

## ชื่อโครงการ Home Automation : Surveillance Camera

หมายเลขอุปกรณ์ 5

ชื่อ นายพีรัตน์ กฤดาธิการกุล, ชื่อเล่น พี,  
รหัสนักศึกษา 55070501035, ที่อยู่อีเมลล์ kitda\_pee@hotmail.com

ชื่อ นาย วรณศ ตั้งคุณวนิชญ, ชื่อเล่น เทียน,  
รหัสนักศึกษา 55070501072, ที่อยู่อีเมลล์ tienworanet@gmail.com

ที่ปรึกษาโครงการ รศ.ดร.พีรพล ศิริพงศ์วุฒิกร

วันที่ – 30 พฤษภาคม 2559

---

ข้าพเจ้าได้อ่านรายงานและตรวจเนื้อหาของรายงานเรียบร้อยแล้ว

## บทคัดย่อ (ภาษาอังกฤษ)

Home Automation: Surveillance Camera is a tool to monitor the safety of property or areas that need security by using raspberry pi, webcam camera and smartphone application which use iOS operating system. Main system in raspberry pi can detect motion by using motion sensor and command webcam camera to take snapshot using it compare with face in face database. So, we can know that in this event what event happened. If system detect strange face, it can send email notification to users. Apart from using algorithm to comparing with face database, we also use temperature sensor, humidity sensor and smoke sensor. All data will be saved in cloud data storage (Dropbox). By using application, we can view event in real-time , query data from Dropbox to show in application. All of this, we have to use knowledge from wireless network and wireless sensor networks in order to receive and send data. From the development of Home Automation: Camera Surveillance, we hope people who interested in security surveillance technology will be able to bring knowledge from this study develop their project for better performance or being a guideline for them.

## บทคัดย่อ(ภาษาไทย)

Home Automation : Surveillance Camera เป็นเครื่องมือชนิดหนึ่งที่ในการเฝ้าระวังความปลอดภัยให้กับทรัพย์สินหรือสถานที่ที่ต้องการรักษาความปลอดภัยโดยการใช้ Raspberry Pi กล้องเว็บแคมและแอปพลิเคชันบนโทรศัพท์เคลื่อนที่สามารถตรวจสอบจับการเคลื่อนไหวที่เกิดขึ้นผ่านทางเข็มซอร์ตรวจจับการเคลื่อนไหว และกล้องเว็บแคมจะมีทำการถ่ายภาพเพื่อนำมาประมวลผล เปรียบเทียบกับฐานข้อมูล ถ้าระบบการทำงานตรวจใบหน้าที่ไม่อยู่ในฐานข้อมูล จะทำการแจ้งเตือนไปยังอีเมลล์ของผู้ใช้ นอกจากการใช้อัลกอริทึมในการเปรียบเทียบใบหน้ากับฐานข้อมูลที่มีอยู่ ระบบยังใช้งานเข็มซอร์ที่ใช้ในการตรวจจับอุณหภูมิ ความชื้นและตรวจจับควัน โดยข้อมูลทั้งหมดจะบันทึกอยู่ใน Cloud Data Storage (Dropbox) ภายในแอปพลิเคชัน ผู้ใช้ระบบสามารถรับชมภาพที่เกิดขึ้นจริงในเวลาอันสั้นๆ รวมถึงการเรียกข้อมูลที่อยู่ใน Dropbox มาแสดงผลในแอปพลิเคชัน ทั้งหมดนี้จะใช้ความรู้ทางด้านระบบเครือข่ายไร้สาย ระบบเครือข่ายเข็มซอร์ไร้สายมาเพื่อใช้ในการรับ-ส่งข้อมูล ซึ่งจากการพัฒนา Home Automation : Surveillance Camera ผู้พัฒนาหวังเป็นอย่างยิ่งว่า บุคคลที่สนใจเกี่ยวกับเทคโนโลยีที่ใช้ในการเฝ้าระวังความปลอดภัย สามารถที่จะนำความรู้ที่ได้จากการศึกษา โครงการนี้ไปพัฒนาต่อยอดเพื่อเพิ่มประสิทธิภาพการทำงานให้สูงขึ้น หรือ นำไปเป็นแนวทางในการพัฒนาต่อไป

## กิตติกรรมประกาศ

โครงการฉบับนี้สำเร็จได้ด้วยความอนุเคราะห์ของบุคคลหลายท่านซึ่งไม่อาจจะนำมากล่าวได้ทั้งหมด ซึ่งผู้มีพระคุณที่ผู้จัดทำข้อความขอบพระคุณ คือ รศ.ดร.พีรพล ศิริพงศ์สุทธิกร อาจารย์ที่ปรึกษาโครงการที่ได้ให้ความรู้ ข้อเสนอแนะ และคำแนะนำที่มีประโยชน์ ตลอดจนการตรวจสอบและแก้ไขข้อบกพร่องต่างๆในการจัดทำ โครงการนี้ และด้วยความเอาใจใส่ทุกขั้นตอนเพื่อให้โครงการฉบับนี้สมบูรณ์ที่สุดผู้จัดทำจึงขอขอบพระคุณเป็นอย่างสูง ไว้ ณ โอกาสนี้ นอกเหนือผู้จัดทำข้อความคุณ พศ.พิพัฒน์ ศุภศิริสันต์, อาจารย์ ไกรกร เศรษฐ์ไกรกุล, อาจารย์ ชกร ตั้งมั่นคง, รศ.ดร.นฤมล วัฒนพงศ์กร, อาจารย์ ราชวิชช์ สโตร์วิกสิต ที่ให้คำแนะนำในการค้นคว้าข้อมูล ความรู้ในด้านต่างๆอีกทั้งยังกรุณาร่วมสละเวลาในการตรวจสอบโครงการฉบับนี้ด้วย รวมถึงคณะอาจารย์ประจำภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่านที่ช่วยประสิทธิ์ประศาสดร์วิชาความรู้ตลอดการเรียนการสอน

ขอขอบพระคุณ คุณพ่อ คุณแม่ และครอบครัว ตลอดจนเพื่อนๆ รุ่นพี่ และน้องในภาควิชาวิศวกรรม คอมพิวเตอร์ ที่อยู่เบื้องหลังความสำเร็จที่ได้ให้ความช่วยเหลือ สนับสนุนและให้กำลังใจตลอดการทำงาน

ผู้จัดทำ

## สารบัญ

หน้า

<b>บทที่ 1 คำนำ</b>	1
1.1 ที่มาของปัจจุบันและแนวทางการแก้ปัจจุบัน	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของงานวิจัย	2
1.4 ขั้นตอนการทำงานและระยะเวลาการดำเนินงาน	3
<b>บทที่ 2 ที่มา ทฤษฎีและงานวิจัยที่เกี่ยวข้อง</b>	7
2.1 เครือข่ายเชื่อมต่อไร้สายแบบเมช (Wireless Mesh Network)	7
2.2 โปรโตคอล SMTP (Simple Mail Transfer Protocol)	8
2.3 Raspberry Pi โมเดล B	9
2.4 กล้องเว็บแคม (Webcam)	10
2.5 Wifi USB Adapter	10
2.6 ภาษา Python	10
2.7 OpenCV	11
2.8 ระบบจดจำใบหน้า (Face Recognition)	12
2.9 Dynamic DNS	13
2.10 Mysql Server	13
2.11 Dropbox	14
2.12 เซ็นเซอร์ตรวจจับการเคลื่อนไหว (PIR Sensor)	16
2.13 เซ็นเซอร์ใช้วัดอุณหภูมิและความชื้น (DHT11)	17
2.14 โมดูลตรวจแก๊สไวไฟ (MQ-2)	17
2.14 Xcode	18
2.15 ภาษา Objective C	18
2.16 ภาษา Swift	18
2.17 หัวข้อทางเทคนิคที่เกี่ยวกับผลิตภัณฑ์	19
<b>บทที่ 3 การออกแบบและประเมินประสิทธิภาพ</b>	20
3.1 อุปกรณ์และเครื่องมือที่ใช้ในการทำโครงการ	20

3.1.1 ฮาร์ดแวร์ (Hardware)	20
3.1.2 ซอฟต์แวร์ (Software)	20
3.1.3 ภาษาที่ใช้ในการเขียนโปรแกรม	20
3.2 โครงสร้างของระบบโดยรวม	21
3.2.1 ส่วนของโมดูลกล้อง (Camera Module)	21
3.2.2 ส่วนของโหนดรูนข้อมูล (Database Node)	21
3.2.3 ส่วนของโมบายแอปพลิเคชัน (Mobile Application)	21
3.3 การออกแบบและการดำเนินการทางด้านซอฟต์แวร์	22
3.3.1 การออกแบบและการดำเนินการทางด้านซอฟต์แวร์ในส่วนของโหนดรูน	22
3.3.2 การออกแบบและการดำเนินการทางด้านซอฟต์แวร์ในส่วนของโมบายแอปพลิเคชัน	28
3.4 การออกแบบและการดำเนินงานทางด้านฮาร์ดแวร์	31
3.5 การออกแบบและการดำเนินงานทางด้านเครื่อข่าย	33
3.6 การออกแบบและการดำเนินงานทางด้านระบบฐานข้อมูล	34
3.7 ผลิตภัณฑ์ที่จะพัฒนาเป็นตัวต้นแบบสำหรับโครงการประเภทผลิตภัณฑ์ทางการค้า	37
<b>บทที่ 4 ผลการดำเนินงาน</b>	38
4.1 การสร้างเครือข่ายของ Surveillance Camera	38
4.2 ระบบการทำงานหลักของโหนดรูนและโหนดรูนข้อมูล	41
4.3 ระบบจดจำใบหน้า (Face Recognition)	44
4.4 ระบบแจ้งเตือนทางอีเมล (Email Notification)	58
4.5 ระบบฐานข้อมูลบนโหนดรูนข้อมูลของ Raspberry Pi	58
4.6 ระบบการส่งข้อมูลไปยัง Dropbox	60
4.7 Dynamic DNS	62
4.8 โมบายแอปพลิเคชัน (Mobile Application)	64
4.8.1 เมนู Camera	64
4.8.2 เมนู Link To Dropbox	66
4.8.3 เมนู About	68

<b>บทที่ ๕ บทสรุป</b>	69
5.1 สรุปผลการดำเนินงาน	69
5.1.1 การสร้างเครือข่ายของ Surveillance Camera	69
5.1.2 ระบบการทำงานหลักของ โหนดกล้องและ โหนดฐานข้อมูล	69
5.1.3 ระบบจดจำใบหน้า (Face Recognition)	69
5.1.4 ระบบการแจ้งเตือนทางอีเมล (Email Notification)	70
5.1.5 ระบบฐานข้อมูลบน โหนดฐานข้อมูลของ Raspberry Pi	70
5.1.6 ระบบการส่งข้อมูลไปยัง Dropbox	70
5.1.7 Dynamic DNS	70
5.1.8 โมบายแอพลิเคชัน (Mobile Application)	70
5.2 ปัญหาที่พบและแนวทางการแก้ไข	71
5.3 ตารางแสดงสถานะของงานในปัจจุบัน	72
<b>การทำงานภายใต้อนาคต (Future Works)</b>	74
<b>บรรณานุกรม</b>	75

## รายการรูปประกอบ

รูปภาพ	หน้า
รูปภาพที่ 2.1 การทำงานของเครือข่ายเซ็นเซอร์ไร้สาย (Wireless Sensor Network)	7
รูปภาพที่ 2.2 การทำงานของพอร์ตโคลอ SMTP	8
รูปภาพที่ 2.3 Raspberry Pi โมเดล B	9
รูปภาพที่ 2.4 โครงสร้างบอร์ด Raspberry Pi ทั้ง 2 โมเดล	9
รูปภาพที่ 2.5 Wifi USB Adapter	10
รูปภาพที่ 2.6 ระบบจดจำใบหน้า (Face Recognition)	12
รูปภาพที่ 2.7 Dynamic DNS ภายใน Router	13
รูปภาพที่ 2.8 MySQL	13
รูปภาพที่ 2.9 Dropbox	14
รูปภาพที่ 2.10 การสร้าง Dropbox API เพื่อใช้เป็น Cloud Storage	15
รูปภาพที่ 2.11 การเชื่อมต่อ Raspberry Pi กับ Dropbox API เพื่อใช้เป็น Cloud Storage	15
รูปภาพที่ 2.12 เซ็นเซอร์ตรวจจับการเคลื่อนไหว (PIR Sensor)	16
รูปภาพที่ 2.13 การทำงานของเซ็นเซอร์ตรวจจับการเคลื่อนไหว	16
รูปภาพที่ 2.14 เซ็นเซอร์ใช้วัดอุณหภูมิและความชื้น (DHT11)	17
รูปภาพที่ 2.15 เซ็นเซอร์ตรวจแก๊สไวไฟ (MQ-2)	17
รูปภาพที่ 2.16 Xcode	18
รูปภาพที่ 3.1 Flowchart ของระบบการทำงานหลักของ โมดูลกล้อง	22
รูปภาพที่ 3.2 Flowchart ของฟังก์ชันการทำงานเมื่อเกิดการอินเตอร์รัพท์จาก PIR Sensor	23
รูปภาพที่ 3.3 Flowchart ของฟังก์ชันการทำงานเมื่อเกิดการอินเตอร์รัพท์จาก MQ-2 Sensor	23
รูปภาพที่ 3.4 Flowchart ของฟังก์ชันการทำงานเมื่อเกิดการอินเตอร์รัพท์จากปุ่มต่างๆ	24
รูปภาพที่ 3.5 : Flowchart ของฟังก์ชันการทำงานการเพิ่มใบหน้าลงฐานข้อมูลใบหน้า	25
รูปภาพที่ 3.6 : Flowchart การทำงานของระบบแปลงโครงสร้างข้อมูลของรูปภาพ	26
รูปภาพที่ 3.7 ภาพ Application Icon : Surveillance Camera	28
รูปภาพที่ 3.8 ภาพตัวอย่างหน้าต่างเมนูในส่วนของ โนมายแอพลิเคชั่น	28
รูปภาพที่ 3.9 รูปภาพโครงสร้างในส่วนของเมนู Camera	29
รูปภาพที่ 3.10 รูปภาพโครงสร้างในส่วนของเมนู Link To Dropbox	29
รูปภาพที่ 3.11 รูปภาพโครงสร้างในส่วนของเมนู About	30
รูปภาพที่ 3.12 วงจรเบื้องต้นของ โหนดกล้อง	31

รูปภาพที่ 3.13 ภาพตำแหน่ง Pin โดยคุณร้านจาก GPIO.BCM	31
รูปภาพที่ 3.14 โครงสร้างทางเครือข่ายเบื้องต้นของระบบกล้องวงจรปิด	33
รูปภาพที่ 3.15 โครงสร้างฐานข้อมูล	34
รูปภาพที่ 3.16 Sequence Diagram การทำงานระหว่าง โหนดกล้องและ โหนดฐานข้อมูล	36
รูปภาพที่ 3.17 Sequence Diagram การทำงานระหว่างสมาร์ทโฟน และ Dropbox	37
รูปภาพที่ 4.1 ภาพแสดงการเปิดใช้งานของพอร์ต wlan0	38
รูปภาพที่ 4.2 การแสดงเครือข่ายในเครือข่ายไร้สายแบบเมมซ	40
รูปภาพที่ 4.3 : แสดงการทำงาน OLSR Routing	41
รูปภาพที่ 4.4 แสดงการทำงานหลักของโหนดกล้องในขณะไม่มีอินเตอร์เฟซ	41
รูปภาพที่ 4.5 ภาพแสดงผลลัพธ์จากการตรวจสอบการเคลื่อนไหว	42
รูปภาพที่ 4.6 ภาพการต่อวงจรของโหนดกล้อง	43
รูปภาพที่ 4.7 ภาพแสดงผลการสำรองไฟล์ฐานข้อมูล	43
รูปภาพที่ 4.8 : ภาพของโหนดฐานข้อมูล	43
รูปภาพที่ 4.9 แสดงผลลัพธ์การทำงานของระบบจดจำใบหน้า	44
รูปภาพที่ 4.10 ภาพฐานข้อมูลใบหน้าที่นำมาใช้ทดสอบ พร้อมค่า label ที่ได้ทำการตั้งไว้	45
รูปภาพที่ 4.11 ภาพที่นำทดสอบการตรวจสอบใบหน้า	45
รูปภาพที่ 4.12 ผลการทดสอบใบหน้าด้วยใบหน้าเดียวกับฐานข้อมูลใบหน้าด้วย Eigenface Recognizer	45
รูปภาพที่ 4.13 ผลการทดสอบใบหน้าด้วยใบหน้าเดียวกับฐานข้อมูลใบหน้าด้วย Fisherface Recognizer	46
รูปภาพที่ 4.14 ผลการทดสอบใบหน้าด้วยใบหน้าเดียวกับฐานข้อมูลใบหน้าด้วย Local Binary Patterns Histograms Face Recognizer	46
รูปภาพที่ 4.15 ผลการทดสอบใบหน้าด้วยชุดใบหน้า A ด้วย Eigenface Recognizer	47
รูปภาพที่ 4.16 ผลการทดสอบใบหน้าด้วยชุดใบหน้า B ด้วย Eigenface Recognizer	47
รูปภาพที่ 4.17 ผลการทดสอบใบหน้าด้วยชุดใบหน้า A ด้วย Fisherface Recognizer	48
รูปภาพที่ 4.18 ผลการทดสอบใบหน้าด้วยชุดใบหน้า B ด้วย Fisherface Recognizer	48
รูปภาพที่ 4.19 ผลการทดสอบใบหน้าด้วยชุดใบหน้า A ด้วย Local Binary Patterns Histograms Face Recognizer	49
รูปภาพที่ 4.20 ผลการทดสอบใบหน้าด้วยชุดใบหน้า B ด้วย Local Binary Patterns Histograms Face Recognizer	49
รูปภาพที่ 4.21 ตัวอย่างภาพใบหน้าของผู้ที่ไม่ใช่สมาชิกที่นำมาทดสอบ ภาพที่ 1	53
รูปภาพที่ 4.22 ผลลัพธ์จากการตรวจสอบใบหน้าในภาพที่ 4.21	54
รูปภาพที่ 4.23 ตัวอย่างภาพใบหน้าของผู้ที่ไม่ใช่สมาชิกที่นำมาทดสอบ ภาพที่ 2	55

รูปภาพที่ 4.24 ผลลัพธ์จากการตรวจสอบใบหน้าในภาพที่ 4.23	55
รูปภาพที่ 4.25 ตัวอย่างภาพใบหน้าของผู้ที่ไม่ใช่สมาชิกที่นำมาทดสอบ ภาพที่ 3	56
รูปภาพที่ 4.26 ผลลัพธ์จากการตรวจสอบใบหน้าในภาพที่ 4.25	56
รูปภาพที่ 4.27 ตัวอย่างภาพใบหน้าของผู้ที่ไม่ใช่สมาชิกที่นำมาทดสอบ ภาพที่ 4	57
รูปภาพที่ 4.28 ผลลัพธ์จากการตรวจสอบใบหน้าในภาพที่ 4.27	57
รูปภาพที่ 4.29 ภาพแสดงการแจ้งเตือนจากระบบการแจ้งเตือนมายังอีเมลล์	58
รูปภาพที่ 4.30 ภาพแสดงตาราง Event Type ในฐานข้อมูล	59
รูปภาพที่ 4.31 ภาพแสดงตาราง Event Log ในฐานข้อมูล	59
รูปภาพที่ 4.32 ภาพแสดงตาราง Temperature Log ในฐานข้อมูล	60
รูปภาพที่ 4.33 ภาพแสดงข้อมูลใน Dropbox	60
รูปภาพที่ 4.34 ภาพแสดงข้อมูลใน Dropbox ไฟล์เดอร์ Database	60
รูปภาพที่ 4.35 ภาพแสดงข้อมูลใน Dropbox ไฟล์เดอร์ EventImage	61
รูปภาพที่ 4.36 ภาพแสดงข้อมูลใน Dropbox ไฟล์เดอร์ FaceDatabase	61
รูปภาพที่ 4.37 ภาพแสดงข้อมูลใน Dropbox ไฟล์เดอร์ TrainingData	61
รูปภาพที่ 4.38 : ภาพแสดง Hostname ที่สร้างไว้สำหรับ Dynamic DNS	62
รูปภาพที่ 4.39 : ภาพแสดงโปรแกรมอัพเดตไฟล์ให้สอดคล้องกับ Dynamic DNS	62
รูปภาพที่ 4.40 : ภาพแสดงการตั้งค่า Dynamic DNS ใน Router	62
รูปภาพที่ 4.41 : ภาพแสดงการตั้งค่า Port Forwarding	63
รูปภาพที่ 4.42 : ภาพแสดงผลลัพธ์จากการใช้งาน Dynamic DNS	63
รูปภาพที่ 4.43 ภาพแสดงผลลัพธ์การ Add Camera	64
รูปภาพที่ 4.44 ภาพแสดงผลลัพธ์การ Edit Camera	64
รูปภาพที่ 4.45 ภาพแสดงผลลัพธ์การ Delete Camera	65
รูปภาพที่ 4.46 ภาพแสดงผลลัพธ์การสตรีมmingเข้ากับกล้อง Webcam จาก Mobile Application	65
รูปภาพที่ 4.47 ภาพแสดงการเชื่อมต่อเข้ากับ Dropbox API	66
รูปภาพที่ 4.48 ภาพแสดงการค้นหาไฟล์ใน Browse Dropbox	67
รูปภาพที่ 4.49 ภาพแสดงการดาวน์โหลดไฟล์ใน Browse Dropbox	67
รูปภาพที่ 4.50 ภาพแสดงผลลัพธ์ของ View Dropbox	68
รูปภาพที่ 4.51 ภาพแสดงผลลัพธ์ของ About	68

## รายการตารางประกอบ

ตาราง	หน้า
ตารางที่ 1.1 : แผนการดำเนินงาน	5
ตารางที่ 3.1 : แสดงโครงสร้างไฟล์การตั้งค่าในโหมดการทำงาน	27
ตารางที่ 3.2 : ตารางการเชื่อมต่อ Pin เข้ากับ Raspberry Pi	32
ตารางที่ 3.3 : ตารางการทำงานในแต่ละส่วนของ Network	33
ตารางที่ 3.4 : ตารางแสดงข้อมูลในตาราง Node Member	34
ตารางที่ 3.5 : ตารางแสดงข้อมูลในตาราง Event Log	35
ตารางที่ 3.6 : ตารางแสดงข้อมูลในตาราง Event Type	35
ตารางที่ 3.7 : ตารางแสดงข้อมูลในตาราง Temp Humidity Log	36
ตารางที่ 4.1 : ผลลัพธ์การทดลองด้วย LHDB Face Recognition	50
ตารางที่ 5.1 : ปัญหาที่พบและแนวทางการแก้ไข	71
ตารางที่ 5.2 : สถานะงานในปัจจุบัน	72

## บทที่ 1 คำนำ

### 1.1 ที่มาของปัญหาและแนวทางการแก้ปัญหา

ทางกลุ่มนักศึกษาได้จัดทำโครงการ Home Automation: Surveillance Camera ขึ้นมาเพื่อเป็นเครื่องมือในการเฝ้าระวังความปลอดภัยให้กับทรัพย์สิน หรือพื้นที่ที่ต้องการรักษาความปลอดภัยโดยการดูภาพผ่านระบบเครือข่ายไร้สาย และยังเป็นเครื่องมือที่ใช้งานบนโทรศัพท์เคลื่อนที่สามารถท์ฟอนระบบปฏิบัติการ ไอโอเอส โดยสามารถรับชมภาพแบบ real-time ที่เกิดขึ้นจริง ทั้งนี้ยังสามารถทราบถึงเหตุการณ์ที่เกิดขึ้นผ่านทางแอพพลิเคชัน และอีเมลล์ เพื่อความสะดวกสบายของผู้ใช้งาน

เนื่องจากปัญหาทางด้านอาชญากรรมมีจำนวนมากขึ้นทุกปี การรักษาความปลอดภัยจึงเป็นสิ่งที่จำเป็นมาก ยิ่งขึ้นในสังคมปัจจุบัน ทางกลุ่มนักศึกษาที่เลือกหัวข้อในปัจจุบันนี้เทคโนโลยีได้ก้าวมาเป็นส่วนหนึ่งของชีวิตประจำวัน ไม่ว่าจะเป็นคอมพิวเตอร์ โทรศัพท์เคลื่อนที่ สมาร์ทโฟน รวมทั้งอินเทอร์เน็ต ไม่ว่าจะเป็นเทคโนโลยี 3G 4G ทำให้สามารถติดต่อ หรือ รับข้อมูลข่าวสารผ่านทางโลกออนไลน์ได้

Home Automation: Surveillance Camera เป็นโครงการประเภทผลิตภัณฑ์ทางการค้า ที่จัดทำขึ้นเพื่อเป็นเครื่องมือที่ใช้ในการเฝ้าระวัง และรักษาความปลอดภัยให้กับผู้ใช้งาน

### 1.2 วัตถุประสงค์

1. เพื่อศึกษาเรียนรู้เกี่ยวกับวิธีการเขียนโปรแกรมที่ใช้ในกระบวนการทำงานของฮาร์ดแวร์ที่เลือกใช้
2. เพื่อศึกษาเรียนรู้เกี่ยวกับการเขียนโปรแกรมสร้างแอพพลิเคชันบนโทรศัพท์เคลื่อนที่สามารถท์ฟอน
3. เพื่อศึกษาการ รับ- ส่ง ข้อมูลนรระบบเครือข่ายไร้สาย (WI-FI) ระหว่าง อุปกรณ์ และ Cloud Storage
4. เพื่อศึกษาการสร้างเครือข่ายขนาดเล็กหรือที่เรียกว่าเครือข่ายไร้สายแบบmesh (Wireless Mesh Network) รวมถึงการบริหารการจัดการต่างๆในเครือข่าย
5. เพื่อพัฒนาระบบเฝ้าระวังการรักษาความปลอดภัยจากอุปกรณ์โทรศัพท์เคลื่อนที่สามารถท์ฟอน
6. เพื่อสร้างสรรค์เทคโนโลยีทางด้านการรักษาความปลอดภัยให้มีความหลากหลายและเหมาะสมต่อการใช้งานยิ่งขึ้น

## 1.3 ขอบเขตของงานวิจัย

Home Automation : Surveillance Camera เป็นโครงงานที่ศึกษาเกี่ยวกับเทคโนโลยีการเฝ้าระวังความปลอดภัยโดยการใช้แอพพลิเคชั่นบนโทรศัพท์เคลื่อนที่สามารถติดต่อผ่านระบบปัญญาติดต่อไร้สาย เช่น ไอโฟน หรือแอนดรอยด์ รับชมภาพจากกล้องวงจรปิดผ่านระบบเครือข่ายไร้สายทั้งยังแจ้งเตือนผู้ใช้งานให้ทราบถึงเหตุการณ์ที่เกิดขึ้นโดยไม่จำเป็นที่จะต้องให้ผู้ใช้งานมอนิเตอร์เข้ามาตรวจสอบตลอดเวลา โดยแบ่งส่วนพัฒนาออกเป็น 2 ส่วน

### 1. ส่วนของกล้อง (Camera Node)

- ทำการเขียนโปรแกรมด้วยภาษา Python โดยส่วนการให้เซ็นเซอร์ต่างๆ ตรวจจับในรูปแบบของการรับอินพุตแบบอินเตอร์รัพท์
- เมื่อเซ็นเซอร์ตรวจจับความเคลื่อนไหวได้ จะทำการถ่ายภาพบันทึกไว้ใน เมมโมรี่ของตัวกล้อง จากนั้นจะทำการนำรูปนั้นมาประมวลผลตรวจจับใบหน้า (Face Recognition) ว่า เป็นผู้บุกรุกหรือไม่ ถ้าหากเป็นผู้บุกรุก ก็มีทำการแจ้งเตือนโดยการส่งอีเมล์ แจ้งเตือนพร้อมรูปภาพที่ถ่ายได้ไปยัง อีเมล์ของผู้ใช้งานระบบ หรือถ้าหากค่าที่รับมาอยู่ในส่วนของเซ็นเซอร์ใช้วัดอุณหภูมิและความชื้น ก็จะทำการตรวจสอบว่าค่าสูงเกิดกว่าอุณหภูมิที่ตั้งไว้หรือไม่ ถ้าสูงกว่าหรือต่ำกว่าที่กำหนด ก็จะทำการส่งการแจ้งเตือนผ่านทางอีเมล์
- สามารถเพิ่มกล้องที่จะทำงานได้เรื่อยๆ ตามลักษณะของเครือข่ายแบบmesh (Mesh Network) และขอบเขตความครอบคลุมของเครือข่ายจะกว้าง ไกลกว่าการใช้จุดเชื่อมต่อ (Access Point) ในการรองรับการเชื่อมต่อ เพียงจุดเดียว และโหนดที่อยู่ใกล้เราหรือที่สุดจะเป็นทางผ่านให้ส่งต่อแพ็กเกจ (Forward Packet) เชื่อมต่อไปยังอินเทอร์เน็ต
- เขียนโปรแกรมให้มีการเกิดเหตุการณ์แล้วจะทำการเก็บล็อก(Log)ข้อมูลลงฐานข้อมูลของโหนดฐานข้อมูล เมื่อเวลาผ่านไปช่วงระยะเวลาหนึ่ง ก็จะทำการรวมข้อมูลส่งไปเก็บยัง Dropbox เพื่อนำไปใช้งานต่อไป

### 2. ส่วนของโมบายแอพพลิเคชั่น (Mobile Application)

- เขียนโปรแกรมที่โมบายแอพพลิเคชั่นเพื่อทำการดาวน์โหลดข้อมูลที่บันทึกไว้ใน Dropbox และแสดงผลลัพธ์ที่อยู่ใน Dropbox ได้
- เขียนโปรแกรมที่โมบายแอพพลิเคชั่นให้สามารถรับข้อมูลที่มีมาดูภาพแบบ Real-Time ได้

## 1.4 ขั้นตอนการทำงานและระยะเวลาการดำเนินงาน

### การดำเนินงานโครงงาน

1. ศึกษา รวบรวมข้อมูลและทำการเลือกอุปกรณ์ที่จะใช้ทำ โปรเจค
2. ศึกษาการใช้งาน Raspberry Pi เป็นต้น
3. ศึกษาการสร้างโหนดกล้อง (Camera Node)
4. ศึกษาการใช้งาน Raspberry Pi เพื่อใช้ในกระบวนการส่งข้อมูล
5. ทำการสร้างเครื่องขยายภาพในบ้านสำหรับ Surveillance Camera ที่สามารถบันทึกข้อมูลไปยัง Dropbox
6. เริ่มทำการเขียนโปรแกรมการทำงานเกี่ยวกับการตรวจสอบความเคลื่อนไหวด้วย Raspberry Pi
7. เริ่มทำการเขียนโปรแกรมการทำงานเกี่ยวกับระบบจดจำใบหน้า
8. เริ่มทำการเขียนโปรแกรมการทำงานเกี่ยวกับระบบการแจ้งเตือนทางอีเมลล์และการเชื่อมต่อไปยัง Dropbox API
9. ทดสอบการทำงานของโหนดกล้อง (Camera Node) และระบบจดจำใบหน้าของ Raspberry Pi พร้อมทั้งแก้ไขปัญหา
10. ทดสอบการทำงานของ Raspberry Pi ให้สามารถแจ้งเตือนทางอีเมลล์
11. ระบบฐานข้อมูลและการเก็บล็อกข้อมูลบน Raspberry Pi
12. ศึกษาการใช้งานโปรแกรมและภาษาที่ใช้ในการเขียนโปรแกรมบนโ้มนายแอพลิเคชั่น
13. ศึกษาการเขียนแอพลิเคชั่น ให้ทำงานร่วมกับ Dropbox
14. พัฒนาแอพลิเคชั่นให้สามารถทำการแสดงผล(Query)ข้อมูลต่างๆจาก Dropbox
15. ส่วนของเมนู Camera (Add /Edit / Delete) , Live Streaming
16. ส่วนของเมนู Link To Dropbox (Browse /Download /View)
17. ส่วนของเมนู About ในโ้มนายแอพลิเคชั่น
18. ทดสอบการทำงานของโ้มนายแอพลิเคชั่นพร้อมทั้งแก้ไขปัญหา

