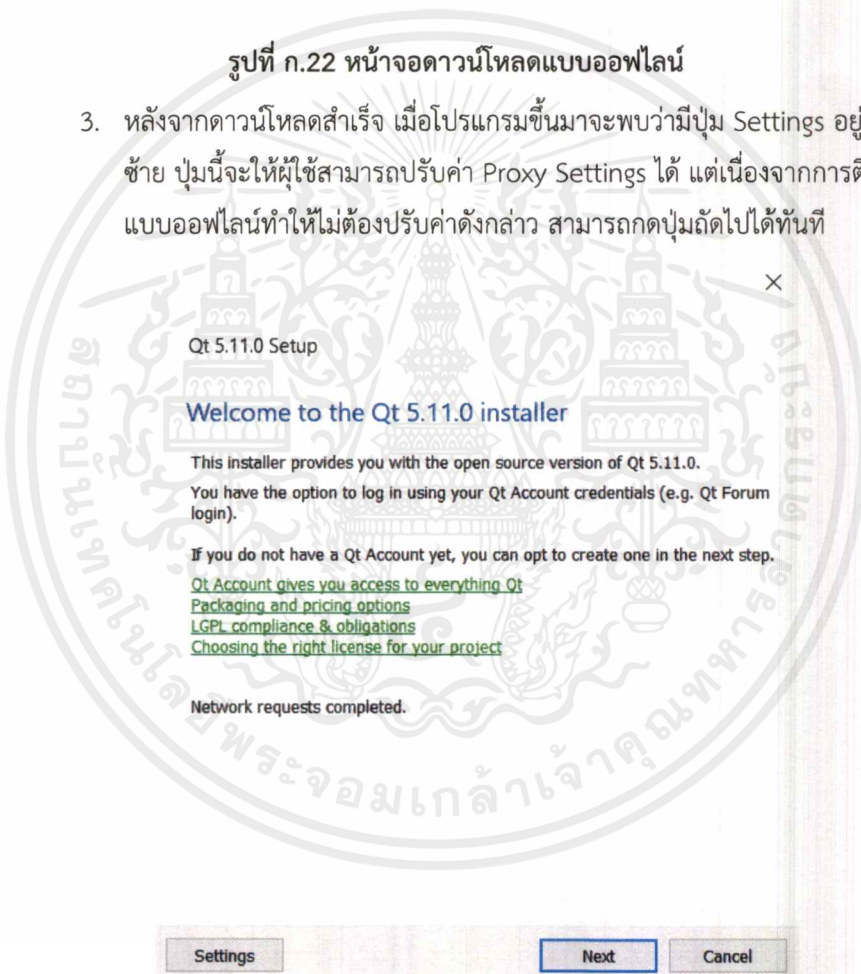
› Qt 5.11.0 for macOS (2.7 GB) [\(info\)](#)

Windows Host

› Qt 5.11.0 for Windows (2.4 GB) [\(info\)](#)

รูปที่ ก.22 หน้าจอดาวน์โหลดแบบออฟไลน์

- หลังจากดาวน์โหลดสำเร็จ เมื่อโปรแกรมขึ้นมาจะพบว่ามีส่วน Settings อยู่ที่มุมล่างซ้าย ปุ่มนี้จะให้ผู้ใช้สามารถปรับค่า Proxy Settings ได้ แต่เนื่องจากการติดตั้งเป็นแบบออฟไลน์ทำให้ไม่ต้องปรับค่าดังกล่าว สามารถกดปุ่มถัดไปได้ทันที



รูปที่ ก.23 หน้าจอปรับค่า Proxy Settings

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ถัดมาเป็นหน้าจอ Log in ให้ทำการกด Skip หรือ Log in/Register ตามสะดวก

Qt 5.11.0 Setup

Qt Account – Your unified login to everything Qt

Please log in to Qt Account

Login Email

Password

[Forgot password?](#)

Need a Qt Account?

Sign-up Valid email address

Password

Confirm Password

I accept the [service terms](#).

Settings Skip Cancel

รูปที่ ก.24 หน้าจอ Log in

5. เลือกโฟลเดอร์ติดตั้งโปรแกรม และทำเครื่องหมายถูกต้องหน้า Associate common file types with QT Creator

Qt 5.11.0 Setup

Installation Folder

Please specify the directory where Qt 5.11.0 will be installed.

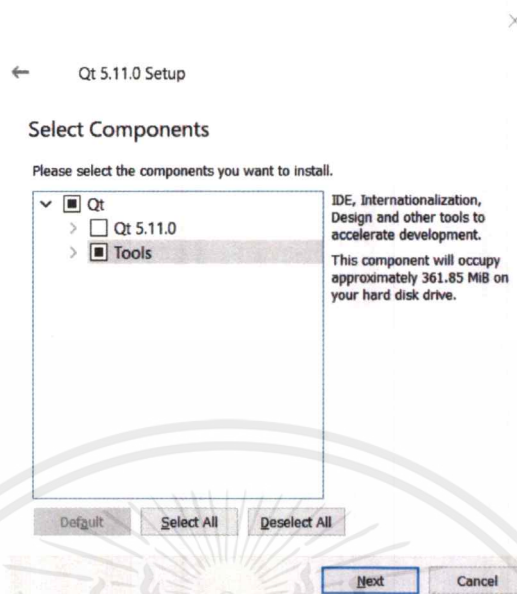
C:\Qt\Qt5.11.0 Browse...

Associate common file types with Qt Creator.

