

แอปพลิชันรวมสินค้าไลฟ์สดด้วยการประมวลผลทางภาพ  
application live shop with image processing

โครงการปริญญาดิษฐ์

ของ

นายกฤตเมธ บัวสิงห์  
นายกิตติศักดิ์ มนพรหมมา

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์  
ปีการศึกษา 2565  
ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม  
คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม

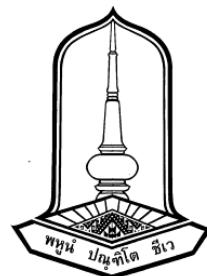
แอปพลิชันรวมสินค้าไลฟ์สดด้วยการประมวลผลทางภาพ  
application live shop with image processing

โครงการปริญญาในพนธ์

ของ

นายกฤตเมธ บัวสิงห์  
นายกิตติศักดิ์ มนพรหมมา

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์  
ปีการศึกษา 2565  
ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม  
คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม



คณะกรรมการสอบโครงการปริญญาดุษฎีบัตร ได้พิจารณาปริญญาดุษฎีบัตรของ นายกฤตเมธ บัวสิงห์ และนายกิตติศักดิ์ มนพรมา แล้วเห็นสมควรรับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาการสารสนเทศ ของมหาวิทยาลัย มหาสารคาม

คณะกรรมการสอบโครงการปริญญาดุษฎีบัตร

..... ประธานสอบ  
(.....)

..... กรรมการ  
(.....)

 ที่ปรึกษาโครงการปริญญาดุษฎีบัตรหลัก  
(ดร.พัฒนพงษ์ ชุมพุวิเศษ)

หลักสูตรวิทยาการคอมพิวเตอร์อนุมัติให้รับโครงการปริญญาดุษฎีบัตรนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา วิทยาศาสตรบัณฑิต สาขาวิชา วิทยาการคอมพิวเตอร์ คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม

.....  
(อาจารย์พชร พฤกษะศรี) (ผู้ช่วยศาสตราจารย์พิมลรัตน์ อ้วนศรีเมือง)  
อาจารย์ผู้ประสานงานวิชาโครงการปริญญาดุษฎีบัตร  
วันที่ เดือน พ.ศ.

## สารบัญ

### หน้า

สารบัญ .....	ก
สารบัญตาราง .....	1
สารบัญรูปภาพ .....	2
สารบัญรูปภาพ(ต่อ) .....	3
สารบัญรูปภาพ(ต่อ) .....	4
บทที่ 1 บทนำ .....	5
1.1 หลักการและเหตุผล .....	5
1.2 วัตถุประสงค์ของโครงการ .....	5
1.3 ขอบเขตของโครงการ .....	5
1.3.1 ผู้ใช้งาน (ใช้งานผ่าน Mobile Application) .....	6
1.3.2 ผู้ดูแลระบบ (ใช้งานผ่าน Desktop Application ที่พัฒนาด้วย Python) .....	8
1.3.3 ระบบ (ระบบพัฒนาเป็น Module ด้วย Python) .....	8
1.4 ภาพรวมของระบบ .....	10
จากภาพรวมระบบจะการทำงานดังนี้ .....	10
1.5 ประโยชน์ที่คาดว่าจะได้รับ .....	11
1.5.1 ช่วยให้ผู้ใช้งานที่เป็นผู้ค้าข่ายมีช่องทางการขายสินค้ามากขึ้น .....	11
1.5.2 ช่วยให้ผู้ใช้งานที่เป็นลูกค้ามีความสะดวกมากขึ้น .....	11
1.6 อุปกรณ์และเครื่องมือที่ใช้ในการดำเนินงาน .....	11
1.6.1 ฮาร์ดแวร์ .....	11
1.6.2 ซอฟต์แวร์ .....	12
1.7 แผนการดำเนินงาน .....	12
1.8 ตัวอย่างโปรแกรม .....	13
1.8.1 แอปพลิเคชั่นบนอุปกรณ์เร็วสายแบบเคลื่อนที่ .....	13
1.8.2 แอปพลิเคชั่นบนคอมพิวเตอร์สำหรับผู้ดูแลระบบ .....	18
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง .....	21

## สารบัญ (ต่อ)

	หน้า
2.1 ทฤษฎีที่เกี่ยวข้อง .....	21
2.1.1 Flutter .....	21
2.1.2 Firebase .....	22
2.1.3 การประมวลผลภาพ (Image Processing) .....	24
2.1.4 OCR หรือ Optical Character Recognition .....	26
2.1.1 Object Detection .....	27
2.1.1.1 YOLO (You Only Look Once) .....	27
2.2 ระบบงานที่เกี่ยวข้อง .....	27
2.2.1 Safety Helmet Detection Based on YOLOv5 .....	27
บทที่ 3 .....	31
3.1 เก็บรวบรวมข้อมูล .....	32
3.2 การเตรียมข้อมูล .....	33
3.2.1 การเตรียมข้อมูลในการ training .....	33
3.2.2 การวาดภาพผลเฉลย .....	33
3.2.3 การทำ Data Augmentation .....	35
3.3 การสร้างโมเดลโดยใช้ yoloV5 .....	37
3.3.1 Convolution Layer .....	37
3.3.2 Rectified Linear Unit (ReLU) .....	38
3.3.3 ReLU leaky .....	39
3.3.4 Pooling layer .....	40
3.3.5 Flatten .....	41
3.3.6 Fully Connected Layer .....	41
3.3.7 การแทนค่าความสูงและความกว้างของ Bounding Box .....	45
3.3.8 การทำ anchor box .....	46
3.3.9 Intersection over Union .....	47
3.3.10 Non-Max Suppression .....	48
3.3.11 โครงสร้างของ Yolov5 .....	49
3.4 การตัดบรรทัดด้วยเทคนิค projection profile .....	56
3.4.1 การเตรียมภาพ .....	56
3.4.2 แปลงภาพสี RGB ไปเป็นภาพระดับเทา .....	56

## สารบัญ (ต่อ)

	หน้า
3.4.3 การแปลงภาพระดับเทา ไปเป็นภาพขาวดำ (Binary image) .....	57
3.4.4 การคาดคะเนจุดที่เป็นที่ว่าง .....	59
3.5 ขั้นตอนการเรียกใช้ Tesseract OCR.....	61
3.6 Mobile application.....	62
3.7 การจัดเก็บข้อมูล .....	62
3.8 วัดประสิทธิภาพ .....	64
อ้างอิง.....	69

## สารบัญตาราง

### หน้า

ตารางที่ 1 แผนการดำเนินงาน .....	12
ตารางที่ 2 ตารางจำนวนและเวลาเฉลี่ยของวิดีโອในแต่ละร้านค้า .....	32
ตารางที่ 3 จำนวนของข้อมูลในแต่ละส่วน .....	34
ตารางที่ 4 จำนวนของข้อมูลก่อนและหลังทำการ Generate .....	35
ตารางที่ 5 ค่า parameter ใน ค่าผลเฉลี่ย y ทั้ง 7 .....	42
ตารางที่ 6 การแทนค่า bx, by, bh, bw .....	45
ตารางที่ 7 parameter ของการทำ anchor box .....	46
ตารางที่ 8 การแปลงรูปเป็นตัวอักษรด้วย tesseract .....	61

## สารบัญรูปภาพ

	หน้า
ภาพประกอบที่ 1.1 ตัวอย่างข้อมูลที่นำมาแยก .....	9
ภาพประกอบที่ 1.2 ภาพรวมของระบบ .....	10
ภาพประกอบที่ 1.3 หน้าแรกของโปรแกรม .....	13
ภาพประกอบที่ 1.4 หน้าล็อกอินเข้าสู่ระบบ .....	14
ภาพประกอบที่ 1.5 หน้าหลักของแอปพลิเคชัน .....	15
ภาพประกอบที่ 1.6 หน้าค้นหาสินค้า .....	16
ภาพประกอบที่ 1.7 หน้ารายละเอียดสินค้า .....	17
ภาพประกอบที่ 1.8 หน้าแรกโปรแกรม .....	18
ภาพประกอบที่ 1.9 หน้าเลือกพื้นที่ภาพ .....	19
ภาพประกอบที่ 1.10 หน้าผลลัพธ์ในการถ่ายภาพหน้าจอ .....	19
ภาพประกอบที่ 1.11 หน้ารายละเอียดข้อมูล .....	20
ภาพประกอบที่ 2.1 ตัวอย่างโครงสร้าง Flutter .....	21
ภาพประกอบที่ 2.2 การตรวจจับและประมวลผลใบหน้าจากภาพ [4] .....	25
ภาพประกอบที่ 2.3 ขบวนการการทำงานของ Optical Character Recognition .....	26
ภาพประกอบที่ 2.4 ภาพการตรวจจับหน่วงนิรภัยของคนงาน [8] .....	28
ภาพประกอบที่ 2.5 การแบ่งช่องของรูปภาพออกเป็น grid [9] .....	28
ภาพประกอบที่ 2.6 ภาพของเทคนิค Anchor Box [9] .....	29
ภาพประกอบที่ 2.7 การทำงานของขบวนการ IOU [9] .....	29
ภาพประกอบที่ 3.1 ขั้นตอนการดำเนินงานของระบบ .....	31
ภาพประกอบที่ 3.2 Streamlabs Desktop .....	32
ภาพประกอบที่ 3.3 ตัวอย่างรูปภาพที่ทำการเก็บ .....	33
ภาพประกอบที่ 3.4 การโหลดภาพผลเฉลยรูปภาพใน Roboflow .....	33
ภาพประกอบที่ 3.5 การโหลดภาพผลเฉลยรูปภาพใน Roboflow (2) .....	34
ภาพประกอบที่ 3.6 ภาพผลเฉลยที่ถูกต้องจากห้อง 3 ร้านค้า .....	34
ภาพประกอบที่ 3.7 การเพิ่มจำนวนรูปภาพใน Traing Set .....	35
ภาพประกอบที่ 3.8 ตัวอย่างรูปภาพที่ Generate ออกมานะ .....	36
ภาพประกอบที่ 3.9 ผลเฉลยของรูปภาพที่ Generate ออกมานะ .....	36
ภาพประกอบที่ 3.10 การทำ Convolution .....	37
ภาพประกอบที่ 3.11 Convolution Layer .....	37
ภาพประกอบที่ 3.12 การคำนวณ ReLU .....	38
ภาพประกอบที่ 3.13 กราฟของ ReLU .....	39
ภาพประกอบที่ 3.14 การคำนวณ ReLU Leaky .....	39

## สารบัญรูปภาพ(ต่อ)

	หน้า
ภาพประกอบที่ 3.15 กราฟของ ReLU Leaky .....	40
ภาพประกอบที่ 3.16 Pooling layer.....	40
ภาพประกอบที่ 3.17 การทำ Flatten .....	40
ภาพประกอบที่ 3.18 Fully Connected Layer .....	41
ภาพประกอบที่ 3.19 รูปภาพ input ที่ใส่ grid .....	41
ภาพประกอบที่ 3.20 ค่าผลเฉลย y ในแต่ละ grid cell .....	43
ภาพประกอบที่ 3.21 ตัวอย่าง grid ที่มีวัตถุในภาพและการแทนค่า .....	44
ภาพประกอบที่ 3.22 ตัวอย่าง grid ที่ไม่มีวัตถุในภาพและการแทนค่า .....	44
ภาพประกอบที่ 3.23 จุดกึ่งกลางของ Bounding Box .....	45
ภาพประกอบที่ 3.24 การแทนค่าใน anchor box และภาพรวมของ anchor box .....	46
ภาพประกอบที่ 3.25 IOU .....	47
ภาพประกอบที่ 3.26 ตัวอย่างการคำนวณค่า IOU .....	47
ภาพประกอบที่ 3.27 ตัวอย่างของ ค่า IOU ในรูปภาพ.....	48
ภาพประกอบที่ 3.28 ตัวอย่างการทำ Non-Max Suppression .....	49
ภาพประกอบที่ 3.29 ตัวอย่างการทำ Non-Max Suppression .....	49
ภาพประกอบที่ 3.30 Layer ของ Yolov5 .....	49
ภาพประกอบที่ 3.31 รูปภาพผลลัพท์ในขั้นตอนที่ 1 .....	50
ภาพประกอบที่ 3.32 รูปภาพผลลัพท์ในขั้นตอนที่ 2 .....	50
ภาพประกอบที่ 3.33 รูปภาพผลลัพท์ในขั้นตอนที่ 3 .....	51
ภาพประกอบที่ 3.34 รูปภาพผลลัพท์ในขั้นตอนที่ 4 .....	51
ภาพประกอบที่ 3.35 รูปภาพผลลัพท์ในขั้นตอนที่ 5 .....	52
ภาพประกอบที่ 3.36 รูปภาพผลลัพท์ในขั้นตอนที่ 6 .....	53
ภาพประกอบที่ 3.37 connection layer .....	53
ภาพประกอบที่ 3.38 ขั้นตอนการทำ NMS และแสดงผลลัพท์ .....	54
ภาพประกอบที่ 3.39 การทำงานในส่วนของการเลือกส่วนที่ต้องการ .....	54
ภาพประกอบที่ 3.40 ผลลัพท์ในการ detect และการเลือกพื้นที่ที่ต้องการ .....	55
ภาพประกอบที่ 3.41 ผลลัพท์ในการ detect ใน desktop application .....	55
ภาพประกอบที่ 3.42 ตัวอย่างภาพนำเข้า .....	56
ภาพประกอบที่ 3.43 ตารางค่าสี R G B ตามลำดับ .....	56
ภาพประกอบที่ 3.44 การแปลงภาพสี RGB ไปเป็นระดับเทา .....	57
ภาพประกอบที่ 3.45 ตัวอย่างผลลัพท์หลังจากการแปลงเป็นภาพระดับเทา .....	57
ภาพประกอบที่ 3.46 ตัวอย่างหลังจากการแปลงเป็นภาพระดับเทา .....	57
ภาพประกอบที่ 3.47 ตัวอย่างการแปลงภาพระดับ เทา เป็นภาพ Binary .....	59