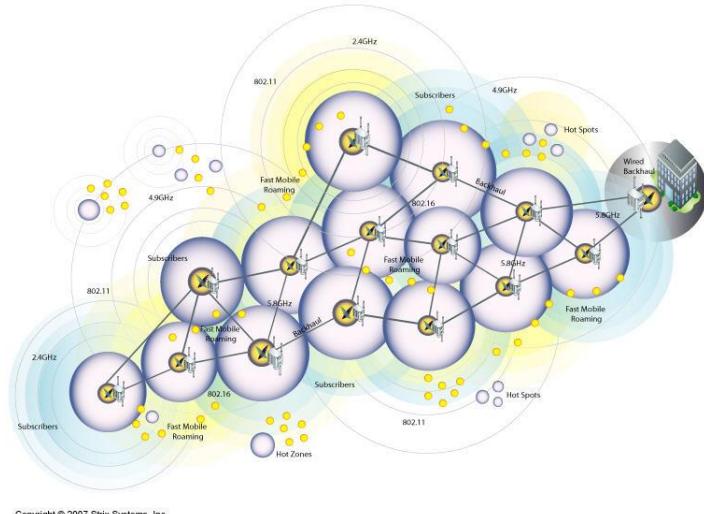
# แผนการดำเนินงาน



7.เริ่มทำการเขียนโปรแกรมการทำงานเกี่ยวกับระบบจดจำใบหน้า	14/10/2015	21/10/2015											
8.เริ่มทำการเขียนโปรแกรมการทำงานเกี่ยวกับระบบการแจ้งเตือนทางอีเมล์ และทำการเชื่อมต่อไปยัง Dropbox API	22/10/2015	25/10/2015											
9.ทดสอบการทำงานของโ INA หอนดก ล้อง และระบบจดจำใบหน้าของ Raspberry Pi พร้อมทั้งแก้ไขปัญหา	22/10/2015	31/10/2015											
10.ทดสอบการทำงานของ Raspberry Pi ให้สามารถ แจ้งเตือนทางอีเมล์	1/11/2015	12/11/2015											
11.ระบบฐานข้อมูล และการเก็บล็อกข้อมูลบน Raspberry Pi	7/11/2015	10/11/2015											
12.ศึกษาการใช้งานโปรแกรมและภาษาที่ใช้ในการเขียนโปรแกรมบนโ INA หมายแอพพลิเคชัน	10/12/2015	17/12/2015											
13.ศึกษาการเขียนแอพพลิเคชันให้ทำงานร่วมกับ Dropbox	20/12/2015	31/1/2015											
14.พัฒนาแอพพลิเคชันให้สามารถทำการแสดงผล(Query) ข้อมูลต่างๆจาก Dropbox	20/12/2015	25/2/2015											
15.ส่วนของเมนู Camera (Add /Edit / Delete) , Live Streaming	1/3/2016	14/4/2016											
16.ส่วนของเมนู Link To Dropbox (Browse /Download /View)	1/3/2016	14/4/2016											
17.ส่วนของเมนู About ในโ INA หมายแอพพลิเคชัน	1/3/2016	3/5/2016											
18.ทดสอบการทำงานของโ INA หมายแอพพลิเคชันพร้อมทั้งแก้ไขปัญหา	1/3/2016	3/5/2016											
<b>การรวมโครงงาน</b>		2/3/2016	3/5/2016										
1.ทดสอบการทำงานร่วมกันระหว่างชาร์ดแวร์ และ โ INA หมายแอพพลิเคชันพร้อมทั้งแก้ไขปัญหา	2/3/2016	3/5/2016											
2.จัดทำเอกสารคู่มือการใช้งาน	20/4/2016	30/4/2016											
3.จัดทำเอกสารรายงานโครงการ	1/5/2016	3/5/2016											
4.จัดทำเอกสารนำเสนอโครงการ	1/5/2016	3/5/2016											

## บทที่ 2 ที่มา ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

### 2.1 เครือข่ายเซ็นเซอร์ไร้สายแบบเมช ( Wireless Mesh Network )



รูปภาพที่ 2.1 การทำงานของเครือข่ายไร้สายแบบเมช (Wireless Mesh Network)

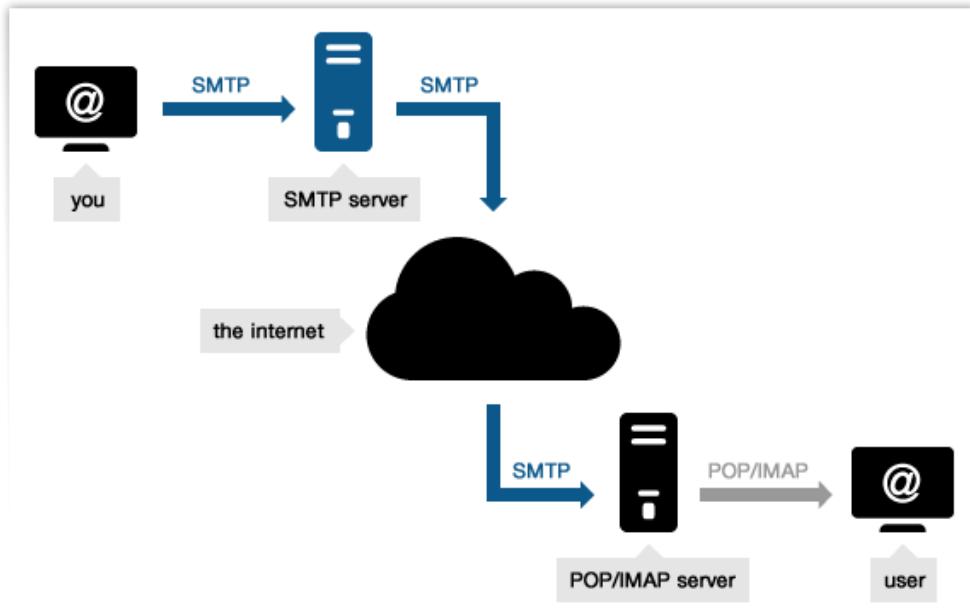
ที่มา: [<https://sites.google.com/site/kkubetaadhoc/2-khwam-sakhay-laea-thima-khxng-payha/prayukt/wireless-mesh-networks-wmn>]

**เครือข่ายเซ็นเซอร์ไร้สายแบบเมช (Wireless Mesh Network)** มีความสามารถในการส่งข้อมูลไปยัง โหนดต่างๆที่สามารถรับข้อมูลได้ และการทำงานในแต่ละ โหนดนั้นสามารถทำโดยอัตโนมัติ โดยจะขึ้นอยู่กับรูปแบบการเชื่อมต่อเครือข่ายหรือกีดีโอด้วยตัวเองที่เหมือนเราที่เดอร์ ที่มีการเชื่อมต่อด้วยสายแลน และมีคุณลักษณะอื่นๆอีกดังนี้

- 1.สามารถใช้พลังงานจากแบตเตอรี่หรือแหล่งไฟที่พลังงานอื่นๆ
- 2.มีความสามารถในการรับมือกับการทำงานผิดพลาดด้วยตัวเอง
- 3.โหนดแต่ละอันสามารถเคลื่อนย้ายได้ง่าย
- 4.มีโอกาสที่จะเกิดการสื่อสารที่ผิดพลาด
- 5.สามารถนำโหนดหลายแบบมาทำงานร่วมกันได้
- 6.รองรับการเพิ่มจำนวนโหนด ในการใช้งานที่จะมีในอนาคต
- 7.มีความสะดวกในการใช้งาน

Physical Layer และ MAC Layer มาตรฐานของ เครือข่ายเซ็นเซอร์ไร้สาย (Wireless Sensor Network) กีดีโอด้วยกิรรม IEEE 802.15.4 ด้วยมาตรฐานนี้ทำให้สามารถส่งข้อมูลโดยใช้พลังงานต่ำ ราคาถูกและในขอบเขตเครือข่ายที่จำกัด นอกจากนี้ยังทำงานบนมาตรฐานสถาปัตยกรรม IEEE 802.11 ได้อีกด้วย

## 2.2 โปรโตคอล SMTP (Simple Mail Transfer Protocol)



รูปภาพที่ 2.2 การทำงานของโปรโตคอล SMTP  
ที่มา :[<http://www.serversmtp.com/en/what-is-smtp>]

**โปรโตคอล SMTP (Simple Mail Transfer Protocol)** [1] เป็นโปรโตคอลที่คู่กับ POP3 เพราะเป็นโปรโตคอลที่ใช้ส่งอีเมลจาก User Agent ของผู้ส่งไปยัง MTA ของผู้ส่งและส่งต่อไปยัง MTA เครื่องอื่นๆที่เป็นจุดผ่านในการเชื่อมต่อไปยังเครื่องของผู้รับ โปรโตคอล STMP จะทำงานร่วมกับโปรโตคอลTCP โดยใช้พอร์ต25 ซึ่งกำลังต่างๆของSMTP จะเป็นลักษณะเดียวกับPOP3 คือเป็นASCII

เมื่อเริ่มต้นการติดต่อ SMTP จะกำหนดให้ User agent ของผู้ส่งต้องส่งคำสั่ง HELLO พร้อมกับรายละเอียดด้านผู้ส่งออกไป หากนี้จะส่งคำสั่ง MAIL เพื่อแจ้งให้เซิร์ฟเวอร์เตรียมรับอีเมล ในส่วนของเซิร์ฟเวอร์เมื่อพร้อมที่จะรับอีเมลก็จะตอบรับกลับมาด้วยคำสั่ง OK หากนี้ที่ด้านส่งก็จะเริ่มส่งโดยใช้คำสั่ง RCPT เพื่อกำหนดอีเมล์แต่ละฉบับที่จะส่งไป ซึ่งการส่งข้อมูลของอีเมลจะถูกระบุด้วยคำสั่ง DATA

เมื่อได้รับคำสั่งต่างๆของผู้ส่งแล้ว เซิร์ฟเวอร์จะมีหน้าที่ตรวจสอบความถูกต้องของคำสั่ง หากนี้จึงทำงานตามคำสั่งและส่งผลตอบกลับมา ส่วนลักษณะของข้อมูลที่ตอบกลับนั้นจะเป็นข้อมูลที่อยู่ในรูปของtext ที่เป็นASCII

ในการส่งอีเมลของโปรโตคอล SMTP นั้น จะใช้วิธีการอ้างถึงเซิร์ฟเวอร์อื่นๆตาม DNS(Domain Name System) เช่นเดียวกับระบบอื่นๆในอินเทอร์เน็ต และยังสามารถส่งอีเมลไปยังผู้รับคนเดียวหรือหลายคนพร้อมกันได้ด้วย

## 2.3 Raspberry Pi โมเดล B



รูปภาพที่ 2.3 Raspberry Pi โมเดล B

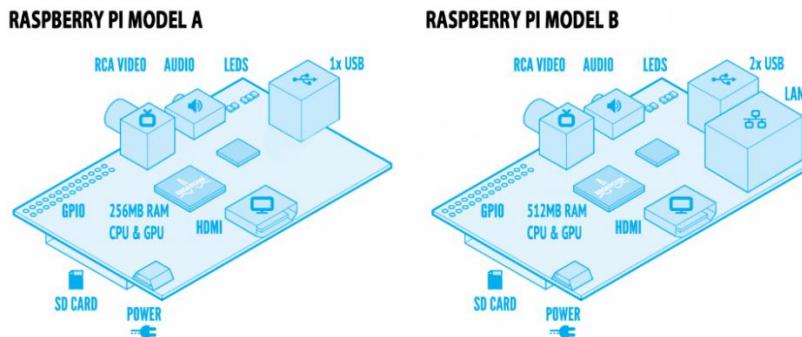
ที่มา : [<https://f.ptcdn.info/026/015/000/1390934978-pi11-o.jpg>]

**Raspberry Pi** [2] คือ บอร์ดคอมพิวเตอร์ขนาดเล็กที่สามารถเชื่อมต่อกับจอมอนิเตอร์ คีย์บอร์ด และเมาส์ ได้ สามารถนำมาประยุกต์ใช้ในการทำโครงการทางด้านอิเล็กทรอนิกส์ การเขียนโปรแกรม หรือเป็นเครื่องคอมพิวเตอร์ตั้งโต๊ะขนาดเล็ก ไม่ว่าจะเป็นการทำงานเอกสาร ท่องอินเทอร์เน็ต ส่งอีเมลล์ หรือเล่นเกมส์ อีกทั้งยังสามารถเล่นไฟล์วีดีโอความละเอียดสูง (High-Definition) ได้อีกด้วย

บอร์ด Raspberry Pi รองรับระบบปฏิบัติการลินุกซ์ (Linux Operating System) ได้หลายระบบ เช่น Raspbian (Debian), Pidora (Fedora) และ Arch Linux เป็นต้น โดยติดตั้งบน SD Card บอร์ด Raspberry Pi นี้ถูกออกแบบมาให้มี CPU GPU และ RAM อยู่ภายในชิปเดียวกัน มีจุดเชื่อมต่อ GPIO ให้ผู้ใช้สามารถนำไปใช้ร่วมกับอุปกรณ์อิเล็กทรอนิกส์อื่นๆ ได้อีกด้วย

### คุณสมบัติทางเทคนิคของบอร์ด

บอร์ด Raspberry Pi ปัจจุบันมีด้วยกัน 2 โมเดล คือ โมเดล A และ โมเดล B ซึ่งทั้ง 2 โมเดลมีคุณสมบัติทางเทคนิคที่ใกล้เคียงกัน แตกต่างกันเพียงบางส่วน รายละเอียดดังภาพที่ 2.4



รูปภาพที่ 2.4 โครงสร้างบอร์ด Raspberry Pi ทั้ง 2 โมเดล

ที่มา: [<http://www.hackthings.com/raspberry-pi-model-a-and-b/>]

## 2.4 กล้องเว็บแคม (Webcam)

กล้องเว็บแคม (Webcam) เป็นอุปกรณ์อินพุตที่สามารถจับภาพเคลื่อนไหวของเราไปปรากฏในหน้าจอคอมพิวเตอร์ และสามารถส่งภาพเคลื่อนไหวนี้ผ่านระบบเครือข่ายไปแสดงผลที่เครื่องอื่นได้ซึ่งเราจะเคลื่อนไหวได้เหมือนอยู่ต่อหน้า ถือว่าเป็นอุปกรณ์ที่มีประโยชน์ในการบันทึกภาพวิดีโอ รวมถึงใช้ในการติดต่อสื่อสารโดยถ้าจะใช้กล้องเว็บแคมคู่กับ Raspberry Pi จะเป็นที่ต้องใช้ กล้องเว็บแคม ตามที่ Raspberry Pi สนับสนุน

โดยในส่วนของโอนคอกล้องในโครงงานนี้ เราได้นำกล้องเว็บแคมมาใช้เป็นเครื่องมือในการตรวจจับภาพ การเคลื่อนไหวก่อนที่จะส่งข้อมูลรูปภาพไปทำการประมวลผลด้วยระบบตรวจสอบใบหน้า(Face Recognition) รวมถึงการแสดงผลภาพที่เกิดขึ้น ณ เวลาต่างๆ ผ่านทางแอปพลิเคชัน

## 2.5 Wifi USB Adapter

เป็น USB ที่ใช้ในการเชื่อมต่อเครือข่ายสามารถนำมาสร้าง โครงสร้างแบบเมッシュ (Mesh Network) , Access Point, Wireless Sensor Network ฯลฯ ให้สามารถทำการติดต่อสื่อสารกันระหว่างโอนคได้



รูปภาพที่ 2.5 WiFi USB Adapter

ที่มา: [[http://ecx.images-amazon.com/images/I/71P8Pfd7npL.\\_SX425\\_.jpg](http://ecx.images-amazon.com/images/I/71P8Pfd7npL._SX425_.jpg)]

โดยจะนำมาใช้ในการสร้างโครงข่าย (Network) ให้โอนคที่ทำหน้าที่ในการเก็บและประมวลผลภาพ สามารถสื่อสาร ส่งข้อมูลมายังจุดเชื่อมต่อเครือข่าย (Access Point) ได้ และใช้ในการสร้างจุดเชื่อมต่อเครือข่าย (Access Point) ให้สามารถติดต่อไปยังอินเทอร์เน็ต ในทางกลับกันก็เป็นส่วนที่ทำให้ผู้ใช้งานสามารถเข้าถึง Raspberry Pi จากภายนอกได้ผ่านทางแอปพลิเคชัน

## 2.6 ภาษา Python

ภาษา Python[3] เป็นภาษาระดับสูงภาษาหนึ่ง ที่มีความสามารถสูงถูกสร้างขึ้นในปี 1989 โดย Guido van Rossum ซึ่งถูกพัฒนาขึ้นมาโดยไม่ขัดติดกับแพลตฟอร์ม กล่าวคือสามารถใช้ภาษา Python ได้ทั้งบนระบบ Unix, Linux, Windows NT, Windows 2000, Windows XP หรือแม้แต่ระบบ FreeBSD ภาษาตัวนี้เป็นภาษาลักษณะ Open Source เหมือนภาษา PHP

### ความสามารถภาษา Python

ในปัจจุบันภาษาที่ใช้ในการพัฒนาเว็บแอ�플ิเคชั่น มีมากหลายภาษา อาทิเช่น ภาษา Perl, PHP, JAVA, ASP, Tcl, Python เป็นต้น สำหรับภาษา Python นับว่ายังใหม่ในการพัฒนาโปรแกรมบนเว็บ แต่ด้วยข้อดีหลายประการของภาษา Python ทำให้มีผู้นิยมใช้มากขึ้นเรื่อยๆ ซึ่งพอสรุปข้อดีของภาษา Python ได้ดังนี้

1. ง่ายต่อการเรียนรู้ โดยภาษา Python มีโครงสร้างของภาษาไม่ซับซ้อนเข้าใจง่าย ซึ่งโครงสร้างภาษา Python จะคล้ายกับภาษา C เพราะภาษา Python สร้างขึ้นมาโดยใช้ภาษา C ทำให้ผู้ที่คุ้นเคยภาษา C สามารถใช้งานภาษา Python ได้ นอกจากนี้โดยตัวภาษาเองมีความยืดหยุ่นสูงทำให้การจัดการกับงานด้านข้อความ และไฟล์ Text
2. ไม่ต้องเสียค่าใช้จ่ายใดๆ ทั้งสิ้น เพราะตัวแปรภาษา Python อยู่ภายใต้ลิขสิทธิ์ GNU
3. ใช้ได้หลากหลายแพลตฟอร์ม ในช่วงแรกภาษา Python ถูกออกแบบใช้งานกับระบบ Unix แต่ในปัจจุบันได้มีการพัฒนาตัวแปลงภาษา Python ให้สามารถใช้กับระบบปฏิบัติการอื่นๆ อาทิเช่น Linux, Windows 95/98/ME, Windows NT, Windows 2000, OS/2
4. ภาษา Python ถูกสร้างขึ้นโดยได้รวมรวมเอาส่วนดีของภาษาต่างๆ เข้ามาไว้ด้วยกัน อาทิเช่น ภาษา C, C++, Java, Perl
5. ภาษา Python เป็นภาษาประเภท Server Side Script คือการทำงานของภาษา Python จะทำงานด้านผู้ Server แล้วส่งผลลัพธ์กลับมายัง Client ทำให้มีความปลอดภัยสูง
6. ใช้พัฒนาเว็บเซอร์วิส (Web Service) โดยที่ภาษา Python สามารถนำมาพัฒนาเว็บเซอร์วิส รวมทั้งใช้บริหารการสร้างเว็บไซต์สำเร็จรูปที่เรียกว่า Content Management Framework (CMF) ตัวอย่าง CMF ที่มีชื่อเสียงมากและเมื่องหลังทำงานด้วย python คือ plone

## 2.7 OpenCV

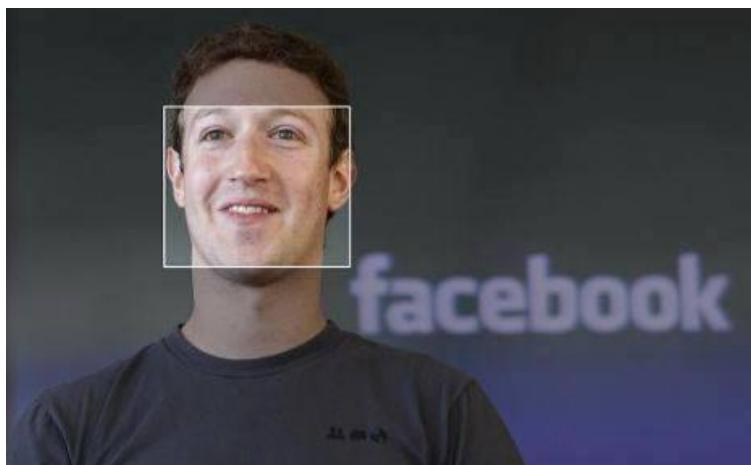
**OpenCV** [4] ย่อมาจาก Open source Computer Vision โดย library เหล่านี้ถูกเขียนด้วยภาษา C/C++ และสามารถใช้ภาษาได้ระบบปฏิบัติการ Linux, Windows และ Mac OS X จุดประสงค์หลักของ OpenCV คือการนำมาพัฒนาโปรแกรมที่เน้น การคำนวณแบบ Real-Time เป้าหมายหลัก ๆ ของ OpenCV ก็เพื่อเตรียมคำสั่งพื้นฐาน ที่ง่ายต่อการใช้งานทางด้านการประมวลผลภาพ (Image Processing) ทางด้าน Computer Vision และทางด้าน Machine Learning โดยจะเน้นไปที่ statistical pattern recognition และ clustering จำพวก K-Mean Clustering ทำให้เราสามารถพัฒนา โปรแกรมและต่อยอดขึ้นไปอีก ได้อย่างรวดเร็ว เนื่องจาก OpenCV มี library มากรถึง 500 พั้งก์ชั่น ตัวอย่างแอพลิเคชั่น ที่นำมาใช้ เช่น การประมวลผลเกี่ยวกับภาพและสัญญาณ (Image and Signal Processing), การตรวจสอบลักษณะ วัตถุจากภาพหรือวิดีโอ(Object Identification) การจดจำใบหน้า(Face Recognition) ม่านตา (iris Recognition) ตรวจสอบขอบหรือด้านของวัตถุ(Edge Detection) ตรวจสอบความเคลื่อนไหว(Motion Detection) เป็นต้น

### โครงสร้างของ OpenCV

- CV ประกอบไปด้วยคำสั่งประมวลผลภาพพื้นฐานและระดับสูง

- MLL ประกอบด้วยคำสั่งเกี่ยวกับ machine learning library และการคำนวณทางสถิติและซึ่งประกอบไปด้วย statistical classifiers และ clustering tools.
- HighGUI เป็น library ส่วนหนึ่งของ OpenCV ที่ใช้สำหรับติดต่อกับระบบปฏิบัติการ ระบบไฟล์ และอุปกรณ์ฮาร์ดแวร์ต่างๆ ฟังก์ชันของ HighGUI เช่น ใช้สำหรับการเปิดหน้าต่างแสดงผล ใช้แสดงภาพที่ใช้สำหรับการอ่านเขียนไฟล์กราฟฟิกใช้สำหรับควบคุมการทำงานของมาส์และคีย์บอร์ด
- CXCore เป็นชุดคำสั่งพื้นฐานโครงสร้างข้อมูล

## 2.8 ระบบจดจำใบหน้า (Face Recognition)



รูปภาพที่ 2.6 ระบบจดจำใบหน้า

ที่มา : [www.thai-fb.com]

**ระบบจดจำใบหน้า (Face Recognition) [5]** เป็นระบบยืนยันตัวบุคคลจากภาพถ่ายหรือวิดีโอ โดยใช้คุณลักษณะจำเพาะของตัวร่าง (Biometric) บนใบหน้า เปรียบเทียบกับใบหน้าในฐานข้อมูล

ระบบนี้ได้รับการพัฒนาอย่างต่อเนื่องกว่าสิบปี จนกระทั่งกระบวนการทำงานของระบบ หรืออัลกอริทึม (Algorithm) มีหลายรูปแบบตามการใช้งาน ทั้งด้านความบันเทิงอย่างระบบจดจำใบหน้าในเฟชบุ๊ก หรือในสมาร์ทโฟน และด้านความปลอดภัย ซึ่งประเทคโนโลยี ติดตั้งระบบจดจำใบหน้าไว้ตามสถานที่สำคัญ เช่น สนามบิน เพื่อรับมือกับผู้ก่อการร้าย หรือติดตามคนร้ายที่หลบหนี

**ระบบ Face Recognition จะแบ่งออกเป็น 2 ขั้นตอนหลักๆ คือ**

### A. การเตรียมความพร้อม (Training)

คือการต้องนำภาพใบหน้าที่ผ่านกระบวนการตรวจใบหน้า (Face Detection) แล้วมาทำการปรับแต่ง แสงเงา ฯลฯ เพื่อให้อัลกอริทึมของเราสามารถดึงคุณสมบัติของใบหน้าที่สำคัญๆ ออกมาให้ได้มากที่สุด ทำการลดขนาดของภาพ และสูตรที่ใช้ทำการจัดประเภท ดึงลักษณะเฉพาะของใบหน้าที่สำคัญๆ ออกมายังฐานข้อมูล

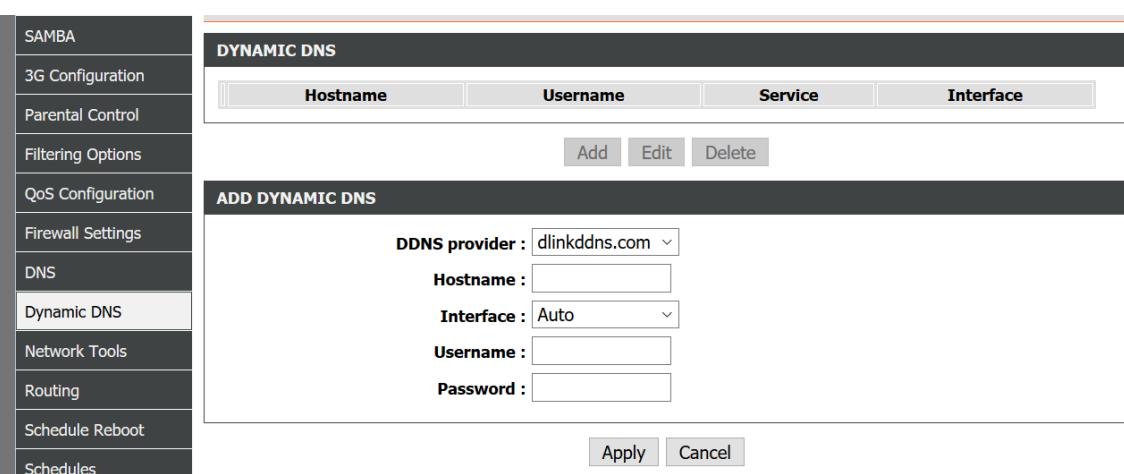
### B. การยืนยันตัวตน (Identify)

ทำเหมือนกับขั้นตอนเตรียมความพร้อมข้อมูล (Training) แต่จะต่างกันตรงที่นำมาใช้ตรวจสอบเพื่อเปรียบเทียบกับอีกใบหน้าหนึ่ง ไม่ได้เก็บลงฐานข้อมูล

## 2.9 Dynamic DNS

เนื่องจากไอพีส่วนใหญ่ที่เราได้รับเพื่อใช้เข้าอินเทอร์เน็ต มักจะเป็นไอพีแบบ Dynamic ที่ทาง ISP ได้ทำการมอบให้ ถ้าหากเราต้องการเข้ามายังระบบเครือข่ายภายในบ้านของเรารากภายนอก โดยไม่ได้ไปฝากไฟล์ที่เว็บไซต์ เราจะต้องมีหมายเลขไอพีหรืออะไรก็ตามที่บ่งบอกถึงความเป็นตัวเรา โดยจะต้องเป็นแบบ Static และเนื่องจากเราไม่สามารถจัดการไอพีที่ทาง ISP มอบหมายให้เราได้ดังนั้น Dynamic DNS จึงเข้ามามีบทบาทในส่วนนี้

Dynamic DNS เป็นเทคโนโลยีที่ทำให้เราสามารถเชื่อมต่อหมายเลขเครือข่ายภายในบ้านได้โดยไม่ต้องใช้ไอพี Address แต่จำเป็นต้องสร้างบัญชีใช้งานไปยังเว็บที่ให้บริการ Dynamic DNS โดยเราจะต้องทำการสร้างโฉสเนมที่บ่งบอกถึงคอมพิวเตอร์ของเรา จากนั้นเมื่อเราต้องการเข้าถึงระบบเครือข่ายภายในบ้านของเรารากภายนอก เราจะต้องใช้โฉสเนมด้วยนั้นๆ พื้นที่เราต้องค่าการให้บริการจากคอมพิวเตอร์ในระบบเครือข่ายภายในบ้านของเรา เพื่อเข้าถึงบริการที่ต้องการ และโดยที่สำคัญคือ เราที่ต้องรักษาในระบบเครือข่ายภายในบ้านของเราต้องสนับสนุนการใช้งาน Dynamic DNS



รูปภาพที่ 2.7 : Dynamic DNS ภายในเราที่เดอร์

## 2.10 MySQL Server



รูปภาพที่ 2.8 MySQL

ที่มา : [http://siam500400009.blogspot.com/2011/02/mysql.html]

**MySQL [7]** เป็นระบบจัดการฐานข้อมูลที่มีหน้าที่เก็บข้อมูล เพื่อใช้ในการเก็บข้อมูลที่ได้จากการงานที่โปรแกรมเมอร์ได้สร้างขึ้น โดยใช้ภาษา SQL โดย MySQL Server จะทำหน้าที่เป็นตัวกลางระหว่างผู้ใช้งานฐานข้อมูล ซึ่ง MySQL เป็นระบบจัดการฐานข้อมูล (database management system DBMS) สำหรับฐานข้อมูลเชิงสัมพันธ์ โดยความสามารถติดต่อกัน MySQL โดยการเขียนโปรแกรมภาษาต่างๆ ได้ เช่น PHP, Perl, Java, C#, C, Ruby, C++ เป็นต้น

โดยในโครงงานนี้เราได้สร้างฐานข้อมูลขึ้นมาเพื่อใช้ในการเก็บข้อมูล รูปภาพ ของผู้ใช้งานต่างๆ เพื่อมาใช้ในการประมวลผลและตัดสินใจจากระบบตรวจสอบใบหน้า(Face Recognition) รวมถึงระบบการแจ้งเตือน(Notification)

## 2.11 Dropbox

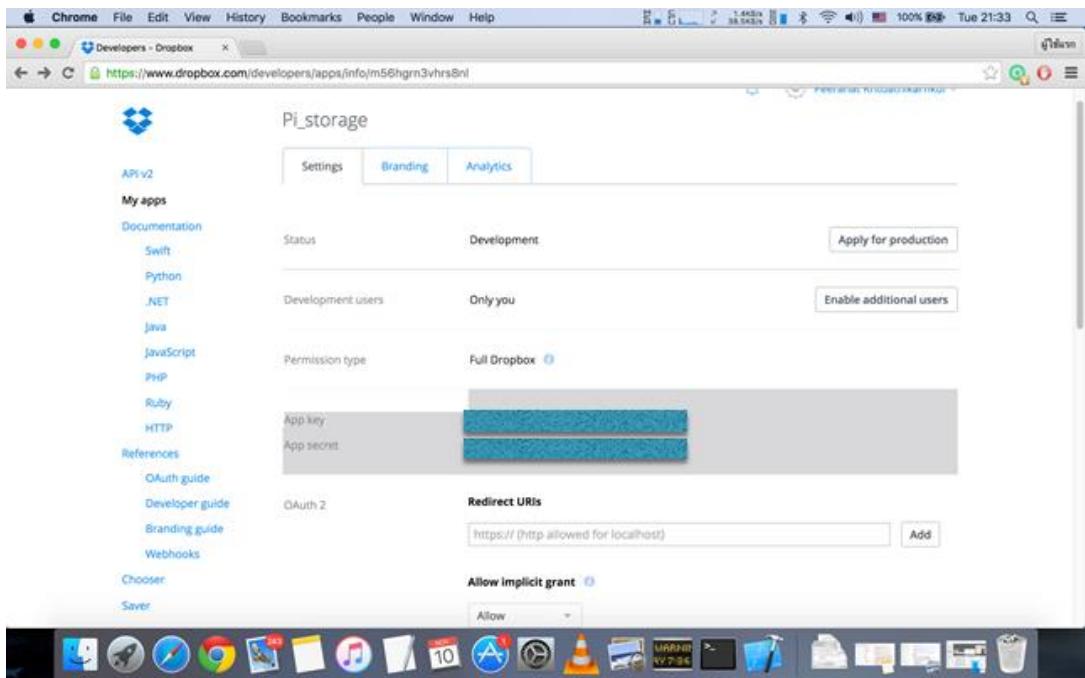


รูปภาพที่ 2.9 Dropbox

ที่มา [<http://edp.edupol.org/genOrganize/publicDoc/ruleForm/manual/computer/dropbox.pdf>]