Next Cancel

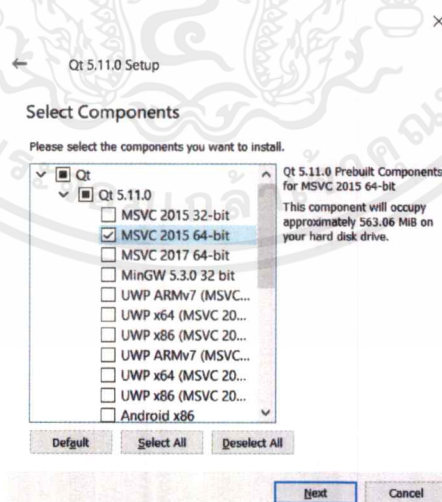
รูปที่ ก.25 เลือกโฟลเดอร์ติดตั้งโปรแกรม

6. หน้าถัดมาจะมี Package ให้เลือกติดตั้ง โดยหัวข้อหลักคือ QT 5.x.x.x และ Tools



รูปที่ ก.26 หน้าจอเลือก Package เพื่อติดตั้ง

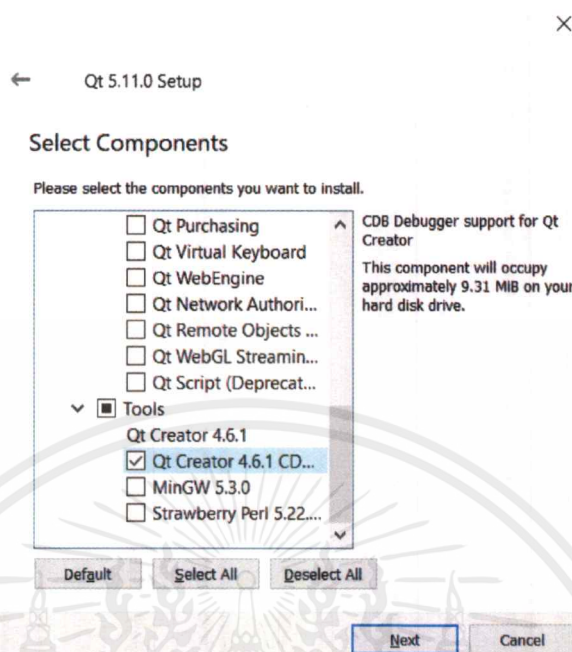
ให้ทำการเลือกขยายตัวเลือก QT 5.x.x.x และ Tools ในช่อง QT 5.x.x.x ให้เลือก MSVC 201x xx-bit โดย เลือกเวอร์ชันตาม Microsoft Visual studio และบิตที่เลือกใช้งานในขั้นตอนก่อนหน้านี ตัวอย่าง สำหรับ Visual Studio 2015 64 bits ให้เลือกที่ MSVC 2015 64 bit



รูปที่ ก.27 เวอร์ชันของ QT ที่ใช้ในการพัฒนาระบบ

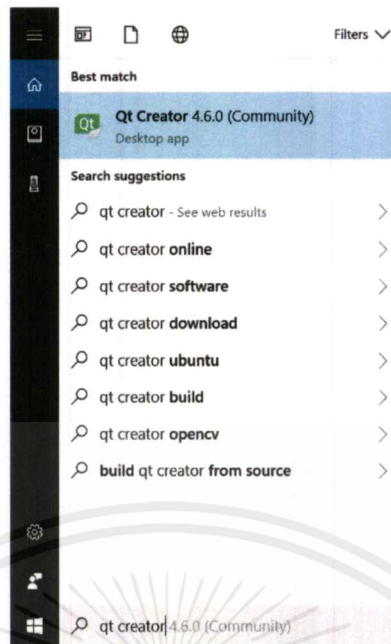
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ต่อมาให้เลื่อนขยายตัวเลือก Tools และเลือก QT Creator



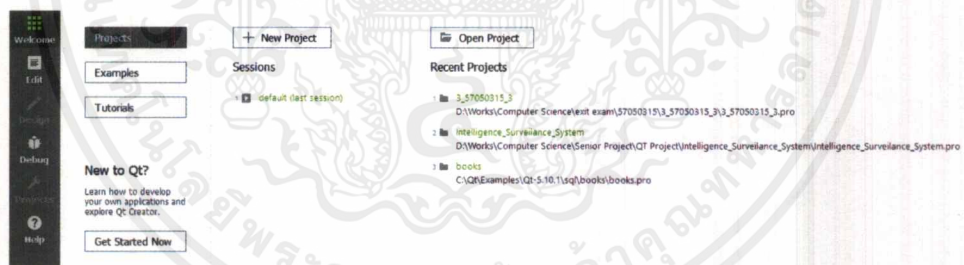
รูปที่ ก.28 เลือก QT Creator

8. ในหน้าต่างมาเป็นหน้าจอ License Agreement ให้เลือกใช้ License QT Installer LGPL License Agreement และเลือก I have read... จากนั้นกดปุ่มถัดไปต่อไปเรื่อย ๆ จนทำการติดตั้งและรอการติดตั้งให้เสร็จสิ้น
9. ในการใช้งาน QT ในการทำหน้าที่จอตต่อผู้ใช้ ให้เปิดโปรแกรม QT Creator ขึ้นมาด้วยการพิมพ์ใน Windows Search ว่า QT Creator



รูปที่ ก.29 ค้นหาโปรแกรม QT Creator

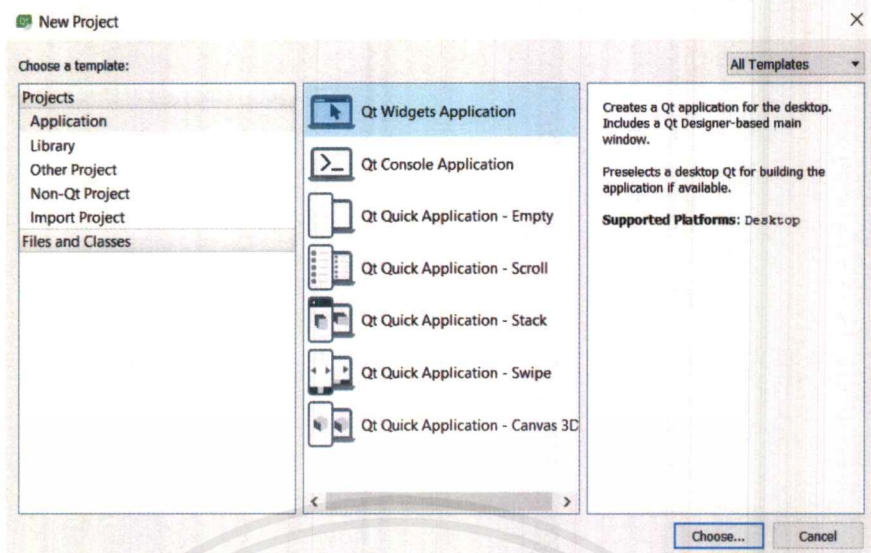
10. ในหน้าจอโปรแกรม บนแถบเมนูซ้ายมือให้เลือกที่ Project และกดปุ่มเครื่องหมาย + New Project เพื่อทำการสร้างงานใหม่