## สารบัญรูปภาพ(ต่อ)

	หน้า
ภาพประกอบที่ 3.48 ตัวอย่างสแกนรูปภาพจากซ้ายไปขวา .....	60
ภาพประกอบที่ 3.49 ตัวอย่างการマーคเส้น .....	60
ภาพประกอบที่ 3.50 ผลลัพท์ในการวางแผนเส้นแยกบรรทัด .....	60
ภาพประกอบที่ 3.51 ตัวอย่างของผลลัพท์ของการแปลงที่ผิดพลาด .....	61
ภาพประกอบที่ 3.52 ตัวอย่างหน้าตา UI mobile application.....	62
ภาพประกอบที่ 3.53 การสร้างโปรเจ็ค firebase .....	63
ภาพประกอบที่ 3.54 การสร้าง Document ใน firebase.....	63
ภาพประกอบที่ 3.55 ตัวอย่างข้อมูลและประเภทของข้อมูล .....	64
ภาพประกอบที่ 3.56 Confusion Matrix.....	64
ภาพประกอบที่ 3.57 สมการหาค่า IoU .....	65
ภาพประกอบที่ 3.58 ตัวอย่างการหาค่า IoU.....	65
ภาพประกอบที่ 3.59 Predicted box และ Ground truth box.....	66
ภาพประกอบที่ 3.60 การหาค่า Precision.....	66
ภาพประกอบที่ 3.61 ตัวอย่างผลลัพท์ในการทำนาย .....	67
ภาพประกอบที่ 3.62 การหาค่า recall.....	67
ภาพประกอบที่ 3.63 กราฟการเปลี่ยนแปลงค่า mAP precision และ recall ของ model .....	67
ภาพประกอบที่ 3.64 สมการ CER .....	68
ภาพประกอบที่ 3.65 ตัวอย่างตัวแปรในสมการ CER.....	68
ภาพประกอบที่ 3.66 ตัวอย่างการหาค่า CER.....	68

## บทที่ 1

### บทนำ

#### 1.1 หลักการและเหตุผล

ปัจจุบันเทคโนโลยีอิเล็กทรอนิกส์ มีความสำคัญในการดำเนินชีวิตของผู้คนในปัจจุบันมากขึ้น ผู้คนสามารถได้ทำการใช้เทคโนโลยีในการค้าขายของออนไลน์ผ่านการไลฟ์และได้เกิดการพัฒนาของ ปัญญาประดิษฐ์ (AI : Artificial Intelligence) คือเครื่องจักรหรือเทคโนโลยีที่มีพัฒนาชั้นที่มี ความสามารถในการทำความเข้าใจ เรียนรู้องค์ความรู้ต่างๆ อาทิเช่น การรับรู้ การเรียนรู้ การให้เหตุผล และการแก้ปัญหาต่างๆ เครื่องมือที่มีความสามารถเหล่านี้ถือว่าเป็น ปัญญาประดิษฐ์ เพราะฉะนั้นจึง สามารถกล่าวได้ว่า AI ถือกำเนิดขึ้นเมื่อเครื่องจักรมีความสามารถที่จะเรียนรู้นั่นเอง

ปัจจุบันการที่ขายพัสดุออนไลน์ในการไลฟ์สดนั้น ใช้เวลาที่ค่อนข้างมากที่จะต้องให้ลูกค้าเข้า มารับชมเพื่อเลือกสินค้าที่ลงทะเบียนและทำให้ลูกค้าบางรายที่ไม่สะดวกมารับชมไม่ได้เลือกซื้อสินค้าที่ ต้องการจึงทำให้พวกร้าวได้ทำการพัฒนาแอปพลิชั่นรวมสินค้าไลฟ์สดขึ้นมาเพื่อแก้ไขปัญหาดังกล่าว

ดังนั้นเพื่อให้เกิดความสะดวกสบายพวกร้าวจึงได้พัฒนาแอปพลิเคชันเพื่อตอบโจทย์การใช้งาน ของแม่ค้าและลูกค้า ก็คือแอปพลิชั่นรวมสินค้าไลฟ์สดซึ่งจะทำการรวมเอาสินค้าต่างๆ ในไลฟ์สดของ แม่ค้าแต่ละคนมารวมไว้ในแอปพลิเคชันเดียว โดยการใช้ image processing ในการตรวจจับข้อมูล สินค้าในไลฟ์สด และนำข้อมูลมาเก็บในฐานข้อมูลและนำไปแสดงผลในแอปพลิเคชัน และเมื่อเลือก สินค้าที่ต้องการจะสามารถดูข้อมูลของสินค้าและไปสู่หน้าไลฟ์สดของแม่ค้าได้ เพื่อให้ลูกค้าได้เลือก สินค้ากันอย่างสะดวกและแม่ค้าก็ไม่เสียลูกค้าที่ไม่สะดวกรับชม

#### 1.2 วัตถุประสงค์ของโครงการ

สร้างแอปพลิชั่นที่สามารถใช้ AI ในการตรวจสอบและดึงข้อมูลของสินค้าในไลฟ์สดขายของเข้า มาในแอปพลิชันได้

#### 1.3 ขอบเขตของโครงการ

โครงการแอปพลิชั่นรวมสินค้าไลฟ์สดทำงานบนระบบปฏิบัติการ android แบ่งการทำงานของ ระบบออกเป็น 3 ส่วน ประกอบไปด้วย

### 1.3.1 ผู้ใช้งาน (ใช้งานผ่าน Mobile Application)

1. ผู้ใช้งานสามารถเข้าสู่ระบบผ่าน Facebook ได้
2. ผู้ใช้งานสามารถดูสถานะการไลฟ์สดของร้านแต่ละร้านค้าได้ ประกอบด้วย
  - สถานะ online / offline
  - รูปภาพในการไลฟ์สด ถ้าสถานะ online รูปจะเป็นสีปกติ และถ้าเป็นสถานะ offline รูปจะเป็นรูปภาพขาวดำ
3. ผู้ใช้งานสามารถดูรายการของสินค้าที่แสดงสินค้าของทุกร้านที่มีสถานะ online ทุกร้าน ใน Mobile Application ได้ โดยรายการที่ปรากฏในไลฟ์สด ประกอบไปด้วย
  - ชื่อร้านค้า
  - รหัสสินค้า
  - ชื่อสินค้า
  - ราคาสินค้า
  - วันเวลาที่ capture รูปภาพ

ตัวอย่าง เช่น ปีรันย่า บิวตี้ ช้อป คือชื่อร้าน ,n39 คือรหัสสินค้า, ติห้อกูปุ่น คือชื่อสินค้า ,39 คือราคาสินค้า, 16/07/2022 16:30 คือ วันเวลาที่ capture รูปภาพ

4. ผู้ใช้งานสามารถเลือกที่สินค้าเพื่อดูรายละเอียดเพิ่มเติมได้โดยรายการที่ปรากฏในไลฟ์สดประกอบด้วย

- ชื่อร้านค้า
- รหัสสินค้า
- ชื่อสินค้า
- ราคาสินค้า
- รูปภาพในการไลฟ์สด
- วันเวลาที่ capture รูปภาพ

ตัวอย่าง เช่น FIRST SHOP V2คือชื่อร้าน ,o7 คือรหัสสินค้า, scrub ขมิ้น คือชื่อสินค้า ,59 คือราคาสินค้า, 14/06/2022 12:30 คือ วันเวลาที่ capture รูปภาพ

5. เมื่อผู้ใช้เลือกสินค้าผู้ใช้งานสามารถกดปุ่ม “chat” ในหน้ารายละเอียดสินค้า เพื่อลิ้งไปยังแชท ของร้านค้าได้

6. เมื่อผู้ใช้เลือกสินค้าผู้ใช้งานสามารถกดปุ่ม “buy” ในหน้ารายละเอียดสินค้า เพื่อลิ้งไปยังแชท ของร้านค้าและมีการนำรหัสสินค้า ชื่อสินค้า ราคาสินค้า ส่งไปให้ร้านค้าอัตโนมัติ

7. ผู้ใช้งานสามารถค้นหาสินค้าจากชื่อสินค้าหรือชื่อร้านค้าได้ ในช่วงที่เวลา  
ร้านค้าໄລฟ์สต็อปตัวอย่าง เช่น

- ชื่อสินค้า
- ชื่อร้านค้า

ตัวอย่าง เช่น ลิป KO(เบอร์ 7) คือชื่อสินค้า, FIRST SHOP V2 คือชื่อร้าน

8. เมื่อร้านค้ามีสถานะ offline ผู้ใช้จะไม่สามารถมองเห็นสินค้าของร้านค้าที่ offline ได้

### 1.3.2 ผู้ดูแลระบบ (ใช้งานผ่าน Desktop Application ที่พัฒนาด้วย Python)

#### 1. ผู้ดูแลระบบสามารถเลือกร้านค้าที่ไฟฟ์สุดได้ 3 ร้านค้ามีร้านค้า

- FIRST SHOP V2
- KANYA SHOP ขายถูกทุกอย่าง
- มหาศจรรย์"วันของAunM

#### 2. ผู้ดูแลระบบสามารถหยุดร้านค้าที่กำลังประมวลผลในระบบและแจ้งเตือนไปยัง Application ของผู้ใช้งานได้

#### 3. ผู้ดูแลระบบสามารถดูข้อมูลรายการสินค้า ของแต่ละร้านค้าได้ ประกอบด้วย

- รหัสสินค้า
- ชื่อสินค้า
- ราคาสินค้า

ตัวอย่าง เช่น บ61 คือรหัสสินค้า, บิบ KO(เบอร์ 8) คือชื่อสินค้า, 59 คือราคาสินค้า

### 1.3.3 ระบบ (ระบบพัฒนาเป็น Module ด้วย Python)

#### 1. ระบบสามารถจับภาพหน้าจอไฟฟ์สุดได้ 3 ร้านค้ามีร้านค้า

- FIRST SHOP V2
- KANYA SHOP ขายถูกทุกอย่าง
- มหาศจรรย์"วันของAunM

#### 2. ระบบสามารถส่งสติ๊มมิ่งภาพหน้าจอไฟฟ์สุดเพื่อนำไปประมวลผลและนำไปแสดงในแต่พื้นที่แสดงผลแต่ละพื้นที่ใน Desktop Application ของผู้ดูแลระบบ ได้ 3 ร้านค้ามีร้านค้า

- FIRST SHOP V2
- KANYA SHOP ขายถูกทุกอย่าง
- มหาศจรรย์"วันของAunM

#### 3. ระบบสามารถตรวจจับกรอบของตัวอักษร รหัสสินค้า ชื่อสินค้า ราคาสินค้า ที่เป็นรายการรวมสินค้าจากไฟฟ์สุดได้โดยที่ขนาดไม่เล็กเกินไป

#### 4. ระบบสามารถตัดบรรทัดของข้อมูลในกรอบที่ตรวจจับมาได้

#### 5. ระบบลบข้อมูลสินค้าของร้านค้าที่มีสถานะ offline ได้

#### 6. ระบบสามารถแยกข้อมูลที่แปลงจากภาพเป็นตัวอักษรได้โดยรายการที่ปรากฏในไฟฟ์สุด ประกอบด้วย

- รหัสสินค้า
- ชื่อสินค้า
- ราคาสินค้า

ตัวอย่าง เช่น ป29 คือรหัสสินค้า, ดีท็อกซ์ปุ่นคือชื่อสินค้า, 39 คือราคาสินค้า

ป29      ดีท็อกซ์ปุ่น 39  
 รหัสสินค้า      ชื่อสินค้า      ราคา

ภาพประกอบที่ 1.1 ตัวอย่างข้อมูลที่นำมาแยก

7. ระบบสามารถนำข้อมูลไปจัดเก็บในฐานข้อมูลได้โดยรายการที่ปรากฏในไฟล์

สตประกอบด้วย

- ชื่อร้านค้า
- รหัสสินค้า
- ชื่อสินค้า
- ราคาสินค้า
- รูปภาพในการไฟล์สตด
- วันเวลาที่ capture รูปภาพ