**Dropbox** เป็น Cloud Data Storage ที่จะเก็บข้อมูลต่างๆอยู่บนเซิฟเวอร์ โดย Dropbox จะเป็นเครื่องมือที่ทำให้เราสามารถเรียกใช้ไฟล์งานต่างๆ ที่เราเก็บไว้บนเซิฟเวอร์ ได้จากทุกที่ ทุกเวลา โดยสามารถที่จะใช้งาน Dropbox ผ่านทาง คอมพิวเตอร์ โน๊ตบุ๊ค คอมพิวเตอร์ส่วนตัวหรือโทรศัพท์มือถือสมาร์ทโฟน โดยเราสามารถที่จะเพิ่ม แก้ไข หรือลบไฟล์งานต่างๆของเราที่อยู่บนเซิฟเวอร์ ได้ ทั้งนี้ยังมีความสามารถการแชร์ เพื่อสามารถแชร์ข้อมูล ไฟล์งาน เพื่อให้บุคคลอื่นสามารถเรียกดู หรือ ดาวน์โหลดมาใช้ได้อีกด้วยเรียกได้ว่าเป็นแหล่งเก็บข้อมูลที่สามารถเรียกใช้งานได้ทุกที่ ทุกเวลา

โดยการที่จะให้ Raspberry Pi สามารถใช้งาน Cloud Storage หรือ Dropbox ได้นั้นจำเป็นที่จะต้องสร้าง Dropbox API เพื่อที่จะนำรหัส App Key และ App Secret มาใช้เชื่อมต่อระหว่างแอปพลิเคชัน กับ Dropbox



รูปภาพที่ 2.10 การสร้าง Dropbox API เพื่อใช้เป็น Cloud Storage

```
FinaloP — pi@raspberrypi: ~/Dropbox-Uploader — ssh pi@192.168.2.2 — 125x30
pi@raspberrypi ~ $ cd Dropbox-Uploader/
pi@raspberrypi ~/Dropbox-Uploader $ ls
CHANGELOG.md LICENSE README.md dropShell.sh dropbox_uploader.sh
pi@raspberrypi ~/Dropbox-Uploader $ ./drop
dropbox_uploader.sh dropShell.sh
(pi@raspberrypi ~/Dropbox-Uploader $ ./dropbox_uploader.sh
Development

This is the first time you run this script.

1) Open the following URL in your Browser, and log in using your account: https://www.dropbox.com/developers/apps
2) Click on "Create App", then select "Dropbox API app"
3) Now go on with the configuration, choosing the app permissions and access restrictions to your DropBox folder
4) Enter the "App Name" that you prefer (e.g. MyUploader2768422947351)

Now, click on the "Create App" button.

When your new App is successfully created, please type the
App Key, App Secret and the Permission type shown in the confirmation page:

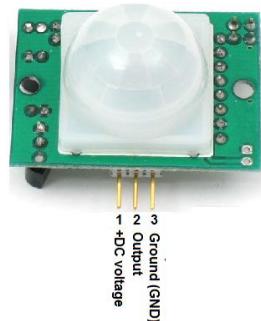
# App key: [REDACTED] xxkbluismfvqyy9
# App secret: [REDACTED] m56hgrn3vhrs8nl

Permission type:
  App Folder [a]: If you choose that the app only needs access to files it creates
  Full Dropbox [f]: If you choose that the app needs access to files already on Dropbox
# Permission type [a/f]: f
Redirect URIs

> App key is m56hgrn3vhrs8nl, App secret is xxkbluismfvqyy9 and Access level is Full Dropbox. Looks ok? [y/n]: y
```

รูปภาพที่ 2.11 การเชื่อมต่อ Raspberry Pi กับ Dropbox API เพื่อใช้เป็น Cloud Storage

## 2.12 เซ็นเซอร์ตรวจจับการเคลื่อนไหว (PIR Sensor)

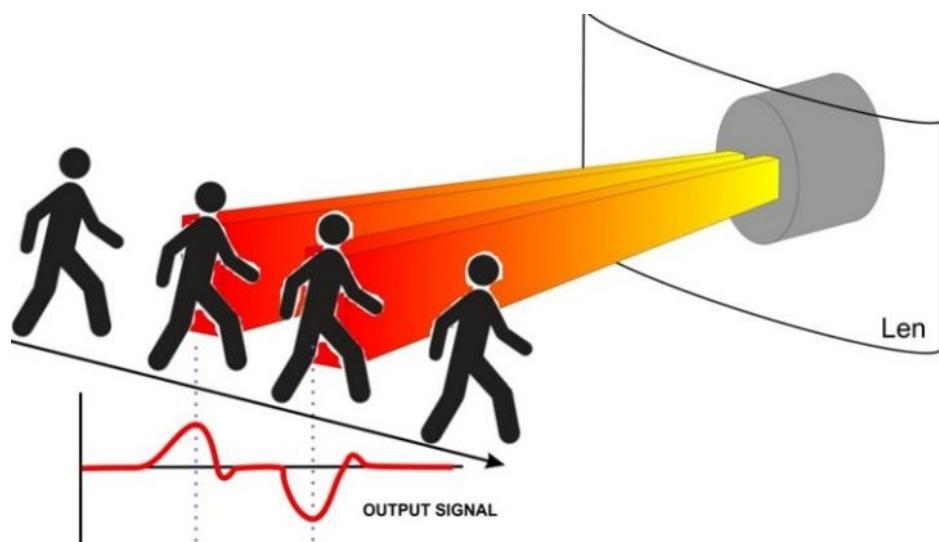


รูปภาพที่ 2.12 เซ็นเซอร์ตรวจจับการเคลื่อนไหว PIR

ที่มา :[<http://www.learningaboutelectronics.com/Articles/Motion-sensor-light-circuit.php>]

เซ็นเซอร์ตรวจจับการเคลื่อนไหว(PIR Sensor) [9] เป็นเซ็นเซอร์ชนิดหนึ่งที่ใช้ตรวจจับคลื่นรังสีอินฟราเรดที่เพริ่งจากมนุษย์ หรือ สัตว์ ที่มีการเคลื่อนไหว ทำให้มีการนำเอาเซ็นเซอร์ตรวจจับการเคลื่อนไหวมาประยุกต์ใช้งานกันเป็นอย่างมากใช้เพื่อตรวจจับการเคลื่อนไหวของสิ่งมีชีวิต หรือ ตรวจจับการบุกรุกในงานรักษาความปลอดภัย

### การทำงานของเซ็นเซอร์ตรวจจับการเคลื่อนไหว(PIR Sensor)



รูปภาพที่ 2.13 การทำงานของเซ็นเซอร์ตรวจจับการเคลื่อนไหว PIR

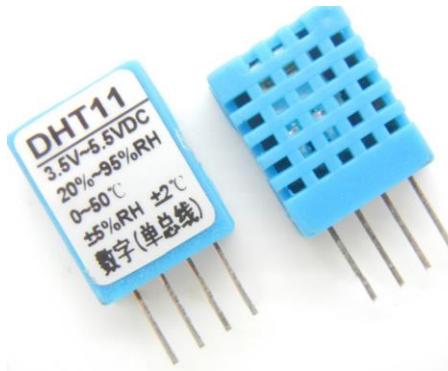
ที่มา [<http://www.thaieasyelec.com/article-wiki/review-product-article/pir-motion-sensor-getting-started.html>]

ภายใน เซ็นเซอร์ตรวจจับการเคลื่อนไหว PIR จะมีอุปกรณ์ตรวจจับรังสีอินฟราเรด อยู่ 2 ชุดด้วยกันดังภาพที่ 2.13 เมื่อมี คน หรือ สัตว์ ที่มีความอบอุ่นในร่างกายเคลื่อนที่ผ่านเข้ามาใน พื้นที่โซนที่เซ็นเซอร์ตรวจจับการเคลื่อนไหวสามารถตรวจจับคลื่นรังสีอินฟราเรดที่เพริ่งออกมาจากสิ่งมีชีวิตได้ เซ็นเซอร์ตรวจจับการเคลื่อนไหวจะเปลี่ยนคลื่นรังสีอินฟราเรดให้กลายเป็นกระแสไฟฟ้า จะเห็นว่าเมื่อมีสิ่งมีชีวิต เคลื่อนที่ผ่าน อุปกรณ์ตรวจจับรังสี

อินฟราเรดตัวที่ 1 จะได้สัญญาณเอาท์พุต ออกมานสูงกว่าแรงดันปกติ และ เมื่อสิ่งมีชีวิตเคลื่อนที่ผ่าน อุปกรณ์ ตรวจจับรังสีอินฟราเรดตัวที่ 2 จะได้แรงดัน เอาท์พุต ต่ำกว่าค่าแรงดันปกติ

โดยในโครงงานนี้เราจะใช้ เช็นเซอร์ตรวจจับการเคลื่อนไหวในการตรวจจับการเคลื่อนไหวที่เกิดขึ้นจาก กลุ่มรังสีอินฟราเรดที่แพร่จาก มุนย์ หรือ สัตว์ โดยจะส่งโลจิกไปเรียกใช้งานกล้องเว็บแคมให้เริ่มการทำงาน

## 2.13 เช็นเซอร์ใช้วัดอุณหภูมิและความชื้น DHT11



รูปภาพที่ 2.14 เช็นเซอร์ใช้วัดอุณหภูมิและความชื้น DHT11

ที่มา [[http://ci.lnwfile.com/\\_ci/\\_raw/my/22/hs.jpg](http://ci.lnwfile.com/_ci/_raw/my/22/hs.jpg)]

**เช็นเซอร์ใช้วัดอุณหภูมิและความชื้น (Temperature and Humidity Sensor)[9]** เป็นเช็นเซอร์ที่เอาไว้ใช้วัดหรือตรวจจับอุณหภูมิความชื้น เพื่อให้สามารถที่จะวัดค่า หรือใช้เพื่อควบคุมอุปกรณ์อื่นๆ ให้ทำงานควบคู่ไปด้วยกันจากปัจจัยของอุณหภูมิความชื้น โดยเช็นเซอร์ตรวจจับอุณหภูมิและความชื้นรุ่น DHT11 มีขาอยู่ 4 Pin โดย Pin1 จะเชื่อมต่อกับไฟฟ้ากระแสตรง 3.5-5 V Pin2 เชื่อมต่อกับไมโครคอนโทรลเลอร์ยูนิต(MCU) Pin3 :Null และ Pin4 จะเชื่อมต่อกับสายดิน (Ground)

## 2.14 โมดูลตรวจแก๊สไวไฟ (MQ-2)



รูปภาพที่ 2.15 เช็นเซอร์ตรวจแก๊สไวไฟ MQ-2

ที่มา [<http://img.priceza.com/img/product/314/314-20160513160111-12369424644517969.jpg>]

**โมดูลตรวจแก๊สไวไฟ (MQ-2)[10]** เป็นโมดูลตรวจวัดแก๊ส ที่ไวต่อแก๊สไวไฟในกลุ่ม LPG, i-butane, propane, methane ,alcohol, Hydrogen รวมไปถึงควันไฟที่เกิดจากการเผาไหม้ด้วย จึงเป็นเช็นเซอร์ที่นิยมนำมาใช้ในการตรวจจับการรั่วของแก๊สต่างๆ เพื่อป้องกันอันตรายที่จะเกิดขึ้นจากการรั่วไหลน้ำได้ ใช้แรงดัน 5V ในการ

ทำงาน ให้อาทพุตทั้งสัญญาณอนาล็อกซึ่งเป็นค่าที่วัดได้จริง และสัญญาณดิจิตอลสามารถปรับตั้งระดับแจ้งเตือน ได้ (ใช้ LM393 เป็นวงจรเบรียบเทียบแรงดัน)

## 2.15 โปรแกรม Xcode



รูปภาพที่ 2.16 Xcode

ที่มา [<http://a5.mzstatic.com/us/r30/Purple5/v4/10/e3/8b/10e38bdc-a1da-5686-0180-d7629dc34590/icon128-2x.png>]

เป็นโปรแกรมที่ใช้พัฒนาแอปพลิเคชันทั้งในระบบ iOS และ Mac OS มีคอมไพล์เลอร์ที่มีประสิทธิภาพสูง ซึ่งชี้ว่า LLVM Compiler 2.0 สามารถคอมไпал์โค้ดได้เร็วกว่า GCC ทำให้สามารถทำงานได้เร็วกว่า มีเครื่องมือที่ใช้ตรวจสอบประสิทธิภาพของแอปพลิเคชันที่พัฒนาเสร็จแล้ว โดยวัดการใช้งาน memory, CPU Time, Overhead ต่างๆ

## 2.16 ภาษา Objective C

เป็นภาษาที่มีคุณสมบัติและรูปแบบของภาษาซีทุกประการ ทำให้เข้าใจง่าย สิ่งที่เพิ่มเติมขึ้นมาคือการมีคุณสมบัติการเป็นออบเจ็ค ทำให้มีคุณสมบัติการสืบทอด การ Encapsulation และอื่นๆ เป็นภาษาที่ใช้พัฒนาแอปพลิเคชันบนระบบ iOS และ Mac OS โดยเฉพาะ

## 2.17 ภาษา Swift

เป็นภาษาโอเพ่นซอร์สที่มีการทำงานที่รวดเร็ว ยิ่งกว่า Objective C และ Python เหมาะสำหรับแอปพลิเคชันที่ต้องการให้มีการทำงานแบบเรียลไทม์ ทำงานร่วมกับภาษาอื่นๆ ได้ ทำให้แอปพลิเคชันที่สร้างมาทำงานได้อย่างไม่สตดด เป็นภาษาที่ทางบริษัท Apple ได้ออกแบบมาเพื่อใช้ในการสร้างแอปพลิเคชันบน iOS และ Mac OS นอกจากนี้ตัวภาษายังมีความปลอดภัยในการเขียนโปรแกรมมากขึ้น โดยจะช่วยลดข้อผิดพลาดในการเขียนโปรแกรม ทำให้มีความง่ายในการเขียนมากกว่า

## 2.18 หัวข้อทางเทคนิคที่เกี่ยวกับผลิตภัณฑ์

กล้องวงจรปิด (Surveillance Camera) ที่ใช้กันอยู่ในปัจจุบันสามารถแบ่งออกได้เป็น โทรทัศน์วงจรปิด (Closed-circuit television ) และ กล้องไอพี (IP Surveillance) ซึ่งระบบกล้องวงจรปิด (Surveillance Camera) แบบ โทรทัศน์วงจรปิด (Closed-circuit television ) มีการทำงานที่ต้องเปิดใช้งานตลอดเวลาเพื่อใช้ในการตรวจสอบความเคลื่อนไหวและพฤติกรรมต่างๆ โดยจะส่งภาพวีดีโອมาแสดงผลโดยผู้ใช้จะต้องมีการตรวจสอบภาพวีดีโอด้วยตัวเอง ทั้งยังจำเป็นต้องใช้สายเคเบิลเพื่อส่งสัญญาณonaลีกอ ซึ่งจำเป็นต้องเดินสายไปยังจุดแสดงผล ซึ่งหากมีกล้องจำนวนมากก็จำเป็นที่ต้องเดินสายมากตามไปด้วยทำให้มีค่าใช้จ่ายเพิ่มขึ้นไป เช่นกัน ทั้งตัวกล้องจะต้องได้รับการจ่ายไฟให้ตลอดเวลา และอีกระบบคือ กล้องไอพี (IP Surveillance) จะใช้ระบบไอพีในการนำสัญญาณแทนผ่านทางสาย LAN หรือแม้กระทั่งทำงานผ่านระบบโทรศัพท์ (IP Camera) ซึ่งสามารถปรับเปลี่ยนการตั้งค่าต่างๆ ของตัวกล้องได้ ไม่ว่าจะเป็นการกำหนดเวลาการบันทึก การปรับค่าแสง การกำหนดการจับภาพ คุณภาพของภาพ และผู้ใช้งานสามารถอัปเดตอัตโนมัติตรวจสอบความเคลื่อนไหวผ่านทางอินเทอร์เน็ตได้

ทางกลุ่มนักศึกษาผู้จัดทำโครงการจึงได้จัดทำกล้องวงจรปิด Home Automation : Surveillance Camera ในระบบของกล้องวงจรปิด (Surveillance Camera) ขึ้นมาเพื่อเป็นเครื่องมือในการเฝ้าระวังความปลอดภัยให้กับทรัพย์สิน หรือพื้นที่ที่ต้องการรักษาความปลอดภัยโดยการคุ้มครองผ่านระบบเครือข่ายโทรศัพท์ และยังเป็นเครื่องมือที่ใช้บนโทรศัพท์เคลื่อนที่สมาร์ทโฟนระบบปฏิบัติการ iOS โดยทางกลุ่มนักศึกษาต้องการที่จะพัฒนา กล้องวงจรปิด (Surveillance Camera) ให้มีประสิทธิภาพในการใช้งานมากยิ่งขึ้นทั้งความสามารถในการตรวจจับการเคลื่อนไหวที่เกิดขึ้น การไลฟ์สตรีมมิ่งเพื่อเข้ามาดูภาพ ทั้งยังแข่งเดือนผู้ใช้งานให้ทราบถึงเหตุการณ์ที่เกิดขึ้นผ่านทางอีเมล์โดยไม่จำเป็นที่ต้องให้ผู้ใช้งานมีอัปเดตเข้ามาตรวจสอบตลอดเวลา มีการบันทึกข้อมูลผ่านทาง Cloud Storage และยังมีการสรุปข้อมูลที่สรุปข้อมูลการตรวจจับ ณ เวลาต่างๆ โดยสามารถคูน้ำพัชท์ทั้งหมดจาก Cloud Storage

## บทที่ 3 การออกแบบและการดำเนินโครงการ

### 3.1 อุปกรณ์และเครื่องมือที่ใช้ในการทำโครงการ

#### 3.1.1 ฮาร์ดแวร์ (Hardware)

- Raspberry Pi Model B (512 MB of Ram)
- Wifi USB Adapter รุ่น TL-WN725N
- กล้องเว็บแคม รุ่น Logitec HD Webcam C310
- เซ็นเซอร์ตรวจจับการเคลื่อนไหว (PIR Sensor)
- เซ็นเซอร์ใช้วัดอุณหภูมิและความชื้น (DHT11 Sensor)
- เซ็นเซอร์ตรวจจับควันไฟ (MQ-2 Sensor)
- iPhone 4S iOS7 or later

#### 3.1.2 ซอฟต์แวร์ (Software)

- Raspbian OS รุ่น 2015-05-05 Wheezy77
- OpenCV รุ่น 4.9 หรือสูงกว่า
- MySQL รุ่น 5.5.47-0+deb7u1
- OLSR Routing รุ่น 0.6.2
- Xcode รุ่น 7.3

#### 3.1.3 ภาษาที่ใช้ในการเขียนโปรแกรม

- ภาษา Python
- ภาษา Objective - C
- ภาษา Swift

## 3.2 โครงสร้างของระบบโดยรวม

ในระบบ Home Automation : Surveillance Camera ได้ทำการแบ่งส่วนการทำงานออกเป็น 3 ส่วนใหญ่ๆ ได้แก่ ส่วนของโมดูลกล้อง ส่วนของโหนดฐานข้อมูล และส่วนของโมบายแอปพลิเคชัน

### 3.2.1 ส่วนของโมดูลกล้อง (Camera Module)

- Raspberry Pi ทำหน้าที่เป็นโมดูลประสานงานระหว่าง เซ็นเซอร์ตรวจจับการเคลื่อนไหว กล้องเว็บแคม และ Wi-Fi Adapter โดยจะทำหน้าที่ในการประมวลผลภาพ (Image Processing) โดยการใช้ระบบจำจำใบหน้า (Face Recognition) เป็นหลัก
- Wi-Fi Adapter ทำหน้าที่ในส่วนของการส่งข้อมูล ไปมาระหว่างโหนดอื่นๆ ในรูปของเครือข่ายไร้สายแบบเมช (Wireless Mesh Network) การที่เลือก Wi-Fi Adapter เพราะว่าต้องการให้ติดต่อสื่อสารกันผ่านคลื่นความถี่ของ Wi-Fi ได้ และมีอัตรา bandwidth รองรับกับ โหนด Live Streaming ที่จะทำส่วนของแอปพลิเคชัน
- เซ็นเซอร์ตรวจจับการเคลื่อนไหว (PIR Sensor) ใช้ในการตรวจจับด้วยสัญญาณความร้อน จากวัตถุใดๆ ก็ตามที่เคลื่อนที่ เมื่อตรวจพบแล้วจะส่งโลจิก มายัง Raspberry Pi ให้ทำการบันทึกภาพ
- กล้องเว็บแคมทำหน้าที่เก็บภาพ การที่เลือกใช้ก็เพราะว่า มีหน้าที่การทำงานเกี่ยวกับการประมวลผลภาพ (Image Processing) มีความต้องการภาพที่มีคุณภาพดีในการประมวลผล

### 3.2.2 ส่วนของโหนดฐานข้อมูล (Database Node)

- Raspberry Pi ทำหน้าที่ในการเป็นระบบฐานข้อมูลของเครือข่ายภายในบ้าน (Home Network) ซึ่งในการทำงานปกติ โหนดกล้องจะทำการบันทึกข้อมูลเหตุการณ์ที่เกิดขึ้นมาลงฐานข้อมูลส่วนนี้ เพื่อรองรับการนำไปใช้สรุปผลและนำไปใช้งานต่อไป
- เมื่อทำการใช้งานไปช่วงระยะเวลาหนึ่งจะทำการส่งฐานข้อมูลไปสำรองที่ Dropbox เพื่อรับให้ส่วนของโมบายแอปพลิเคชันดึงไปใช้งาน

### 3.3.3 ส่วนของโมบายแอปพลิเคชัน (Mobile Application)

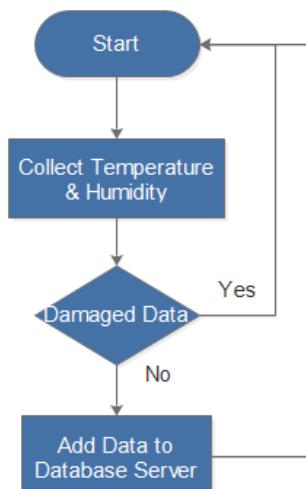
- โมบายแอปพลิเคชันทำหน้าที่เป็นอินเทอร์เฟซ (Interface) ติดต่อกับผู้ใช้ระบบ ทำให้สามารถใช้งานได้ง่าย โดยจะเป็นอินเทอร์เฟซที่จะสามารถทำการเชื่อมต่อไปยังโมดูลกล้องที่ต้องเพื่อทำการดูภาพวิดีโอ และเป็น อินเทอร์เฟซที่จะทำการแจ้งเตือนผู้ใช้งานว่ามีอะไรเกิดขึ้น รวมทั้งยังสามารถแสดงผลสรุปการทำงานในด้านต่างๆ ออกมาให้ผู้ใช้งานทราบได้

### 3.3 การออกแบบและการดำเนินการทางด้านซอฟแวร์

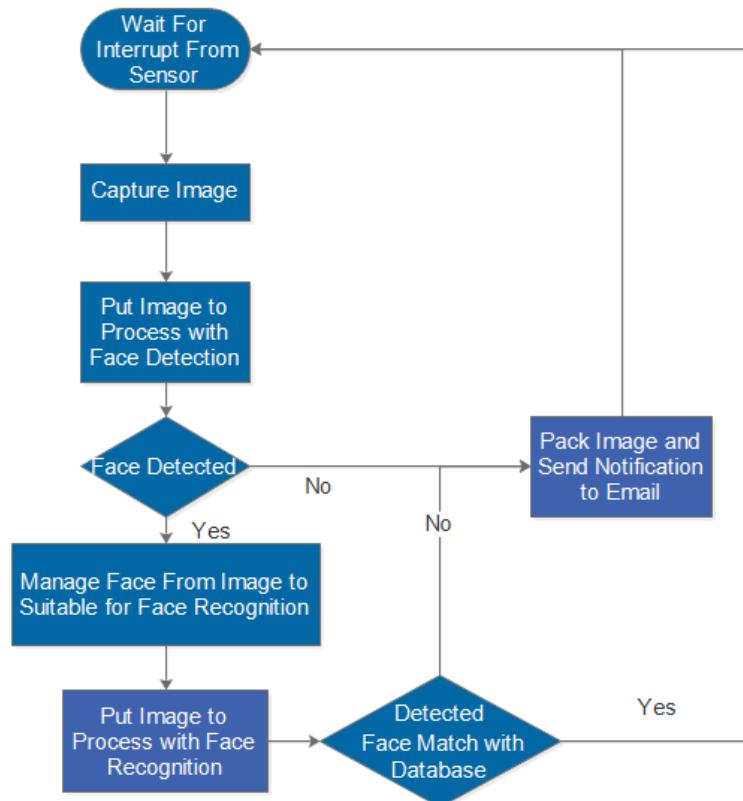
#### 3.3.1 การออกแบบและการดำเนินการทางด้านซอฟแวร์ในส่วนของโภนคล้อง

##### 1. ส่วนของระบบการทำงานหลัก

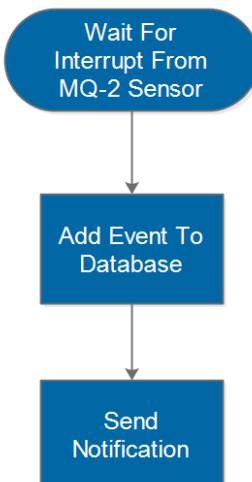
โปรแกรมในส่วนนี้จะเป็นการทำงานหลักของระบบ ซึ่งก็คือระบบการตรวจจับความเคลื่อนไหวและการวิเคราะห์ภาพ โดยขั้นตอนการทำงานคือการใช้เซ็นเซอร์ตรวจจับการเคลื่อนไหวในการตรวจจับการเคลื่อนไหวโดยใช้รังสีอินฟราเรด และใช้เซ็นเซอร์อินฯในการตรวจจับค่าอื่นๆ โดยจะทำการเปลี่ยนโปรแกรมในทำการรับสัญญาณ สูง/ต่ำ ในรูปแบบของอินเตอร์รัพท์โดยเมื่อมีอินเตอร์รัพท์เกิดขึ้นก็จะทำการถ่ายภาพ จากนั้นจะทำการแยกว่ามาจากอินเตอร์รัพท์ของเซ็นเซอร์ตรวจจับการเคลื่อนไหว(PIR Sensor) หรือถ้าเซ็นเซอร์อื่นๆจะทำการส่งรูปภาพผ่านระบบ Notification เพื่อมาแจ้งเตือนทันที แต่ถ้าหากมาจากเซ็นเซอร์ตรวจจับการเคลื่อนไหวจะทำการประมวลผลรูปภาพโดยเรียกใช้ฟังก์ชัน การตรวจสอบใบหน้า (Face Detection) มาจัดการตรวจสอบว่ามีใบหน้าอยู่ในภาพหรือไม่ ถ้าหากไม่พบใบหน้าก็จะทำการส่งมาแจ้งเตือนว่า ตรวจพบการเคลื่อนไหวแต่ไม่สามารถตรวจจับใบหน้าได้ผ่านระบบแจ้งเตือน (Notification) กลับกันหากตรวจจับใบหน้าได้ จะนำรูปภาพนั้นนำมาทำการแบ่งส่วนเพื่อให้ได้เฉพาะรูปภาพใบหน้าทุกใบหน้าอອกมา จากนั้นจึงจะนำไปบันทึกภาพนั้นมาเข้าฟังก์ชัน ระบบจดจำใบหน้า(Face Recognition) เพื่อทำการตรวจสอบว่าใช่ใบหน้าของคนที่อยู่ในฐานข้อมูลของเราหรือไม่ ถ้าใช้ก็จะทำการทำงานในรอบนี้และไปริ่มด้านการทำงานในโหมดรอการทำงานอีกครั้ง แต่ถ้าตรวจจับใบหน้าที่ไม่ตรงกับฐานข้อมูล ก็จะทำการส่งการแจ้งเตือนพร้อมรูปภาพผ่านระบบแจ้งเตือนทันที และถ้าหากตรวจพบการเคลื่อนไหวแต่ไม่พบใบหน้า ตรงส่วนนี้สามารถเลือกได้ว่าจะให้ทำการแจ้งเตือนผ่านทางอีเมลหรือไม่ เพราะในการใช้งานสามารถแบ่งเป็นการใช้งานในที่ส่วนตัว และที่ส่วนรวม ถ้าหากใช้งานในที่ส่วนตัวทางผู้ออกแบบคิดว่าควรจะทำการแจ้งเตือนถึงแม้จะไม่พบใบหน้าก็ตาม และถ้าหากใช้ในที่ส่วนรวมการส่งการแจ้งเตือนทุกๆการเคลื่อนจะเป็นการเปลี่ยนแปลงแนวคิดที่ ดังนั้นจึงควรทำการวิเคราะห์ตรวจสอบใบหน้าก่อนถ้าพบผู้ที่ไม่ตรงกับฐานข้อมูลจึงจะส่งการแจ้งเตือน โดยสามารถทำการแก้ไขได้ที่ไฟล์การตั้งค่าที่มีชื่อว่า config.py ภายในตัว Raspberry Pi



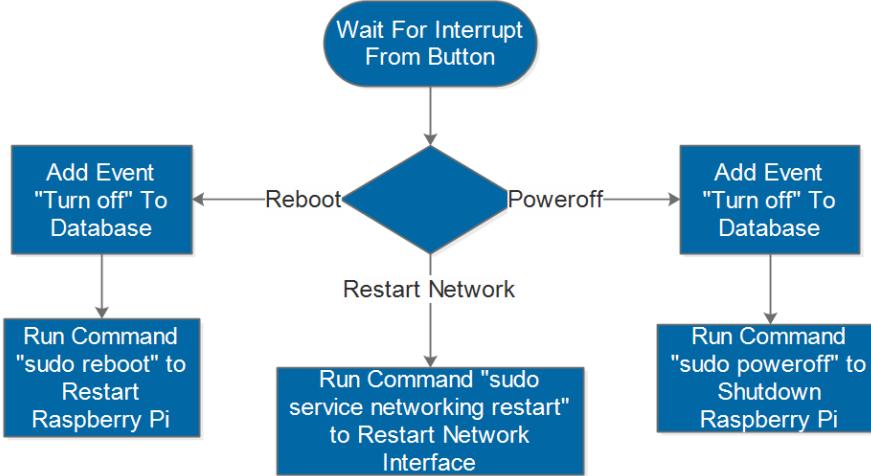
รูปภาพที่ 3.1 : Flowchart ของระบบการทำงานหลักของโภนคล้อง



รูปภาพที่ 3.2 : Flowchart ของฟังก์ชันการทำงานเมื่อเกิดการอินเตอร์รัพท์จากเซ็นเซอร์ตรวจสอบจับการเคลื่อนไหว



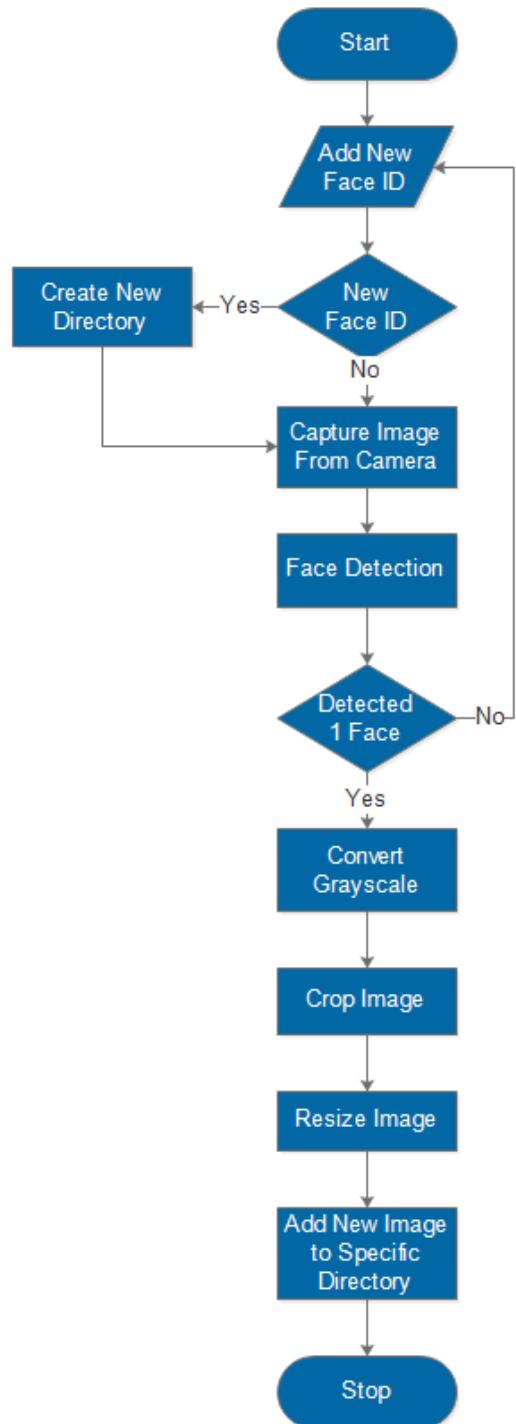
รูปภาพที่ 3.3 : Flowchart ของฟังก์ชันการทำงานเมื่อเกิดการอินเตอร์รัพท์จากเซ็นเซอร์ตรวจแก๊สไวไฟ(MQ-2)



รูปภาพที่ 3.4 : Flowchart ของฟังก์ชันการทำงานเมื่อเกิดการอินเตอร์รัพท์จากปุ่มด่างๆ