รูปที่ ก.30 หน้าจอโปรแกรม QT Creator

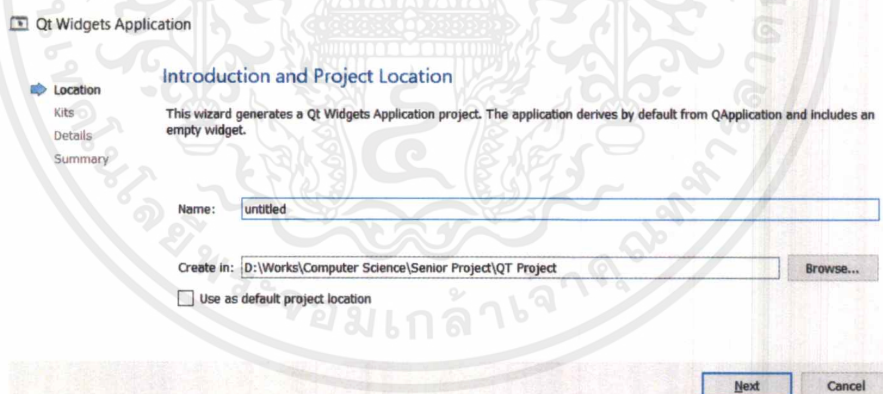
11. กด New Project จะมีหน้าต่างใหม่แสดงขึ้น ในส่วนของ Project ให้เลือกเป็น Application และจากการที่จะนำ QT มาใช้ในการสร้างส่วนติดต่อกับผู้ใช้งานจึงเลือก QT Widgets Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



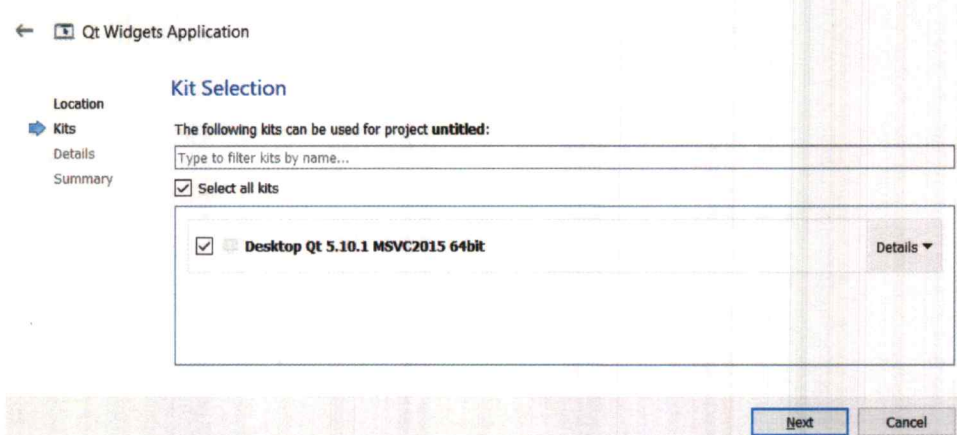
รูปที่ ก.31 หน้าจอเลือกประเภทของโปรเจกต์ที่สร้างขึ้น

12. เมื่อกดปุ่ม Choose... หน้าต่างจะปิดและเปิดอีกหน้าต่างขึ้นมาเป็นหน้าต่างตั้งค่า Project ตั้งค่าชื่อ Project และเลือกโฟลเดอร์ที่ Project จะถูกเก็บไว้ จากนั้นกดปุ่ม Next



รูปที่ ก.32 เลือกโฟลเดอร์ที่เก็บ Project

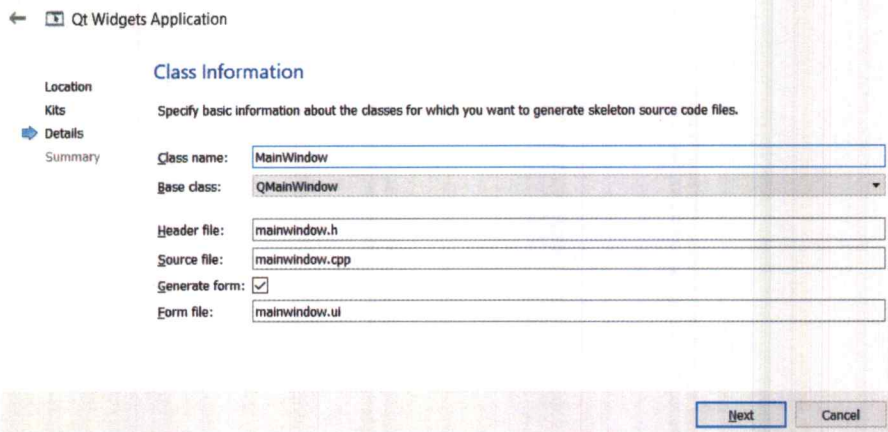
13. ถัดมาคือการเลือก Tools ที่จะใช้ ในที่นี้จะมีเพียง QT 5.x.x.x MSVC 2015 64 bit เพียงอันเดียวเท่านั้นให้ทำเครื่องหมายถูกต้องหน้า Tools ดังกล่าวและกดปุ่ม Next



รูปที่ ก.33 เลือก Tool ที่ใช้ในการพัฒนา

14. ในหน้าต่างถัดมาจะแสดงรายละเอียดของคลาส (Class) ซึ่งมีรายละเอียดดังนี้

- 1) Class Name : เป็นชื่อของคลาสหลักที่ใช้ในโปรแกรม
- 2) Base Class : เป็นการระบุคลาสที่คลาสหลักจะทำการ Inheritance มาใช้งานในโปรแกรม ซึ่งแนะนำให้ปรับเป็น QMainWindow เอาไว้ซึ่งเป็นคลาสที่ใช้แสดงผลในหน้าจอหลัก
- 3) Header file : เป็นไฟล์ Header ของ Class หลักซึ่งชื่อต้องสอดคล้องกับ Class Name ในขั้นต้น
- 4) Source file : ไฟล์คำสั่งหลักของโปรแกรม ซึ่งชื่อต้องสอดคล้องกับ Class Name ในขั้นต้นเช่นกัน
- 5) Generate form : ให้ใส่เครื่องหมายถูกต้องเอาไว้ เนื่องจากต้องใช้ส่วนโค้ด (Code) ควบคู่กับไฟล์หน้าจอติดต่อผู้ใช้ที่เป็นไฟล์แยกต่างหาก (ไฟล์ .ui)
- 6) From File : หากเลือกใช้ Generate Form ตัวเลือกนี้จะใช้งานได้ เลือกชื่อไฟล์นามสกุล ui ซึ่งต้องสอดคล้องกับชื่อ Class Name ที่จะใช้ในการทำเป็นไฟล์รวมข้อมูลของกราฟิกทุกอย่างอยู่ในไฟล์นี้และถูกใช้โดยไฟล์อื่น ๆ ในงาน