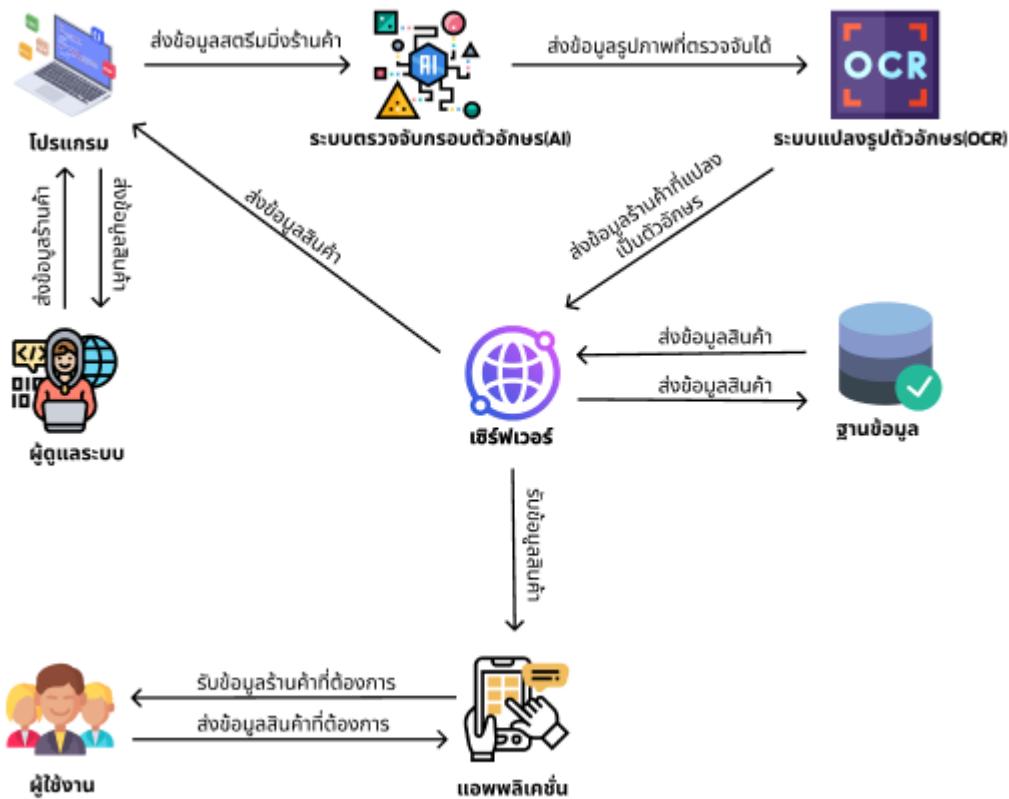
ตัวอย่าง เช่น ของฝากใต้ขายส่งราคากู คือชื่อร้าน, มม3 คือรหัสสินค้า, เม็ดมะม่วงหัก คือชื่อสินค้า, 89บ คือราคาสินค้า, 14/06/2022 14:30 คือ วันเวลาที่ capture รูปภาพ

8. ระบบจะทำงานแบบ real-time เป็น Database ของ Firebase

9. การวัดประสิทธิภาพประเมินจากความถูกผิดของการตรวจจับกรอบของ  
ข้อความและความถูกผิดของการทำ OCR(Optical Character Recognition) โดยวัดด้วย CER  
(character error rate)

10. ระบบสามารถแจ้งเตือนผู้ใช้งานได้ เมื่อร้านค้าที่มีสถานะ online
11. ระบบสามารถตรวจจับข้อความของไฟล์สตดที่จบแล้ว และหยุดการ  
ประมวลผลของร้านค้าที่จบไฟล์และแจ้งเตือนไปยัง Mobile Application ผู้ใช้งานได้
12. ระบบไม่สามารถตรวจจับไฟล์สตดที่ไม่มีกรอบข้อความได้
13. ระบบไม่สามารถตรวจจับกรอบตัวอักษรและตัวอักษรที่มีสีเหมือนกันได้

#### 1.4 ภาพรวมของระบบ



ภาพประกอบที่ 1.2 ภาพรวมของระบบ

จากภาพรวมระบบจะการทำงานดังนี้

1. ผู้ดูแลระบบต้องทำการเลือกร้านค้าที่ต้องการจำนวน 3 ร้านค้า ในโปรแกรมที่พัฒนาด้วย python ที่ปุ่ม “capture” ในตัวโปรแกรมเพื่อทำการครอบหน้าจอในส่วนที่จะนำไปสตรีม และสามารถกดปุ่ม “clear” เพื่อลบร้านค้าที่ต้องการได้

2. หลังจากได้หน้าจอสตรีมที่ต้องการระบบจะทำการตรวจจับกรอบตัวอักษรในสตรีมโดยใช้อัลกอริทึม Yolov5 และนำข้อมูลที่เป็นรูปภาพกรอบตัวอักษรมาตัดเป็นแฉว่าโดยจะใช้เทคนิคในการตรวจ pixel ในหน้าจอที่ลละเอียด เมื่อตรวจเจอช่องว่างระหว่างบรรทัด จะทำการตัดรูปเพื่อแยกข้อมูลออกจากกันและนำชุดข้อมูลสินค้าที่เป็นรูปภาพที่ได้ไปส่งต่อไปยังกระบวนการ แปลงภาพเป็นตัวอักษร

3. กระบวนการแปลงภาพเป็นตัวอักษรจะทำการนำชุดข้อมูลที่เป็นภาพมาแปลง เป็นชุดข้อมูลที่เป็นตัวอักษร โดยใช้ Tesseract ในกระบวนการแปลง

4. หลังจากได้ชุดข้อมูลที่เป็นตัวอักษร จะนำข้อมูลไปเก็บไว้ใน Database ของ firebase ที่มีลักษณะการทำงานแบบ real-time ผ่านตัว server โดยข้อมูลที่ส่งไปจะมี ชื่อร้านค้า รหัสสินค้า ชื่อสินค้า ราคสินค้า รูปภาพของไฟล์สต์ที่ทำการบันทึกมา ตัวอย่างของข้อมูลชื่อร้านค้า คือ ปีรันย่า บิวตี้ช้อป รหัสสินค้าคือ ก22, ชื่อสินค้าคือ ดีทอกซ์ญี่ปุ่น, ราคสินค้าคือ 39 บาท เป็นต้น

5. server จะทำการส่งข้อมูลสินค้าประกอบไปด้วย ชื่อร้านค้า รหัสสินค้า ชื่อสินค้า ราคสินค้า และรูปภาพของไฟล์สต์ ใน Database ไปแสดงผลใน Desktop Application ของผู้ดูแลระบบ

6. server จะทำการส่งข้อมูลสินค้าประกอบไปด้วย ชื่อร้านค้า รหัสสินค้า ชื่อสินค้า ราคสินค้า และรูปภาพของไฟล์สต์ ใน Database ไปแสดงผลใน mobile application ของผู้ใช้งาน

7. ผู้ใช้ต้องเข้าสู่ระบบผ่านเฟสบุ๊คจะสามารถถูก application เพื่อเลือกสินค้าที่ต้องการและรอแจ้งเตือนร้านที่กำลังไฟล์สต์ได้

8. เมื่อผู้ใช้งานกดปุ่ม “chat” ในหน้ารายละเอียดสินค้าเพื่อลิ้งไปยังแซท ของร้านค้าและผู้ใช้งานสามารถกดปุ่ม “buy” ในหน้ารายละเอียดสินค้าเพื่อลิ้งไปยังแซท ของร้านค้าและจะมีการนำรหัสสินค้า ชื่อสินค้า ราคสินค้า ส่งไปให้ร้านค้าอัตโนมัติ

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ช่วยให้ผู้ใช้งานที่เป็นผู้ค้าขายมีช่องทางการขายสินค้ามากขึ้น

1.5.2 ช่วยให้ผู้ใช้งานที่เป็นลูกค้ามีความสะดวกมากขึ้น

## 1.6 อุปกรณ์และเครื่องมือที่ใช้ในการดำเนินงาน

1.6.1 ฮาร์ดแวร์

คอมพิวเตอร์ 2 เครื่อง มีคุณสมบัติดังนี้

### 1. Acer Nitro5

- ระบบปฏิบัติการ Windows 11
- AMD Ryzen 5 3550H with Radeon Vega Mobile Gfx 2.10 GHz
- Installed RAM 16.0 GB (13.9 GB usable)

### 2. ROG STRIX

- ระบบปฏิบัติการ Windows 11
- Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz 2.59 GHz
- Installed RAM 16.0 GB (15.9 GB usable)

### 1.6.2 ซอฟต์แวร์

1. Python ใช้สำหรับการพัฒนาโปรแกรม
  2. Tesseract ใช้สำหรับการประมวลผลภาพ
  3. firebase ใช้เป็นฐานข้อมูลในการเก็บข้อมูลที่ใช้ระหว่างพัฒนาโปรแกรม
  4. Flutter เป็น Framework หลักสำหรับการพัฒนาแอปพลิเคชัน
  5. Android Studio ใช้สำหรับการสร้างแอปพลิเคชัน

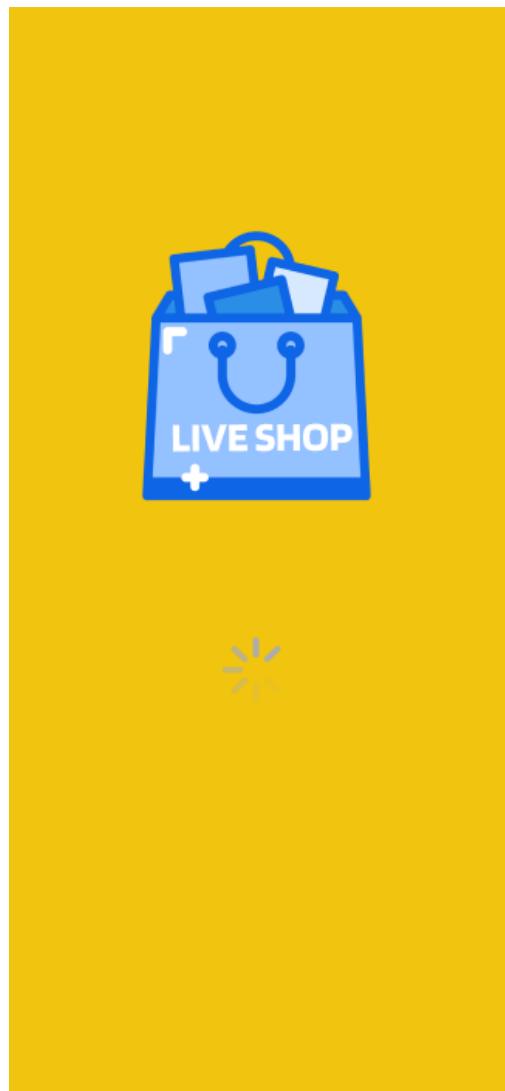
## 1.7 แผนการดำเนินงาน

# โครงการปริญญาบัณฑิตชั้นนำ ดำเนินงาน ณ คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม ระหว่างเดือน พ.ค. 2565 ถึง ก.พ. 2566