## 2. ส่วนของการเพิ่มใบหน้าลงฐานข้อมูลใบหน้า (Face Database)

ในส่วนนี้จะเป็นโปรแกรมในส่วนของการเพิ่มใบหน้าที่ต้องการเป็นสมาชิกที่ไม่ต้องการให้มีการเดือนลงฐานข้อมูล โดยขั้นตอนการเพิ่มใบหน้านั้นจะเริ่มจากการระบุว่าใบหน้าที่เราจะทำการเพิ่มนั้นเป็นใบหน้าของผู้ที่มีข้อมูลอยู่แล้วหรือไม่ ถ้าเป็นก็ให้ทำการกรอกไอดีของคนนั้น ถ้าไม่ก็จะเป็นการสร้างไดเรกทอรี (Directory) ใหม่ ต่อมาก็จะทำการ ถ่ายภาพจากกล้อง จะต้องนำรูปภาพที่มีใบหน้านั้นๆมาแล้ว ยังต้องทำการแปลง ปรับแต่งภาพให้เหมาะสมกับการใช้งานในระบบจดจำใบหน้า โดยลำดับการทำงานจะเริ่มจาก การนำรูปภาพที่มีใบหน้าที่ต้องการอยู่มาทำการ ตรวจสอบใบหน้าเพื่อหาตำแหน่งใบหน้าในรูปภาพ ถ้าหากมีจำนวนใบหน้ามากกว่า 1 ก็จะทำการยกเลิกเพื่อให้ทำการเริ่มต้นการทำงานใหม่ เมื่อเก็บภาพที่มีใบหน้าเพียง 1 ใบหน้าได้แล้ว จะทำการแปลงภาพเป็น Grayscale เพื่อเตรียมการใช้งานต่อ จากนั้นก็จะทำการตัดรูปภาพนำเฉพาะส่วนใบหน้านั้นๆออกมา เช่นเป็นไฟล์ใหม่ โดยหาร ตัดส่วนรูปภาพก็จะใช้การคำนวณทางคณิตศาสตร์เข้าช่วย เพราะว่า รูปภาพที่เราถ่าย อาจมีใบหน้าไม่เท่ากัน และจะทำการตรวจใบหน้าเฉพาะใบหน้าเท่านั้น ไม่รวมตรงหมหรือลำคอ เพื่อให้ได้ผลลัพธ์ที่มีใบหน้าขนาดเท่ากันทุกรูปและ เราสามารถถูกใจได้ว่า เป็นใบหน้าของใคร จึงต้องทำการปรับแต่งภาพก่อน จากนั้นจึงนำภาพที่ผ่านฟังก์ชันตัดส่วนรูปภาพ แล้วมาทำการปรับเปลี่ยนขนาดด้วยขนาดที่สอดคล้องกับสูตรคณิตศาสตร์ที่ใช้ในขั้นตอนการตัดส่วนรูปภาพ เมื่อเสร็จสิ้นการแต่งภาพก็จะทำการบันทึกใบหน้าที่ปรับแต่งแล้วลงในไดเรกทอรีที่ได้เตรียมไว้ การที่ไม่ทำการบันทึกลงฐานข้อมูลโดยตรงนั้น เพราะว่าจะทำให้ภาพมีคุณภาพแย่ลงจากการบีบอัดเข้าฐานข้อมูล ทำให้ไม่สามารถนำมาใช้เบรย์บิทท์บินใบหน้าได้

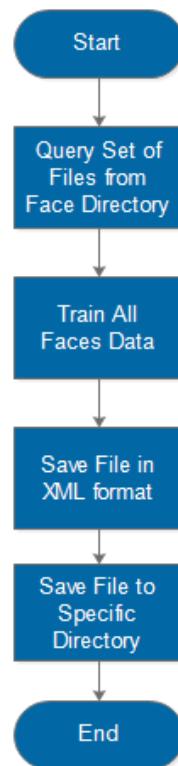


รูปภาพที่ 3.5 : Flowchart ของฟังก์ชันการทำงานการเพิ่มใบหน้าลงฐานข้อมูลใบหน้า

### 3. ระบบการแปลงใบหน้าในฐานข้อมูลใบหน้า (Face Database) ให้เป็นข้อมูลที่สามารถนำมาเปรียบเทียบใบหน้าได้

จากการที่ฐานข้อมูลใบหน้าของนั้นเป็นรูปภาพ การที่เราจะทำการนำรูปภาพนั่นมาเปรียบเทียบก็รูปภาพนั่นได้นั้น จะต้องทำการแปลงภาพด้านลับให้เป็นค่าที่คอมพิวเตอร์ สามารถนำมาใช้เปรียบเทียบได้ก่อน และแน่นอนว่าถ้าหากเรานำรูปภาพเพียง 1 ภาพของเรามาทำฐานข้อมูลใบหน้า อาจจะไม่เพียงพอ เนื่องจากใบหน้าของเรานั้นมีหลายมุมมอง เราต้องทำให้โปรแกรมนั้นเข้าใจว่าหน้าในมุมนี้เป็นหน้าของใคร ก็จะเป็นจะต้องเก็บใบหน้า มุมนั้นมาทำเป็นฐานข้อมูลใบหน้าด้วย

โดยการทำงานของระบบนี้จะเริ่มจากการดึงไฟล์รูปภาพทั้งหมดที่เราต้องการมาทำฐานข้อมูลรวมกัน จากนั้นจึงทำการเลือก อัลกอริทึมการทำงาน ที่จะทำการสร้าง เนื่องจะอัลกอริทึมที่จะใช้ในกระบวนการจดจำใบหน้า (Face Recognition) นั้นมีหลายอย่าง ดังนั้นหากเราเลือกใช้แบบไหนแล้วก็จะต้องทำการสร้างข้อมูลที่สอดคล้องกับ อัลกอริทึมนั้นๆอ กมา เมื่อเราทำการแปลงรูปภาพตามอัลกอริทึมแล้วก็จะได้เป็นไฟล์นามสกุล XML ออกมาเพื่อ นำมาใช้งานต่อไป ข้อมูลในไฟล์นามสกุล XML นี้จะถูกเก็บในรูปแบบของโครงสร้างข้อมูล ในที่นี้จะเป็นข้อมูล โครงสร้างรูปภาพ โดยข้อมูลที่ได้นั้นจะเป็นตัวเลขทำให้สามารถนำไปทำการเปรียบเทียบได้ต่อไป



รูปภาพที่ 3.6 : Flowchart การทำงานของระบบแปลงโครงสร้างข้อมูลของรูปภาพ

#### 4. ส่วนของการตั้งค่าที่สามารถปรับแก้ได้ต่างๆ

เพื่อให้เกิดความสะดวกแก่ผู้ใช้งานที่ต้องการปรับแก้ข้อมูลในส่วนต่างๆ จึงได้จัดทำไฟล์ที่เป็นสมุด Configuration ของระบบการทำงานทั้งหมด โดยสิ่งที่สามารถปรับแก้ได้จะเป็นไปดังตารางต่อไปนี้

Parameters	Meaning
NodeID	Node ID ประจำโหนດการทำงาน
Status	สถานะของโหนดการทำงานประกอบด้วย “on” และ “off”
UserMySQL	Username สำหรับการล็อกอินเข้าฐานข้อมูล MySQL Server
PasswordMySQL	Password สำหรับการล็อกอิน เข้าฐานข้อมูล MySQL Server
DatabaseName	ชื่อฐานข้อมูลที่ต้องการจะเข้าถึง
HostDatabase	Ip Address ของฐานข้อมูลที่ต้องการจะเข้าถึง
FromAddr	อีเมลล์ที่ต้องการให้ทำหน้าเป็นเหมือนระบบอัตโนมัติอย่างแจ้งเตือน
PassOfSender	Password สำหรับอีเมลล์ที่ต้องการให้ทำหน้าเป็นเหมือนระบบอัตโนมัติอย่างแจ้งเตือน
ToAddr	อีเมลล์ที่ต้องการให้แจ้งเตือน
ImageLocation	ตำแหน่งของไฟล์รูปภาพเหตุการณ์ในฐานข้อมูล Dropbox
TempPin	Pin รับข้อมูลของเซ็นเซอร์ตรวจจับอุณหภูมิและความชื้น
PIR_Pin	Pin รับข้อมูลของเซ็นเซอร์ตรวจจับความเคลื่อนไหว
ledPower	Pin ส่งเอาท์พุตแสดงสถานะการทำงานของโหนด
button_poweroff	Pin รับอินพุตอินเตอร์รัพท์ของปุ่มการทำงาน “Power Off”
button_restart	Pin รับอินพุตอินเตอร์รัพท์ของปุ่มการทำงาน “Reboot”
button_reconnect_internet	Pin รับอินพุตอินเตอร์รัพท์ของปุ่มการทำงาน “Restart Network”
mq2_Pin	Pin รับอินพุตอินเตอร์รัพท์ของเซ็นเซอร์ตรวจจับก๊าซไวไฟ (MQ-2)
notiFaceNotDetected	ตัวแปรที่ใช้กำหนดว่าเมื่อตรวจสอบการเคลื่อนไหวแต่ไม่พบใบหน้า จะทำการแจ้งเตือนหรือไม่ สำหรับคัดสินใจว่าใช้ใบหน้าที่ส่วนตัว หรือใบหน้าที่ส่วนรวม โดยค่า “1” คือทำการส่งการแจ้งเตือน หมายสำหรับใช้ใบหน้าที่ส่วนตัว ค่า “0” คือไม่ทำการส่ง หมายสำหรับพื้นที่ส่วนรวมหรือที่สาธารณะ เพราะจะได้ไม่ต้องทำการแจ้งเตือนทุกความเคลื่อนไหว จะแจ้งเตือนเฉพาะเจอบนหน้าไม้รู้จักเท่านั้น
Front_Dir_Image	ตำแหน่งของไฟล์รูปภาพเหตุการณ์ที่บันทึกภายในโหนดการทำงาน
DropboxCommand	ใช้สำหรับกำหนดตำแหน่งของฟังก์ชันการทำงานของ Dropbox ที่จะทำการดึงมาใช้ รองรับสำหรับผู้ที่ติดตั้งฟังก์ชันการทำงาน Dropbox ในตำแหน่งอื่นๆ
training_file	ตำแหน่งของไฟล์ฐานข้อมูลใบหน้าที่จะนำมาใช้

### 3.3.2 การออกแบบและการดำเนินการทางด้านซอฟแวร์ในส่วนของโภนบายแอพลิเคชั่น



รูปภาพที่ 3.7 : ภาพ Application Icon : Surveillance Camera

การทำงานในส่วนของโภนบายแอพลิเคชั่นจะเริ่มต้นจากหน้าต่างเมนูดังภาพที่ 3.8

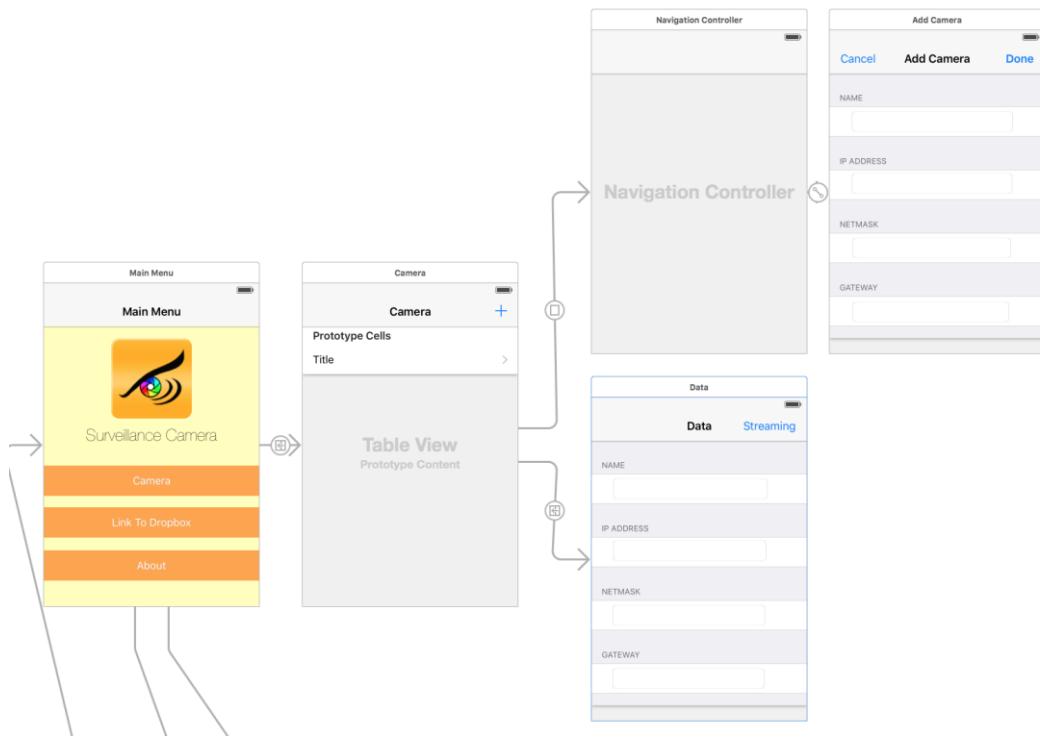


รูปภาพที่ 3.8 : ภาพตัวอย่างหน้าต่างเมนูในส่วนของโภนบายแอพลิเคชั่น

ได้ทำการแบ่งการทำงานออกเป็น 3 เมนู ได้แก่

#### 1. Camera

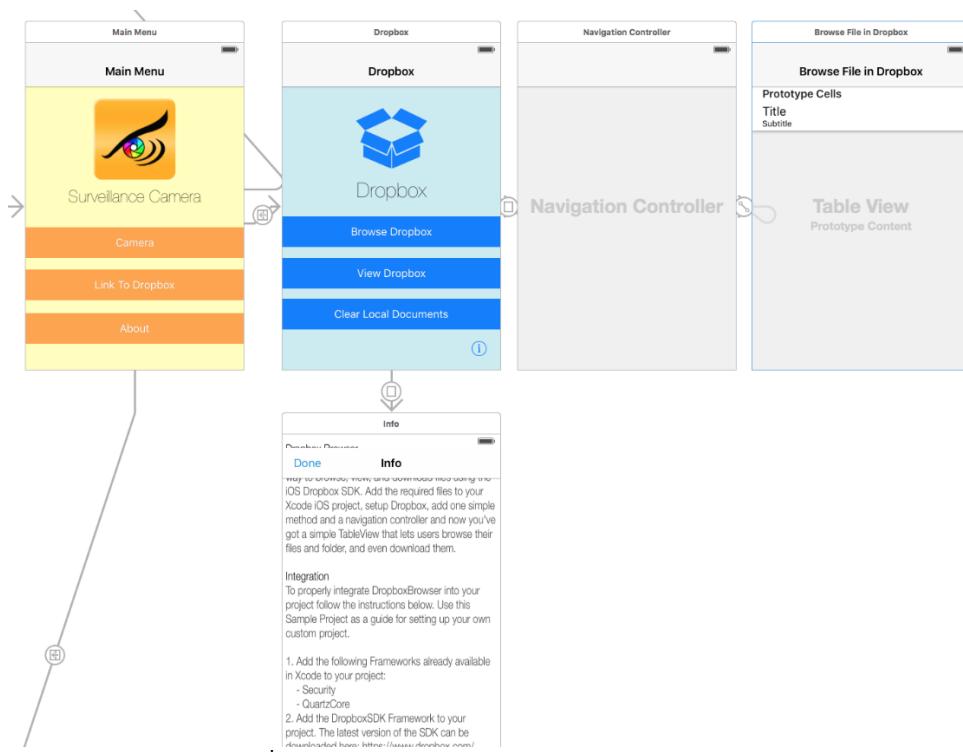
เป็นโภนดกการทำงานที่จะแสดงรายชื่อ โภนดกล้องที่อยู่ในเครือข่ายที่เราได้ที่ทำการระบุ IP Address ของ อุปกรณ์แล้ว โดยเราจะสามารถทำการ เพิ่ม/แก้ไข หรือ ลบรายการ โภนดกล้อง ได้ และเมื่อทำการคลิกเข้าไปใน โภนดกล้อง ก็จะสามารถแก้ไขข้อมูลและใช้งานในส่วนของการติดต่อได้



รูปที่ 3.9 : รูปภาพโครงสร้างในส่วนของเมนู Camera

## 2. Link To Dropbox

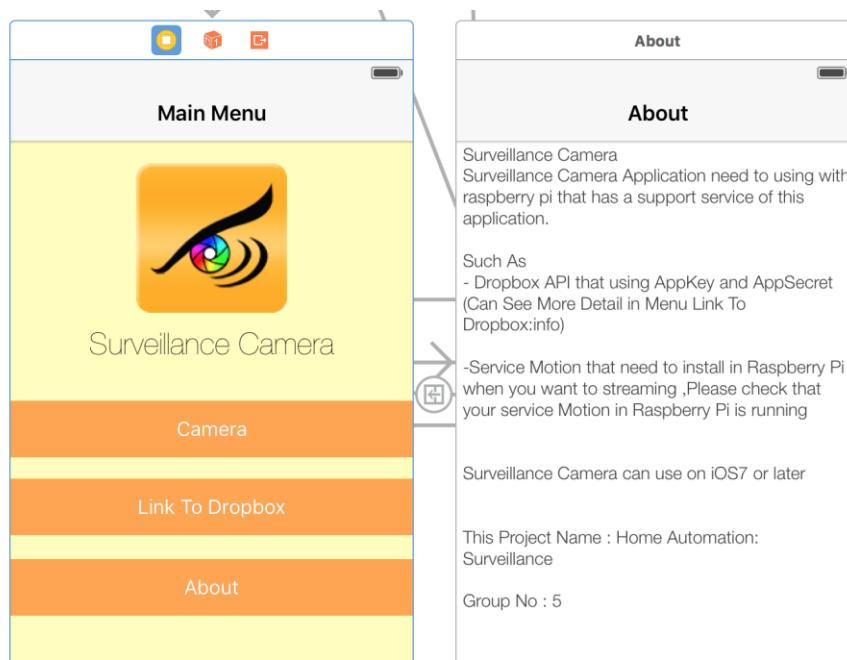
เป็นโหมดคือใช้สำหรับการเชื่อมต่อคลังข้อมูลของ Dropbox โดยจะสามารถเรียกดูรายการไฟล์ที่อยู่ใน Dropbox ได้ สามารถที่จะทำการถ่ายภาพไฟล์โดยการใส่กีบไวร์ด ชื่อไฟล์ สกุลไฟล์ และสามารถทำการดาวน์โหลดไฟล์ได้ รวมถึงการแสดงผลผ่านทางเว็บบราวเซอร์



รูปที่ 3.10 : รูปภาพโครงสร้างในส่วนของเมนู Link To Dropbox

### 3. About

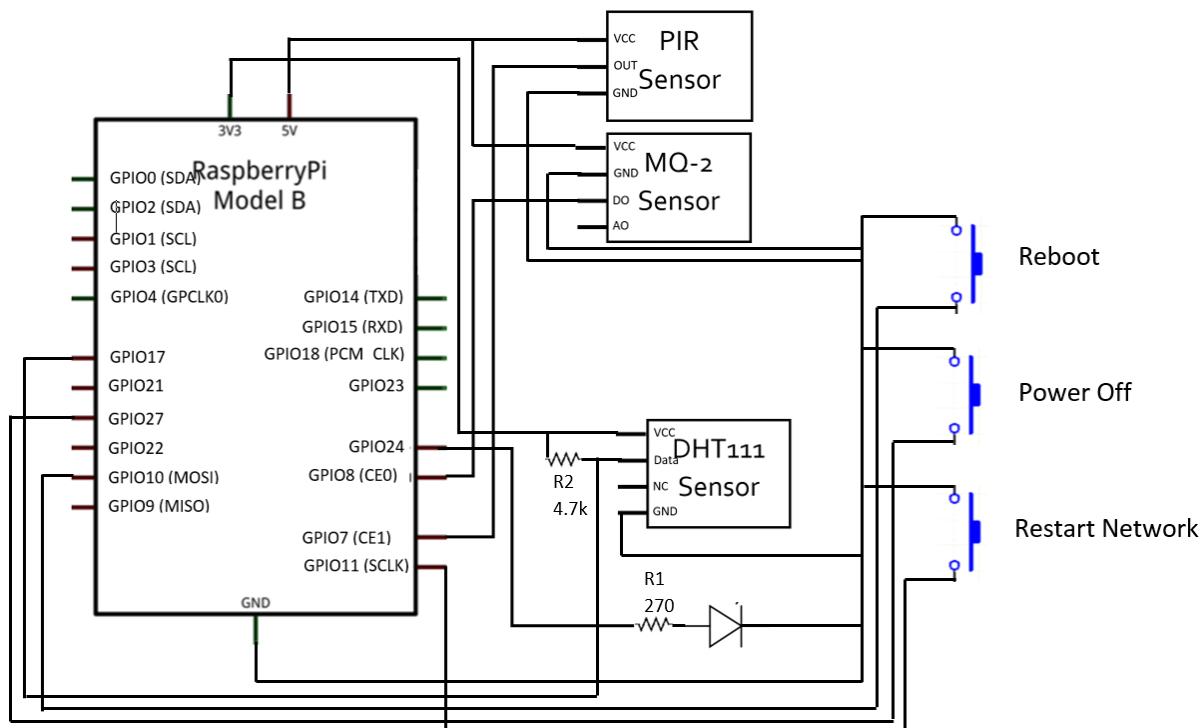
ເປັນໄທ່ມຄຣາຍລະເອີດຂໍ້ມູນຂອງໄມນາຍແອພລິເຄຊົ້ນ



ຮູບທີ 3.11 : ຮູບກາພໂຄຮງສ້າງໃນສ່ວນຂອງເມນຸ About

### 3.4 การออกแบบและการดำเนินงานทางด้านสารด้วยวิธีการอ่านค่าจากเซ็นเซอร์

สำหรับ โหนดกล้อง ได้ทำการออกแบบวงจรเบื้องต้นไว้ดังนี้



รูปภาพที่ 3.12 : วงจรเบื้องต้นของโหนดกล้อง



รูปภาพที่ 3.13 : ภาพตำแหน่ง Pin โดยอ้างพื้นฐานจาก GPIO.BCM

ที่มา : <http://raspberrypiht.blogspot.com/2014/02/gpio-raspberry-pi.html>

การเชื่อมต่อ Pin ต่างๆของ Raspberry Pi จะใช้ฟีล์ชันการดู Pin จาก GPIO.BCM ดังภาพที่ 3.10 และแต่ละ Pin มีการทำงานต่างๆตามตารางด้านล่าง

Pin	หน้าที่
7	รับอินพุตอินเตอร์รัพท์จากเซ็นเซอร์ตรวจจับการเคลื่อนไหว
8	รับดิจิตอลอินพุตจากเซ็นเซอร์ตรวจจับแก๊สไวไฟ
10	รับอินพุตจากปุ่ม “Reboot”
11	รับอินพุตจากปุ่ม “Networking Restart”
17	รับอินพุตจาก Temperature Sensor
22	รับอินพุตจากปุ่ม “Power Off”
24	เป็นเอาท์พุตส่งออกไปเพื่อแสดงสถานะการทำงาน

ตารางที่ 3.2 : การเชื่อมต่อ Pin เข้ากับ Raspberry Pi

ได้ทำการต่อ Raspberry Pi เข้ากับเซ็นเซอร์ต่างๆ 3 ชนิด ได้แก่

1. เซ็นเซอร์ตรวจจับความเคลื่อนไหว
2. เซ็นเซอร์ตรวจจับอุณหภูมิและความชื้น
3. เซ็นเซอร์ตรวจจับแก๊สไวไฟ (MQ-2)

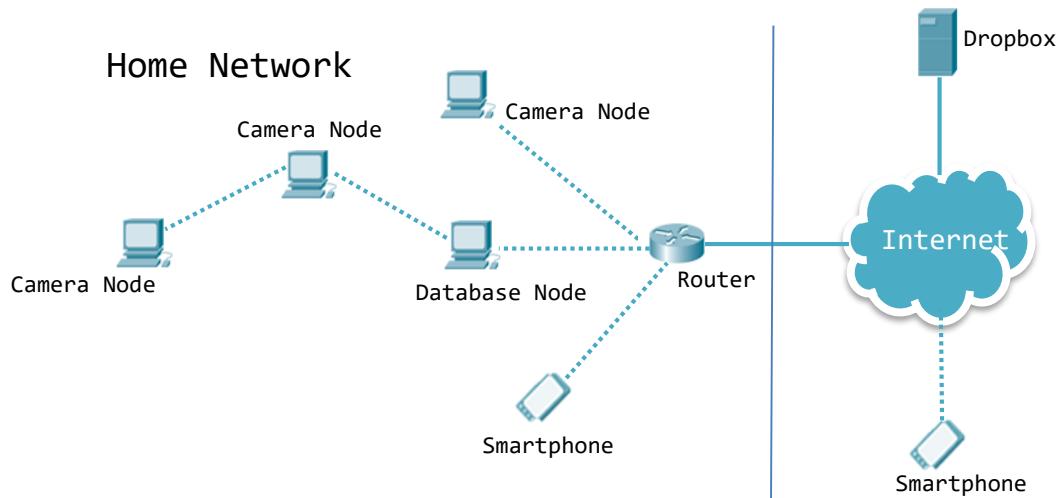
นอกจากนี้ เนื่องจากตัว Raspberry Pi เมื่อใช้งานไปเรื่อยๆเพื่ออาจจะเกิดปัญหาการบูตไม่ขึ้นเนื่องจากไม่ได้ทำการปิดเครื่องตามขั้นตอน แต่ทำการกดปลั๊กแทน ส่งผลให้มีโอกาสสักพัก SD Card จะไม่สามารถใช้งานได้ และในบางครั้งเมื่อทำการปิดเครื่อง Raspberry Pi ขึ้นมาแล้ว แต่ยังต้องรีเฟสสำหรับการเชื่อมต่อเครือข่ายไม่ทำงาน จำเป็นต้องทำการกดปลั๊ก หรือทำการต่อจอยเพื่อรีบูตใหม่ ทำให้เกิดความไม่สะดวกในด้านต่างๆ ดังนั้นจึงได้ทำการเพิ่มปุ่มการทำงานทั้งหมด 3 อย่าง ดังนี้

- 1.รีบูต เพื่อทำการรีบูตเครื่องในกรณีที่เมื่อทำการรีบูตอินเตอร์เฟสแล้ว อินเตอร์เฟสยังคงไม่ทำงาน เช่นเดิม
- 2.ปิดเครื่อง เพื่อเป็นการทำการปิดเครื่องโดยการใช้คำสั่ง关机ในระบบปฏิบัติการ
- 3.รีบูตอินเตอร์เฟสเชื่อมต่อกับเครือข่าย เพื่อทำการรีบูตการปิดการเชื่อมต่อเครือข่ายต่างๆ

### 3.5 การออกแบบและการดำเนินงานทางด้านเครือข่าย

ได้ทำการออกแบบเครือข่ายให้ระบบเครือข่ายของกล้องวงจรปิดเชื่อมต่อกับ Home Network ทุกโหนด โดยไม่ว่าโหนดไหนก็จะสามารถส่งข้อมูลไปยังอินเทอร์เน็ตได้หากอยู่ในระยะลัญญาณการทำงานของ Home เรายังสามารถเชื่อมต่อได้โดยไม่ต้องผ่านเราเตอร์เพื่อให้เกิดความคงสภาพการทำงานของระบบบางส่วน แม้จะมีโหนดใดโหนดหนึ่งหยุดการทำงาน

โดยจะทำการตั้งค่าไอพีแบบ Static ให้กับโหนดการทำงานทุกโหนด โดยไอพีที่ตั้งค่าจะไม่ตรงกับ ไอพีแบบ Dynamic ที่ทางเราเตอร์แจกว่า โดยจะมีแผนผังโครงสร้างเครือข่ายคร่าวๆดังภาพที่ 3.14



รูปภาพที่ 3.14 : โครงสร้างทางเครือข่ายเบื้องต้นของระบบกล้องวงจรปิด

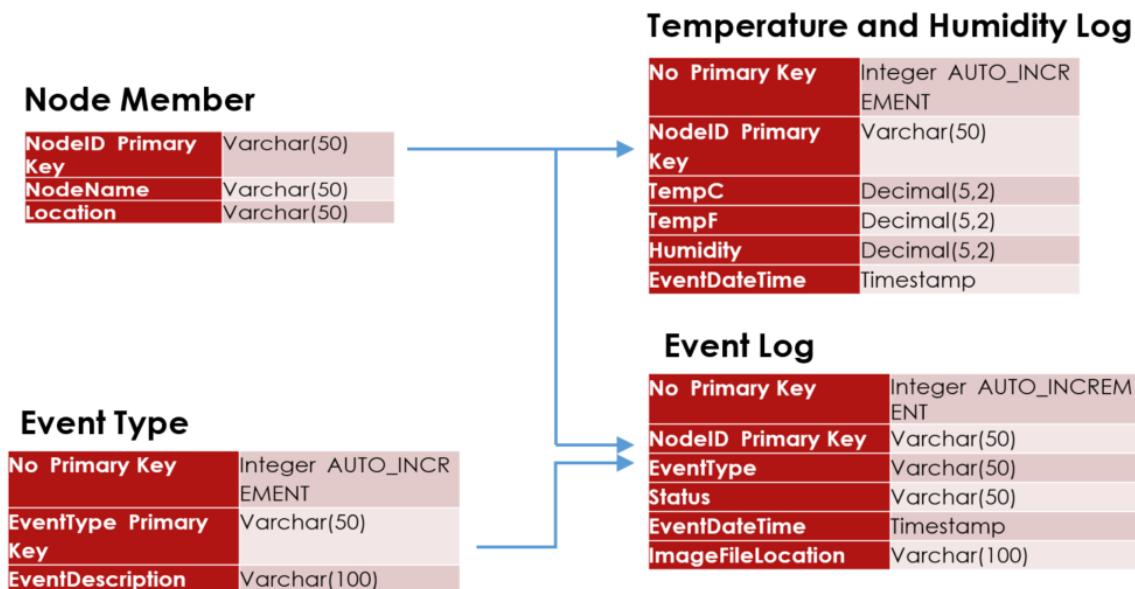
โดยจะสรุปการทำงานในแต่ละส่วนของ Network ได้ดังนี้

<b>โหนดกล้อง (Camera Node)</b>	<ul style="list-style-type: none"> <li>ระบบตรวจสอบใบหน้า (Face Recognition)</li> <li>ระบบตรวจจับการเคลื่อนไหว (Detect Motion)</li> <li>ระบบแจ้งเตือนอีเมล์ (Email Notification)</li> </ul>
<b>โหนดร้านข้อมูล (Database Node)</b>	<ul style="list-style-type: none"> <li>ฐานข้อมูล (Database)</li> <li>ระบบสำรองฐานข้อมูลไปยัง Dropbox</li> </ul>
<b>สมาร์ทโฟน (Smartphone)</b>	<ul style="list-style-type: none"> <li>ระบบ Live Streaming</li> <li>ระบบเชื่อมต่อสู่ Dropbox เพื่อดาวน์โหลดและดูข้อมูล</li> </ul>
<b>Dropbox</b>	<ul style="list-style-type: none"> <li>เป็นฐานข้อมูลไว้ให้สมาร์ทโฟนดึงข้อมูลมาทำงานสรุปข้อมูล</li> </ul>
<b>การเข้ามาภายในระบบ เครือข่ายภายในบ้าน</b>	<ul style="list-style-type: none"> <li>จะทำการเลือกใช้บริการ Dynamic DNS จาก <a href="http://www.noip.com">www.noip.com</a> เพื่อเข้า Raspberry Pi จากภายนอกระบบเครือข่ายภายในบ้าน</li> </ul>

ตารางที่ 3.3 : ตารางการทำงานในแต่ละส่วนของ Network

### 3.6 การออกแบบและการดำเนินงานทางด้านระบบฐานข้อมูล

ได้ทำการออกแบบระบบฐานข้อมูลแบบโครงสร้าง (Data Structure) ไว้ใช้เก็บข้อมูลต่างๆที่เกิดจากการใช้งานอุปกรณ์ บันทึกข้อมูลเหตุการณ์ต่างๆลงฐานข้อมูล MySQL ของ Raspberry Pi ตัวที่ทำการแชร์อินเทอร์เน็ต ผ่านสายอีเทอร์เน็ตและเมื่อระบบทำงานไปชักช่วงเวลาหนึ่ง ก็จะทำการบันทึกข้อมูลเป็นไฟล์นามสกุล db ส่งไปเก็บยัง Dropbox เพื่อให้ส่วนของแอพพลิเคชั่นนำไปใช้งานต่อไป



รูปภาพที่ 3.15 : โครงสร้างฐานข้อมูล

#### ตารางในฐานข้อมูลและการสร้างฐานข้อมูล

ตารางที่ 1 : Node Member

ชื่อคอลัมน์	ประเภทคอลัมน์	คำอธิบาย
NodeID <i>Primary Key</i>	Varchar(50)	ไอดีของโหนດกล้อง
NodeName	Varchar(50)	ชื่อโหนดกล้อง
Location	Varchar(50)	ตำแหน่งที่ติดตั้งโหนดกล้อง