รูปที่ ก.34 หน้าแสดงรายละเอียดของคลาส

15. ในหน้าต่างสุดท้ายจะแสดงข้อมูลสรุปผลว่าการสร้าง Project จะได้ข้อมูลอะไรมาบ้าง โดยในส่วนของ version control หากใช้ Github สามารถเลือกปรับแต่งให้เชื่อมต่อกับ Git ได้โดยตรง และกด Finish



รูปที่ ก.35 หน้าสรุปผลการสร้าง Project

16. เมื่อสร้าง Project เสร็จสิ้นจะได้ไฟล์หลัก ๆ มาทั้งหมด 5 ไฟล์ด้วยกันคือ .pro 1 ไฟล์, .ui 1 ไฟล์, .cpp 2 ไฟล์ และ .h 1 ไฟล์
- โดยแต่ละไฟล์มีหน้าที่ดังนี้

.pro : ไฟล์นี้คือไฟล์ควบคุมโครงสร้างทุกอย่างของ Project ตั้งแต่การ link libraries, version control ไปจนถึงการทำงานของ compiler โดยให้ทำการ link libraries ทั้งหมดที่ต้องใช้ของ OpenCV ไว้ที่นี่ด้วยคำสั่ง INCLUDEPATH += และ LIBS += เช่น

INCLUDEPATH +=

D:\Programs\OpenCV\3.4.1\opencv\build\include

LIBS += -

LD:\Programs\OpenCV\3.4.1\opencv\build\x64\vc14\lib\

opencv_imgproc341\

opencv_imgproc341d\

opencv_objdetect341\

opencv_objdetect341d\

opencv_optflow341\

opencv_optflow341d\

opencv_video341\

opencv_video341d\

opencv_videoio341\

opencv_videoio341d\

opencv_videostab341\

opencv_videostab341d\

opencv_world341\

opencv_world341d\

opencv_xfeatures2d341\

opencv_xfeatures2d341d\

opencv_ximgproc341\

opencv_ximgproc341d\

opencv_xobjdetect341\

opencv_xobjdetect341d\

.ui : ไฟล์นี้เป็นไฟล์สำหรับเก็บข้อมูลหน้าจอติดต่อผู้ใช้งานด้านกราฟิกทุกอย่างไว้ในไฟล์สำหรับถูกรียกใช้ด้วยไฟล์อื่น ๆ ในระบบ และใช้ในการออกแบบหน้าจอติดต่อผู้ใช้อีกด้วย

.h : ไฟล์นี้เป็นไฟล์ Header สำหรับการใช้งานสามารถใช้งานได้เหมือนการเขียนโปรแกรมภาษา C และ C++ ทั่วไป

.cpp : ในการสร้าง Project ครั้งแรกจะประกอบไปด้วย 2 ไฟล์ด้วยกันคือ main.cpp และ Project_name.cpp โดย main.cpp เป็นไฟล์เริ่มต้นการทำงานของโปรแกรม โดยเมื่อรันโปรแกรมไฟล์นี้จะถูกเรียกก่อนไฟล์ cpp ตัวอื่น ซึ่งโดยปกติจะไม่แตะต้องไฟล์นี้ การเขียนโปรแกรมและงานอื่น ๆ จะไปทำในไฟล์ Project_name.cpp แทนโดยที่ ไฟล์ ui และไฟล์นี้จะติดต่อกันอยู่โดยสามารถเรียกข้อมูลผ่าน object ui ได้เลย

หมายเหตุ : ในการเขียนโปรแกรมใน QT Widget ควรใช้ Timer หรือ Thread ในการเขียน มิฉะนั้นหน้าจอตีตต่อกับผู้ใช้จะไม่ตอบสนองจนกว่าการประมวลผลจะเสร็จสิ้น

ก.6 การสร้าง Custom Libraries Package สำหรับใช้ในระบบ

ในขั้นตอนนี้มีความจำเป็นเนื่องจากระบบต้องการ libraries บางตัวที่ไม่มีใน pre-built libraries ที่มากับ libraries หลังจากดาวน์โหลดมาในขั้นต้น เช่น Cuda libraries และ QT libraries ซึ่งในระบบใช้ทั้งสอง libraries ภายในระบบ การสร้าง Custom Libraries Package ทำได้ดังขั้นตอนต่อไปนี้

1. ทำการดาวน์โหลด CMake จากเว็บไซต์ <https://cmake.org/download/> โดยเลือก platform ในที่นี้คือ Windows และเลือก x64 หรือ x86 ตามระบบปฏิบัติการ และเลือกเป็นแบบ Installer

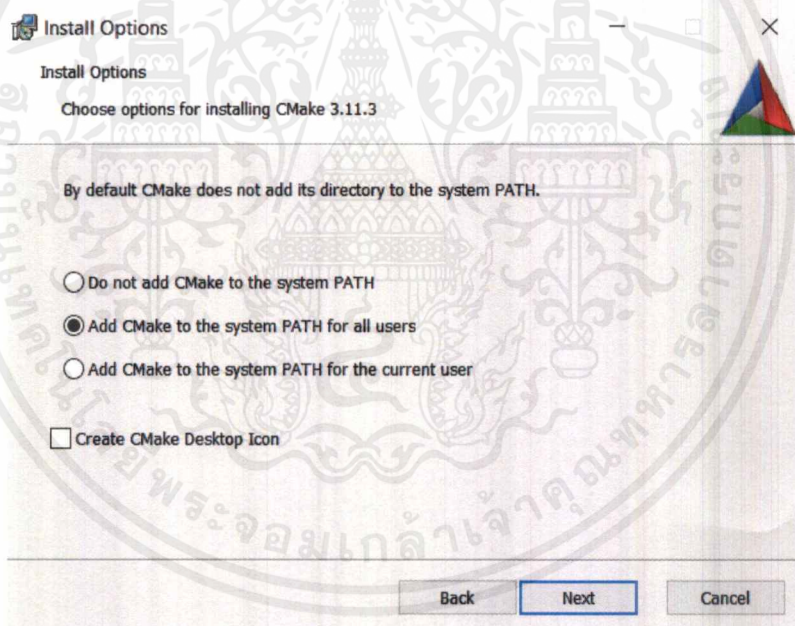
Platform	Files
Unix/Linux Source (has \n line feeds)	cmake-3.11.3.tar.gz cmake-3.11.3.tar.Z
Windows Source (has \r\n line feeds)	cmake-3.11.3.zip

Binary distributions:

Platform	Files
Windows win64-x64 Installer: Installer tool has changed. Uninstall CMake 3.4 or lower first!	cmake-3.11.3-win64-x64.msi
Windows win64-x64 ZIP	cmake-3.11.3-win64-x64.zip
Windows win32-x86 Installer: Installer tool has changed. Uninstall CMake 3.4 or lower first!	cmake-3.11.3-win32-x86.msi
Windows win32-x86 ZIP	cmake-3.11.3-win32-x86.zip

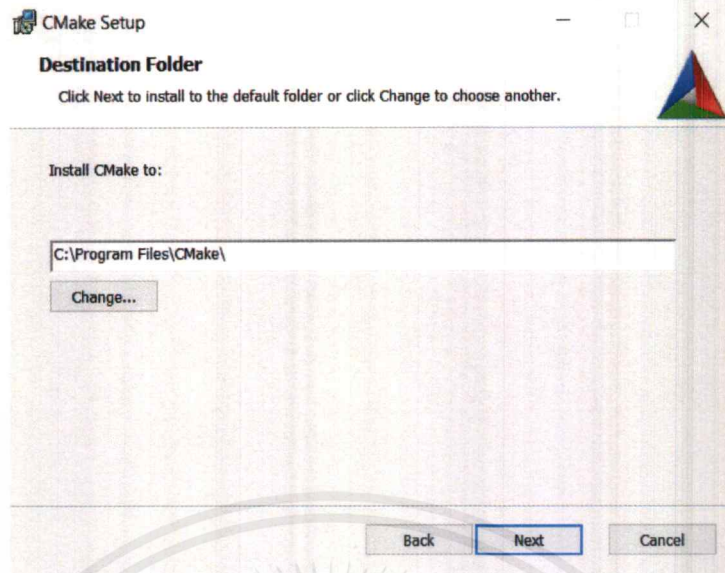
รูปที่ ก.36 หน้าจอตัวติดตั้ง CMake

2. ในขั้นตอนของการติดตั้งในส่วนของ Install Options ให้เลือก Add CMake to the System PATH for all user หรือ for current user



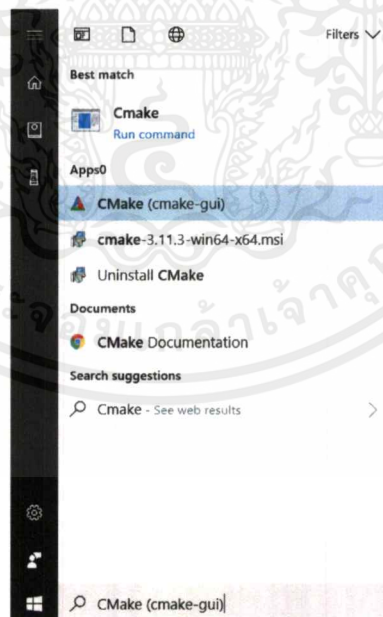
รูปที่ ก.37 เลือก Option ก่อนติดตั้ง CMake

3. เลือกฟลैตเตอร์ที่ต้องการติดตั้ง จากนั้นกด Next เพื่อทำการติดตั้ง



รูปที่ ก.38 เลือกโฟลเดอร์ที่ต้องการติดตั้งโปรแกรม

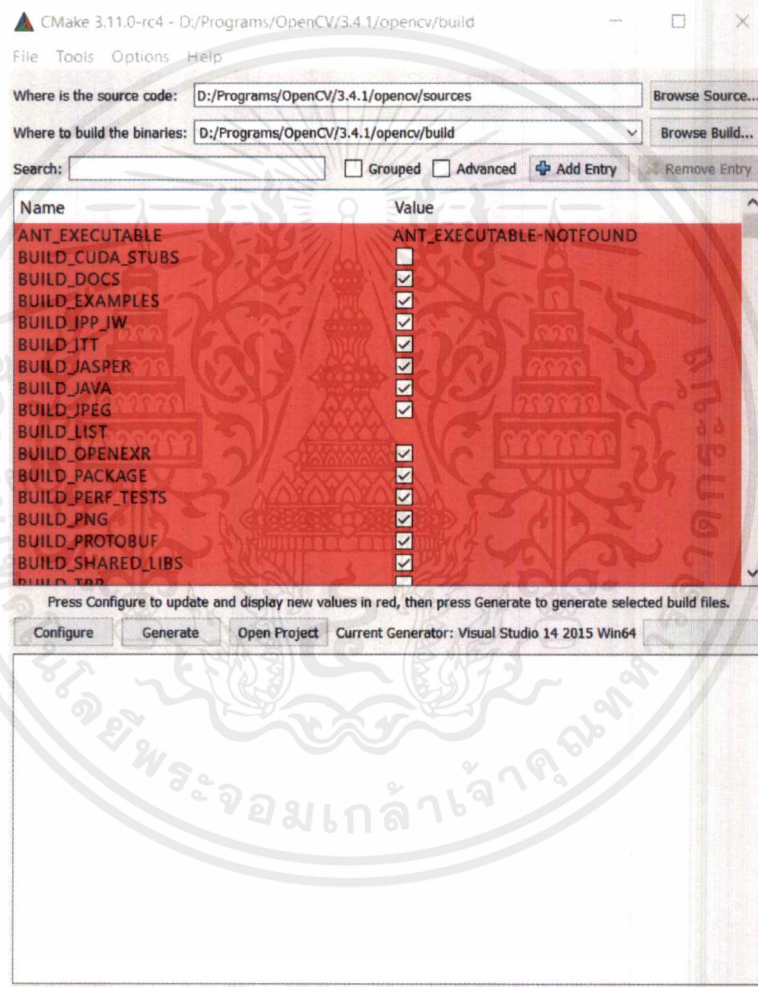
4. เปิดโปรแกรม CMake ขึ้นมาด้วยการค้นหาใน Windows search ว่า CMake และเลือก CMake-GUI