## ตารางที่ 1 แผนการดำเนินงาน

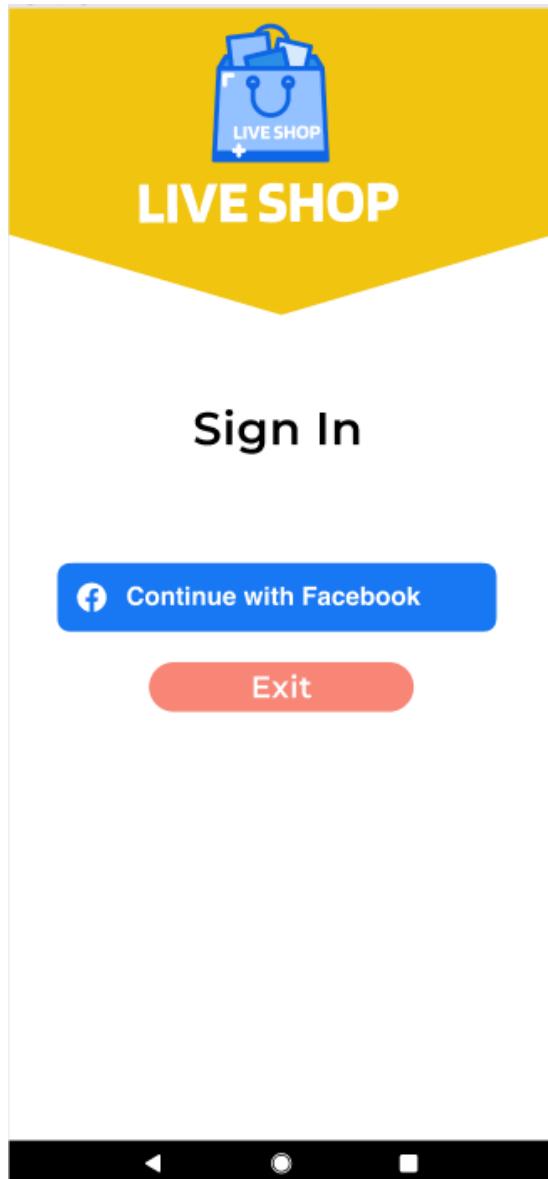
## 1.8 ตัวอย่างโปรแกรม

### 1.8.1 แอปพลิเคชั่นบนอุปกรณ์เรือสายแบบเคลื่อนที่



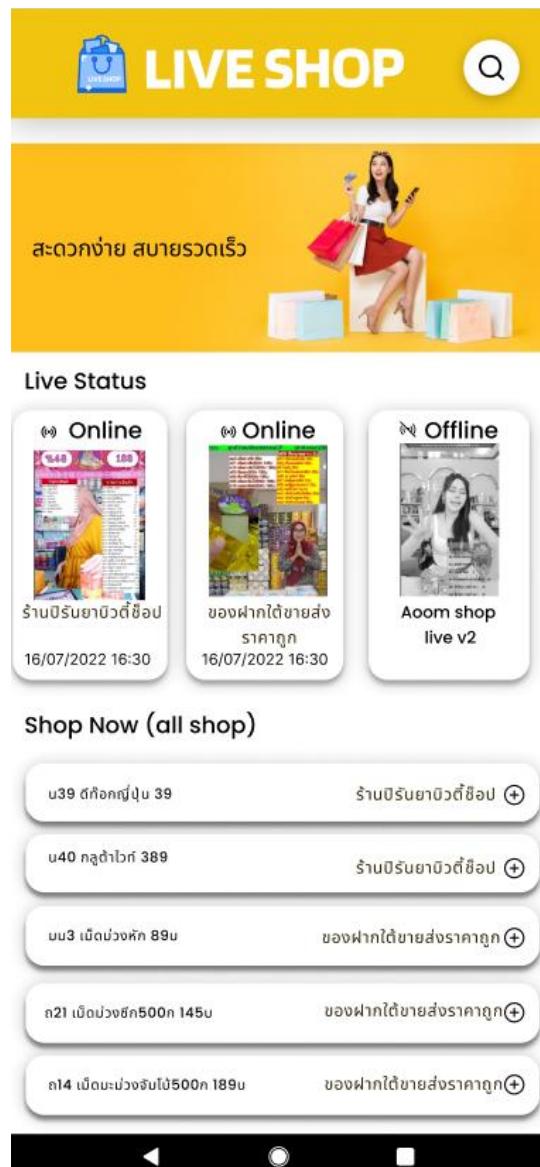
ภาพประกอบที่ 1.3 หน้าแรกของโปรแกรม

หน้าแรกของโปรแกรม จะมีการรอโหลดข้อมูลจาก Database เมื่อทำการโหลดข้อมูลเสร็จจะนำพาไปยังหน้าล็อกอินทันที



#### ภาพประกอบที่ 1.4 หน้าล็อกอินเข้าสู่ระบบ

หน้าล็อกอินเข้าสู่ระบบ จะมีปุ่ม 2 ปุ่ม “Continue with Facebook” เพื่อเข้าสู่ระบบผ่านแอปพลิเคชันเฟสบุ๊คเมื่อเข้าสู่ระบบสำเร็จจะนำไปยังหน้าหลักของแอปพลิเคชันและ ปุ่ม “Exit” เพื่อออกจากแอปพลิเคชัน



### ภาพประกอบที่ 1.5 หน้าหลักของแอปพลิเคชัน

หน้าหลักของแอปพลิเคชัน จะมีปุ่ม ”ค้นหา” กดเพื่อไปยังหน้าค้นหาสินค้า และ จะมีแท็บ ”Live Status” แสดงสถานะการไลฟ์สดของร้านทั้งสามร้านว่าร้านไหนกำลังออนไลน์หรืออฟฟ์ไลน์อยู่ และวันเวลาที่อัพเดทรูปภาพและจะมีแท็บ ”Shop Now” แสดงรายการสินค้าของร้านที่กำลังไลฟ์สดอยู่ ทั้งหมด



### ກາພປະກອບທີ 1.6 ທັນທີ່ ທັນທີ່ ທັນທີ່

ທັນທີ່ ທັນທີ່ ທັນທີ່ ຈະມີຊ່ອງເພື່ອໃສ່ຂໍ້ມູນຊ່ອງສິນຄ້າ ອີ່ຮ້າບປັບປຸງຕື່ອບ ທີ່ຕ້ອງການຄັນຫາ ແລະ ເມື່ອໃສ່  
ຂໍ້ມູນລົງໄປຈະມີສ່ວນໃນກາຮແສດງພລັພທ໌ ຂອງກາຮຄັນຫາເປັນຮາຍກາຮສິນຄ້າ ທີ່ມີມາແລະ ເມື່ອເລືອກຮາຍກາຮ  
ສິນຄ້າຈະໄປຢັງທັນທີ່ ແລະ ເອີ້ດສິນຄ້າທັນທີ່



### ภาพประกอบที่ 1.7 หน้ารายละเอียดสินค้า

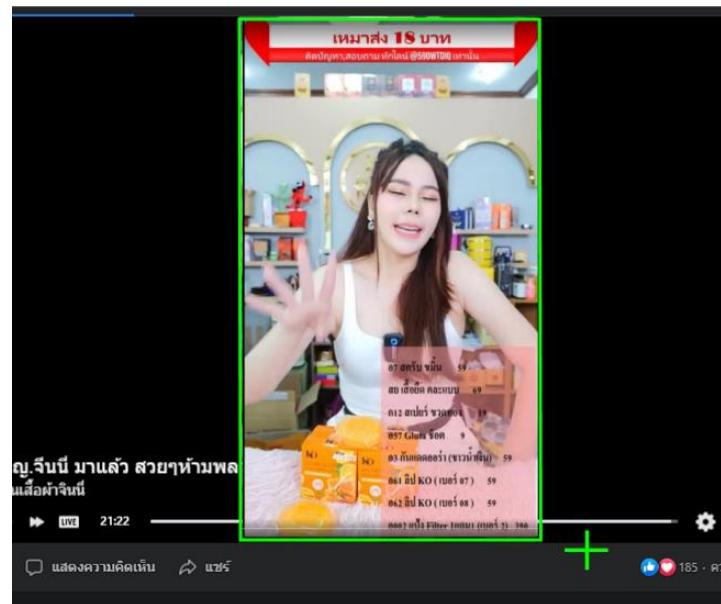
หน้ารายละเอียดสินค้า จะแสดงรูปภาพที่แคปมาจากไลฟ์สดและวันเวลาอัพเดทรูปภาพและแสดงชื่อร้านค้า ข้อมูลของสินค้าประกอบไปด้วย รหัสสินค้า ชื่อสินค้า ราคา และจะมีปุ่ม 2 ปุ่ม “Chat” เพื่อลิ้งไปยังแชทของร้านค้าและ “Buy” เพื่อลิ้งไปยังแชทของร้านค้าและนำข้อมูลสินค้าส่งไปอัตโนมัติ

### 1.8.2 แอปพลิเคชันบนคอมพิวเตอร์สำหรับผู้ดูแลระบบ



ภาพประกอบที่ 1.8 หน้าแรกโปรแกรม

หน้าแรกโปรแกรม มีพื้นที่ในการแสดงผลสำหรับสตรีม 3 ส่วน และ มีปุ่ม 3 ปุ่ม “capture” เพื่อเลือกครอบจากหน้าไฟล์สตูดิโอที่ต้องการนำมาประมวลผล ปุ่ม “clear” เพื่อทำการลบสตรีมที่ต้องการยกเลิก ประมวลผล และปุ่ม “detail” เพื่อทำการแสดงข้อมูลรหัสสินค้า ชื่อสินค้า และราคาสินค้า ที่ระบบทำการประมวลผลแปลงรูปภาพเป็นข้อความออกมายังเป็น Popup ที่บริเวณหน้าจอ



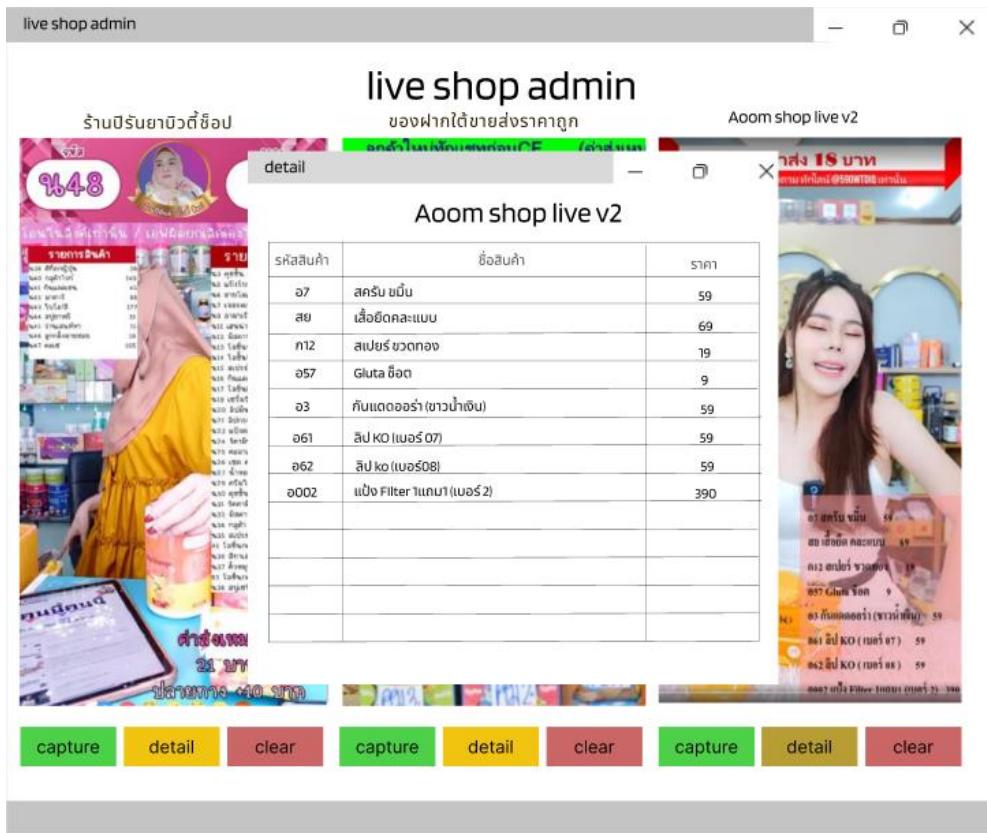
ภาพประกอบที่ 1.9 หน้าเลือกพื้นที่ภาพ

เมื่อกดปุ่ม “capture” แล้วต้องลาก cursor ครอบพื้นที่ที่ต้องการสตรีมจากไฟล์สดเพื่อนำไป ประมวลผล



ภาพประกอบที่ 1.10 หน้าผลลัพธ์ในการถ่ายภาพหน้าจอ

## ผลลัพธ์ในการถ่ายภาพหน้าจอ ของปุ่ม Capture



### ภาพประกอบที่ 1.11 หน้ารายระเอียดข้อมูล

หน้ารายละเอียดข้อมูล เมื่อกดที่ปุ่ม "detail" ในส่วนของสต็อคเมื่อนั้นๆ จะทำการแสดงข้อมูลรหัสสินค้า ชื่อสินค้า และราคาสินค้า ที่ระบบทำการประมวลผลแปลงรูปภาพเป็นข้อความ ออกมาโดยเป็น Popup ที่ปริเวณหน้าจอ

## บทที่ 2

### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

#### 2.1 ทฤษฎีที่เกี่ยวข้อง

##### 2.1.1 Flutter

Flutter [1] คือ Framework ที่ใช้สร้าง UI สำหรับ mobile application ที่สามารถทำงานได้ทั้ง IOS และ Android ในเวลาเดียวกันโดยที่ใช้ source code ตัวเดียวกัน โดยภาษาที่ใช้ใน Flutter นั้นจะเป็นภาษา dart ซึ่งถูกพัฒนาโดย Google และ ยังเป็น open source ที่สามารถใช้งานได้ฟรี ตัวอย่าง syntax ของภาษา dart ที่ใช้ใน Flutter ซึ่งจะมีความคล้ายกับภาษา Java เนื่องจาก dart เป็นภาษาที่รองรับ OOP และมีแนวคิด เช่นเดียวกับภาษา Java

```
1 import 'package:flutter/material.dart';
2
3 void main() {
4     runApp(new MaterialApp(
5         home: new MyApp(),
6     ));
7 }
8
9 class MyApp extends StatelessWidget {
10     @override
11     Widget build(BuildContext context) {
12         return new Scaffold(
13             appBar: new AppBar(
14                 title: new Text("Example App"),
15                 backgroundColor: Colors.blue,
16             ),
17             backgroundColor: Colors.blue,
18             body: new Center(
19                 child: new Column(
20                     mainAxisAlignment: MainAxisAlignment.center,
21                     children: <Widget>[
22                         new Icon(Icons.favorite, color: Colors.redAccent, size: 200.0,
23                         ),
24                     ],
25                 ),
26             ),
27         );
28     }
29 }
```

#### ภาพประกอบที่ 2.1 ตัวอย่างโครงสร้าง Flutter

ซึ่งหากสังเกตจาก จะเห็นว่า Flutter นั้นจะมี Widget พื้นฐานมาให้ เพื่อทำให้ การออกแบบ UI มีความง่าย และสะดวกยิ่งขึ้น โดย Widget พื้นฐานของ Flutter หลัก ๆ จะมีอยู่ 2