ตารางที่ 3.4 : ตารางแสดงข้อมูลในตาราง Node Member

**ตารางที่ 2 : Event Log**

ชื่อคอลัมน์	ประเภทคอลัมน์	คำอธิบาย
No <i>Primary Key</i>	Integer <i>AUTO_INCREMENT</i>	ลำดับหมายเลขจากเหตุการณ์ที่เก็บมาทั้งหมด
NodeID <i>Primary Key</i>	Varchar(50)	ไอดีของโหนดกล้อง
EventType	Varchar(50)	ประเภทของเหตุการณ์ที่เกิดขึ้น
Status	Varchar(50)	สถานะขณะเก็บข้อมูลลงตาราง “Event Log” ว่าโหนดกล้องทำงานอยู่ ถ้าหากเป็นกรณีปิดเครื่องจะมีการบันทึกลงตาราง “Event Log” ให้มีสถานะเป็น off
EventDateTime	Timestamp	เวลาที่ทำการบันทึกข้อมูลลงตาราง “Event Log”
ImageFileLocation	Varchar(100)	บอกตำแหน่งของไฟล์ในไดร์ฟหรือภายใน Dropbox

ตารางที่ 3.5 : ตารางแสดงข้อมูลในตาราง Event Log

**ตารางที่ 3 : Event Type**

ชื่อคอลัมน์	ประเภทคอลัมน์	คำอธิบาย
No <i>Primary Key</i>	Integer <i>AUTO_INCREMENT</i>	ลำดับหมายเลข
EventType <i>Primary Key</i>	Varchar(50)	ไอดีของเหตุการณ์แต่ละประเภท
EventDescription	Varchar(100)	คำอธิบายว่า “EventType” แต่ละประเภท

ตารางที่ 3.6 : ตารางแสดงข้อมูลในตาราง Event Type

### ประเภทของเหตุการณ์จะแบ่งได้คร่าวๆ ได้เป็น

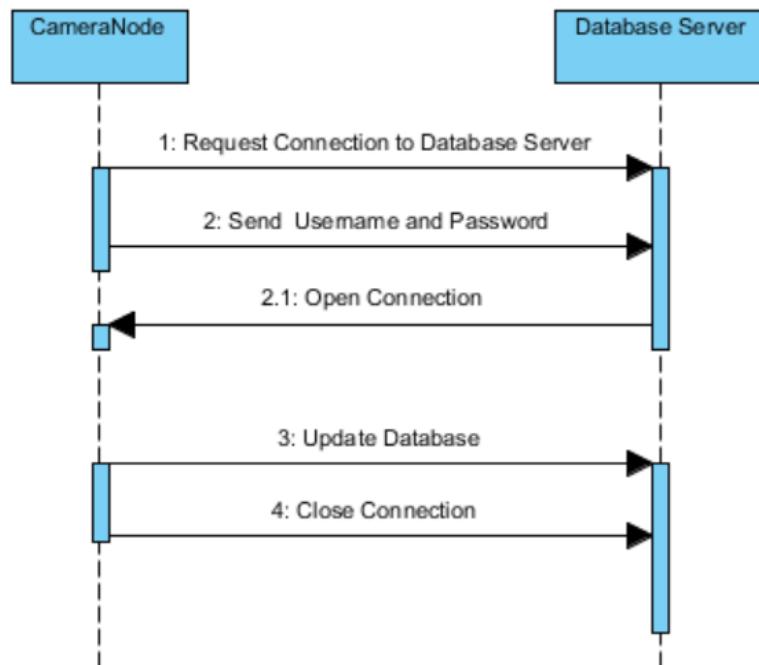
1. “Turn On”
2. “Turn Off”
3. “Temperature Alert(High)”
4. “Temperature Alert(Low)”
5. “Motion Detected , No Face Detected”
6. “Motion Detected, Face Detected : Member Face”
7. “Motion Detected , NonMember Face”
8. “Smoke Detected”

ตารางที่ 4 : Temp Humidity Log

ชื่อคอลัมน์	ประเภทคอลัมน์	คำอธิบาย
No <i>Primary Key</i>	Integer <i>AUTO_INCREMENT</i>	หมายเลขลำดับของล็อก (Log) ข้อมูล
NodeID <i>Primary Key</i>	Varchar(50)	ไอดีของโหนดที่ทำการเก็บล็อก(Log)
TempC	Decimal(5,2)	อุณหภูมิแบบองศาเซลเซียส
TempF	Decimal(5,2)	อุณหภูมิแบบองศาฟาเรนไฮต์
Humidity	Decimal(5,2)	ความชื้น
EventDateTime	Timestamp	เวลาที่ทำการบันทึกล็อก(Log)

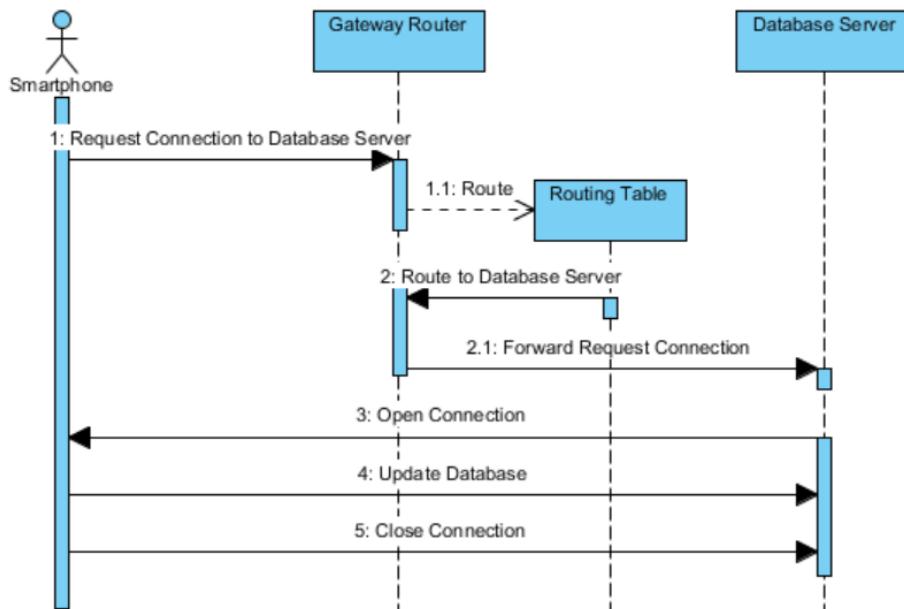
ตารางที่ 3.7 : ตารางแสดงข้อมูลในตาราง Temp Humidity Log

Sequence Diagram แสดงการทำงานระหว่างโหนดกล้องและโหนดฐานข้อมูล



รูปภาพที่ 3.16 : รูปภาพแสดง Sequence Diagram การทำงานระหว่างโหนดกล้องและโหนดฐานข้อมูลขณะรับส่งข้อมูล

### Sequence Diagram แสดงการทำงานระหว่างสมาร์ทโฟน และ Dropbox



รูปที่ 3.17 : รูปภาพแสดง Sequence Diagram การทำงานระหว่างสมาร์ทโฟนและ Dropbox ขนะรับส่งข้อมูล

### 3.7 ผลิตภัณฑ์ที่จะพัฒนาเป็นตัวต้นแบบสำหรับโครงงานประเพณีผลิตภัณฑ์ทำการค้า ความสามารถของผลิตภัณฑ์ต้นแบบที่จะพัฒนามีดังนี้

#### ส่วนของโมดูลกล้อง (Camera Module)

1. การตรวจจับความเคลื่อนไหว (Motion Detection)
2. การตรวจจับใบหน้า (Face Detection)
3. การจดจำใบหน้า (Face Recognition)
4. การแจ้งเตือนทางอีเมล (Email Notification)
5. การเชื่อมต่อ กับ Cloud Data Storage
6. ใช้โครงสร้างเครือข่ายไร้สายแบบmesh
7. เก็บล็อก (Log) ข้อมูล
8. ตรวจจับอุณหภูมิและความชื้น
9. ตรวจจับควันไฟ

#### ส่วนของ Mobile Application

1. ไลฟ์สตรีมมิ่ง (Live Streaming)
2. การเชื่อมต่อคลังข้อมูลครีอปบีก (Link To Dropbox)

## บทที่ 4

### ผลการดำเนินงาน

#### 4.1 การสร้างเครือข่ายของ Surveillance Camera

ทางด้านการดำเนินงานการสร้างเครือข่าย Home Network

##### 1) เริ่มจากการติดตั้งไดรเวอร์ของ Wireless Adapter

###### สำหรับ TP-Link WN725N

1. ได้ทำการเลือกใช้ TP-Link WN725N เป็นอุปกรณ์สร้างการเชื่อมต่อแบบ เครือข่ายไร้สายแบบเมช (“Wireless Mesh Network) ดังนั้นจึงเริ่มต้นจากการติดตั้งไดรเวอร์ของอุปกรณ์ลงบน Raspberry Pi โดยว่าอร์ชั่นที่ทำการดาวน์โหลด และทำการติดตั้งจะขึ้นอยู่กับเวอร์ชั่น ของ Raspbian OS โดยการใช้คำสั่ง uname -a เพื่อทำการตรวจสอบ kernel version จากนั้นทำการดาวน์โหลดไดรเวอร์ให้ตรงเวอร์ชั่น โดยสามารถตรวจสอบไฟล์ที่จะดาวน์โหลดได้จาก <http://www.raspberrypi.org/forum/viewtopic.php?p=462982>

คำยการใช้คำสั่งดังนี้ใน Terminal (ในที่นี้คือเวอร์ชั่น #827)

- a. wget <https://dl.dropboxusercontent.com/u/80256631/8188eu-20151113.tar.gz>
- b. tar -xvf 8188eu-20151113.tar.gz
- c. ./install.sh

2. ทำการรีสตาร์ทอุปกรณ์ และทำการต่อ TP-Link WN725N กับตัว Raspberry Pi ทำการใช้คำสั่งดังนี้ เพื่อ ตรวจสอบว่าตัว Adapter ไร้สาย ใช้งานได้แล้ว โดยจะมีอินเตอร์เฟส “wlan0” แสดงขึ้นมา

- a. Ifconfig

```
pi@raspberrypi ~ $ ifconfig
eth0      Link encap:Ethernet HWaddr b8:27:eb:02:b6:8e
          inet addr:192.168.1.41 Bcast:192.168.1.255 Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:4582 errors:0 dropped:0 overruns:0 frame:0
          TX packets:122 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:216040 (210.9 KiB) TX bytes:17515 (17.1 KiB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:40 errors:0 dropped:0 overruns:0 frame:0
          TX packets:40 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:3696 (3.6 KiB) TX bytes:3696 (3.6 KiB)

wlan0    Link encap:Ethernet HWaddr c4:e9:84:08:bd:45
          UP BROADCAST MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
```

รูปภาพที่ 4.1 : ภาพแสดงการเปิดใช้งานของพอร์ต wlan0

2) ทำการตั้งค่าเครือข่ายเซ็นเซอร์ไร้สาย ของ Raspberry Pi แต่ละ โหนดให้เป็น เครือข่ายเดียวกัน และเชื่อมต่อ กับระบบเครือข่ายภายในบ้าน

#### สำหรับทุกโหนด [11]

1. ทำการแก้ไขการตั้งค่าระบบเครือข่ายในไฟล์เดอร์ /etc/network/interface ด้วย

a. Sudo nano /etc/network/interface

b. ตั้งค่า wlan0 ให้เป็น ไอพีแบบ Static โดย ไอพีที่ตั้งต้องไม่ตรงกับ ไอพีแบบ Dynamic ที่ DHCP Server แจกมาจากการที่เตอร์

```
auto wlan0
#allow-hotplug wlan0
iface wlan0 inet static
    address 192.168.4.x
    netmask 255.255.255.0
    gateway 192.168.4.1
wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
```

2. ทำการแก้ไขไฟล์ /etc/wpa\_supplicant/wpa\_supplicant.conf โดยการเพิ่ม

network={

```
    ssid="homeauto" #ชื่อระบบเครือข่ายภายในบ้านที่ทำการเชื่อมต่อไปยังอินเตอร์เน็ต
    psk="raspberry"
```

}

3. ทำการ Reboot raspberry pi หรือใช้คำสั่ง sudo service networking restart จะพบว่าทุกโหนดที่ตั้งค่าแบบเดียวกัน สามารถ ping หากันได้แล้ว และสามารถเชื่อมต่อไปยังอินเทอร์เน็ต ได้เช่นกัน

3) ตั้งค่าการหาเส้นทางในเครือข่ายเซ็นเซอร์ไร้สาย ด้วยการค้นหาเส้นทางแบบ OLSR

1. ทำการติดตั้ง olsr ด้วยคำสั่ง

a. Sudo apt-get install olsrd

2. ทำการตั้งค่า configure ใน folder /etc/olsrd/olsrd.conf

a. Sudo nano /etc/olsrd/olsrd.conf

b. ตั้งค่าอินเทอร์เฟซที่จะใช้ olsrd เป็น wlan0

Interface "wlan0"{

Ip4Broadcast 255.255.255.255

}

c. ตั้งค่า Quality ของการ routing ที่ LinkQualityLevel ให้มีค่าเท่ากับ 2

3. ทำการตั้งค่าแต่ละ โหนด ให้สามารถ forward packet ได้

a. Sudo nano /etc/sysctl.conf

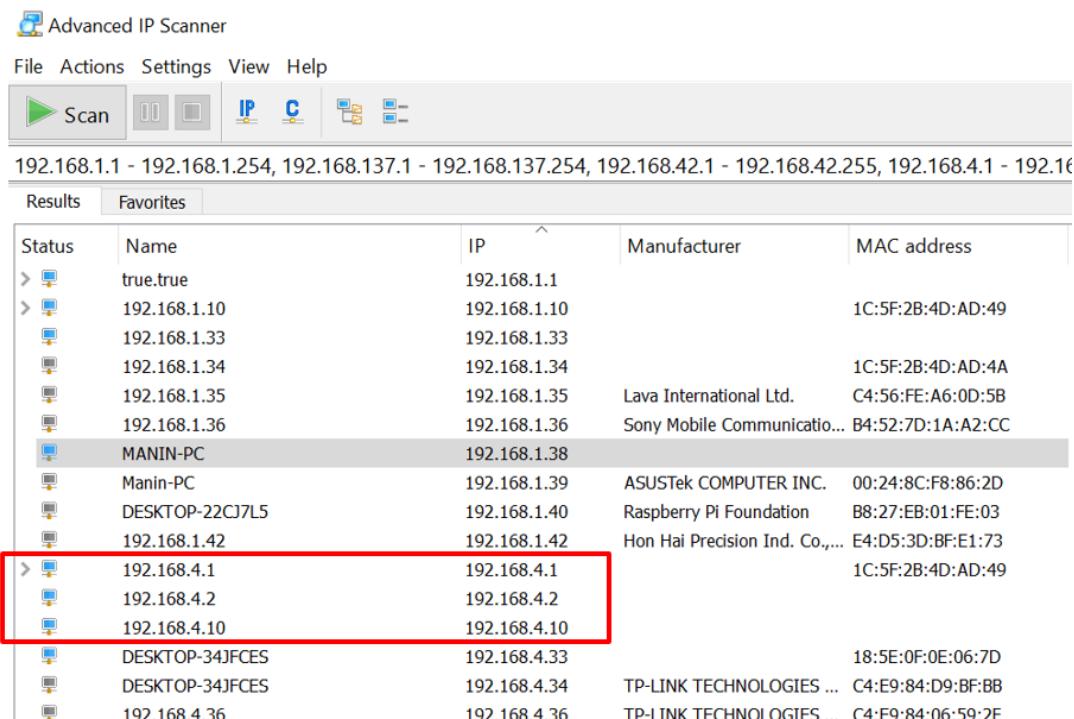
b. นำ comment บรรทัด net.ipv4.ip\_forward=1 ออก เพื่อตั้งค่าให้เริ่มทำงานตอน boot เครื่องทุกครั้ง

4. ทำการตั้งค่าให้ olsrd routing ทำงานทุกครั้งที่ปิดเครื่อง raspberry pi

a. Sudo nano /etc/rc.local

b. เพิ่มบรรทัด sudo olsrd

จากการที่ได้ตั้งค่าไอพีแบบ Static ของอุปกรณ์ทุกตัวให้อยู่ในวงเดียวกัน และผ่านการทำงานกับระบบเครือข่ายภายในบ้านเราที่เตอร์มีขั้นตอนการทำดังนี้



Status	Name	IP	Manufacturer	MAC address
>	true.true	192.168.1.1		
>	192.168.1.10	192.168.1.10		1C:5F:2B:4D:AD:49
	192.168.1.33	192.168.1.33		
	192.168.1.34	192.168.1.34		1C:5F:2B:4D:AD:4A
	192.168.1.35	192.168.1.35	Lava International Ltd.	C4:56:FE:A6:0D:5B
	192.168.1.36	192.168.1.36	Sony Mobile Communicatio...	B4:52:7D:1A:A2:CC
	MANIN-PC	192.168.1.38		
	Manin-PC	192.168.1.39	ASUSTek COMPUTER INC.	00:24:8C:F8:86:2D
	DESKTOP-22CJ7LS	192.168.1.40	Raspberry Pi Foundation	B8:27:EB:01:FE:03
	192.168.1.42	192.168.1.42	Hon Hai Precision Ind. Co.,...	E4:D5:3D:BF:E1:73
>	192.168.4.1	192.168.4.1		1C:5F:2B:4D:AD:49
	192.168.4.2	192.168.4.2		
	192.168.4.10	192.168.4.10		
	DESKTOP-34JFCES	192.168.4.33		18:5E:0F:0E:06:7D
	DESKTOP-34JFCES	192.168.4.34	TP-LINK TECHNOLOGIES ...	C4:E9:84:D9:BF:BB
	192.168.4.36	192.168.4.36	TP-LINK TECHNOLOGIES ...	C4:E9:84:06:59:2F

รูปภาพที่ 4.2 : การแสกนเครือข่ายในเครือข่ายไร้สายแบบเมฆ

จากการที่ได้จะเห็นว่า ไอพี 192.168.4.1 จะเป็น ไอพีของในวง 192.168.4.x ที่ได้ทำการตั้งค่าไว้ในแต่ล่ะไฟน์ในเครือข่าย และจะได้ผลการเชื่อมต่อเพื่อหาเส้นทาง ภายในระบบเครือข่าย ไร้สายแบบเมฆดังภาพที่ 4.3

```

SPP: insert candidate 192.168.4.10, cost 0.000
SPP: exploring node 192.168.4.10, cost 0.000
SPP:   exploring edge 192.168.4.2, cost 1.000
SPP: insert candidate 192.168.4.2, cost 1.000
SPP:   better path to 192.168.4.2, cost 1.000, via <none>, hops 1
SPP:   exploring edge 192.168.4.3, cost 1.000
SPP: insert candidate 192.168.4.3, cost 1.000
SPP:   better path to 192.168.4.3, cost 1.000, via <none>, hops 1
SPP: delete candidate 192.168.4.10, cost 0.000
SPP: append path 192.168.4.10, cost 0.000, via -
SPP: exploring node 192.168.4.2, cost 1.000
SPP:   exploring edge 192.168.4.3, cost 2.000
SPP:   exploring edge 192.168.4.10, cost 2.000
SPP: delete candidate 192.168.4.2, cost 1.000
SPP: append path 192.168.4.2, cost 1.000, via 192.168.4.2
SPP: exploring node 192.168.4.3, cost 1.000
SPP:   exploring edge 192.168.4.2, cost 2.000
SPP:   exploring edge 192.168.4.10, cost 2.000
SPP: delete candidate 192.168.4.3, cost 1.000
SPP: append path 192.168.4.3, cost 1.000, via 192.168.4.3
--- 22:04:59.629894 ----- DIJKSTRA

--- 22:04:59.630214 ----- LINKS
IP address      hyst      LQ      ETX
192.168.4.3    0.000  1.000/1.000  1.000
192.168.4.2    0.000  1.000/1.000  1.000

--- 22:04:59.632214 ----- TWO-HOP NEIGHBORS
IP addr (2-hop)  IP addr (1-hop)  Total cost
192.168.4.3    192.168.4.2    2.000
192.168.4.2    192.168.4.3    2.000

```

รูปภาพที่ 4.3 : แสดงการทำงาน OLSR Routing

## 4.2 ระบบการทำงานหลักของโหนดกล้องและโหนดฐานข้อมูล

### โหนดกล้อง(Camera Node)

ในระบบการทำงานหลักได้ทำการตั้งค่าโปรแกรมให้เริ่มทำงานตั้งแต่เริ่มการทำงาน ที่ /etc/rc.local โดยการทำงานจะมีการทำงานแบบอินเตอร์รัพท์เป็นหลัก โดยจะนำสัญญาณที่ได้จากอุปกรณ์เซ็นเซอร์ต่างๆมาทำการกระดูนฟังก์ชันอินเตอร์รัพท์และจะมีเพียงส่วนเดียวเท่านั้นที่จะไม่ทำงานแบบอินเตอร์รัพท์นั้นคือการทำงานของเซ็นเซอร์ตรวจจับอุณหภูมิและความชื้น โดยจะทำการเก็บข้อมูลทุกๆอุณหภูมิทุกๆ 10 วินาที ดังภาพที่ 4.4

```

pi@homeautocamera2 ~/Workingfile $ sudo python mainInterrupt.py
Temperature : 28 C
Humidity : 19 %
Cannot Connect To Database Server.
Connection Closed
Temperature : 28 C
Humidity : 19 %
Cannot Connect To Database Server.
Connection Closed
Error: 2
Error: 1
Temperature : 28 C
Humidity : 19 %
Cannot Connect To Database Server.
Connection Closed
^CYep

```

รูปภาพที่ 4.4 : แสดงการทำงานหลักของโหนดกล้องในขณะไม่มีอินเตอร์รัพท์

ในสภาวะปกติที่ไม่มีการส่งค่าอินเตอร์รัพท์เข้ามา ก็จะทำการอ่านค่าอุณหภูมิ และทำการบันทึกลงฐานข้อมูล ในโหนดรูปฐานข้อมูลแต่ถ้าหาก โหนดรูปฐานข้อมูลไม่ทำงาน ก็จะทำงานต่อไปได้ปกติ ในส่วนอื่นๆ ลักษณะต่อมาในส่วนของเซ็นเซอร์ตรวจจับการเคลื่อนไหว ได้ทำการเขียนโปรแกรมให้รับค่าแบบ อินเตอร์รัพท์ เมื่อตรวจพบและจะทำการเก็บภาพโดยใช้กล้องเว็บแคม นำมายังเครื่องที่ตรวจจับใบหน้าในภาพต่อไป ดังภาพที่ 4.5

```
pi@homeautocamera2 ~/Workingfile $ sudo python mainInterrupt.py
Error: 1
Motion Detected
/home/pi/Workingfile/motioncapture/2016-05-02_22:15:07.jpg
--- Opening /dev/video0...
Trying source module v4l2...
/dev/video0 opened.
No input was specified, using the first.
--- Capturing frame...
Skipping 15 frames...
Capturing 1 frames...
Corrupt JPEG data: 1 extraneous bytes before marker 0xd7
Captured 16 frames in 1.00 seconds. (16 fps)
--- Processing captured image...
Setting output format to JPEG, quality 95
Enabling text shadow.
Setting title "Surveillance Camera System".
Setting subtitle "Room 1".
Writing JPEG image to '/home/pi/Workingfile/motioncapture/2016-05-02_22:15:07.jpg'.
Found 0 faces
()
No face detect but motion in the area.
Send Notification to Email
Temperature : 27 C
Humidity : 19 %
Cannot Connect To Database Server.
Connection Closed
-> Uploading "/home/pi/Workingfile/motioncapture/2016-05-02_22:15:07.jpg" to "/EventImage/2016-05-02_22:15:07.jpg"... DONE
Error: 1
Cannot Connect to Database Server.
Connection Closed
Error: 1
Yep
```

รูปภาพที่ 4.5 : ภาพแสดงผลลัพธ์จากการตรวจจับการเคลื่อนไหว

เมื่อทำการตรวจจับการเคลื่อนไหว ได้จะทำการ แจ้งข้อความว่า “Motion Detected” และจะทำการเก็บภาพ นำมายังเครื่องที่ทางใบหน้า ถ้าหากไม่พบหรือเจอกับใบหน้าไม่รู้จัก ก็จะทำการส่งอีเมลล์ ซึ่งในส่วนนี้สามารถทำการตั้งค่าได้ในไฟล์ส่วนการตั้งค่าตั้งที่กล่าวไว้ในหัวข้อที่ 3.3.1 ข้อย่อยที่ 4 จากนั้นจะบันทึกข้อมูลลงโหนดรูปฐานข้อมูล และบันทึกข้อมูลลง Dropbox ต่อไป แต่ถ้าหากพบใบหน้าที่รู้จัก ก็จะทำการเก็บบนที่กักข้อมูลลงโหนดรูปฐานข้อมูลแต่เพียงอย่างเดียว

โดยจากการดำเนินงาน รูป่างของ โหนดกล้องจะเป็นดังภาพที่ 4.6



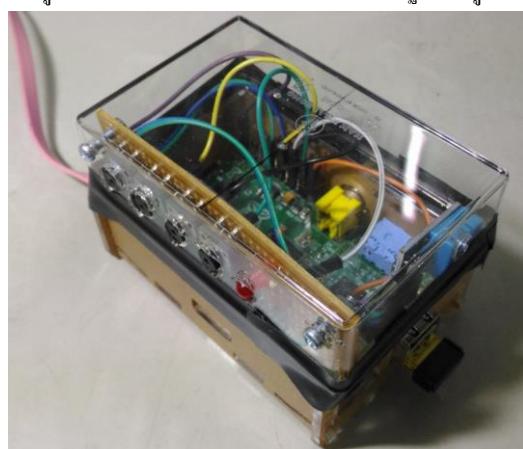
รูปภาพที่ 4.6 : ภาพของ โหนดกล้อง

### โหนดฐานข้อมูล (Database Node)

ในระบบการทำงานหลักของ โหนดฐานข้อมูล ในเบื้องต้นจะมีเพียงเซ็นเซอร์ตรวจจับอุณหภูมิและความชื้นอยู่เพียงอย่างเดียว โดยจะทำการเก็บค่าอุณหภูมิทุกๆ 5 วินาที และทุกๆนาทีจะมีการสำรองฐานข้อมูลไปยัง Dropbox ดังภาพที่ 4.7

```
pi@databasehomeauto ~/Workingfile $ sudo python main_system_database_node.py
Error: 1
Dump Database....
Storing Database to Dropbox....
> Uploading "/home/pi/Workingfile/homeauto_database.sql" to "/Database/homeauto_database.sql"... DONE
Error: 1
Error: 1
Error: 1
Error: 1
```

รูปภาพที่ 4.7 : ภาพแสดงผลการสำรองไฟล์ฐานข้อมูล



รูปภาพที่ 4.8 : ภาพของ โหนดฐานข้อมูล

### 4.3 ระบบจดจำใบหน้า (Face Recognition)

จากการเปรียบเทียบใบหน้ากับโครงสร้างข้อมูลในไฟล์นามสกุล XML พบว่าถ้าหากตรงกันก็จะทำการแจ้งว่ารู้จักใบหน้าดังภาพที่ 4.9 โดยเกณฑ์การตัดสินว่าเป็นภาพบุคคลเดียวกันหรือไม่จะดูที่ผลลัพธ์ที่ออกมายในรูปของลabe (Label) และค่าความคล้าย (Confidence) โดย ลabe คือส่วนที่ได้กำหนดไว้ก่อนหน้าว่าเป็นค่าโครงใบหน้าที่สอดคล้องกับใบหน้าในฐานข้อมูลใบหน้าใหม่นักที่สุด และค่าความคล้าย (Confidence) เปรียบเสมือนค่าความคล้ายที่ได้จากการเปรียบเทียบข้อมูล โดยตามปกติแล้วค่าความคล้าย (Confidence) ที่ดีจะอยู่ในช่วงต่ำกว่า 2000 สำหรับอัลกอริทึม Eigenface

```

--- Capturing frame...
Skipping 15 frames...
Capturing 1 frames...
Corrupt JPEG data: 3 extraneous bytes before marker 0xd7
Captured 16 frames in 1.00 seconds. (16 fps)
--- Processing captured image...
Setting output format to JPEG, quality 95
Enabling text shadow.
Setting title "Surveillance Camera System".
Setting subtitle "Room 1".
Writing JPEG image to '/home/pi/workfile/motioncapture/2015-11-15 22:30:15'
Found 1 faces
[1205 123 192 192]
Label 1 Confidence 1710.70480473
Recognized Face!
Motion Not Detected
Motion Not Detected

```

รูปภาพที่ 4.9 : แสดงผลลัพธ์การทำงานของระบบจดจำใบหน้า

#### ได้ทำการทดลองใช้อัลกอริทึมสำหรับตรวจสอบใบหน้านรูปภาพหลายแบบ

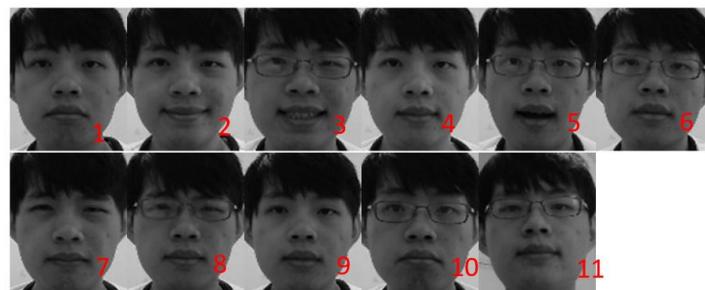
สำหรับเกณฑ์การตัดสินว่าใบหน้าที่นำมาเทียบนั้นมีความคล้ายกับฐานข้อมูลหรือไม่จะดูที่ค่าความคล้าย (Confidence) ที่ได้ออกมาจากการตรวจสอบดังภาพที่ 4.7 ข้างต้น สำหรับอัลกอริทึมที่ต่างกัน ค่าความคล้าย (Confidence) ที่เป็นตัวชี้วัดจะแตกต่างกันด้วย โดยสำหรับอัลกอริทึมที่นำมาทดลอง ค่า Threshold ตัวชี้วัดจะเป็นไปตามตารางนี้

Algorithm	ค่า Threshold ที่ถือว่าใบหน้าคล้ายใบหน้าในฐานข้อมูล
Eigenface Recognizer	<= 2000
Fisherface Recognizer	<= 2000
Local Binary Patterns	<= 120
Histograms Face Recognizer	

ตารางที่ 4.7 : แสดงผลลัพธ์การทำงานของระบบจดจำใบหน้า

ได้ผลลัพธ์การทดลองดังนี้

### 1. ฐานข้อมูลใบหน้าที่นำมาใช้



รูปภาพที่ 4.10 : ภาพฐานข้อมูลใบหน้าที่นำมาใช้ทดสอบ พร้อมค่า label ที่ได้ทำการตั้งไว้

### 2. รูปภาพที่นำมาใช้ทดสอบ



รูปภาพที่ 4.11 : ภาพที่นำมาทดสอบการตรวจสอบใบหน้า

### 3. ผลการทดสอบ

ภาพผลการทดลองเปรียบเทียบการใช้อัลกอริทึมแบบต่างๆ ทำการตรวจสอบใบหน้า

#### Eigenface Recognizer

ตรวจสอบด้วยใบหน้าเดิมกับฐานข้อมูลใบหน้า

```
pi@homeautocamera2 ~/Workingfile
Found 11 faces
[[ 8 198 132 132]
 [141 23 137 137]
 [275 18 145 145]
 [414 23 139 139]
 [553 22 140 140]
 [694 28 129 129]
 [143 196 134 134]
 [284 197 131 131]
 [557 193 133 133]
 [ 6 26 132 132]
 [417 194 139 139]]
Labels 2 conf 1425.1687439
Labels 9 conf 552.443890652
Labels 11 conf 1140.93353388
Labels 6 conf 1274.9120483
Labels 4 conf 690.492142865
Labels 10 conf 1054.87324473
Labels 6 conf 1113.34260923
Labels 1 conf 1344.44832433
Labels 7 conf 677.538726333
Labels 10 conf 1479.33430175
Labels 10 conf 582.626703275
```

รูปภาพที่ 4.12 : ผลการทดสอบใบหน้าด้วยใบหน้าเดิมกับฐานข้อมูลใบหน้าด้วย Eigenface Recognizer