รูปที่ ก.39 ค้นหาโปรแกรม Cmake ใน Windows Search

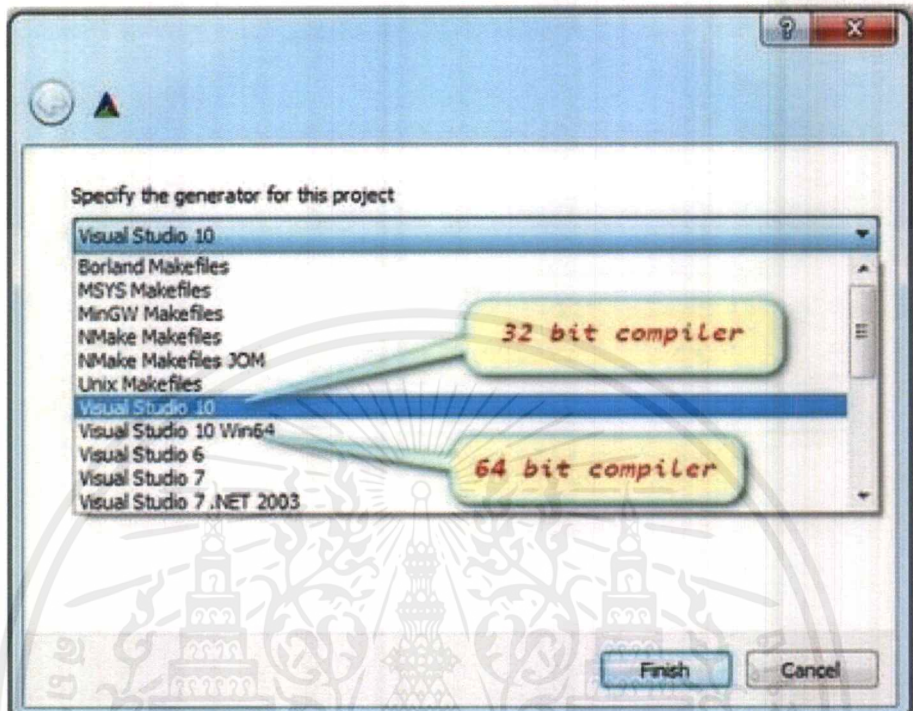
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ในหน้าของโปรแกรม จะมีส่วนของ source code folder และ binaries folder ในส่วน source code ให้เลือกโฟลเดอร์ 3.4.1/opencv/sources แบบ full path ซึ่งเป็นที่ source folder อยู่ ตัวอย่าง
- D:/Programs/OpenCV/3.4.1/opencv/sources ส่วน Binaries folder ให้เลือกที่ได้ตามต้องการ เพราะในส่วนนี้จะเป็นส่วนที่ libraries ที่ถูก build ขึ้นมาใหม่จะอยู่ ตัวอย่าง D:/Programs/OpenCV/3.4.1/opencv/build Libraries ที่ถูก build ใหม่จะอยู่ใน build folder



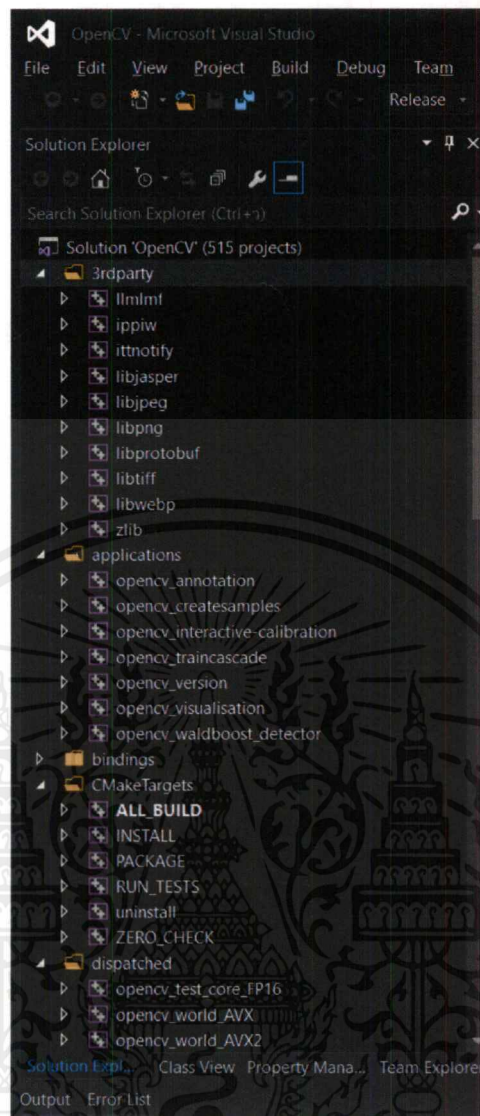
รูปที่ ก.40 เลือกโฟลเดอร์ติดตั้ง

6. เมื่อเลือก path เสร็จเรียบร้อยทั้งสองโฟลเดอร์แล้ว ให้กดที่ Configure เพื่อตั้งค่า libraries จะมีหน้าจอให้เลือก generator สำหรับ Project ขึ้นมา ให้เลือก Visual Studio ที่ใช้งานอยู่ และ Version ที่ใช้ 64 bit หรือ 32 bit และกด Finish



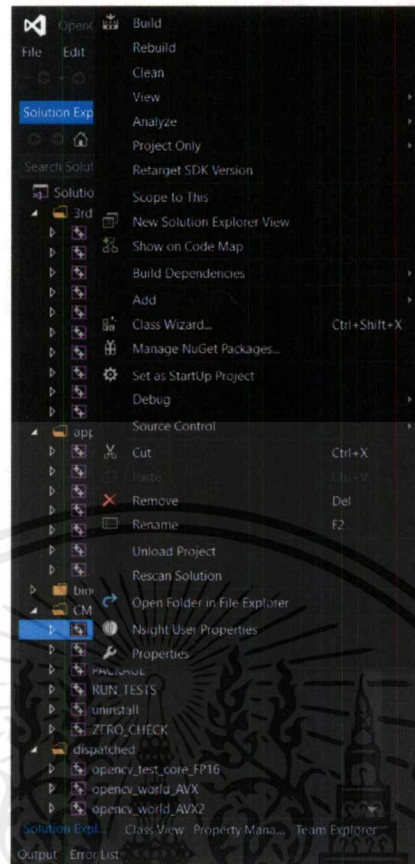
รูปที่ ก.41 หน้าจอให้เลือก generator

7. เมื่อ Configure เสร็จสิ้นจะมีข้อความบ่งบอกที่ช่องด้านล่าง จากนั้น ให้เลื่อนค้นหาและเครื่องหมายถูกที่ BUILD_opencv_world , WITH_CUDA (ถ้าหากใช้รองรับเฉพาะการ์ดจอคอมพิวเตอร์ของ Nvidia เท่านั้น) และ WITH_QT จากนั้นกดที่ Generate เพื่อสร้างข้อมูล libraries สำหรับ build เมื่อ generate เสร็จสิ้นจะมีข้อความแสดงบนหน้าจอด้านล่าง จากนั้นให้กดที่ Open Project จะทำการเปิด Visual Studio ที่ได้ตั้งค่าไว้ในตอน Configure ขึ้นมาพร้อม Solution จำนวนมาก