ชนิดคือ StatelessWidget และ StatefulWidget โดยที่ StatelessWidget จะใช้สร้าง Widget ที่ไม่มีการจัดการสถานะการทำงานใดหรือหน้าจอจะไม่มีการเปลี่ยนแปลง เช่น การแสดงข้อความ, Icon หรือรูปภาพที่ไม่มี animation เข้ามาเกี่ยวข้องเป็นต้น ส่วน StatefulWidget จะใช้สร้าง Widget ที่มีการจัดการสถานการณ์การทำงานต่างๆ หรือมีปุ่มที่มี action เช่น การสร้าง Icon ที่มีการใส่ animation ให้สามารถขยับไปมาได้, ปุ่มกดต่างๆ บนหน้า UI เป็นต้น

จุดเด่นหลัก ๆ ของ Flutter คือ ระบบ Hot Reload โดยเมื่อมีการทดสอบ, การสร้าง, การ add features หรือการกระทำต่าง ๆ กับ UI จะต้องมีการ reload เพื่อให้หน้า UI update ซึ่งระบบ Hot Reload จะเข้ามาช่วยในส่วนของการ reload โดยจุดเด่นของระบบนี้คือการย่นระยะเวลาที่ใช้ในการ reload ให้เหลือเพียงเสี้ยววินาทีเท่านั้น ทำให้การพัฒนา UI ของ application มีความรวดเร็วขึ้นอย่างมาก และยังมีจุดเด่นอีก 1 ที่ช่วยให้การพัฒนาเป็นไปได้่ายขึ้นไม่ว่าจะเป็น Build-In ที่ช่วยในการออกแบบ UI ให้มีความสวยงามยิ่งขึ้นอย่าง Material Design และ Cupertino (iOS-flavor), มี Framework ที่ช่วยให้การทำ animation ต่าง ๆ หรือ gesture ของ UI เป็นเรื่องง่ายยิ่งขึ้น และยังสามารถใช้งานร่วมกับ IDE ที่กำลังเป็นที่นิยมอยู่ในปัจจุบันอย่าง VS Code และ Android Studio ได้

ข้อเสียหลัก ๆ คือ การใช้ภาษา dart ในการเขียน ซึ่งคนส่วนใหญ่อาจจะยังไม่คุ้นเคยกับ syntax ของภาษา dart ประกอบกับ community ยังเล็กเนื่องจาก Flutter ยังเปิดตัวมาได้ไม่นานนัก เมื่อเทียบกับ Framework ตัวอื่น ๆ อย่าง React Native ที่มี community ค่อนข้างใหญ่จึงทำให้ document ต่าง ๆ ยังไม่ยอดเยี่ยมเท่าที่ควร ทำให้เวลา มีปัญหาเกี่ยวกับการใช้งานอาจจะต้องใช้เวลาในการหาวิธีแก้

### 2.1.2 Firebase

เป็นหนึ่งในผลิตภัณฑ์ของ Google โดย Firebase [2] คือ Platform ที่รวมเครื่องมือต่าง ๆ สำหรับการจัดการในส่วนของการเก็บข้อมูล ซึ่งทำให้สามารถ สร้าง Mobile Application ได้อย่างมีประสิทธิภาพ และยังลดเวลาและค่าใช้จ่ายของการทำ Server side หรือการวิเคราะห์ข้อมูลให้อีกด้วย โดยมีทั้งเครื่องมือที่ฟรี และเครื่องมือที่มีค่าใช้จ่าย Firebase มีบริการให้ใช้หลายอย่าง สามารถแบ่งเป็นหมวดหมู่ดังนี้

#### 1.) Better Apps

- Cloud Firestore จัดเก็บและซิงค์ข้อมูลระหว่างผู้ใช้และอุปกรณ์ในระดับโลกโดยใช้ฐานข้อมูล NoSQL ที่โไฮสต์บคลาวด์ Cloud Firestore ให้การซิงโครไนซ์แบบสดและการสนับสนุนออฟไลน์พร้อมกับการสืบค้นข้อมูลที่มีประสิทธิภาพ การผสานรวมกับผลิตภัณฑ์ Firebase อื่นๆ ช่วยให้คุณสร้าง แอปแบบไฮบริดได้อย่างแท้จริง

- Authentication จัดการผู้ใช้ของคุณด้วยวิธีที่ง่ายและปลอดภัย Firebase Auth มีหลายวิธีในการตรวจสอบสิทธิ์รวมถึง Email และ password ผู้ให้บริการบุคคลที่สาม เช่น Google หรือ Facebook และใช้ระบบบัญชีที่คุณมีอยู่โดยตรง สร้างอินเทอร์เฟชของคุณเองหรือใช้ประโยชน์จาก UI ที่อเปนซอร์ส ที่ปรับแต่งได้อย่างเต็มที่

- Hosting ลดความซับซ้อนของเว็บไซต์ของคุณด้วยเครื่องมือที่สร้างขึ้นเฉพาะสำหรับเว็บ แอปสมัยใหม่ เมื่อคุณอัปโหลดเนื้อหาเว็บของคุณเราจะส่งเนื้อหาเหล่านั้นไปยัง CDN ทั่วโลกของเราโดย อัตโนมัติและมอบให้ SSL ฟรีเพื่อให้ผู้ใช้ของคุณได้รับประสบการณ์ที่ปลอดภัยเชื่อถือได้และมี เวลาแฝงต่ำไม่ว่าจะอยู่ที่ใดก็ตาม

- Realtime Database Realtime Database คือฐานข้อมูลตั้งเดิมของ Firebase เป็นโซลูชันที่มีประสิทธิภาพและมีเวลาแฝงต่ำสำหรับแอปบนอุปกรณ์เคลื่อนที่ที่ต้องการสถานะการซิงค์ระหว่าง คลาวน์แบบเรียลไทม์ เราขอแนะนำ Cloud Firestore แทน Realtime Database สำหรับนักพัฒนา ส่วนใหญ่ที่เริ่มโปรเจกต์ใหม่

## 2.) Improve app quality

- Crashlytics ลดเวลาในการแก้ไขปัญหาของคุณด้วยการเปลี่ยนข้อขัดข้องจากหิมะถล่มให้ เป็นรายการปัญหาที่จัดการได้ รับข้อมูลเชิงลึกที่ชัดเจนและนำไปปฏิบัติได้ว่าปัญหาใดที่ต้องจัดการก่อน โดยเห็นผลกระทบของผู้ใช้ในแดชบอร์ด Crashlytics การแจ้งเตือนแบบเรียลไทม์จะช่วยให้คุณมีความ เสถียรแม่นยำในขณะเดินทาง Crashlytics เป็นตัวรายงานข้อขัดข้องหลักของ Firebase

- Performance Monitoring วินิจฉัยปัญหาประสิทธิภาพของแอปที่เกิดขึ้นบนอุปกรณ์ของ ผู้ใช้ ใช้การติดตามเพื่อตรวจสอบประสิทธิภาพของบางส่วนของแอปและดูมุมมองสรุปในคอนโซล Firebase อยู่หนึ่งเดียวเพื่อติดตามและตรวจสอบความล่าช้าของแอปและตรวจสอบคำขอ HTTP โดยไม่ต้องเขียนโค้ดใด ๆ

- Test Lab เรียกใช้การทดสอบอัตโนมัติและกำหนดเองสำหรับแอปของคุณบนอุปกรณ์ เสมือนและจริงที่ไฮสต็อคโดย Google ใช้ Firebase Test Lab ตลอดจนการพัฒนาของคุณเพื่อค้นหา จุดบกพร่องและความไม่สอดคล้องกันเพื่อให้คุณสามารถนำเสนอประสบการณ์ที่ยอดเยี่ยมบนอุปกรณ์ หลากหลายประเภท

## 3.) Grow your business

- Google Analytics วิเคราะห์คุณลักษณะและพฤติกรรมของผู้ใช้ในแดชบอร์ดเดียวเพื่อทำการตัดสินใจอย่างชาญฉลาดเกี่ยวกับแผนงานผลิตภัณฑ์ของคุณ รับข้อมูลเชิงลึกแบบเรียลไทม์จาก รายงานหรือส่องออกข้อมูลเหตุการณ์ตัวเป็น Google BigQuery สำหรับการวิเคราะห์ที่กำหนดเอง

- Remote Config กำหนดวิธีการแสดงผลแอปของคุณสำหรับผู้ใช้แต่ละคน เปลี่ยนรูปลักษณ์ เปิดตัวฟีเจอร์ที่ลับน้อยเรียกใช้การทดสอบ A / B ส่งมอบเนื้อหาที่กำหนดเองให้กับผู้ใช้บางรายหรือทำ

การอัปเดตอื่น ๆ โดยไม่ต้องปรับใช้เวอร์ชันใหม่ทั้งหมดนี้ทำได้จากคอนโซล Firebase ตรวจสอบผลกระทบของการเปลี่ยนแปลงของคุณและทำการปรับเปลี่ยนในเวลาไม่เกินนาที

- Cloud Messaging ส่งข้อความและการแจ้งเตือนไปยังผู้ใช้ข้ามแพลตฟอร์มทั้ง Android, iOS และเว็บได้ฟรี สามารถส่งข้อความไปยังอุปกรณ์เดียวกันอุปกรณ์หรือหัวข้อเฉพาะหรือกลุ่มผู้ใช้ Firebase Cloud Messaging (FCM) ปรับขนาดเป็นแอปที่ใหญ่ที่สุดโดยส่งข้อความหลายส่วนล้านข้อความต่อวัน

### 2.1.3 การประมวลผลภาพ (Image Processing)

การประมวลผลภาพ (Image Processing) [3] เป็นการนำภาพมาประมวลผลหรือคิดคำนวณด้วย คอมพิวเตอร์แล้วใช้กรัมวิธีใดๆ มากระทำกับข้อมูลภาพเพื่อให้ได้ภาพที่มีคุณสมบัติตามต้องการทั้งในเชิง คุณภาพและปริมาณ โดยมีขั้นตอนต่างๆ ที่สำคัญ คือ การทำให้ภาพมีความคมชัดมากขึ้น การกำจัดสัญญาณรบกวนออกจากภาพ การแปลงส่วนของวัตถุที่สนใจออกมาจากภาพ เพื่อนำภาพวัตถุที่ได้ ไปวิเคราะห์หาข้อมูลเชิงปริมาณ เช่น ขนาด รูปร่าง และทิศทางการเคลื่อนของวัตถุในภาพ คอมพิวเตอร์มีความสามารถในการคำนวณและประมวลผลข้อมูลจำนวนมหาศาลได้ในเวลาอันสั้นจึงมีประโยชน์อย่างมากในการเพิ่มประสิทธิภาพการประมวลผลภาพและวิเคราะห์ข้อมูลที่ได้จากการในระบบ ลักษณะของการใช้งานจะมีเป็น

#### 1. การรู้จำภาพ (Image Recognition)

การรู้จำภาพ (Image Recognition) การรู้จำภาพเป็นกระบวนการในการระบุและการตรวจจับวัตถุหรือคุณสมบัติที่มีอยู่ในระบบดิจิตอล ซึ่งได้จากการหรือวิดีโอแนวคิดนี้ได้ถูกนำมาใช้ในแอ�플ิเคชั่นมากมายที่เมื่อนักในระบบสำหรับการทำงานของระบบอัตโนมัติในโรงงานการตรวจสอบด้านเก็บเงินด้วยการรักษาความปลอดภัยและการรักษาความปลอดภัย อัลกอริทึมการรู้จำภาพทั่วไป

#### 2. การสกัดคุณลักษณะเด่น (Feature Extraction)

การสกัดคุณลักษณะเด่น (Feature Extraction) เป็นการหาลักษณะเด่นที่ได้จากการโดยผลลัพธ์ที่ได้จากการนำไปหาลักษณะเด่นจะปั่งบอกรถึงลักษณะ เช่น ความสูง ความยาว สีหรือเวคเตอร์ ที่เป็นคุณลักษณะ ที่สำคัญของรูปภาพ เป็นต้น นอกเหนือจากข้อมูลคุณลักษณะดังกล่าว ในปัจจุบันยัง มีวิธีการการสกัดหา ลักษณะเด่นที่นิยมใช้กันอย่างแพร่หลายที่ใช้ในการต่อภาพคือ การวางแผนตัวของเส้นขอบ(Edge) การหามุม (Corner) และการหาพื้นที่น่าสนใจ (Blob) เป็นต้น



ภาพประกอบที่ 2.2 การตรวจจับและประมวลผลใบหน้าจากภาพ [4]