### Fisherface Recognizer

ตรวจสอบด้วยใบหน้าเดียวกับฐานข้อมูลใบหน้า

```
pi@homeautocamera2 ~/Workingfile
Found 11 faces
[[ 8 198 132 132]
 [141 23 137 137]
 [275 18 145 145]
 [414 23 139 139]
 [553 22 140 140]
 [694 28 129 129]
 [143 196 134 134]
 [284 197 131 131]
 [557 193 133 133]
 [ 6 26 132 132]
 [417 194 139 139]]
Labels 2 conf 1407.92036763
Labels 9 conf 540.953952924
Labels 11 conf 1140.18022003
Labels 6 conf 1274.79279217
Labels 4 conf 531.175077546
Labels 10 conf 1053.70943175
Labels 6 conf 1081.77368772
Labels 1 conf 1258.82645829
Labels 7 conf 456.671325556
Labels 10 conf 1472.44723672
Labels 10 conf 582.570882182
```

รูปภาพที่ 4.13 : ผลการทดสอบใบหน้าด้วยใบหน้าเดียวกับฐานข้อมูลใบหน้าด้วย Fisherface Recognizer

### Local Binary Patterns Histograms Face Recognizer

ตรวจสอบด้วยใบหน้าเดียวกับฐานข้อมูลใบหน้า

```
Found 11 faces
[[ 8 198 132 132]
 [141 23 137 137]
 [275 18 145 145]
 [414 23 139 139]
 [553 22 140 140]
 [694 28 129 129]
 [143 196 134 134]
 [284 197 131 131]
 [557 193 133 133]
 [ 6 26 132 132]
 [417 194 139 139]]
Labels 8 conf 43.2568525188
Labels 9 conf 23.6173901352
Labels 2 conf 38.8145208827
Labels 8 conf 27.5699779502
Labels 4 conf 27.5428824052
Labels 11 conf 40.5681010413
Labels 6 conf 29.4140778443
Labels 3 conf 31.637199275
Labels 7 conf 32.4617743815
Labels 1 conf 25.9272388421
Labels 10 conf 28.895988274
```

รูปภาพที่ 4.14 : ผลการทดสอบใบหน้าด้วยใบหน้าเดียวกับฐานข้อมูลใบหน้าด้วย Local Binary Patterns Histograms Face Recognizer

### Eigenface Recognizer

ตรวจสอบด้วยใบหน้าของผู้ที่ไม่ใช่สมาชิกของฐานข้อมูลใบหน้า

#### ภาพ A



```
pi@homeautocamera2 ~/Workingfil
Found 16 faces
[[467 31 45 45]
 [328 56 50 50]
 [195 83 57 57]
 [409 113 59 59]
 [507 120 62 62]
 [251 168 55 55]
 [35 99 64 64]
 [453 272 65 65]
 [439 440 73 73]
 [319 194 69 69]
 [554 194 68 68]
 [199 171 67 67]
 [153 243 77 77]
 [275 458 76 76]
 [382 327 78 78]
 [155 356 80 80]]
Labels 8 conf 3539.09701771
Labels 7 conf 3042.61896744
Labels 8 conf 3387.06911873
Labels 7 conf 4343.86136421
Labels 7 conf 4656.67774503
Labels 7 conf 4127.37300304
Labels 5 conf 4320.5811135
Labels 8 conf 3610.86078145
Labels 8 conf 4450.08085213
Labels 7 conf 3096.45583713
Labels 7 conf 3631.63159107
Labels 7 conf 3492.68685988
Labels 7 conf 3040.56043187
Labels 7 conf 2643.43768438
Labels 8 conf 2691.56830277
Labels 7 conf 3079.08611782
```

รูปภาพที่ 4.15 : ผลการทดสอบใบหน้าด้วยชุด  
ใบหน้า A ด้วย Eigenface Recognizer

#### ภาพ B



```
pi@homeautocamera2 ~/Workingfil
Found 11 faces
[[541 193 41 41]
 [391 198 45 45]
 [479 227 43 43]
 [286 280 41 41]
 [754 252 53 53]
 [561 260 50 50]
 [458 331 53 53]
 [547 395 55 55]
 [278 366 59 59]
 [170 359 71 71]
 [387 391 73 73]]
Labels 8 conf 2290.60990791
Labels 7 conf 1810.30050254
Labels 7 conf 1700.56609488
Labels 7 conf 2088.50899829
Labels 7 conf 1819.5371585
Labels 7 conf 2273.90711728
Labels 7 conf 2158.4585702
Labels 7 conf 2313.17118827
Labels 7 conf 2122.938211
Labels 8 conf 2529.31822311
Labels 7 conf 2299.19355378
```

รูปภาพที่ 4.16 : ผลการทดสอบใบหน้าด้วยชุด  
ใบหน้า B ด้วย Eigenface Recognizer

### Fisherface Recognizer

ตรวจสอบด้วยใบหน้าของผู้ที่ไม่ใช่สมาชิกของฐานข้อมูลใบหน้า

#### ภาพ A



```
pi@homeautocamera2 ~/Workingfil
Found 16 faces
[[467 31 45 45]
[328 56 50 50]
[195 83 57 57]
[409 113 59 59]
[507 120 62 62]
[251 168 55 55]
[35 99 64 64]
[453 272 65 65]
[439 440 73 73]
[319 194 69 69]
[554 194 68 68]
[199 171 67 67]
[153 243 77 77]
[275 458 76 76]
[382 327 78 78]
[155 356 80 80]]
Labels 8 conf 2809.57399656
Labels 7 conf 2939.52981973
Labels 8 conf 3083.81140468
Labels 8 conf 3920.03110122
Labels 8 conf 4399.41999786
Labels 7 conf 4092.79735392
Labels 8 conf 3637.77123431
Labels 8 conf 3422.1344472
Labels 8 conf 4051.64299823
Labels 7 conf 3063.85889256
Labels 8 conf 3288.58121111
Labels 7 conf 3381.54973258
Labels 8 conf 3038.57183401
Labels 7 conf 2181.78003404
Labels 7 conf 2452.51816995
Labels 7 conf 2760.95136832
```

รูปภาพที่ 4.17 : ผลการทดสอบใบหน้าด้วยชุดใบหน้า A ด้วย Fisherface Recognizer

#### ภาพ B



```
pi@homeautocamera2 ~/Workingfi
Found 11 faces
[[541 193 41 41]
[391 198 45 45]
[479 227 43 43]
[286 280 41 41]
[754 252 53 53]
[561 260 50 50]
[458 331 53 53]
[547 395 55 55]
[278 366 59 59]
[170 359 71 71]
[387 391 73 73]]
Labels 8 conf 1761.82830011
Labels 7 conf 1289.4515532
Labels 7 conf 1699.11224204
Labels 7 conf 1781.19228786
Labels 7 conf 1306.00038871
Labels 7 conf 2266.88709164
Labels 7 conf 1591.38130794
Labels 7 conf 2288.30660727
Labels 7 conf 1397.14207328
Labels 7 conf 2097.11128465
Labels 7 conf 1989.34875567
```

รูปภาพที่ 4.18 : ผลการทดสอบใบหน้าด้วยชุดใบหน้า B ด้วย Fisherface Recognizer

### Local Binary Patterns Histograms Face Recognizer

ตรวจสอบด้วยใบหน้าของผู้ที่ไม่ใช่สมาชิกของฐานข้อมูลใบหน้า

#### ภาพ A



```
pi@homeautocamera2 ~/Workingfil
Found 16 faces
[[467 31 45 45]
 [328 56 50 50]
 [195 83 57 57]
 [409 113 59 59]
 [507 120 62 62]
 [251 168 55 55]
 [ 35 99 64 64]
 [453 272 65 65]
 [439 440 73 73]
 [319 194 69 69]
 [554 194 68 68]
 [199 171 67 67]
 [153 243 77 77]
 [275 458 76 76]
 [382 327 78 78]
 [155 356 80 80]]
Labels 4 conf 260.140161727
Labels 2 conf 198.732862605
Labels 5 conf 149.820875705
Labels 5 conf 154.697677267
Labels 8 conf 131.38185251
Labels 2 conf 191.976638804
Labels 5 conf 150.366894198
Labels 3 conf 144.771802008
Labels 3 conf 148.292686677
Labels 9 conf 161.451195853
Labels 8 conf 129.566491672
Labels 9 conf 149.407416282
Labels 3 conf 145.876521103
Labels 9 conf 139.599387883
Labels 5 conf 128.001685572
Labels 2 conf 139.91248649
```

รูปภาพที่ 4.19 : ผลการทดสอบใบหน้าด้วยชุดใบหน้า A ด้วย Local Binary Patterns Histograms Face Recognizer

#### ภาพ B



```
pi@homeautocamera2 ~/Workingfil
Found 11 faces
[[541 193 41 41]
 [391 198 45 45]
 [479 227 43 43]
 [286 280 41 41]
 [754 252 53 53]
 [561 260 50 50]
 [458 331 53 53]
 [547 395 55 55]
 [278 366 59 59]
 [170 359 71 71]
 [387 391 73 73]]
Labels 3 conf 198.233991543
Labels 7 conf 122.781864546
Labels 3 conf 280.851880521
Labels 6 conf 152.794561018
Labels 7 conf 210.120977706
Labels 5 conf 189.948264495
Labels 5 conf 211.312011458
Labels 7 conf 228.987066124
Labels 5 conf 143.931328263
Labels 5 conf 183.034053921
Labels 5 conf 145.657497306
```

รูปภาพที่ 4.20 : ผลการทดสอบใบหน้าด้วยชุดใบหน้า B ด้วย Local Binary Patterns Histograms Face Recognizer

### สรุปการเปรียบเทียบทั้ง 3 อัลกอริทึม

จากการทดสอบจะเห็นว่าอัลกอริทึมให้ผลลัพธ์ที่แบ่งแยกเจนมากที่สุด กืออัลกอริทึม

Local Binary Patterns Histograms Face Recognizer จะเห็นว่า เมื่อนำภาพที่ไม่ใช่สมาชิกฐานข้อมูลมาตรวจสอบ จะมีค่าความคล้าย (Confidence) อยู่ในช่วงไม่น้อยกว่า 120 ซึ่งแตกต่างจากอัลกอริทึมแบบอื่นๆ ที่มีช่วงค่าความคล้าย (Confidence) ที่ทับซ้อนกันด้วย

ทดสอบการใช้งานจริงด้วยใบหน้าของบุคคลที่มีข้อมูลใบหน้าอยู่ในฐานข้อมูลจำนวน 100 ครั้ง ได้ผลดังนี้

ลำดับครั้งของการทดสอบ	ค่า Confidence
1	64.03503489
2	120.3220566327
3	94.133856017
4	78.727232339
5	85.6865547382
6	105.889100623
7	91.2259819266
8	89.4934054288
9	81.4218142772
10	97.4395928283
11	147.317046828
12	100.542674114
13	90.4650660807
14	83.8707889019
15	95.4044803986
16	105.8626386
17	96.8875963452
18	80.5827444855
19	100.077555134
20	92.6672707849
21	94.6712134078
22	137.318021797
23	90.3023422468
24	177.9715187
25	103.027885355
26	123.8237181945
27	88.6418824145
28	96.2966292238
29	164.927751022
30	103.051189745
31	91.1713439809
32	117.443729202

33	127.907906203
34	148.077320082
35	115.091402983
36	71.2820663421
37	81.0421887285
38	69.9695708211
39	80.9067233633
40	70.8206227133
41	72.2350939428
42	87.7987913512
43	83.7987913513
44	81.6264109307
45	96.7626677026
46	167.513041239
47	82.9977857825
48	102.696681956
49	88.1332722826
50	80.2531735747
51	82.5570933565
52	75.9332010149
53	71.9082440511
54	74.4866895833
55	86.5006922475
56	66.6159383356
57	73.4672236739
58	66.8300808304
59	62.959106291
60	64.1806964442
61	84.4331151619
62	67.7822161626
63	66.8812569598
64	63.3416914473
65	66.5305971381
66	74.9785677221
67	77.1162687582
68	65.4376693505
69	69.7517766499
70	70.2891206922
71	72.4032139041
72	76.0871615412
73	72.9690099135

74	76.4151166825	
75	77.0562451361	
76	65.0102950211	
77	63.9080751517	
78	132.207990401	
79	108.052808844	
80	83.8950145598	
81	104.05474594	
82	70.043806176	
83	88.1722554764	
84	76.4068588696	
85	73.8706346363	
86	78.4351983608	
87	87.718187307	
88	79.5613458647	
89	79.3967648721	
90	94.7031721468	
91	65.5109725098	
92	82.8075421859	
93	77.2258494517	
94	73.2701736727	
95	68.4048437649	
96	66.58591899	
97	79.677836581	
98	71.8712321863	
99	68.6866125089	
100	70.2245663236	
ค่าความคล้าย (Confidence) น้อยกว่า 120	91	ครึ่ง
ค่าความคล้าย (Confidence) มากกว่า 120	9	ครึ่ง
สรุปความถูกต้องโดยเฉลี่ย	91	เปอร์เซ็นต์

ตารางที่ 4.1 : ผลลัพธ์การทดสอบด้วย LHDB Face Recognition

ทดสอบการใช้งานด้วยใบหน้าของบุคคลที่ไม่ได้เป็นสมาชิกของระบบฐานข้อมูลใบหน้า



รูปภาพที่ 4.21 : ตัวอย่างภาพใบหน้าของผู้ที่ไม่ใช่สมาชิกที่นำมาทดสอบ ก้าพที่ 1

ที่มา : [http://24.media.tumblr.com/tumblr\\_m6oxghQKOw1r4mt8qo1\\_1280.jpg](http://24.media.tumblr.com/tumblr_m6oxghQKOw1r4mt8qo1_1280.jpg)

จากภาพที่ 4.21 นี้ใบหน้าทั้งหมด 72 ใบหน้า สามารถตรวจจับใบหน้าและทำการวิเคราะห์ได้ทั้งหมด 71 ใบหน้า  
ได้ค่าความคล้ายดังภาพดังต่อไปนี้

```
pi@homeautocamera2 ~/Workingfile $ sudo python testrec.py
/home/pi/Workingfile/10.jpg
Face: 1 Confidence: 557.861927519 -----Incorrectly Recognized!!!!-----
Face: 2 Confidence: 564.333622945 -----Incorrectly Recognized!!!!-----
Face: 3 Confidence: 377.13600677 -----Incorrectly Recognized!!!!-----
Face: 4 Confidence: 470.862372327 -----Incorrectly Recognized!!!!-----
Face: 5 Confidence: 321.342885692 -----Incorrectly Recognized!!!!-----
Face: 6 Confidence: 525.989561677 -----Incorrectly Recognized!!!!-----
Face: 7 Confidence: 479.429551343 -----Incorrectly Recognized!!!!-----
Face: 8 Confidence: 409.15628557 -----Incorrectly Recognized!!!!-----
Face: 9 Confidence: 371.743805535 -----Incorrectly Recognized!!!!-----
Face: 10 Confidence: 419.449647518 -----Incorrectly Recognized!!!!-----
Face: 11 Confidence: 274.832916225 -----Incorrectly Recognized!!!!-----
Face: 12 Confidence: 261.41191908 -----Incorrectly Recognized!!!!-----
Face: 13 Confidence: 288.422901251 -----Incorrectly Recognized!!!!-----
Face: 14 Confidence: 297.279216383 -----Incorrectly Recognized!!!!-----
Face: 15 Confidence: 387.290335679 -----Incorrectly Recognized!!!!-----
Face: 16 Confidence: 363.225097943 -----Incorrectly Recognized!!!!-----
Face: 17 Confidence: 423.992574915 -----Incorrectly Recognized!!!!-----
Face: 18 Confidence: 345.230439308 -----Incorrectly Recognized!!!!-----
Face: 19 Confidence: 306.157820573 -----Incorrectly Recognized!!!!-----
Face: 20 Confidence: 341.788724599 -----Incorrectly Recognized!!!!-----
Face: 21 Confidence: 494.915761518 -----Incorrectly Recognized!!!!-----
Face: 22 Confidence: 373.357167637 -----Incorrectly Recognized!!!!-----
Face: 23 Confidence: 391.972843745 -----Incorrectly Recognized!!!!-----
Face: 24 Confidence: 341.690424564 -----Incorrectly Recognized!!!!-----
Face: 25 Confidence: 311.387234366 -----Incorrectly Recognized!!!!-----
Face: 26 Confidence: 279.858894013 -----Incorrectly Recognized!!!!-----
Face: 27 Confidence: 401.677135724 -----Incorrectly Recognized!!!!-----
Face: 28 Confidence: 227.292983577 -----Incorrectly Recognized!!!!-----
Face: 29 Confidence: 307.48183698 -----Incorrectly Recognized!!!!-----
Face: 30 Confidence: 274.724456524 -----Incorrectly Recognized!!!!-----
Face: 31 Confidence: 331.294636104 -----Incorrectly Recognized!!!!-----
Face: 32 Confidence: 410.043366897 -----Incorrectly Recognized!!!!-----
Face: 33 Confidence: 275.601123262 -----Incorrectly Recognized!!!!-----
Face: 34 Confidence: 262.536022296 -----Incorrectly Recognized!!!!-----
Face: 35 Confidence: 295.66570542 -----Incorrectly Recognized!!!!-----
Face: 36 Confidence: 344.73838647 -----Incorrectly Recognized!!!!-----
Face: 37 Confidence: 254.596903008 -----Incorrectly Recognized!!!!-----
Face: 38 Confidence: 267.45404793 -----Incorrectly Recognized!!!!-----
Face: 39 Confidence: 224.080577009 -----Incorrectly Recognized!!!!-----
Face: 40 Confidence: 227.53920607 -----Incorrectly Recognized!!!!-----
Face: 41 Confidence: 250.123429615 -----Incorrectly Recognized!!!!-----
Face: 42 Confidence: 307.551693036 -----Incorrectly Recognized!!!!-----
Face: 43 Confidence: 188.975312441 -----Incorrectly Recognized!!!!-----
Face: 44 Confidence: 360.704637571 -----Incorrectly Recognized!!!!-----
Face: 45 Confidence: 283.802094379 -----Incorrectly Recognized!!!!-----
Face: 46 Confidence: 313.203076636 -----Incorrectly Recognized!!!!-----
Face: 47 Confidence: 254.245193982 -----Incorrectly Recognized!!!!-----
Face: 48 Confidence: 263.504817161 -----Incorrectly Recognized!!!!-----
Face: 49 Confidence: 255.691399122 -----Incorrectly Recognized!!!!-----
Face: 50 Confidence: 236.086709384 -----Incorrectly Recognized!!!!-----
Face: 51 Confidence: 260.799485794 -----Incorrectly Recognized!!!!-----
Face: 52 Confidence: 246.294620231 -----Incorrectly Recognized!!!!-----
Face: 53 Confidence: 236.610790347 -----Incorrectly Recognized!!!!-----
Face: 54 Confidence: 199.246966872 -----Incorrectly Recognized!!!!-----
Face: 55 Confidence: 268.124784676 -----Incorrectly Recognized!!!!-----
Face: 56 Confidence: 263.712553904 -----Incorrectly Recognized!!!!-----
Face: 57 Confidence: 242.098126067 -----Incorrectly Recognized!!!!-----
Face: 58 Confidence: 201.698951253 -----Incorrectly Recognized!!!!-----
Face: 59 Confidence: 285.828336623 -----Incorrectly Recognized!!!!-----
Face: 60 Confidence: 215.935185297 -----Incorrectly Recognized!!!!-----
Face: 61 Confidence: 241.699470975 -----Incorrectly Recognized!!!!-----
Face: 62 Confidence: 234.406511413 -----Incorrectly Recognized!!!!-----
Face: 63 Confidence: 275.718493718 -----Incorrectly Recognized!!!!-----
Face: 64 Confidence: 199.410981812 -----Incorrectly Recognized!!!!-----
Face: 65 Confidence: 292.926067142 -----Incorrectly Recognized!!!!-----
Face: 66 Confidence: 209.835251986 -----Incorrectly Recognized!!!!-----
Face: 67 Confidence: 259.098344674 -----Incorrectly Recognized!!!!-----
Face: 68 Confidence: 193.478734792 -----Incorrectly Recognized!!!!-----
Face: 69 Confidence: 195.39744368 -----Incorrectly Recognized!!!!-----
Face: 70 Confidence: 260.731955212 -----Incorrectly Recognized!!!!-----
Face: 71 Confidence: 162.042154542 -----Incorrectly Recognized!!!!-----
```

รูปภาพที่ 4.22 : ผลลัพธ์จากการตรวจสอบใบหน้าในภาพที่ 4.21

พบว่า ค่าความคล้ายทั้งหมดที่ได้อ่านในช่วงที่มีค่ามากกว่า 120 ทั้งหมด แสดงว่าใบหน้าทั้งหมดที่ตรวจสอบได้ไม่ใช่  
ใบหน้าของสมาชิกในฐานข้อมูลใบหน้าที่มีอยู่



รูปภาพที่ 4.23 : ตัวอย่างภาพใบหน้าของผู้ที่ไม่ใช่สมาชิกที่นำมาทดสอบ ภาพที่ 2

ที่มา : [http://koratstartup.com/wp-content/uploads/2015/12/3464\\_1uniqlo\\_portraits\\_squares.jpg](http://koratstartup.com/wp-content/uploads/2015/12/3464_1uniqlo_portraits_squares.jpg)

จากการที่ 4.23 มีใบหน้าทั้งหมด 23 ใบหน้า สามารถตรวจจับใบหน้าและทำการวิเคราะห์ได้ทั้งหมด 27 ใบหน้า ได้ค่าความคล้ายดังภาพดังต่อไปนี้

```
pi@homeautocamera2:~/Workingfile$ sudo python testrec.py
/home/pi/Workingfile/13.jpg
Face: 1 Confidence: 628.504140107 -----Incorrectly Recognized!!!!!
Face: 2 Confidence: 519.852118138 -----Incorrectly Recognized!!!!!
Face: 3 Confidence: 560.473793715 -----Incorrectly Recognized!!!!!
Face: 4 Confidence: 425.066324659 -----Incorrectly Recognized!!!!!
Face: 5 Confidence: 374.321965208 -----Incorrectly Recognized!!!!!
Face: 6 Confidence: 440.458962802 -----Incorrectly Recognized!!!!!
Face: 7 Confidence: 378.809564757 -----Incorrectly Recognized!!!!!
Face: 8 Confidence: 468.969466877 -----Incorrectly Recognized!!!!!
Face: 9 Confidence: 330.796410233 -----Incorrectly Recognized!!!!!
Face: 10 Confidence: 410.352563687 -----Incorrectly Recognized!!!!!
Face: 11 Confidence: 835.978687132 -----Incorrectly Recognized!!!!!
Face: 12 Confidence: 529.226316017 -----Incorrectly Recognized!!!!!
Face: 13 Confidence: 380.037610848 -----Incorrectly Recognized!!!!!
Face: 14 Confidence: 631.779400605 -----Incorrectly Recognized!!!!!
Face: 15 Confidence: 613.381020568 -----Incorrectly Recognized!!!!!
Face: 16 Confidence: 314.199238296 -----Incorrectly Recognized!!!!!
Face: 17 Confidence: 347.932074474 -----Incorrectly Recognized!!!!!
Face: 18 Confidence: 474.238932355 -----Incorrectly Recognized!!!!!
Face: 19 Confidence: 468.921497173 -----Incorrectly Recognized!!!!!
Face: 20 Confidence: 246.121026259 -----Incorrectly Recognized!!!!!
Face: 21 Confidence: 420.157186249 -----Incorrectly Recognized!!!!!
Face: 22 Confidence: 870.742114069 -----Incorrectly Recognized!!!!!
Face: 23 Confidence: 331.945220669 -----Incorrectly Recognized!!!!!
Face: 24 Confidence: 353.567070707 -----Incorrectly Recognized!!!!!
Face: 25 Confidence: 294.937861259 -----Incorrectly Recognized!!!!!
Face: 26 Confidence: 332.78237339 -----Incorrectly Recognized!!!!!
Face: 27 Confidence: 962.763777769 -----Incorrectly Recognized!!!!!
```

รูปภาพที่ 4.24 : ผลลัพธ์จากการตรวจสอบใบหน้าในภาพที่ 4.23

พบว่า ค่าความคล้ายทั้งหมดที่ได้ออกมายืนยันช่วงที่มีค่ามากกว่า 120 ทั้งหมด แสดงว่าใบหน้าทั้งหมดที่ตรวจสอบได้ไม่ใช่ใบหน้าของสมาชิกในฐานข้อมูลใบหน้าที่มีอยู่



รูปภาพที่ 4.25 : ตัวอย่างภาพใบหน้าของผู้ที่ไม่ใช่สมาชิกที่นำมาทดสอบ ภาพที่ 3

ที่มา : [http://thumb101.shutterstock.com/display\\_pic\\_with\\_logo/172762/277752143/stock-photo-collage-of-portrait-of-many-smiling-faces-277752143.jpg](http://thumb101.shutterstock.com/display_pic_with_logo/172762/277752143/stock-photo-collage-of-portrait-of-many-smiling-faces-277752143.jpg)

จากภาพที่ 4.25 มีใบหน้าทั้งหมด 25 ใบหน้า สามารถตรวจจับใบหน้าและทำการวิเคราะห์ได้ทั้งหมด 26 ใบหน้า  
ได้ค่าความคล้ายดังภาพดังต่อไปนี้

```
pi@homeautocamera2 ~/Workingfile $ sudo python testrec.py
/home/pi/Workingfile/4.jpg
Face: 1 Confidence: 252.343824636 -----Incorrectly Recognized!!!!-----
Face: 2 Confidence: 195.518537179 -----Incorrectly Recognized!!!!-----
Face: 3 Confidence: 223.616633507 -----Incorrectly Recognized!!!!-----
Face: 4 Confidence: 160.779293785 -----Incorrectly Recognized!!!!-----
Face: 5 Confidence: 153.297023469 -----Incorrectly Recognized!!!!-----
Face: 6 Confidence: 308.540322405 -----Incorrectly Recognized!!!!-----
Face: 7 Confidence: 190.726841102 -----Incorrectly Recognized!!!!-----
Face: 8 Confidence: 182.810746991 -----Incorrectly Recognized!!!!-----
Face: 9 Confidence: 139.167254798 -----Incorrectly Recognized!!!!-----
Face: 10 Confidence: 166.854685595 -----Incorrectly Recognized!!!!-----
Face: 11 Confidence: 209.781536242 -----Incorrectly Recognized!!!!-----
Face: 12 Confidence: 188.06981969 -----Incorrectly Recognized!!!!-----
Face: 13 Confidence: 277.855541049 -----Incorrectly Recognized!!!!-----
Face: 14 Confidence: 155.77283098 -----Incorrectly Recognized!!!!-----
Face: 15 Confidence: 156.820157922 -----Incorrectly Recognized!!!!-----
Face: 16 Confidence: 140.249165768 -----Incorrectly Recognized!!!!-----
Face: 17 Confidence: 198.627203261 -----Incorrectly Recognized!!!!-----
Face: 18 Confidence: 242.811829781 -----Incorrectly Recognized!!!!-----
Face: 19 Confidence: 168.685176725 -----Incorrectly Recognized!!!!-----
Face: 20 Confidence: 180.300181484 -----Incorrectly Recognized!!!!-----
Face: 21 Confidence: 145.401755438 -----Incorrectly Recognized!!!!-----
Face: 22 Confidence: 211.756000536 -----Incorrectly Recognized!!!!-----
Face: 23 Confidence: 198.627369051 -----Incorrectly Recognized!!!!-----
Face: 24 Confidence: 236.979690824 -----Incorrectly Recognized!!!!-----
Face: 25 Confidence: 159.509956371 -----Incorrectly Recognized!!!!-----
Face: 26 Confidence: 156.589342272 -----Incorrectly Recognized!!!!-----
```

รูปภาพที่ 4.26 : ผลลัพธ์จากการตรวจสอบใบหน้าในภาพที่ 4.25

พบว่า ค่าความคล้ายทั้งหมดที่ได้อ่านในช่วงที่มีค่ามากกว่า 120 ทั้งหมด แสดงว่าใบหน้าทั้งหมดที่ตรวจสอบได้ไม่ใช่  
ใบหน้าของสมาชิกในฐานข้อมูลใบหน้าที่มีอยู่



รูปภาพที่ 4.27 : ตัวอย่างภาพใบหน้าของผู้ที่ไม่ใช่สมาชิกที่นำมาทดสอบ ภาพที่ 4

ที่มา : <http://previews.123rf.com/images/alphaspirit/alphaspirit1506/alphaspirit150600034/40435390-Pортraits-collage-of-people-faces-who-scream-Stock-Photo-collage.jpg>

จากภาพที่ 4.27 มีใบหน้าทั้งหมด 25 ใบหน้า สามารถตรวจจับใบหน้าและทำการวิเคราะห์ได้ทั้งหมด 25 ใบหน้า ได้ค่าความคล้ายดังภาพดังต่อไปนี้

```
pi@homeautocamera2:~/Workingfile$ sudo python testrec.py
/home/pi/Workingfile/14.jpg
Face: 1 Confidence: 422.073372471 -----Incorrectly Recognized!!!
Face: 2 Confidence: 512.073308699 -----Incorrectly Recognized!!!
Face: 3 Confidence: 259.183318694 -----Incorrectly Recognized!!!
Face: 4 Confidence: 276.649592279 -----Incorrectly Recognized!!!
Face: 5 Confidence: 159.688551481 -----Incorrectly Recognized!!!
Face: 6 Confidence: 251.567680906 -----Incorrectly Recognized!!!
Face: 7 Confidence: 185.534247071 -----Incorrectly Recognized!!!
Face: 8 Confidence: 233.162883405 -----Incorrectly Recognized!!!
Face: 9 Confidence: 312.5253266 -----Incorrectly Recognized!!!
Face: 10 Confidence: 162.558259815 -----Incorrectly Recognized!!!
Face: 11 Confidence: 231.090073021 -----Incorrectly Recognized!!!
Face: 12 Confidence: 204.581463343 -----Incorrectly Recognized!!!
Face: 13 Confidence: 182.701355892 -----Incorrectly Recognized!!!
Face: 14 Confidence: 145.586728995 -----Incorrectly Recognized!!!
Face: 15 Confidence: 192.989962556 -----Incorrectly Recognized!!!
Face: 16 Confidence: 173.947244214 -----Incorrectly Recognized!!!
Face: 17 Confidence: 164.675155506 -----Incorrectly Recognized!!!
Face: 18 Confidence: 251.828343081 -----Incorrectly Recognized!!!
Face: 19 Confidence: 209.285925781 -----Incorrectly Recognized!!!
Face: 20 Confidence: 186.143674202 -----Incorrectly Recognized!!!
Face: 21 Confidence: 156.126154378 -----Incorrectly Recognized!!!
Face: 22 Confidence: 218.212232924 -----Incorrectly Recognized!!!
Face: 23 Confidence: 215.195004127 -----Incorrectly Recognized!!!
Face: 24 Confidence: 198.329444662 -----Incorrectly Recognized!!!
Face: 25 Confidence: 210.041065086 -----Incorrectly Recognized!!!
```

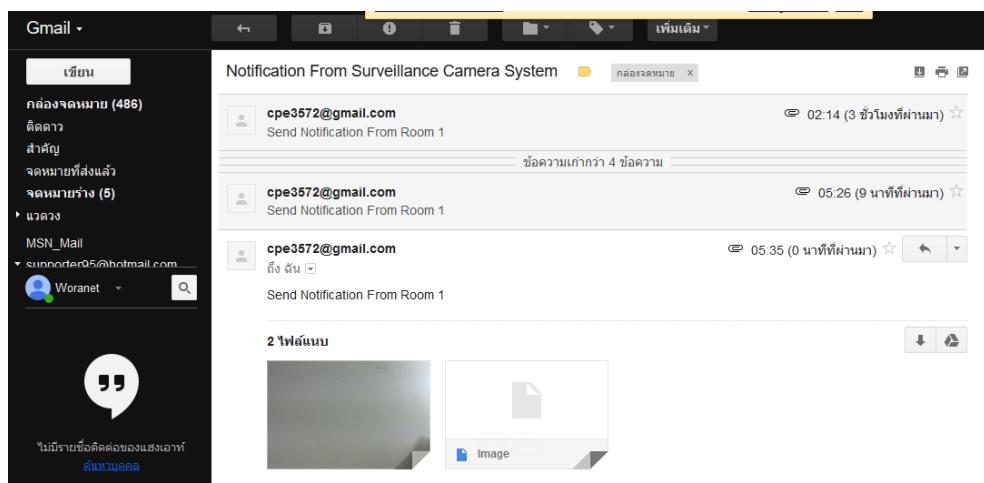
รูปภาพที่ 4.28 : ผลลัพธ์จากการตรวจสอบใบหน้าในภาพที่ 4.27

พบว่า ค่าความคล้ายทั้งหมดที่ได้ออกมามีค่ามากกว่า 120 ทั้งหมด แสดงว่าใบหน้าทั้งหมดที่ตรวจสอบได้ไม่ใช่ใบหน้าของสมาชิกในฐานข้อมูลใบหน้าที่มีอยู่