รูปที่ ก.42 หน้าจอหลังจากที่เปิด Visual studio ที่ตั้งค่าไว้

8. จากนั้นให้เลือกเป็น version x64/x86 ตามการทำงาน และเปลี่ยนเป็นโหมด Debug จากนั้นในโฟลเดอร์ CMakeTargets ให้คลิกขวาที่ ALL_BUILD และเลือก Build ในขั้นตอนนี้อาจใช้เวลาานมากถึง 20 ชั่วโมงและขนาดไฟล์ libraries อาจมีขนาดถึง 10 GB ขึ้นอยู่กับว่าเลือก libraries ไตบ้างในขั้นตอน generate เมื่อ build debug เสร็จสิ้นให้เปลี่ยนเป็นโหมด Release และทำเช่นเดียวกัน รอจนเสร็จสิ้นและปิดโปรแกรมได้



รูปที่ ก.43 ทำการ Build

9. จากนั้นเมื่อ build เสร็จทั้งสองโหมดแล้ว ให้ทำการไปยังโฟลเดอร์ที่ได้เลือกเอาไว้ ในขณะที่อยู่ในหน้าจอ CMake โดยจะพบว่า มีโฟลเดอร์อยู่พอสมควร โดยโฟลเดอร์ทั้งหมดยกเว้น bin และ lib ให้ทำการคัดลอกและนำไปไว้ในโฟลเดอร์ build ของ Opencv 3.4.1 ตัวอย่าง 3.4.1\opencv\build และโฟลเดอร์ bin และ lib ให้นำไปไว้ในโฟลเดอร์ของ opencv visual studio libraries ซึ่งชื่อของโฟลเดอร์จะแตกต่างกันไปตามเวอร์ชันของระบบปฏิบัติการและโปรแกรม Visual Studio ที่ใช้ ตัวอย่าง 3.4.1\opencv\build\x64\vc14

สำหรับ Visual Studio 2015 และระบบปฏิบัติการแบบ 64 บิต ทำการวางทับทุกอย่างที่ถูกถาม จะเป็นการเสร็จสิ้นขั้นตอนการติดตั้งและตั้งค่าทั้งหมดของเครื่องมือที่ต้องใช้ภายในระบบเฟียร์วังอัจฉริยะ

ภาคผนวก ข

ผลการทดลองของการสร้างโมเดลด้วย

เทคนิคการเรียนรู้ของเครื่อง

เนื่องจากในการเลือกใช้โมเดลที่นำมาพัฒนาระบบเฝ้าระวังอัจฉริยะ ได้มีการใช้เทคนิคของการเรียนรู้ของเครื่องแบบมีผู้สอนทั้ง 2 เทคนิค คือ ซัพพอร์ตเวกเตอร์แมชชีน และโครงข่ายประสาทเทียม มาพัฒนาเพื่อให้ได้โมเดลที่มีประสิทธิภาพมากที่สุด

ข.1 ผลลัพธ์ของโมเดลที่พัฒนาด้วยเทคนิคซัพพอร์ตเวกเตอร์แมชชีน

ตารางที่ ข.1 ผลลัพธ์ที่ได้จากเคอร์เนล Linear ที่กำหนดให้ C-value = 0.1

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	342	258	598	1702	57	36.383	44.416	70.483
2	347	253	1016	1284	57.833	25.459	35.354	56.241
3	320	280	598	1702	53.333	34.858	42.16	69.724
4	329	271	664	1636	54.833	33.132	41.306	67.759
5	323	277	857	1443	53.833	27.373	36.292	60.897
6	313	287	701	1599	52.167	30.868	38.786	65.931
7	307	293	564	1736	51.167	35.247	41.741	70.448
8	329	271	778	1522	54.833	29.72	38.547	63.828
9	341	259	872	1428	56.833	28.112	37.617	61
10	335	265	835	1465	55.833	28.632	37.853	62.069
							Average	64.838

ตารางที่ ข.2 ผลลัพธ์ที่ได้จากเคอร์เนล Polynomial ที่กำหนดให้ $\gamma = 0.05$, degree = 3, coef = 1, C-value = 0.1

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	463	137	350	1950	77.167	56.95	65.535	83.207
2	431	169	369	1931	71.833	53.875	61.571	81.448
3	459	141	374	1926	76.5	55.102	64.061	82.241
4	459	141	384	1916	76.5	54.448	63.617	81.897
5	438	162	376	1924	73	53.808	61.952	81.448
6	448	152	336	1964	74.667	57.143	64.74	83.172
7	439	161	382	1918	73.167	53.471	61.787	81.276
8	410	190	308	1992	68.333	57.103	62.215	82.828
9	452	148	404	1896	75.333	52.804	62.088	80.966
10	436	164	342	1958	72.667	56.041	63.28	82.552
							Average	82.1035

ตารางที่ ข.3 ผลลัพธ์ที่ได้จากเคอร์เนล Polynomial ที่กำหนดให้ $\gamma = 0.075$, $\text{degree} = 4$, $\text{coef} = 1$, $\text{C-value} = 0.1$

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	475	125	289	2011	79.167	62.173	69.648	85.724
2	474	126	306	1994	79	60.769	68.696	85.103
3	459	141	289	2011	76.5	61.364	68.101	85.172
4	488	112	320	1980	81.333	60.396	69.318	85.103
5	468	132	273	2027	78	63.158	69.799	86.034
6	475	125	353	1947	79.167	57.367	66.527	83.517
7	457	143	256	2044	76.167	64.095	69.611	86.241
8	479	121	315	1985	79.833	60.327	68.723	84.966
9	455	145	290	2010	75.833	61.074	67.658	85
10	449	151	291	2009	74.833	60.676	67.015	84.759
							Average	85.1619

ตารางที่ ข.4 ผลลัพธ์ที่ได้จากเคอร์เนล RBF ที่กำหนดให้ $\gamma = 0.05$, $\text{C-value} = 0.1$

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	479	121	126	2174	79.833	79.174	79.502	91.483
2	473	127	109	2191	78.833	81.271	80.033	91.862
3	527	73	212	2088	87.833	71.313	78.716	90.172
4	437	163	77	2223	72.833	85.019	78.456	91.724
5	470	130	101	2199	78.333	82.312	80.273	92.034
6	64	536	3	2297	10.667	95.522	19.191	81.414
7	484	116	127	2173	80.667	79.214	79.934	91.621
8	478	122	138	2162	79.667	77.597	78.618	91.034
9	496	104	116	2184	82.667	81.046	81.848	92.414
10	452	148	100	2200	75.333	81.884	78.472	91.448
							Average	90.5206