ตัวอย่างการ นำการประมวลผลภาพไปใช้งาน เช่น

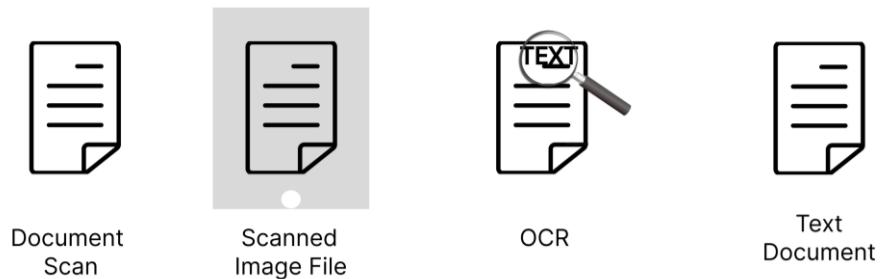
- ระบบตรวจจับใบหน้าในกล้องดิจิตอล โดยกล้องจะมีระบบตรวจว่าส่วนไหนของภาพมี ลักษณะคล้ายใบหน้า แล้วก็องก์จะทำการโฟกัสตำแหน่งที่ตรวจจับเพื่อภาพมีความคมชัดมากขึ้น เช่น ระยะห่างระหว่างคิวมุมปาก โหนกแก้ม จมูก โครงหน้า
- ระบบตรวจจับการเคลื่อนไหวเพื่อรักษาความปลอดภัย โดยกล้องจะเปรียบเทียบภาพก่อน หน้าและภาพปัจจุบันถ้ามีส่วนใดส่วนใดเปลี่ยนแปลงระบบจะบันทึกเฉพาะภาพที่มีการเปลี่ยนแปลงทำให้ประยุตต์ เนื้อที่ในการเก็บภาพและสามารถตรวจสอบได้ภาพหลัง
- ระบบอ่านคิวบาร์โค้ด โดยระบบจะอ่านรหัสจากบาร์โค้ด แล้วแปลงเป็นข้อมูลซึ่งสามารถ นำมาใช้งานได้ เช่น QR code, Microsoft tag ภาพที่นำมาประมวลผลนั้นเป็นได้ทั้งภาพนิ่งและ ภาพเคลื่อนไหว ในการนำภาพนิ่งเข้ามาประมวลผลนั้น โปรแกรมจะอ่านไฟล์ภาพขึ้นมาแล้วถอดรหัสจาก ค่าสีของภาพ แต่จะเป็นตัวเลขเพื่อนำมาประมวลผลสำหรับการประมวลผลภาพเคลื่อนไหวนั้น โปรแกรมจะมองเห็นเป็นภาพนิ่งหลายภาพที่เรียงต่อกัน จึงมีประโยชน์อย่าง มากในการเพิ่ม ประสิทธิภาพการประมวลผลภาพและวิเคราะห์ข้อมูลได้จากภาพในระบบต่าง ๆ ดังกล่าวข้างต้น การประมวลผลภาพสามารถจำแนกตาม วัตถุประสงค์ในการใช้งานดังนี้
- การปรับปรุงคุณภาพของภาพ (Image Enhancement and Restoration) เช่น ปรับภาพ ให้คมชัดมากขึ้นการปรับ Contrast หรือการปรับเน้นเส้นขอบภาพ (Edge Enhancement) หรือการ กรองสัญญาณรบกวน (Image Filtering) เพื่อกำจัดสัญญาณรบกวนได้
- การบีบอัดภาพ (Image Compression) เนื่องจากข้อมูลภาพนั้นมีขนาดใหญ่มากโดยเฉพาะ ภาพสี ซึ่งจะ ทำให้การจัดเก็บหรือรับส่งไฟล์ข้อมูลขนาดใหญ่เสียเวลา many เพื่อทำให้ขนาดภาพเล็กลงจึง ทำให้

การบีบอัด ข้อมูลภาพ เช่น JPEG หรือ GIF และตัวอย่างการบีบอัดข้อมูลวิดีโอ เช่น MPEG หรือ AVI เป็นต้น

- การวิเคราะห์ข้อมูลภาพ (Image Analysis) เป็นการสังเคราะห์ ข้อมูลที่มีความหมายจากภาพ แล้วนำไปใช้งาน ซึ่งอาจต้องมีรู้จักรูป (Object recognition) การแยกส่วนวัตถุ 317 The Tenth National conference on Computing and Information Technology NCCIT4 (segmentation) ตรวจจับการ เคลื่อนไหวที่ของวัตถุ (motion detection)

#### 2.1.4 OCR หรือ Optical Character Recognition

OCR หรือ Optical Character Recognition [4] หรือ "การรู้จำอักษรด้วยแสง" เป็นเทคโนโลยีที่แปลงเอกสาร รูปภาพ หรือแม้แต่ ลายเซ็นอิเล็กทรอนิกส์ ให้อยู่ในรูปแบบตัวอักษร หรือข้อความ (Plain Text) ไฟล์ดิจิทัลที่สามารถสืบค้นได้ด้วยคำค้นหา (Keyword) ยกตัวอย่าง ไฟล์เอกสารที่มาในรูปแบบ PDF มีข้อดีก็คือไม่สามารถปลอมแปลง แก้ไขได้ แต่ถ้าต้องการแก้ไขข้อความใด ๆ ในนั้น ก็เป็นเรื่องยากลำบาก วิธีแก้ไขปัญหาก็คือ แปลงไฟล์ PDF ให้เป็น OCR ซึ่งวิธีการที่ทุกคนคุ้นเคยก็คือ การนำแผ่นเอกสาร มาสแกนข้อความ ตัวอักษร เส้นตารางและอื่น ๆ ให้เป็นไฟล์ Word หรือ PDF ด้วยเครื่องสแกนเนอร์ หรือกล้องถ่ายภาพ



#### ภาพประกอบที่ 2.3 ขั้นตอนการการทำงานของ Optical Character Recognition

ส่วนการทำงานของ OCR นั้น หลัก ๆ คือ ใช้การจดจำรูปแบบ เพื่อกำหนดอักษรของแต่ละประเภทไฟล์ จากนั้น ซอฟต์แวร์จะทำการอ่านข้อความและอักษร แล้วแปลงเป็นไฟล์ที่สืบค้นได้ นอกจากนี้ การทำงานของ OCR ยังขึ้นอยู่กับซอฟต์แวร์ที่ใช้ประมวลผล เพราะนอกเหนือจากการแปลงรูปภาพ อักษรจะได้ ๆ เป็นข้อความแล้ว ซอฟต์แวร์บางตัวสามารถจัดวางรูปแบบข้อความในไฟล์ OCR ได้

### 2.1.1 Object Detection

Object Detection [5] การตรวจจับวัตถุ คือ เทคโนโลยีในทางคอมพิวเตอร์ หลักการที่เกี่ยวกับ Computer Vision และ Image Processing ที่ใช้ในงาน AI ตรวจจับวัตถุชนิดที่กำหนด เช่น มุขย์ รถยนต์ อาคาร ที่อยู่ในรูปภาพ หรือวิดีโองาน Object Detection การตรวจจับวัตถุในรูปภาพ สามารถเจาะลึกลงไปได้อีกหลายชั้น เช่น การทำ ตรวจจับหน้าคน ตรวจจับคนเดินถนน สามารถประยุกต์ใช้ได้หลากหลาย เช่น ใช้ในงานรักษาความปลอดภัย และรถยนต์เรือคันขับ เป็นต้น

#### 2.1.1 YOLO (You Only Look Once)

เป็นอัลกอริทึม [6] ที่นำแนวความคิดของการทำนายตำแหน่งและขนาดของ กล่องจากความน่าจะเป็นที่กล่องนั้น จะเป็นกรอบล้อม วัตถุ แต่สิ่งที่ YOLO มีความสามารถและความเร็วเหนือกว่าอัลกอริทึมอื่น เช่น Faster R-CNN ที่มีการ ทำงานในลักษณะที่จะทำนายตำแหน่งของกรอบล้อม วัตถุและค่อยนิ่งไว้วัตถุในกล่องนั้น ไปผ่านแบบจำลองเพื่อ ทำนายวัตถุในกล่อง แต่ YOLO นั้นจะทำนายทั้งกรอบ ล้อมวัตถุ และความน่าจะเป็นของวัตถุบางส่วนที่อยู่ใน กรอบอ กมาพร้อมกันที่เดียว YOLO จัดว่าเป็นเทคนิคการตรวจจับวัตถุในภาพซึ่ง เป็นซอฟต์แวร์เปิดสำหรับงานปัญญาประดิษฐ์แบบ โครงข่ายประสาท (Neural Network) ที่พัฒนาด้วย โปรแกรมภาษา C++ และสามารถทำงานบน หน่วยประมวลผล CUDA ของ GPU ได้เป็นอย่างดี หมายเหตุ กับ การประมวลผลภาพแบบ Real Time ภาพจากกล้องหรือ วิดีโอ ปัจจุบัน YOLO มี การพัฒนามาแล้ว 5 เวอร์ชันปัจจุบันคือ YOLO v5 โครงสร้างโครงข่ายประสาทเทียมของ YOLO แต่ละ เวอร์ชันจะมี Convolution Box ที่ แตกต่างกัน แต่ หลักการโดยทั่วไปแล้ว YOLO จะแบ่งภาพออกเป็น Grid Cell เล็กๆ และแต่ละ Grid Cell จะถูกทำนาย ผ่าน แบบจำลอง เพื่อหาตำแหน่งจุดกึ่งกลางของวัตถุ และ ความน่าจะเป็นที่จะมีวัตถุใด ใน Grid Cell

## 2.2 ระบบงานที่เกี่ยวข้อง

### 2.2.1 Safety Helmet Detection Based on YOLOv5

"Safety Helmet Detection Based on YOLOv5" [7] เป็นงานวิจัยของ Fangbo Zhou, Huailin Zhao , Zhen Nie มหาวิทยาลัย Shanghai Institute of Technology [7] ที่พัฒนาระบบ ตรวจสอบความปลอดภัยของคนงาน ซึ่งระบบจะทำการตรวจจับภาพของคนงาน และ จะแสดงกรอบที่มีคำอธิบาย โดยได้นำรูปภาพจากอินเทอร์เน็ตจำนวน 6045 ภาพมาทำการทดสอบ โดยศีรษะของคนงานที่ไม่สวมหมวกนิรภัยจะมีคำอธิบายว่า "Alarm" และสำหรับศีรษะของคนงานที่สวมหมวกนิรภัยจะมีคำอธิบายว่า "Helmet" โดยใช้ algorithm ที่ทันสมัยที่สุดในการตรวจสอบ algorithm งาน ตรวจจับวัตถุถูกใช้อย่างแพร่หลายในความเป็นจริง เป้าหมายของการตรวจจับคือการค้นหาวัตถุทั้งหมด

ที่น่าสนใจในภาพ ซึ่งจะมี 2 งานย่อยคือการกำหนดหมวดหมู่'และการระบุตำแหน่งของวัตถุ แม้ว่า อัลกอริธึมการตรวจสอบแบบดั้งเดิมสามารถทำงานได้ผลลัพธ์ที่ดี ในบางสถานการณ์ในสภาพแวดล้อมที่ แปรปรวน เช่น การเปลี่ยนแปลงของสภาพอากาศ การกระจายตัวของคนงานที่ไม่สม่ำเสมอ และหากกันน็อคแบบต่างๆ ทำให้ความแม่นยำนั้นรับประทานได้ยากว่ามันสามารถทำงานได้ถูกต้อง จึง ได้นำตัวYOLOv5มาใช้ซึ่งมีความเร็วและความแม่นยำสูง



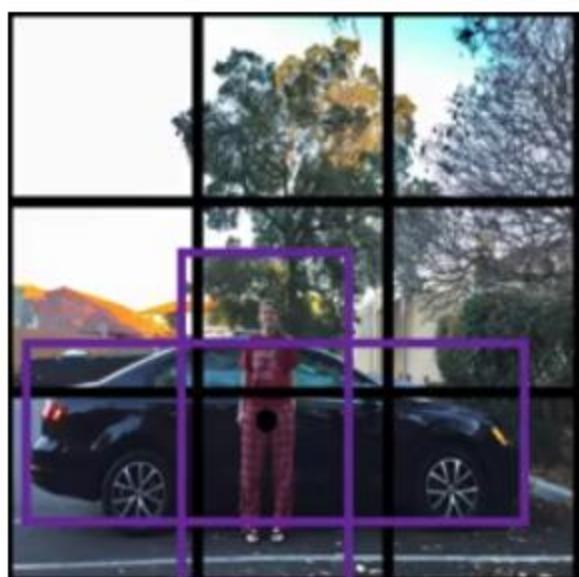
ภาพประกอบที่ 2.4 ภาพการตรวจสอบหมากนิรภัยของคนงาน [8]

โดยการทำงานของYOLOv5 [8]คือจากรูป 1 รูปเต็มๆ จะทำการแบ่ง Grid cell ออกมานเป็น  $n \times n$  grid ยิ่งแบ่งมากก็จะละเอียด trade off กับการคำนวณ



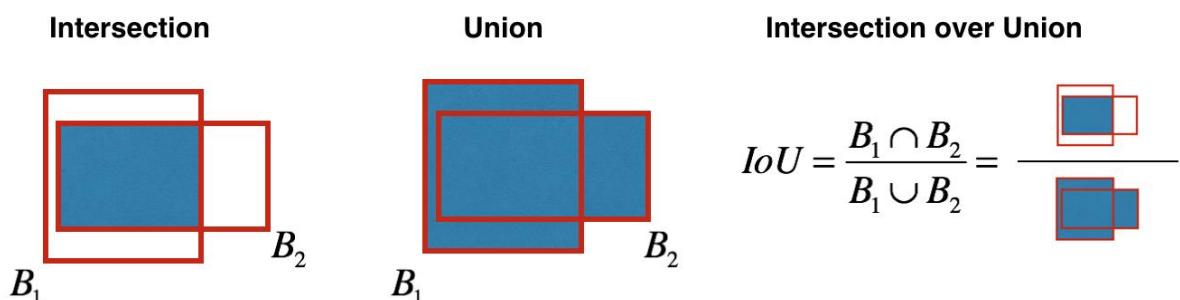
ภาพประกอบที่ 2.5 การแบ่งช่องของรูปภาพออกเป็น grid [9]