## สรุป

จากการทดสอบพบว่าอัตราเฉลี่ยความถูกต้องของการตรวจสอบใบหน้าสำหรับใบหน้าสมาชิก มีความแม่นยำอยู่ที่ 91 เปอร์เซ็นต์ และการตรวจสอบใบหน้าของผู้ที่ไม่ใช่สมาชิกมีความแม่นยำอยู่ที่ 100 เปอร์เซ็นต์ นอกเหนือนี้ยังพัฒนาการตรวจจับใบหน้า (Face Detection) จากภาพมากกว่าใบหน้าที่พนจริงๆ ซึ่งสามารถมาจากการที่ภายในภาพมีส่วนที่ทำให้อัลกอริทึมสำหรับการตรวจจับใบหน้า (Face Detection) คาดว่าเป็นใบหน้า ออาทิตย์ เดือนพฤษภาคม 2562 ถึงเดือนมิถุนายน 2562 ที่ดำเนินการ

## 4.4 ระบบการแจ้งเตือนทางอีเมลล์ (Email Notification)



รูปภาพที่ 4.29 : ภาพแสดงการแจ้งเตือนจากการแจ้งเตือนมาทางอีเมลล์

จากการเขียนโค้ดส่งข้อมูลด้วยโปรโตคอล SMTP มีการแจ้งเตือนทั้งในกรณีที่ตรวจพบใบหน้าที่ไม่คล้ายกับฐานข้อมูล และกรณีที่ตรวจพบการเคลื่อนไหวแต่ไม่สามารถจับภาพได้ชั่วขณะ ก็จะทำการส่งอีเมลล์ของระบบที่ได้สร้างเตรียมไว้ นماซึ่ง อีเมลล์ปลายทาง ในที่นี้ อีเมลล์ของระบบ คือ cpe3572@gmail.com และ อีเมลล์ปลายทางคือ tienworanet@gmail.com

## 4.5 ระบบฐานข้อมูลบนโภนดฐานข้อมูลของ Raspberry Pi

### วิธีการติดตั้งซอฟต์แวร์ และการตั้งค่าที่จำเป็น

1. ทำการติดตั้งแพ็กเกจ การทำงาน MySQL Server ลงบน Raspberry Pi ที่ทำหน้าที่เป็นโภนดฐานข้อมูล

```
Sudo apt-get install mysql-server
```

2. ทำการติดตั้งแพ็กเกจที่ใช้ในการเข้าถึงฐานข้อมูลบน Raspberry Pi ที่ทำหน้าที่บันทึกฐานข้อมูล

```
Sudo apt-get install mysql-client python-mysqldb
```

3. ทำการตั้งค่าการรีโมทเข้าฐานข้อมูล MySQL Server จากที่ได้

```
sudo nano /etc/mysql/my.cnf
```

ทำการแก้ไข IP Address ที่เข้าถึง MySQL Server ได้จากเดิม 127.0.0.1 เป็น 0.0.0.0

sudo service mysql restart

#### 4. ทำการเข้า MySQL Server และทำการสร้าง Username ที่สามารถเข้าใช้ MySQL Server จากทุก IP Address

Mysql -uroot -hlocalhost -p

เมื่อเข้าระบบแล้วใช้คำสั่ง MySQL ดังนี้

```
GRANT ALL PRIVILEGES ON *.* TO 'USERNAME'@'%' IDENTIFIED BY  
'PASSWORD';
```

flush privileges;

ได้ทำการเขียนโค้ดเชื่อมต่อกับฐานข้อมูลในภาษา Python โดยจะทำการการเข้าถึงฐานข้อมูลได้จากทุกที่เข้ามายังฐานข้อมูล MySQL Server ที่ได้ตั้งไว้ชั่วโมงนี้คือ 192.168.42.10 โดยตัวอย่างผลลัพธ์ตารางที่ MySQL มีดังนี้

```
mysql> select * from eventtype
-> ;
+----+-----+-----+
| No | EventType | EventDescription |
+----+-----+-----+
| 1 | E001 | Turn On |
| 2 | E002 | Turn Off |
| 3 | E003 | Temperature Alert(High) |
| 4 | E004 | Temperature Alert(Low) |
| 5 | E005 | Motion Detected , No Face Detected |
| 6 | E006 | Motion Detected , Face Detected : Member Face |
| 7 | E007 | Motion Detected , Face Detected : Nonmember Face |
+----+-----+-----+
7 rows in set (0.01 sec)

mysql>
```