ตารางที่ ข.5 ผลลัพธ์ที่ได้จากเคอร์เนล RBF ที่กำหนดให้ $\gamma = 0.025$, $\text{C-value} = 0.1$

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	458	142	143	2157	76.333	76.206	76.269	90.172
2	477	123	194	2106	79.5	71.088	75.059	89.069
3	477	123	150	2150	79.5	76.077	77.751	90.586
4	513	87	232	2068	85.5	68.859	76.282	89
5	482	118	137	2163	80.333	77.868	79.081	91.207
6	478	122	127	2173	79.667	79.008	79.336	91.414
7	454	146	115	2185	75.667	79.789	77.673	91
8	508	92	251	2049	84.667	66.93	74.761	88.172
9	434	166	147	2153	72.333	74.699	73.497	89.207
10	447	153	159	2141	74.5	73.762	74.129	89.241
							Average	89.9068

ตารางที่ ข.6 ผลลัพธ์ที่ได้จากเคอร์เนล RBF ที่กำหนดให้ $\gamma = 0.075$, $C\text{-value} = 0.1$

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	498	102	113	2187	83	81.506	82.246	92.586
2	480	120	82	2218	80	85.409	82.616	93.034
3	479	121	88	2212	79.833	84.48	82.091	92.793
4	578	22	596	1704	96.333	49.233	65.163	78.69
5	502	98	100	2200	83.667	83.389	83.528	93.172
6	535	65	173	2127	89.167	75.565	81.804	91.793
7	504	96	92	2208	84	84.564	84.281	93.517
8	491	109	95	2205	81.833	83.788	82.799	92.966
9	510	90	93	2207	85	84.577	84.788	93.69
10	512	88	112	2188	85.333	82.051	83.66	93.103
							Average	91.5344

ตารางที่ ข.7 ผลลัพธ์ที่ได้จากเคอร์เนล Chi2 ที่กำหนดให้ $\gamma = 0.025$, $C\text{-value} = 2$

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	572	28	485	1815	95.333	54.115	69.04	82.31
2	563	37	347	1953	93.833	61.868	74.569	86.759
3	585	15	535	1765	97.5	52.232	68.023	81.034
4	575	25	387	1913	95.833	59.771	73.623	85.793
5	572	28	448	1852	95.333	56.078	70.617	83.586
6	575	25	419	1881	95.833	57.847	72.145	84.69
7	577	23	476	1824	96.167	54.796	69.813	82.793
8	568	32	416	1884	94.667	57.724	71.718	84.552
9	565	35	381	1919	94.167	59.725	73.092	85.655
10	560	40	469	1831	93.333	54.422	68.754	82.448
							Average	83.962

ตารางที่ ข.8 ผลลัพธ์ที่ได้จากเคอร์เนล Chi2 ที่กำหนดให้ $\gamma = 0.05$, $C\text{-value} = 2$

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	579	21	773	1527	96.5	42.825	59.323	72.621
2	130	470	2	2298	21.667	98.485	35.52	83.724
3	582	18	747	1553	97	43.792	60.342	73.621
4	295	305	22	2278	49.167	93.06	64.341	88.724
5	558	42	502	1798	93	52.642	67.229	81.241
6	122	478	6	2294	20.333	95.313	33.516	83.31
7	590	10	1333	967	98.333	30.681	46.769	53.69
8	154	446	4	2296	25.667	97.468	40.634	84.483
9	580	20	1011	1289	96.667	36.455	52.944	64.448
10	374	226	28	2272	62.333	93.035	74.651	91.241
							Average	77.7103

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ข.9 ผลลัพธ์ที่ได้จากเคอร์เนล Chi2 ที่กำหนดให้ $\text{gamma} = 0.075$, $C\text{-value} = 0.1$

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	491	109	295	2005	81.833	62.468	70.851	86.069
2	433	167	88	2212	72.167	83.109	77.252	91.207
3	433	167	76	2224	72.167	85.069	78.089	91.621
4	418	182	77	2223	69.667	84.444	76.347	91.069
5	431	169	94	2206	71.833	82.095	76.622	90.931
6	433	167	93	2207	72.167	82.319	76.909	91.034
7	161	439	17	2283	26.833	90.449	41.388	84.276
8	481	119	202	2098	80.167	70.425	74.981	88.931
9	261	339	41	2259	43.5	86.424	57.871	86.897
10	438	162	77	2223	73	85.049	78.565	91.759
							Average	89.3794

ตารางที่ ข.10 ผลลัพธ์ที่ได้จากเคอร์เนล Chi2 ที่กำหนดให้ $\text{gamma} = 0.05$, $C\text{-value} = 0.1$

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	499	101	85	2215	83.167	85.445	84.291	93.586
2	481	119	63	2237	80.167	88.419	84.091	93.724
3	505	95	108	2192	84.167	82.382	83.265	93
4	497	103	97	2203	82.833	83.67	83.249	93.103
5	500	100	101	2199	83.333	83.195	83.264	93.069
6	482	118	76	2224	80.333	86.38	83.247	93.31
7	501	99	92	2208	83.5	84.486	83.99	93.414
8	498	102	89	2211	83	84.838	83.909	93.414
9	517	83	108	2192	86.167	82.72	84.408	93.414
10	503	97	236	2064	83.833	68.065	75.131	88.517
							Average	92.8551

ข.2 ผลลัพธ์ของโมเดลที่พัฒนาด้วยเทคนิคซัพพอร์ตเวกเตอร์แมชชีน

ตารางที่ ข.11 ผลลัพธ์ที่ได้จากโครงข่ายประสาทเทียม ที่กำหนดให้ Sigmoid = 0.1, Hidden layer = 50, Learning rate = 0.1

	TP	FN	FP	TN	Recall	Precision	F-measure	Accuracy
1	309	291	133	2167	51.5	69.91	59.309	85.379
2	290	310	201	2099	48.333	59.063	53.162	82.379
3	305	295	166	2134	50.833	64.756	56.956	84.103
4	301	299	137	2163	50.167	68.721	57.996	84.966
5	322	278	133	2167	53.667	70.769	61.043	85.828
6	330	270	142	2158	55	69.915	61.567	85.793
7	288	312	130	2170	48	68.9	56.582	84.759
8	276	324	132	2168	46	67.647	54.762	84.276
9	283	317	104	2196	47.167	73.127	57.346	85.483
10	305	295	151	2149	50.833	66.886	57.765	84.621
							Average	84.7587

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้