และในแต่ละ grid จะมี Label ทุกช่อง เช่น [Pc, bx, by, bh, bw, c1, c2, ..., cn] โดยที่ Pc คือ ความน่าจะเป็นที่มีวัตถุอยู่ใน Grid นั้นๆ ถ้าไม่มีค่าจะได้ค่า 0 ถ้ามีจะได้ค่า 1 bx, by คือตำแหน่ง ตรงกลางของ Object ว่าอยู่พิกัดไหนส่วน bh, bw คือขนาดความสูงและกว้างของ Object ว่าสูง, กว้าง ขนาดไหน และ c1,c2,..,cn คือ ผลลัพธ์ว่าเป็น class อะไร ถ้าโจทย์มีแค่ detect หมวดนิรภัย ก็จะมี class เดียว และ grid นั้นมีค่าเป็น 1 แต่ถ้ามีหลาย Object ก็จะมีเลขต่อๆไป



ภาพประกอบที่ 2.6 ภาพของเทคนิค Anchor Box [9]

จากภาพประกอบที่ 2.6 ภาพของเทคนิค Anchor Box [9] จะรับได้เฉพาะ 1 grid คือ 1 Object แต่กรณีถ้ามีหลาย Object เราจะใช้หลักการที่เรียกว่า Anchor Box โดยเราสามารถกำหนด



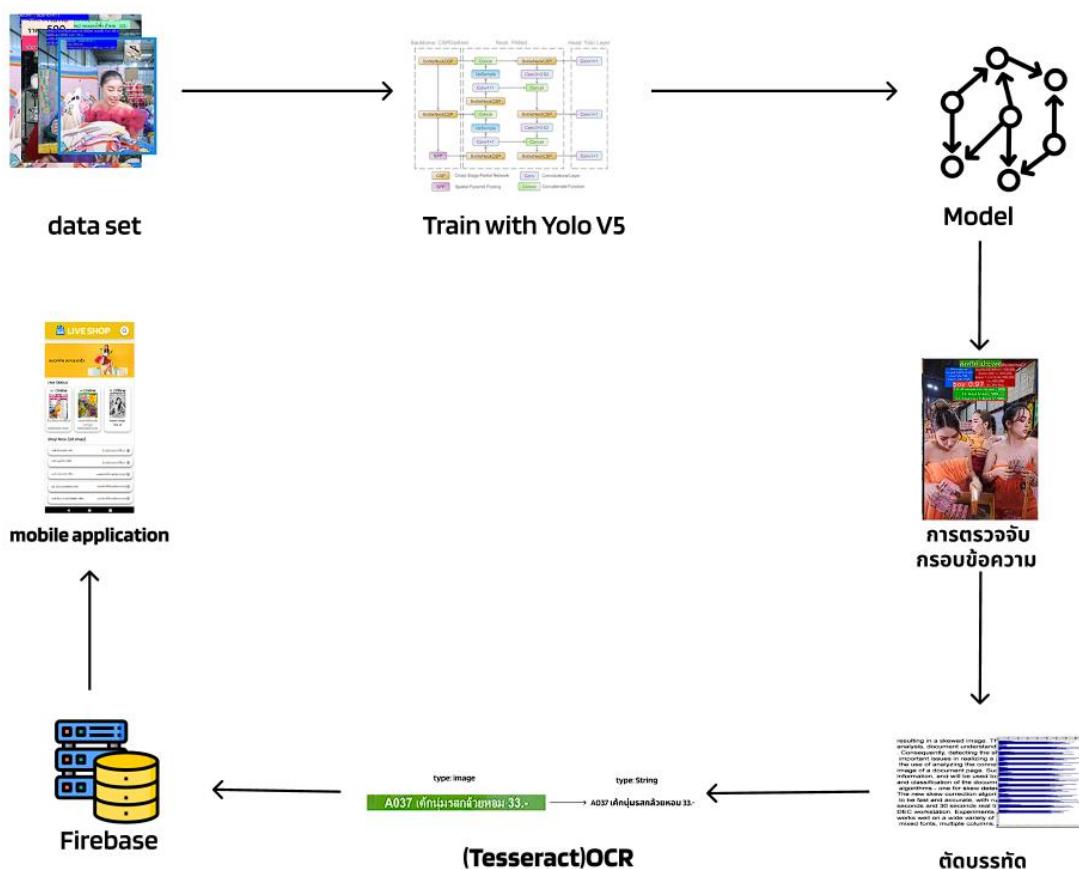
ภาพประกอบที่ 2.7 การทำงานของขบวนการ IOU [9]

จำนวน Box นี้ได้ ก็จะมี Label แบบด้านบน 2 อันใน 1 Grid ได้ และตัว YOLO ก็จะคำนวณให้ว่ารูปนั้นใกล้ Anchor อันไหนสุดจากค่า IOU ก็จะถูกกำหนดไปที่ Anchor นั้น

IOU จะมา มีส่วนหลักๆ ห่างส่วนใน YOLO เช่นเวลาไม่เดลทำนายอภิภูมิ มันอาจจะทำนายรูปรถเดียวกันแต่ห่างกันไป ได้ จึงต้องใช้ metric IOU ในเดือกดังต่อไปนี้เป็นตัวแทนของ Object โดยหลักการคำนวณคือ หาส่วนที่ Intersect หารส่วนที่ Union กัน ถ้ากล่องใดมีค่านี้สูง ก็จะกำหนดแสดงว่ามันคือ Object เดียวกัน

## บทที่ 3

### ขั้นตอนการดำเนินงาน

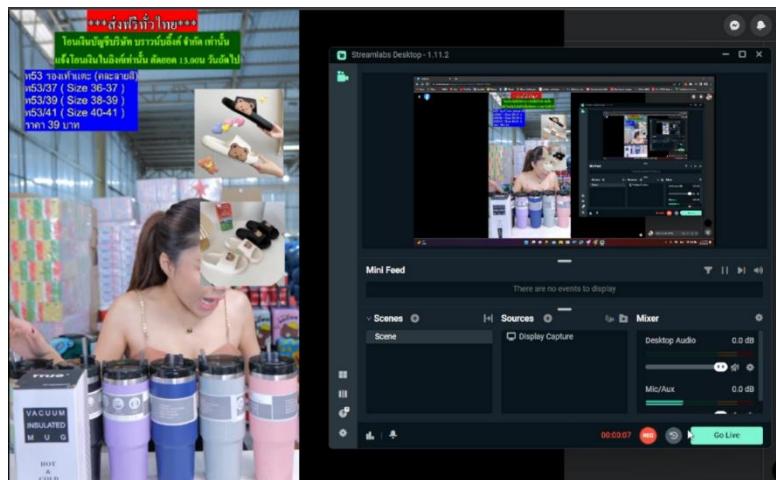


ภาพประกอบที่ 3.1 ขั้นตอนการดำเนินงานของระบบ

สำหรับในบทนี้จะกล่าวถึงขั้นตอนในการดำเนินงานของโครงการปริญญา妮พนธ์ซึ่งจะทำให้ทราบถึงการวิเคราะห์และการออกแบบแอปพลิเคชันโดยละเอียดว่ามีแนวทางในการดำเนินงานหรือมีขั้นตอนในการทำงานของแอปพลิเคชันอย่างไรบ้างโดยขั้นตอนในการดำเนินงานมีรายละเอียดดังนี้ การรวบรวมข้อมูล การ tren โมเดลด้วย YoloV5 การตัดบรรทัดข้อความ การแปลงรูปภาพเป็นตัวอักษร การเก็บข้อมูลลงในฐานข้อมูล และการนำข้อมูลไปแสดงผลใน mobile application

### 3.1 เก็บรวบรวมข้อมูล

เก็บข้อมูลโดยการบันทึกคลิปวิดีโอโดยใช้ Streamlabs Desktop ในการบันทึกคลิปวิดีโอ แล้วนำวิดีโอมาเปิดและบันทึกเป็นภาพนิ่ง ทำการบันทึกวิดีโอด้วยจำนวน 3 ร้านค้าได้แก่ 1.ร้าน FIRST SHOP V2 2.ร้านKANYA SHOP ขายถุงทุกอย่าง 3.ร้านมหัศจรรย์"วันของAunM เงื่อนไขในการเก็บ โดยมีการเก็บวิดีโอเป็น 1 ร้านจะมี 3 วิดีโอลีฟ์สด 1 วิดีโอะจะแบ่งเป็น 6 คลิปวิดีโอ 1 คลิปวิดีโอ จะแบ่งเป็นคลิปละ 3 นาที



ภาพประกอบที่ 3.2 Streamlabs Desktop

ตารางที่ 2 ตารางจำนวนและเวลาเฉลี่ยของวิดีโoinแต่ละร้านค้า

ชื่อร้านค้า	เวลาเฉลี่ยของวิดีโอ	จำนวน
ร้าน FIRST SHOP V2	3 นาที	18 วิดีโอ
ร้าน KANYA SHOP ขายถุงทุกอย่าง	3 นาที	18 วิดีโอ
ร้านมหัศจรรย์"วันของAunM	3 นาที	18 วิดีโอ

### 3.2 การเตรียมข้อมูล

#### 3.2.1 การเตรียมข้อมูลในการ training

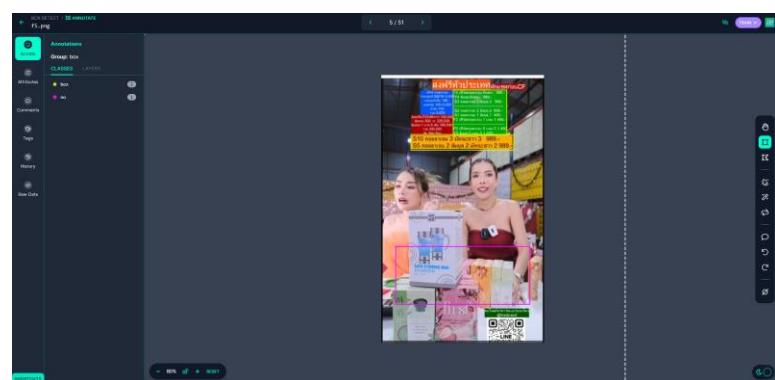
เตรียมข้อมูลในการ training โดยบันทึกรูปภาพจาก Video ไฟล์สติ๊กเก็บมาได้โดยจะทำการบันทึกเฉพาะช่วงที่มีการเปลี่ยนแปลงของข้อความบนหน้าจอไฟล์สติ๊ก โดยจะได้จาก ร้าน FIRST SHOP V2 จำนวน 33 ภาพ จากร้าน KANYA SHOP ขายถูกทุกอย่าง จำนวน 28 ภาพ จากร้านมหาศจรรย์ "วันของ AuuM" จำนวน 33 ภาพ รวมกันทั้งหมด 92 รูปภาพ



ภาพประกอบที่ 3.3 ตัวอย่างรูปภาพที่ทำการเก็บ

#### 3.2.2 การวัดภาพผลเฉลย

ทำการวัดภาพผลเฉลยรูปภาพผ่าน Roboflow framework สำหรับใช้จัดเก็บ เตรียมชุดข้อมูล และสร้างแบบจำลองต่างๆ ที่สามารถใช้งานผ่าน web browser โดยจะทำการวัดภาพผลเฉลยในรูปภาพที่เตรียมมาทั้งหมด โดยจะจัดหมวดหมู่ในที่ที่สนใจเป็น “box” และในพื้นที่ที่ไม่สนใจเป็น “no”



ภาพประกอบที่ 3.4 การวัดภาพผลเฉลยรูปภาพใน Roboflow



ภาพประกอบที่ 3.5 การวัดภาพผลเฉลยรูปภาพใน Roboflow (2)

**C415** กล่องเก็บของลายรัก(คละสี) 79.-  
**เสื้อ1** เสื้อคละแบบไม่ໄกร 69.-  
**A037** เสื้อกันหนาวสก็อตยาวห่ม 33.-  
**E589** สาหร่ายเหมเปะรสน้ำมีค้า 25.-

ข1 บัวเหงื่อยุ1 แพ็ค 9.-  
 ข2 คุกี้หนับ Makiato รสชาติเบรชชี่/ช็อก 60.  
 ข3 2 ช็อกสติ๊ก ยอกบานด์ 60.-  
 ข4 วินเทจลัม ถุงน้ำเงิน 500 กรัม 100.-  
 ข9 กระเช้าพิเศษวันแม่ ลูกปัก 80  
 ข10 กระเช้าให้หัวรับแขกปัก 80

ห53 รองเท้าแตะ (คละสี)  
 ห53/37 ( Size 36-37 )  
 ห53/39 ( Size 38-39 )  
 ห53/41 ( Size 40-41 )  
 ราคา 39 บาท  
 ห083 แก้วเยดิเก็บความเย็น (คละสี) ราคา 100 บาท