รูปภาพที่ 4.30 : ภาพแสดงตาราง Event Type ในฐานข้อมูล

```
Database changed
mysql> select * from eventlog;
+----+-----+-----+-----+-----+
| No | NodeID | EventType | Status | EventDateTime | ImageFileLocation |
+----+-----+-----+-----+-----+
| 1 | N000 | E001 | on | 2016-03-17 16:30:56 | NULL |
| 2 | N000 | E002 | off | 2016-03-17 16:31:03 | NULL |
| 3 | N000 | E001 | on | 2016-03-17 16:31:30 | NULL |
| 4 | N000 | E002 | off | 2016-03-17 16:31:32 | NULL |
| 5 | N000 | E001 | on | 2016-03-17 16:31:34 | NULL |
| 6 | N000 | E002 | off | 2016-03-17 16:31:36 | NULL |
+----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

รูปภาพที่ 4.31: ภาพแสดงตาราง Event Log ในฐานข้อมูล

182	N001	26.00	78.00	25.00	2016-03-24	17:03:24
183	N001	26.00	78.00	20.00	2016-03-24	17:04:15
184	N001	25.00	77.00	20.00	2016-03-24	17:04:26
185	N001	26.00	78.00	20.00	2016-03-24	17:05:07
186	N001	25.00	77.00	28.00	2016-03-24	18:53:44
187	N001	30.00	86.00	34.00	2016-03-24	18:54:05
188	N001	30.00	86.00	23.00	2016-03-24	18:54:41
189	N001	30.00	86.00	33.00	2016-03-24	18:55:01
190	N001	30.00	86.00	22.00	2016-03-24	18:55:12
191	N001	30.00	86.00	23.00	2016-03-24	18:55:33
192	N001	30.00	86.00	23.00	2016-03-24	18:55:49
193	N001	30.00	86.00	24.00	2016-03-24	19:46:05
194	N000	25.00	77.00	20.00	2016-03-28	00:04:02
195	N000	24.00	75.00	21.00	2016-03-28	00:04:07
196	N000	24.00	75.00	21.00	2016-03-28	00:04:18
197	N000	25.00	77.00	23.00	2016-03-28	00:04:33
198	N000	24.00	75.00	25.00	2016-03-28	00:04:43
199	N000	25.00	77.00	20.00	2016-03-28	00:05:05
200	N000	25.00	77.00	20.00	2016-03-28	00:05:15
201	N000	25.00	77.00	20.00	2016-03-28	00:05:31
202	N000	24.00	75.00	21.00	2016-03-28	00:05:36
203	N000	25.00	77.00	20.00	2016-03-28	00:05:46
204	N001	25.00	77.00	20.00	2016-03-28	00:11:41
205	N001	25.00	77.00	20.00	2016-03-28	00:26:47
206	N001	24.00	75.00	21.00	2016-03-28	00:50:02
207	N001	24.00	75.00	21.00	2016-03-28	01:12:10
208	N001	24.00	75.00	21.00	2016-03-28	01:41:19
209	N001	24.00	75.00	21.00	2016-03-28	01:42:53
210	N001	23.00	73.00	21.00	2016-03-28	01:44:22
211	N001	24.00	75.00	21.00	2016-03-28	01:45:16
212	N001	24.00	75.00	21.00	2016-03-28	01:45:51
213	N001	23.00	73.00	37.00	2016-03-28	01:46:29
214	N001	24.00	75.00	21.00	2016-03-28	01:46:44
215	N001	23.00	73.00	21.00	2016-03-28	02:08:57

รูปภาพที่ 4.32 : ภาพแสดงตาราง Temperature Log ในฐานข้อมูล

## 4.6 ระบบการส่งข้อมูลไปยัง Dropbox

The screenshot shows the Dropbox web interface. On the left, there's a sidebar with links like Recents, Files, Team, Paper, Photos, Sharing, Links, Events, and File requests. The main area shows a folder structure under 'Dropbox > Apps > Homeauto3572'. Inside, there are four sub-folders: 'Database', 'EventImage', 'FaceDatabase', and 'TrainingData'. The 'Database' folder has a blue 'Share' button next to it. At the bottom left, there's a link to 'Deleted Files'.

รูปภาพที่ 4.33 : ภาพแสดงข้อมูลใน Dropbox

ให้ทำการเชื่อมต่อ กับ Dropbox โดยเรียกใช้ API ที่ทำการ เชิญไฟล์ ของ Dropbox โดยจะทำการแบ่งไฟล์เดอร์การทำงานไว้ 4 ไฟล์เดอร์ ได้แก่

### 1.Database

จะทำการเก็บไฟล์ฐานข้อมูลที่ส่งมาจากโอนคฐานข้อมูลโดยจะมีลักษณะดังภาพ

The screenshot shows the 'Database' folder from the previous image. It contains a single file named 'homeauto\_database.sql'. Below the file name, it says '28 mins ago' and 'Shared with --'. There are standard file operations icons above the file list.

รูปภาพที่ 4.34: ภาพแสดงข้อมูลใน Dropbox ไฟล์เดอร์ Database

## 2. EventImage

ทำการเก็บภาพที่ทำการตรวจสอบแล้วพบว่าไม่สามารถตรวจสอบใบหน้าหรือ พบใบหน้าไม่รู้จัก

Name	Modified	Shared with
2016-05-02_09:01:41.jpg	Today 9:01 AM	--
2016-05-02_09:01:57.jpg	Today 9:02 AM	--
2016-05-02_09:02:33.jpg	Today 9:02 AM	--
2016-05-02_09:02:50.jpg	Today 9:03 AM	--
2016-05-02_09:02:57.jpg	Today 9:03 AM	--
2016-05-02_09:03:02.jpg	Today 9:03 AM	--
2016-05-02_09:03:26.jpg	Today 9:03 AM	--
2016-05-02_09:04:01.jpg	Today 9:04 AM	--
2016-05-02_09:23:30.jpg	Today 9:23 AM	--
2016-05-02_09:24:22.jpg	Today 9:24 AM	--

รูปภาพที่ 4.35 : ภาพแสดงข้อมูลใน Dropbox โฟลเดอร์ EventImage

## 3. FaceDatabase

ทำการเก็บภาพใบหน้าที่ผ่านการปรับแต่งร้องนำมานำมาใช้ในการเรียนรู้ร้องนำข้อมูล(Training)

Name	Modified	Shared with
0.jpg	30 secs ago	--
1.jpg	28 secs ago	--
2.jpg	26 secs ago	--
3.jpg	24 secs ago	--
4.jpg	22 secs ago	--
5.jpg	20 secs ago	--
6.jpg	19 secs ago	--
7.jpg	16 secs ago	--
8.jpg	15 secs ago	--

รูปภาพที่ 4.36 : ภาพแสดงข้อมูลใน Dropbox โฟลเดอร์ FaceDatabase

## 4.TrainingData

ทำการเก็บไฟล์ร้องนำสำหรับการตรวจสอบใบหน้า

Name	Modified	Shared with
training5.xml	2 mins ago	--

รูปภาพที่ 4.37 : ภาพแสดงข้อมูลใน Dropbox โฟลเดอร์ TrainingData

## 4.7 Dynamic DNS

### ขั้นตอนการใช้งาน Dynamic DNS

1. ทำการสมัครใช้บริการเว็บไซต์ที่ให้บริการ Dynamic DNS
2. ทำการสร้างโฉสเนมที่ต้องการ ในที่นี่คือ homeauto3572.no-ip.biz ดังภาพที่ 4.28

The screenshot shows the No-IP Dynamic DNS service interface. On the left, there's a sidebar with options like Dashboard, Dynamic DNS (selected), Hostnames, Groups, Dynamic Update Client, My Services, Account, and Support Center. A green button 'Upgrade to Enhanced' is visible. The main area is titled 'Dynamic DNS' and shows 'Hostnames'. It lists two entries: 'Manage Hostnames' and 'homeauto3572.ddns.net' and 'homeauto3572.no-ip.biz'. Each entry has columns for Hostname, IP / Target, Type, and Status. The 'homeauto3572.no-ip.biz' entry shows 'Expires in 6 days' and a 'Confirm' button. To the right, there's a sidebar for 'Service Level' with a note about free hostnames expiring every 30 days and enhanced ones never expiring, along with an 'Upgrade to Enhanced' button. At the bottom right, it says 'Hostname Count'.

รูปภาพที่ 4.38 : ภาพแสดง Hostname ที่สร้างไว้สำหรับ Dynamic DNS

3. ทำการติดตั้งโปรแกรมติดตามไอพีของทางผู้ให้บริการ เพื่อให้บริการ Dynamic DNS ของเราทำการอัปเดตไอพีที่ได้รับมาแบบ DHCP จาก ISP แบบอัตโนมัติ

The screenshot shows the DUC v4.1.1 application window. The title bar says 'DUC v4.1.1'. The menu bar includes File, Edit, Tools, and Help. The main area is titled 'Status' and displays three items with green checkmarks: 'Client ID: C3A32135533', 'Account: tienworanet', 'Edit' button; 'Updating: 0 Groups & 1 hosts', 'Edit Hosts' button; and 'IP: [REDACTED]', 'Next Check: 4m 50s', 'Refresh Now' button. At the bottom, it says '11:52 AM: Updated 1 items successfully.' and '0 Notices, 0 Alerts'.

รูปภาพที่ 4.39 : ภาพแสดงโปรแกรมอัปเดตไอพีให้สอดคล้องกับ Dynamic DNS

4. ทำการตั้งค่าเราท์เตอร์ที่สนับสนุนการทำงานของ Dynamic DNS ของเราให้ตรงกับที่ได้สมัครเอาไว้

The screenshot shows the 'Dynamic DNS Setup' configuration page. It has several input fields and dropdown menus. Under 'Active Dynamic DNS', there's a checked checkbox and a dropdown menu set to 'www.no-ip.com'. Below it are dropdowns for 'Default' and 'N/A', and a text input field containing 'homeauto3572.no-ip.biz'. Other fields include 'User Name' (tienworanet) and 'Password' (redacted). There's also a checkbox for 'Enable Wildcard Option' which is unchecked. The entire form is contained within a light blue border.

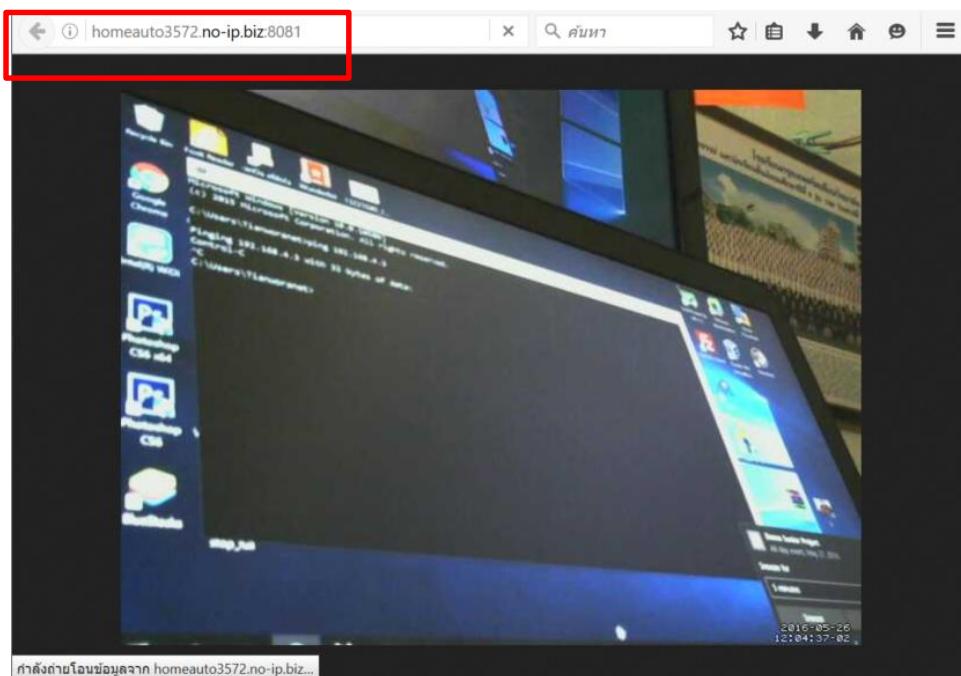
รูปภาพที่ 4.40 : ภาพแสดงการตั้งค่า Dynamic DNS ในเราเตอร์

5. ทำการตั้งค่าในส่วนของ Port Forwarding ที่ต้องการให้เข้าถึงได้จากภายนอกมายังบริการภายในระบบเครือข่ายภายในบ้านของเรา

#	Active	Service Name	Start Port	End Port	Port Translation Start Port	End Port	Server IP Address	Modify
0	<input checked="" type="checkbox"/>	WWW	80	80	80	80	192.168.1.10	
1	<input checked="" type="checkbox"/>	SSH	22	22	22	22	192.168.1.10	
2	<input checked="" type="checkbox"/>	SurveillanceCamera	8081	8081	8081	8081	192.168.1.37	
3	<input type="checkbox"/>	-	-	-	-	-	-	

รูปภาพที่ 4.41 : ภาพแสดงการตั้งค่า Port Forwarding

6. ทำการเข้าถึงโหนดกล่องของเราด้วยโฮสเนมของระบบเครือข่ายภายในบ้านของเราและพอร์ตที่เราได้ตั้งค่าเอาไว้จะได้ผลดังภาพที่ 4.32



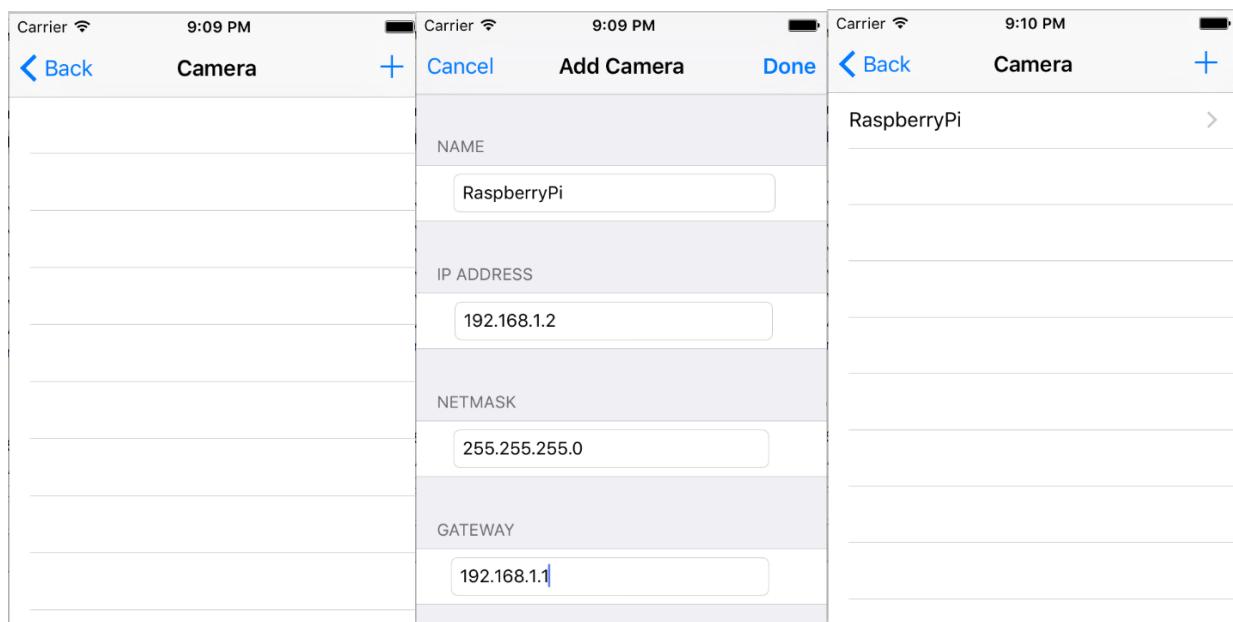
รูปภาพที่ 4.42 : ภาพแสดงผลลัพธ์จากการใช้งาน Dynamic DNS

## 4.8 ໂມບາຍແອພລິເຄົ່ນ (Mobile Application)

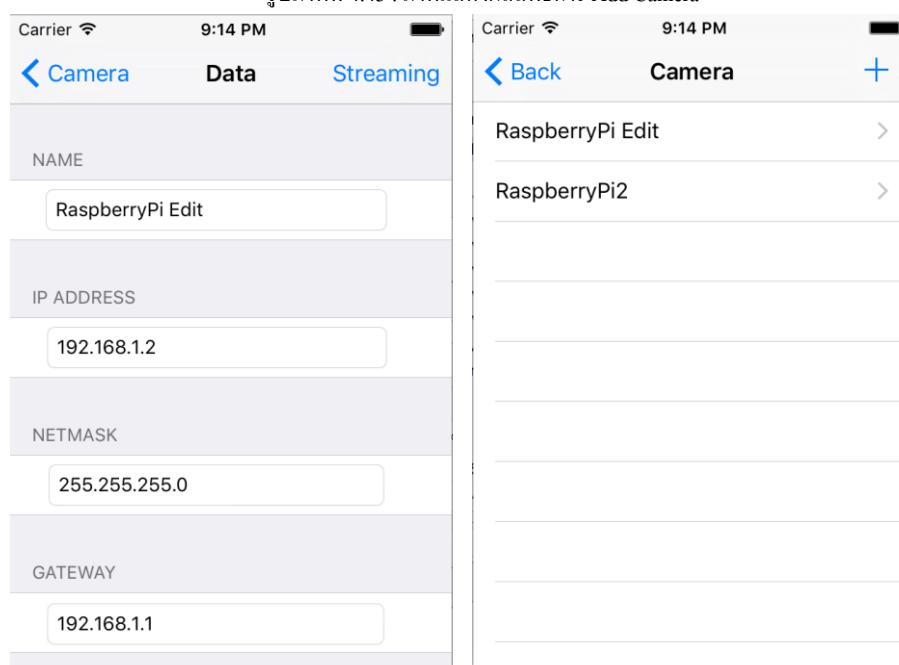
ພລດັບທີ່ໃນສ່ວນຂອງໂມບາຍແອພລິເຄົ່ນຈະມີອູ້ຄ້ວຍກັນ 3 ເມື່ອ

### 4.8.1. Camera

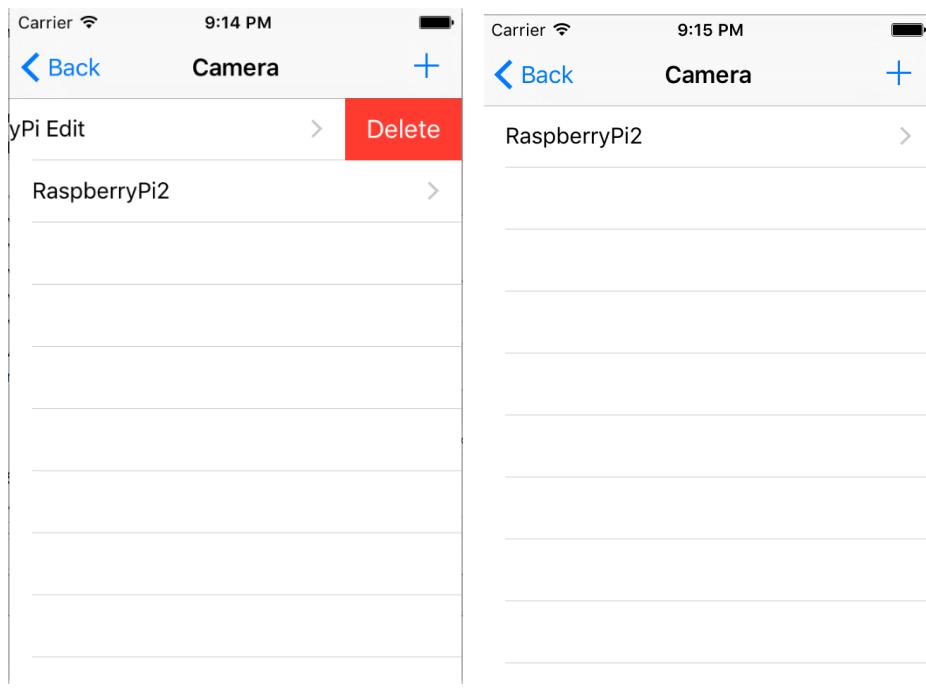
ໃນສ່ວນຂອງເມື່ອ Camera ດາວກາດທຳການ Add/ Edit/ Delete ໂດຍຈະທຳການຮັບຂໍ້ອມຸດໄອຟີ Address ເພື່ອທີ່ຈະໃຊ້ໃນການສຕຣີມມຶ່ງ ເພື່ອດູກາພທີ່ເກີດຂຶ້ນ



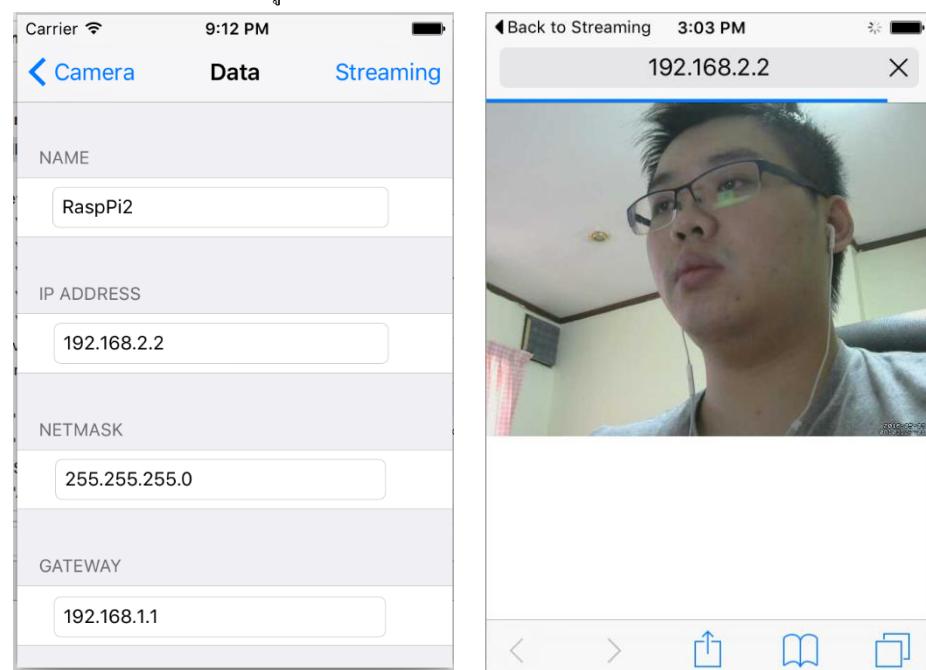
ຮູບກາພທີ່ 4.43 : ກາພແສດງພລດັບທີ່ກາຮ Add Camera



ຮູບກາພທີ່ 4.44 : ກາພແສດງພລດັບທີ່ກາຮ Edit Camera



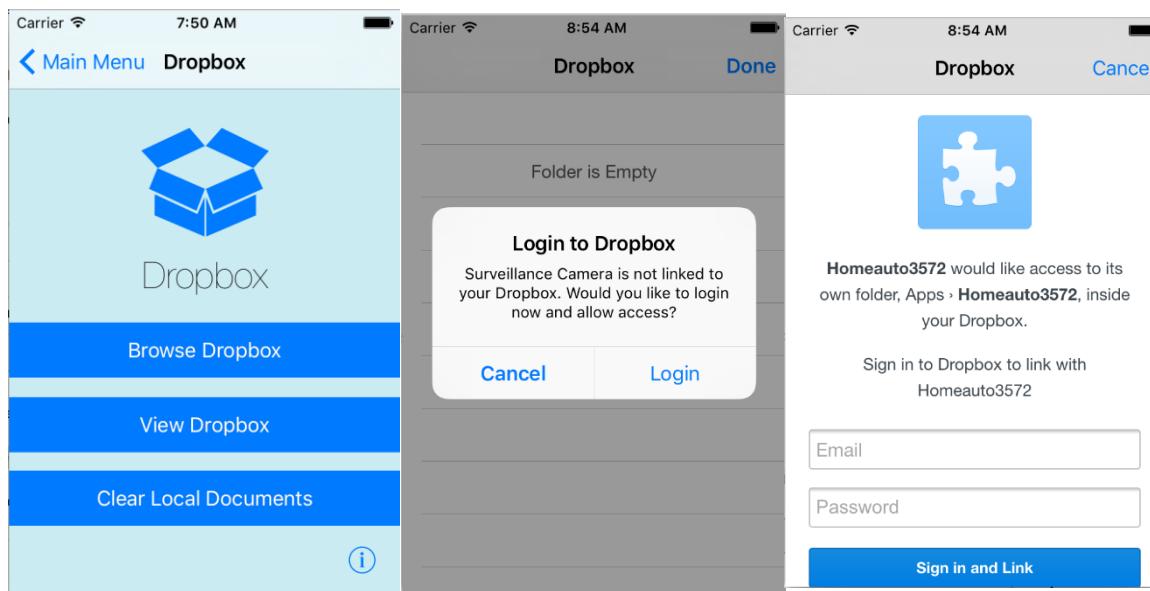
รูปภาพที่ 4.45 : ภาพแสดงผลลัพธ์การ Delete Camera



รูปภาพที่ 4.46 : ภาพแสดงผลลัพธ์การสตรีมเมิ่งเข้ากับกล้อง Webcam จาก Mobile Application

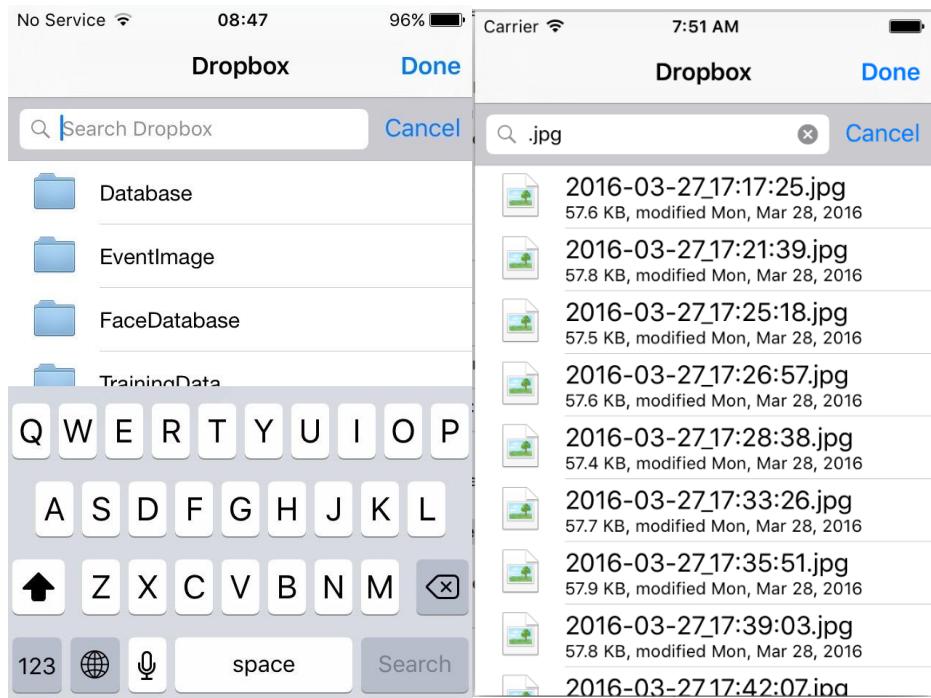
#### 4.8.2. เมนู Link To Dropbox

ในส่วนของเมนู Link To Dropbox จะมีอยู่ด้วยกัน 3 ระบบ คือ Browse Dropbox , View Dropbox และ Clear Local Document ซึ่งจำเป็นที่จะต้องทำการเชื่อมต่อเข้ากับ Dropbox API Account จึงจะสามารถใช้งานในส่วนนี้ได้

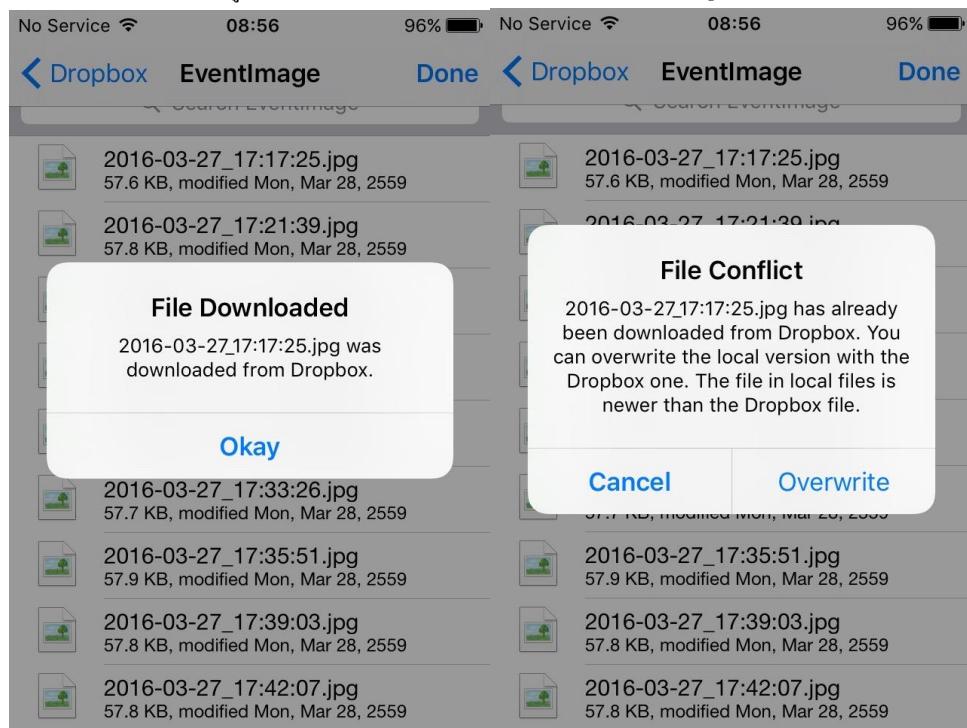


รูปภาพที่ 4.47 : ภาพแสดงการเชื่อมต่อเข้ากับ Dropbox API

ในส่วนของระบบ Browse Dropbox หลังจากทำการเชื่อมต่อ Dropbox API แล้วจะทำการดึงข้อมูลรายการที่มีอยู่ใน Dropbox มาแสดงผลในแอปพลิเคชันซึ่ง สามารถที่จะทำการค้นหาไฟล์โดยใส่คำค้นหา ชื่อไฟล์ หรือ สกุลไฟล์ โดยสามารถคลิกที่ไฟล์เพื่อทำการดาวน์โหลดมาบันทึกไว้ โดยสามารถทำการตรวจสอบได้ว่าไฟล์นั้นได้ทำการดาวน์โหลดไปแล้วหรือไม่ ต้องการให้บันทึกแทนที่หรือไม่

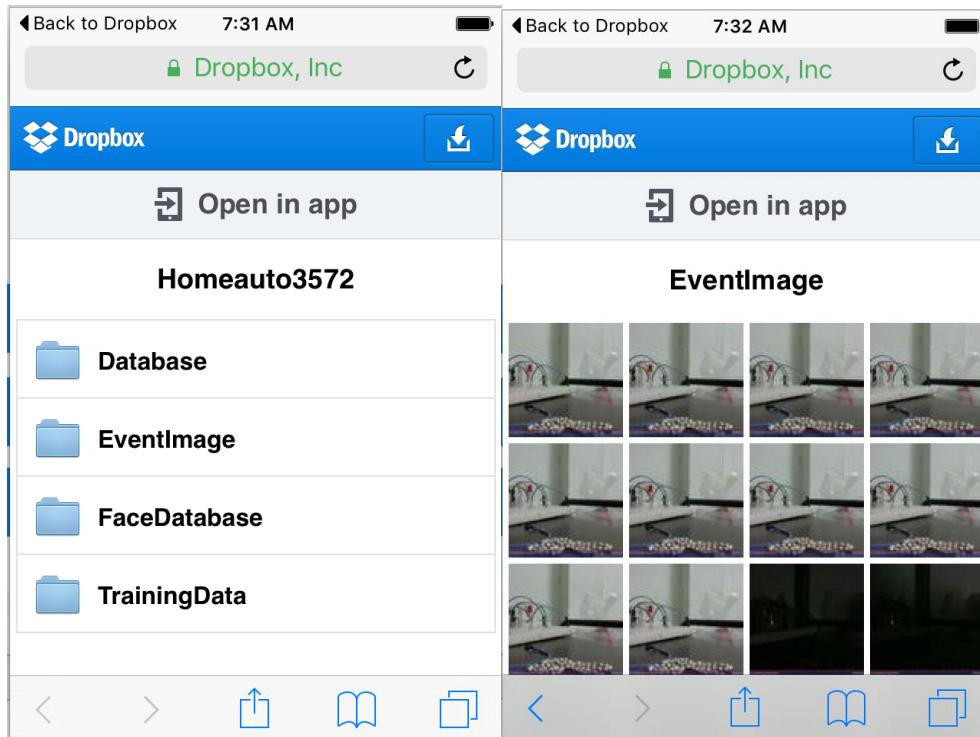


รูปภาพที่ 4.48 : ภาพแสดงการค้นหาไฟล์ใน Browse Dropbox



รูปภาพที่ 4.49 : ภาพแสดงการดาวน์โหลดไฟล์ใน Browse Dropbox

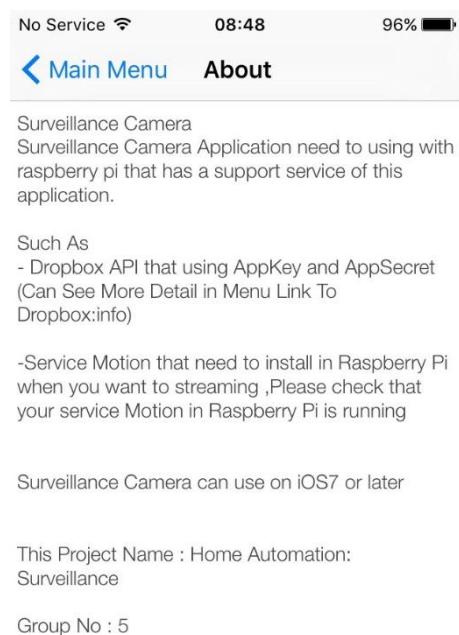
ในส่วนของระบบ View Dropbox นั้นจะทำการเชื่อมต่อผ่านทางเว็บบราวเซอร์เข้าไปที่ Dropbox ใช้เพื่อแสดงผลข้อมูลที่มีอยู่ใน Dropbox และทำการพิริว่าผ่านทางเว็บบราวเซอร์



รูปภาพที่ 4.50 : ภาพแสดงผลลัพธ์ของ View Dropbox

#### 4.8.3 About

ในส่วนของเมนู About เป็นรายละเอียด ของแอปพลิเคชัน



รูปภาพที่ 4.51 : ภาพแสดงผลลัพธ์ของ About

## บทที่ 5

### บทสรุป

#### 5.1 สรุปผลการดำเนินงาน

##### 5.1.1 การสร้างเครือข่ายของ Surveillance Camera

จากการดำเนินงาน การสร้างเครือข่ายของ Surveillance Camera ถือว่าประสบผลสำเร็จตามที่คาดหวังไว้ และจากการที่เปลี่ยนมาเป็นการใช้งานแบบการเชื่อมต่อไร้สายทุกโหนด และการเพิ่มโภคการทำงานที่รองรับการเกิดข้อผิดพลาดเวลาเจอปัญหาการต่างๆ ทำให้ระบบมีความคงสภาพการใช้งานมากขึ้น หากโหนดไหนหยุดทำงาน ระบบก็จะทำงานต่อไปเท่าที่ทำได้ เช่น โหนดที่ทำหน้าที่เป็นฐานข้อมูลเกิดใช้งานไม่ได้ แต่โหนดอื่นๆ จะยังสามารถแจ้งเตือนอีเมล์และบันทึกข้อมูลต่างๆลง Dropbox ได้ตามเดิม แต่ทั้งนี้ทั้งนั้นถ้าหากอินเทอร์เน็ตภายในระบบเครือข่ายภายในบ้าน เกิดใช้งานไม่ได้ แต่โหนดที่ทำหน้าที่เป็นฐานข้อมูล ใช้งานได้ก็จะทำการเก็บข้อมูลลงฐานข้อมูล แต่เพียงอย่างเดียว และเมื่ออินเทอร์เน็ตกลับมาใช้งานได้อีกครั้ง ก็จะทำการส่งฐานข้อมูล ไปสำรองไว้ที่ Dropbox ได้ตามเดิม แต่รูปภาพเหตุการณ์จะไม่สามารถรับรวมไปส่งย้อนหลังได้ ดังนั้นในกรณีนี้ อาจจะเกิดความไม่สอดคล้องกันของข้อมูลขึ้นได้

##### 5.1.2 ระบบการทำงานหลักของโหนดกล้องและโหนดฐานข้อมูล

จากการดำเนินงาน ได้ทำการเขียนโปรแกรมให้ระบบการทำงานหลังมีเพียงการตรวจสอบอุณหภูมิและความชื้นตามช่วงเวลา ส่วนอื่นๆ ได้ทำการเขียนโปรแกรมแบบอินเตอร์รัพท์ โดยผลที่ออกมายังเป็นที่น่าพอใจ เพราะโปรแกรมจะไม่ต้องงานลูปเช็คเชิงซ้อนทุกตัวลดเวลา และถ้าหากเกิดอินเตอร์รัพท์พร้อมกันหลายอย่าง ก็ยังสามารถทำงานพร้อมกันได้ และในส่วนของการเขียนโปรแกรมถือว่ามีความสามารถทำงานเป็นที่น่าพอใจ อาทิ การเกิดอินเตอร์รัพท์ส่วนของเซ็นเซอร์ตรวจจับการเคลื่อนไหวหลายครั้งติดกัน ทำให้ต้องไปเรียกใช้กล้องเว็บแคมติดต่อกัน และเนื่องจากมีการใช้เซลชั่นอยู่ก่อน เซลชั่นที่ทำการเรียกที่หลังจะไม่สามารถทำงานได้และจะเกิดความผิดพลาดในการทำงาน แต่เนื่องจากการได้มีการเขียนโปรแกรมให้รองรับการทำงานต่อแม้จะเกิดความผิดพลาดในการทำงานบางอย่างขึ้นมา ระบบจึงทำงานต่อได้อย่างไม่มีปัญหา

##### 5.1.3 ระบบจดจำใบหน้า (Face Recognition)

จากการดำเนินงานทดลองอัลกอริทึมตรวจสอบใบหน้าแบบอื่นๆ พบว่าอัลกอริทึมที่ให้ผลเป็นที่น่าพอใจที่สุดคือ Local Binary Patterns Histograms และถือว่ายังนั้นการนำระบบตรวจสอบใบหน้ามาทำเป็นกล้องวงจรปิดที่ทำการตรวจสอบการเคลื่อนไหว แล้วเก็บภาพมาวิเคราะห์ ถือว่ายังไม่สามารถให้ผลที่น่าพอใจนัก เพราะถ้าหากทำการเก็บภาพ ได้ภาพที่เบลอ คนในภาพเคลื่อนที่เร็วเกินไป ถึงแม้จะตรวจจับใบหน้าได้ แต่ไม่สามารถตรวจสอบได้ว่าเป็นสามาชิกหรือไม่ ทั้งนี้ขึ้นอยู่กับวางแผนที่ให้กล้องทำการเก็บภาพ ให้มีความเป็นไปได้ในการเก็บภาพหน้าตรงมากที่สุด เพราะถ้าหากเก็บภาพหน้าอิ่ยง ค่าความคล้ายของคลาดเคลื่อนไปได้ และถึงแม้จะทำฐานข้อมูลใบหน้าที่มีลักษณะหน้าอิ่ยงรวมอยู่ด้วย ผลที่ได้คือระบบตรวจสอบใบหน้าจะได้ผลที่ไม่แน่นอน ระบบจดจำใบหน้าแบบวิเคราะห์รูปภาพในโปรเจกนี้จะหมายความว่าระบบที่บังคับให้ผู้ใช้งานอยู่เฉยกันเพื่อทำการตรวจสอบมากกว่า ดังนั้นผลลัพธ์โดยรวมที่ออกมารับรับจะดำเนินการตามที่ระบุไว้ในระดับปานกลาง

#### **5.1.4 ระบบการแจ้งเตือนทางอีเมลล์ (Email Notification)**

จากการดำเนินงาน ถือว่าประสบผลสำเร็จเป็นที่น่าพอใจในระดับหนึ่ง มีระบบการแจ้งเตือนพร้อมรูปภาพที่ทำการตรวจสอบ มาทดสอบเพื่อให้ผู้ใช้งานระบบทราบทางอีเมลล์

#### **5.1.5 ระบบฐานข้อมูลบนโหนดรูปแบบ Raspberry Pi**

จากการทำงาน ฐานข้อมูลที่ออกแบบมาได้ออกแบบมาให้รองรับการใช้งานโดยไม่ต้องติดต่อสัญญาณภายนอกต่อไปยังเดิม โดยการจัดทำ และเขียนโค้ดประกอบในส่วนต่างๆ ได้เสร็จสิ้นทั้งหมดแล้ว ผลลัพธ์ออกมาเป็นที่น่าพอใจโดยได้ทำการเก็บข้อมูลจากเซ็นเซอร์ทุกด้าน ได้แก่ เซ็นเซอร์ตรวจจับการเคลื่อนไหว เซ็นเซอร์ตรวจจับอุณหภูมิ ความชื้น และเซ็นเซอร์ตรวจจับแก๊สไวไฟ จัดเก็บค่าต่างๆ สามารถนำไปแสดงผลยังโหนดรูปช่องข้อมูล ของโดยไม่ต้องติดต่อสัญญาณภายนอกต่อไปยังเดิม ความแม่นยำ รวดเร็ว และลดต้นทุนการซื้อตัวเครื่องที่ต้องติดต่อสัญญาณภายนอกต่อไป

#### **5.1.6 ระบบการส่งข้อมูลไปยัง Dropbox**

จากการทำงาน ระบบการส่งข้อมูลที่ทำการตรวจสอบได้มาที่โหนดรูปแบบฐานข้อมูลหรือโหนดที่ทำหน้าที่เป็นโหนดรูปแบบฐานข้อมูลแต่เพียงอย่างเดียว และจะทำการส่งฐานข้อมูลระบบไปสำรองไว้ยัง Dropbox ในทุกๆ 50 วินาที ผลลัพธ์ที่ออกมายังถือว่าทำออกมาได้ประสบผลสำเร็จตามเป้าหมาย ตอบสนองต่อระบบในส่วนอื่นๆ เช่น ไม่บ้ายแอพลิเคชันได้เป็นอย่างดี

#### **5.1.7 Dynamic DNS**

จากการทำงาน พบว่ายังพบปัญหาการทำงาน โดยสามารถเข้าถึงและรับชมภาพได้สักระยะหนึ่งจากนั้นการเชื่อมต่อจะขาดหาย จะต้องทำการรีสตาร์ทเพื่อทำการเชื่อมต่ออีกครั้งด้วยการรีเฟรชหน้าต่างการใช้งาน โดยรวมแล้วถือว่ายังต้องมีการปรับปรุงและแก้ไขปัญหาทางด้าน Quality of Service ให้ผลลัพธ์ออกมาดีขึ้น

#### **5.1.8 โมบายแอพลิเคชัน (Mobile Application)**

จากการทำงาน โมบายแอพลิเคชันในส่วนของเมนู Camera สามารถทำการ Add Camera / Edit / Delete รายการที่จะทำการไลฟ์สตรีมมิ่งได้ โดยจะเชื่อมต่อผ่านทางเว็บบริการจากภายนอกแอพลิเคชัน

ในส่วนของเมนู Link to Dropbox สามารถที่จะเชื่อมต่อคลังข้อมูลใน Dropbox ได้ สามารถเรียกข้อมูลที่อยู่ใน Dropbox มาแสดงในแอพลิเคชันได้ สามารถทำการค้นหาไฟล์จากการใส่คีย์เวิร์ดไม่ว่าจะเป็น ชื่อไฟล์ สกุลไฟล์ รวมถึงสามารถดาวน์โหลดไฟล์และสามารถทำการแสดงผลจากการเว็บบริการ

และ ในส่วนของ About ก็สามารถทำการแสดงรายละเอียดข้อมูลของแอพลิเคชันได้ ผลลัพธ์ในส่วนแอพลิเคชันนี้ถือว่าสามารถใช้งานได้ แต่ควรจะปรับปรุงให้เหมาะสมกับการทำงานมากยิ่งขึ้นในอนาคต

## 5.2 ปัญหาที่พบและแนวทางการแก้ไข

	ปัญหาที่พบ	การแก้ไขปัญหา
1	การตรวจสอบใบหน้าที่ผิดพลาดของอัลกอริทึมในระบบจดจำใบหน้า	ได้ทำการสร้างฐานข้อมูลใบหน้าที่มีคุณภาพมากขึ้น โดยจะบันทึกภาพใบหน้าของคน 1 คน ในลักษณะหลายมุมมองมากขึ้น และพยายามนำมากขึ้น และผลลัพธ์ที่ออกมานั้นถือว่ามีโอกาสตรวจจับได้ตรงสูงมากขึ้น แต่ก็ยังมีโอกาสเกิดข้อผิดพลาด บ้าง
2	ปัญหาการสร้างจุดเชื่อมต่อ (Access Point) ด้วยอุปกรณ์ TP-WN722N ทำให้ไม่สามารถเครือข่ายแบบmeshได้	ทำการยกเลิกการใช้งานจุดเชื่อมต่อ (Access Point) และ DHCP Server เพราะไม่มีความจำเป็นที่ผู้ใช้งานจะต้องเข้าถึงเครือข่าย Surveillance Camera โดยตรง และทำการเปลี่ยนกลับไปใช้อุปกรณ์ TP-WN725N ดังเดิม
3	ปัญหารื่องการตั้งค่า Raspberry Pi เป็นเกตเวย์ระหว่างอีเซอร์เน็ตและระบบเครือข่ายไร้สายของระบบ Surveillance จะมีข้อเสียคือถ้าหากเกิดหยุดการทำงานให้นดื่นๆจะไม่สามารถติดต่อ กับระบบเครือข่ายภายในบ้านผ่านระบบเครือข่ายไร้สาย เพื่อความสะดวก และคงสภาพการทำงานบางส่วนได้แม้จะมีการทำงานส่วนหนึ่งไม่สามารถทำได้ ก็จะสามารถทำส่วนอื่นๆได้ต่อไป	ได้ทำการยกเลิกการใช้งาน Raspberry Pi 1 ตัวเป็นเกตเวย์และทำการเปลี่ยนให้ทุกโน๊ตมาใช้การเชื่อมต่อ กับระบบเครือข่ายภายในบ้านผ่านระบบเครือข่ายไร้สาย เพื่อความสะดวก และคงสภาพการทำงานบางส่วนได้แม้จะมีการทำงานส่วนหนึ่งไม่สามารถทำได้ ก็จะสามารถทำส่วนอื่นๆได้ต่อไป
4	ปัญหารื่องเซ็นเซอร์ตรวจจับความเคลื่อนไหว ตรวจพบการเคลื่อนไหว ทั้งๆ ที่ความเป็นจริงแล้วไม่มีการเคลื่อนไหว	ทำการตรวจสอบโดยการใช้งาน และแก้ไขด้วยภาษา C บน Raspberry Pi ของตัวเซ็นเซอร์อยู่ในลักษณะคงอยู่ และมีคุณภาพที่ไม่ดี ดังนั้นจึงทำการแก้ไขโดยใช้ภาษา C ที่มีขนาดพอเหมาะในการเชื่อมต่อ
5	ความไม่สะดวกในการใช้งานจริง เมื่อผู้ใช้พบว่าอินเตอร์เฟสที่เชื่อมต่อเครือข่ายไม่ทำงาน และการผิดเครื่องจำเป็นต้องทำการรีโมทเข้าไปเพื่อปิดการใช้งานอุปกรณ์อย่างปลดภัย	เพิ่มปุ่มการทำงาน ได้แก่ ปุ่มรีบูตการเชื่อมต่อเครือข่าย ปุ่มรีบูตเครื่อง ปุ่มปิดเครื่อง ปุ่มข้อนการทำงานตั้งค่า
6.	ปัญหาจากการเขียนโปรแกรมแอพลิเคชันโดยใช้ Android เนื่องจากเวอร์ชันแพ็กเกจที่เผยแพร่อยู่ในอินเทอร์เน็ตไม่รองรับการ	ทำการเปลี่ยนจาก Android มาทำการใช้ระบบ iOS แทน โดยใช้โปรแกรม Xcode ซึ่งทำการทดสอบ ทั้ง Android และ iOS ว่าสามารถใช้งานในส่วนที่ต้องการได้หรือไม่

	ใช้งาน	หลังจากทดลอง ก็ได้เปลี่ยนมาใช้ iOS เนื่องจาก มีปัญหาทางด้านแพ็กเกจ Android เวอร์ชันในส่วนที่ต้องการเขียนโค้ด
7.	ปัญหาจากการทำแอพลิเคชั่นเนื่องจากทางกลุ่มผู้จัดทำไม่มีความชำนาญ	ทำการศึกษาเพิ่มเติมจากทางอินเทอร์เน็ต อ้างอิงจากแหล่งข้อมูลทางอินเทอร์เน็ต
8.	ปัญหาจากการทำแอพลิเคชั่น เนื่องจาก เวอร์ชัน หรือการที่ไม่มีโค้ดแบบ Open Source ในส่วนที่ระบบต้องการ เช่น SSH , Live Streaming	พยายามทำการปรับเปลี่ยน แก้ไข เพื่อให้สามารถใช้งานได้ Live Streaming ไม่สามารถเปิด Media Player จากภายในแอพลิเคชั่น ได้ จึงทำการใช้ HTTP Streaming ผ่านทางเบราว์เซอร์แทนในส่วนของ SSH ไม่มีโค้ด Open Source มีแต่การใช้แอพลิเคชั่นตัวอื่นทำการ VPN แทน ซึ่งกำลังหาทางอื่นในการปรับแก้ไขเพื่อให้สามารถใช้งานได้จริงจากภาพในแอพลิเคชั่นที่สร้างขึ้น

ตารางที่ 5.1 : ปัญหาที่พบและแนวทางการแก้ไข

### 5.3 ตารางแสดงสถานะของงานในปัจจุบัน

	หัวข้องาน	สถานะงาน
1	ศึกษา รวบรวมข้อมูลและทำการเลือกอุปกรณ์ที่จะใช้ทำโปรเจค	เสร็จสมบูรณ์
2	ศึกษาการใช้งาน Raspberry Pi เป็นต้น	เสร็จสมบูรณ์
3	ศึกษาการสร้างโหนดกล้อง (Camera Node)	เสร็จสมบูรณ์
4	ศึกษาการใช้งาน Raspberry Pi เพื่อใช้ในกระบวนการส่งข้อมูล	เสร็จสมบูรณ์
5	ทำการสร้างระบบเครือข่ายภายในบ้าน(Home Network)ที่สามารถบันทึกข้อมูลไปยัง Dropbox	เสร็จสมบูรณ์
6	เริ่มทำการเขียนโปรแกรมการทำงานเกี่ยวกับการตรวจสอบความเคลื่อนไหวด้วย Raspberry Pi	เสร็จสมบูรณ์
7	เริ่มทำการเขียนโปรแกรมการทำงานเกี่ยวกับระบบจดจำใบหน้า	เสร็จสมบูรณ์
8	เริ่มทำการเขียนโปรแกรมการทำงานเกี่ยวกับระบบการแจ้งเตือนทางอีเมลล์และทำการสร้าง Dropbox API	เสร็จสมบูรณ์
9	ทดสอบการทำงานของโหนดกล้องและระบบจดจำใบหน้าของ Raspberry Pi พร้อมทั้งแก้ไขปัญหา	เสร็จสมบูรณ์
10	ทดสอบการทำงานของ Raspberry Pi ให้สามารถแจ้งเตือนทางอีเมลล์	เสร็จสมบูรณ์
11	ระบบดำเนินส และการเก็บล็อกข้อมูลบน Raspberry Pi	เสร็จสมบูรณ์
12	ศึกษาการใช้งานโปรแกรมและภาษาที่ใช้ในการเขียนโปรแกรมบนโน้ตบุ๊ก แอพลิเคชั่น	เสร็จสมบูรณ์

13	ศึกษาการเขียนแอปพลิเคชัน ให้ทำงานร่วมกับ Cloud Storage	เซิร์ฟสมบูรณ์
14	พัฒนาแอปพลิเคชันให้สามารถทำการแสดงผล(Query)ข้อมูลต่างๆจาก Cloud Storage	เซิร์ฟสมบูรณ์
15	ส่วนของเมนู Camera (Add /Edit / Delete) Live Streaming	เซิร์ฟสีนบางส่วน
16	ส่วนของเมนู Link To Dropbox (Browse /Download /View)	เซิร์ฟสมบูรณ์
17	ส่วนของเมนู About ใน โภນขายแอปพลิเคชัน	เซิร์ฟสมบูรณ์
18	ทดสอบการทำงานของ โภນขายแอปพลิเคชันพร้อมทั้งแก้ไขปัญหา	เซิร์ฟสีนบางส่วน

ตารางที่ 5.2 : สถานะงานในปัจจุบัน

## การทำงานภายใต้อนาคต (Future Works)

- เนื่องจากปัญหาข้อจำกัดทางด้านความสามารถของตัว Raspberry Pi ทำให้การประมวลผลจำนวนมาก เป็นไปได้ช้า จึงทำให้เกิดความจำกัดในการวิเคราะห์รูปภาพ ตรวจสอบใบหน้ารูปภาพ การตรวจสอบใบหน้าในรูปภาพที่ใช้ในโครงการ เป็นการใช้อัลกอริทึมการวิเคราะห์ใบหน้าแบบ 2 มิติ และผลที่ได้จากการวิเคราะห์ใบหน้าแบบ 2 มิติ ถือว่ามีข้อจำกัด เพราะทุกอย่างจะขึ้นอยู่กับฐานข้อมูลใบหน้าที่สร้างไว้ ถ้าลักษณะ มุมมอง องศา ไม่ตรงกับฐานข้อมูลใบหน้า จะถือว่าตรวจสอบแล้วได้ใบหน้าไม่ตรงกับฐานข้อมูล ดังนั้นสิ่งที่จะเพิ่มต่อไปในอนาคต คือการเปลี่ยนการตรวจสอบใบหน้าจากรูปภาพแบบ 2 มิติมาเป็น 3 มิติ เพราะอัลกอริทึมแบบ 3 มิติ จะสามารถคิดได้เองว่า ในหน้าแนวตรงในฐานข้อมูลนั้น เมื่อทำการหันด้านซ้ายหรือด้านขวาแล้วควรจะมีใบหน้าเป็นแบบไหน ซึ่งตรงส่วนนี้เนื่องจากปัญหาข้อจำกัดของ Raspberry Pi ที่กล่าวมาข้างต้น ดังนั้นถ้าหากจะทำการวิเคราะห์แบบ 3 มิติ จะต้องทำการเพิ่มส่วนประมวลผลกลางของระบบซึ่งเป็นคอมพิวเตอร์ที่มีประสิทธิภาพในการประมวลผลมากกว่าเดิม จากเดิมที่ให้ Raspberry Pi ทำการตรวจสอบใบหน้าเอง โดยจะเปลี่ยนเป็นให้ Raspberry Pi ทำการเก็บภาพแล้วให้คอมพิวเตอร์ส่วนกลางประมวลผลแทน
- จากสิ่งที่ได้ทำในโครงการ เราจำเป็นต้องทำการกรอกข้อมูลไอดีวิตัวเองในโมบายแอปพลิเคชัน เพื่อเข้าถึงโอนดการทำงานต่างๆ ดังนั้นในอนาคตหากมีการพัฒนาต่อ จะทำการเพิ่มความอัจฉริยะในระบบส่วนนี้โดยจะให้มีความเป็นอัตโนมัติ โดยเมื่อเปิดใช้โมบายแอปพลิเคชัน จะสามารถติดต่อกับโอนดการทำงานได้ทันที ในส่วนนี้คาดว่าจะใช้คอมพิวเตอร์ส่วนกลางมาทำหน้าที่ในการบริหารจัดการส่วนนี้
- ในส่วนของโมบายแอปพลิเคชัน จะทำให้สามารถใช้งานในส่วนของเพิ่มฐานข้อมูลใบหน้า การนำรูปภาพมาสร้างไฟล์นามสกุล XML ทำให้สามารถทำการเชื่อมต่อเข้าไปภายใน Raspberry Pi เพื่อควบคุมบริการและการทำงานต่างๆ ได้
- ปรับเปลี่ยนไปใช้โซลูชันในการเข้าถึงการทำงานในส่วนของ Live Streaming เพื่อใช้งานในส่วนของโอนดกล้องทั้งนี้ปรับเปลี่ยนให้เหมาะสมกับการใช้งาน มีตัวแยกจ่ายไอพี Address ไม่จำเป็นต้องให้ผู้ใช้ทำการกรอกข้อมูลทุกครั้ง

## บรรณานุกรม

[1] SMTP คือ [ออนไลน์], สืบค้นจาก :

[wich246.tripod.com/smtp.htm](http://wich246.tripod.com/smtp.htm) [17 พฤษภาคม 2558]

[2] Raspberry Pi คือ [ออนไลน์], สืบค้นจาก :

[http://www.thaieeasyelec.com/article-wiki/embedded-electronics-application/%E0%B8%9A%E0%B8%97%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%9E%E0%B8%B1%E0%B8%92%E0%B8%99%E0%B8%B2%E0%B9%82%E0%B8%9B%E0%B8%A3%E0%B9%81%E0%B8%81%E0%B8%A3%E0%B8%A1%E0%B8%9A%E0%B8%99-raspberry-pi-%E0%B8%94%E0%B9%89%E0%B8%A7%E0%B8%A2-qt.html](http://www.thaieasyelec.com/article-wiki/embedded-electronics-application/%E0%B8%9A%E0%B8%97%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%9E%E0%B8%B1%E0%B8%92%E0%B8%99%E0%B8%B2%E0%B9%82%E0%B8%9B%E0%B8%A3%E0%B9%81%E0%B8%81%E0%B8%A3%E0%B8%A1%E0%B8%9A%E0%B8%99-raspberry-pi-%E0%B8%94%E0%B9%89%E0%B8%A7%E0%B8%A2-qt.html) [17 พฤษภาคม 2558]

[3] ความรู้เบื้องต้นเกี่ยวกับภาษา Python [ออนไลน์], สืบค้นจาก :

[http://python.cmsthailand.com/basic\\_python.html](http://python.cmsthailand.com/basic_python.html) [17 พฤษภาคม 2558]

[4] การติดตั้ง OpenCV 2.1 บน Windows ที่ใช้ร่วมกับ Microsoft Visual Studio 2010 สำหรับงานประมวลผลภาพ [ออนไลน์], สืบค้นจาก :

<http://ibookengineering.blogspot.com/2013/08/opencv-21-windows-microsoft-visual.html>

[17 พฤษภาคม 2558]

[5] สารคดี สถาพรบุ๊คส์,"ระบบจดจำใบหน้า (Facial Recognition)" [ออนไลน์], สืบค้นจาก :

<https://www.facebook.com/FeatureSatapornbooks/photos/a.520204658014864.1073741829.520092828026047/59486557420007/?type=3> [18 พฤษภาคม 2558]

[6] Face Recognition คือ [ออนไลน์], สืบค้นจาก :

<http://kokzard.blogspot.com/2012/04/face-recognition-algorithm.htm> [18 พฤษภาคม 2558]

[7] Mysql คือ [ออนไลน์], สืบค้นจาก :

<http://siam500400009.blogspot.com/2011/02/mysql.html> [19 พฤษภาคม 2558]

[8] PIR Sensor คือ [ออนไลน์], สืบค้นจาก :

[http://www.thaieeasyelec.com/article-wiki/review-product-article/pir-motion-sensor-getting-started.html](http://www.thaieasyelec.com/article-wiki/review-product-article/pir-motion-sensor-getting-started.html)

[18/11/2015]

[9] DHT11 Humidity & Temperature Sensor, D-Robotics , 2010

[10] mq-2 sensor คือ [ออนไลน์], สืบค้นจาก :

<http://thaisensormodule.com/index.php/gas-sensor/mq2> [19 พฤษภาคม 2558]

[11] Setting up a Raspberry Pi as a WiFi access point [ออนไลน์], สืบค้นจาก :

<https://learn.adafruit.com/setting-up-a-raspberry-pi-as-a-wifi-access-point/install-software>

[19 พฤษภาคม 2558]

[12] How to use Dropbox with Raspberry Pi [ออนไลน์], สืบค้นจาก :

<http://raspi.tv/2013/how-to-use-dropbox-with-raspberry-pi> [20 ตุลาคม 2558]

[13] tutorial send email python [ออนไลน์], สืบค้นจาก :

<http://naelshiab.com/tutorial-send-email-python/> [20 ตุลาคม 2558]

[14]How to Use Dropbox API in IOS App [ออนไลน์], สืบค้นจาก :

<http://www.appcoda.com/dropbox-api-tutorial/> [1 มกราคม 2559]

[15]Build a Raspberry Pi Webcam Server in Minutes [ออนไลน์], สืบค้นจาก :

<https://pimylifeup.com/raspberry-pi-webcam-server/> [1 มกราคม 2559]

[16] [iOS/iPhone] Tutorials - สอนเขียน iPhone App ฟรี เขียน iPad App เรียน iPhone ฟรีเขียนโปรแกรม iPhone

[ออนไลน์], สืบค้นจาก : <http://www.thaicreate.com/mobile/ios.html> [1 มกราคม 2559]