ภาพประกอบที่ 3.6 ภาพผลเฉลยที่ถูกต้องจากทั้ง 3 ร้านค้า

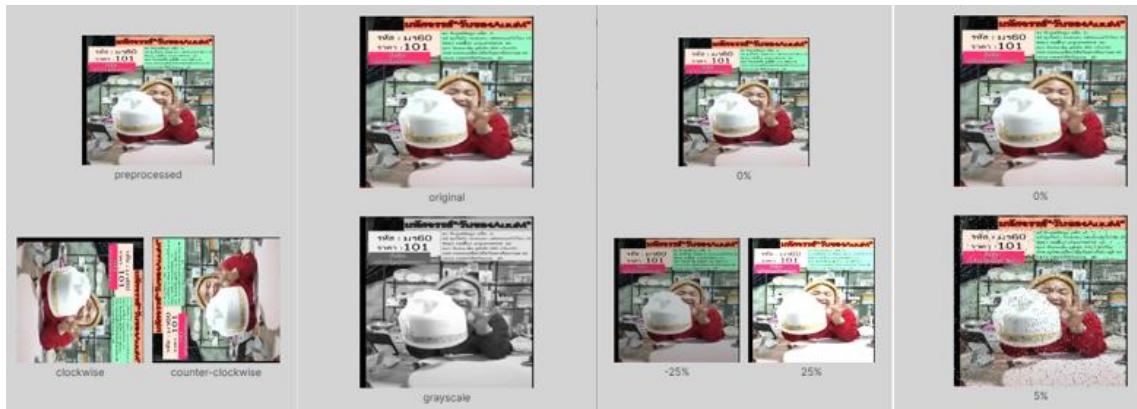
ตารางที่ 3 จำนวนของข้อมูลในแต่ละส่วน

Training Set	Validation Set	Testing Set
62	17	13

เมื่อทำการวัดภาพผลเฉลยรูปภาพทั้งหมดจะทำการแบ่งส่วนของข้อมูลออกเป็นดังนี้โดยจำนวนของ Training Set จะแบ่งเป็น 62 รูปภาพ Validation Set 17 รูปภาพและ Testing Set มีจำนวน 13 รูปภาพ โดยจำนวนของ Training Set จะมีจำนวนที่มากกว่าเนื่องจาก ผลเฉลยที่สนใจมีรูปแบบที่คล้ายกัน

### 3.2.3 การทำ Data Augmentation

ในส่วนนี้จะนำข้อมูลไปทำการ Generate รูปภาพเพิ่มเพื่อเพิ่มจำนวนของข้อมูลที่จะนำไป train โดยจะทำการ หมุนภาพ 90 องศา, ทำให้เป็นภาพสีเทา, เพิ่มแสงสีแดง, เพิ่มสีสีรุ้งรุ่งในรูปภาพ และปรับขนาดของรูปภาพเป็น 416 x 416 pixel ทุกรูปภาพ



ภาพประกอบที่ 3.7 การเพิ่มจำนวนรูปภาพใน Traing Set

จะทำการ Generate ทั้งหมด 3 ครั้ง โดยจะได้รูปภาพจากการ Generate ทั้งหมด 186 ภาพจะมีรูปภาพที่ใช้ในการ train ทั้งหมด 558

ตารางที่ 4 จำนวนของข้อมูลก่อนและหลังทำการ Generate

	Training Set	Validation Set	Testing Set
ก่อน Generate	62	17	13
หลัง Generate	186	17	13
หลัง Generate 3 ครั้ง	558	17	13



ภาพประกอบที่ 3.8 ตัวอย่างรูปภาพที่ Generate ออกมานะ

```
0 0.6838942307692307 0.12259615384615384 0.4951923076923077 0.07211538461538461
1 0.28846153846153844 0.8161057692307693 0.5 0.1875
```

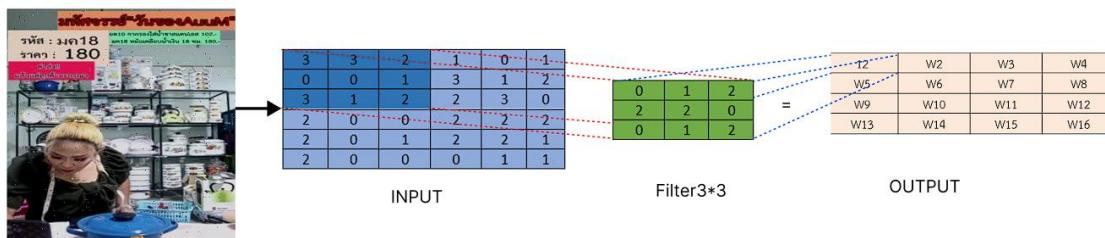
ภาพประกอบที่ 3.9 ผลเฉลยของรูปภาพที่ Generate ออกมานะ

โดยในสิ่งที่ผลเฉลยอธิบายจะมีดังนี้ 1. คลาสของสิ่งที่สนใจในรูป 2. ตำแหน่งเริ่มต้นของกรอบสิ่งที่ detect เจอ 3. ตำแหน่งสิ้นสุดของกรอบสิ่งที่ detect เจอ โดยจะนำตำแหน่งที่ได้มาหาร กับค่า ความกว้างและความสูงของรูปภาพ คือ 416\*416 จึงได้ค่าออกมาเป็นทศนิยมเพื่อจะได้นำไปปรับใช้กับหน้าจอได้หลายขนาด

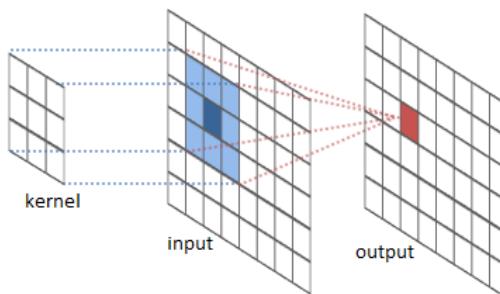
### 3.3 การสร้างโมเดลโดยใช้ yoloV5

ในส่วนนี้จะเป็นการอธิบายถึงโครงสร้างของ yoloV5 และการเทรนข้อมูลโดยจะมีขั้นตอนการทำงานแบบเป็นส่วนหลักๆดังนี้

ในส่วนที่ 1 จะเป็นสำหรับการคัดกรองภาพเพื่อดึงลักษณะเด่นของรูปภาพอกรูปโดยจะมีขั้นตอนการทำงานดังนี้



ภาพประกอบที่ 3.10 การทำ Convolution



ภาพประกอบที่ 3.11 Convolution Layer

#### 3.3.1 Convolution Layer

การทำ Convolution Layer เพื่อสกัดเอาร่วงต่างๆ ของภาพอกรูป เช่น เส้นขอบของวัตถุต่างๆ เพื่อให้มีความสามารถเรียนรู้ลักษณะของภาพได้อย่างมีประสิทธิภาพและแม่นยำโดยขั้นตอนจะมีรูปภาพที่รับเข้ามาเป็น Matrix input ขนาด  $6 \times 6$  เป็นรูปภาพของเราและมี Filter ขนาด  $3 \times 3$  จะนำเฉพาะ Filter  $3 \times 3$  กับ ช่องแรกของ Matrix input และ มาคูณ กับ Filter matrix และนำผลที่ได้แต่ละค่า มาบวกกัน และนำมายังช่องแรกของ Matrix output ซึ่งเป็นผลลัพธ์ โดยในภาพผลลัพธ์ในช่องแรกโดย output จะเท่ากับ

$$W1 = (I1 * f1) + (I2 * f2) + (I3 * f3) + (I4 * f4) + (I5 * f5) + (I6 * f6) + (I7 * f7) + (I8 * f8) + (I9 * f9)$$

เมื่อแทนค่าค่าในสมการจะได้ดังค่าดังต่อไปนี้

$$W1 = (3 * 0) + (3 * 1) + (2 * 2) + (0 * 2) + (0 * 2) + (1 * 0) + (3 * 0) + (1 * 1) + (2 * 2) = 12$$

โดยน้ำถดมาจะเลื่อนกรอบขนาด  $3 \times 3$  ใน Matrix แรกไปทางขวา 1 ช่อง และทำแบบเดิม ผลลัพธ์ที่ได้นำไปใส่ใน格子 1 ช่อง 2 ของ Matrix output โดยจะทำไปเรื่อยๆ จนสุดแล้วเลื่อน กรอบ  $3 \times 3$  ลงมา ด้านล่าง 1 ช่อง และทำแบบเดิม จนกระทั่งเติมค่าใน Matrix ผลลัพธ์จนเต็ม

### 3.3.2 Rectified Linear Unit (ReLU)

Rectified Linear Unit (ReLU) เป็นฟังก์ชัน Activation Function ที่นิยมใช้ใน Deep Learning เพื่อปรับค่าผลลัพธ์ให้เป็นค่าบวก โดยถ้า  $x$  มีค่าเป็นลบจะแทนค่า  $x$  ตัวมันกลับเป็น 0 โดยได้จากสมการต่อไปนี้

$$f(x) = \max(0, x)$$

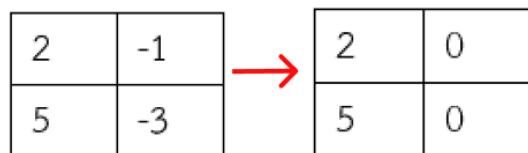
ตัวอย่างการคำนวณReLU

$$x = 2: f(2) = \max(0, 2) = 2$$

$$x = -1: f(-1) = \max(0, -1) = 0$$

$$x = 5: f(5) = \max(0, 5) = 5$$

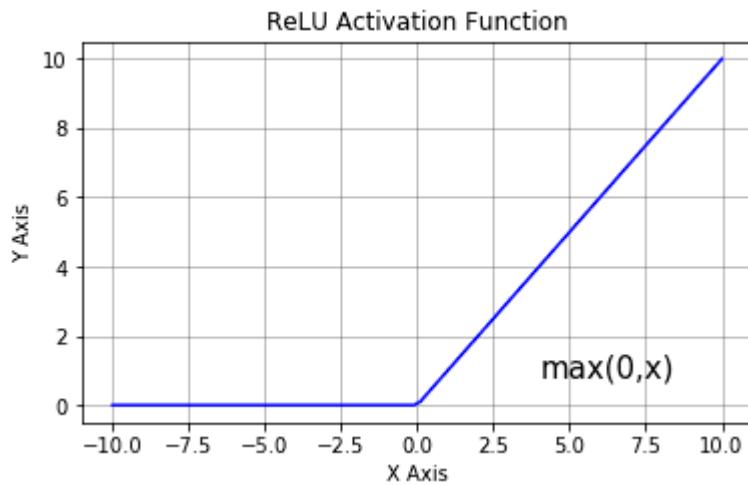
$$x = -3: f(-3) = \max(0, -3) = 0$$



2	-1
5	-3

2	0
5	0

ภาพประกอบที่ 3.12 การคำนวณ ReLU



ภาพประกอบที่ 3.13 กราฟของ ReLU

### 3.3.3 ReLU leaky

ฟังก์ชัน ReLU นั้นมีจุดขาดที่เรียกว่า ReLU leaky โดยจุดขาดนี้เกิดจากเมื่อค่านิพจน์ที่ผ่านมา มีค่าติดลบมาก ๆ จะทำให้ gradient หายไปและไม่สามารถปรับค่าได้ จึงมีการพัฒนา ReLU leaky เพื่อแก้ไขปัญหานี้ ReLU leaky จะไม่ให้ค่าลบเป็น 0 แต่จะแทนที่ด้วยค่าที่เล็กๆ โดยที่ค่านี้จะกำหนดได้ โดยทั่วไปจะใช้ค่าเดียวกับ  $\alpha$  ซึ่งเป็นค่าเล็กๆ อยู่ในช่วง 0 ถึง 1 ซึ่งช่วยให้ gradient ยังคงมีค่าได้ และช่วยให้โมเดลมีความสามารถในการเรียนรู้ได้ดีขึ้นโดยมีสมการดังต่อไปนี้

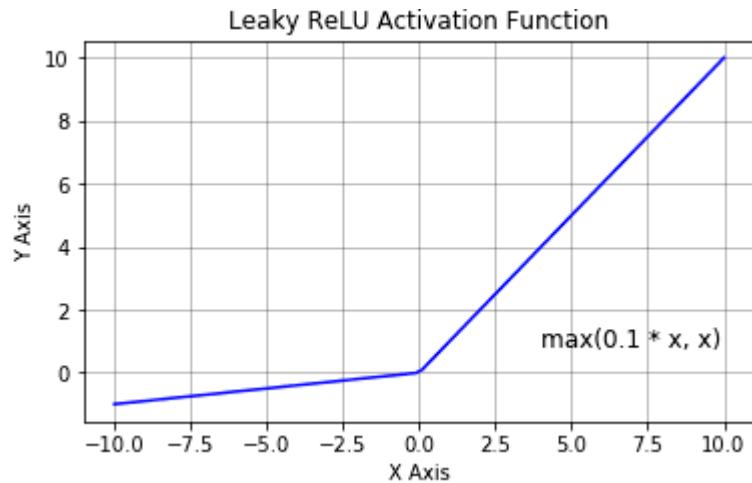
$$f(x) = \max(\alpha * x, x)$$

ตัวอย่างการคำนวณ ReLU leaky  
โดยจะกำหนดให้  $\alpha = 0.1$

$$\begin{aligned} x = 2: f(2) &= \max(0.1 * 2, 2) = 2 \\ x = -1: f(-1) &= \max(0.1 * -1, -1) = -0.1 \\ x = 5: f(5) &= \max(0.1 * 5, 5) = 5 \\ x = -3: f(-3) &= \max(0.1 * -3, -3) = -0.3 \end{aligned}$$

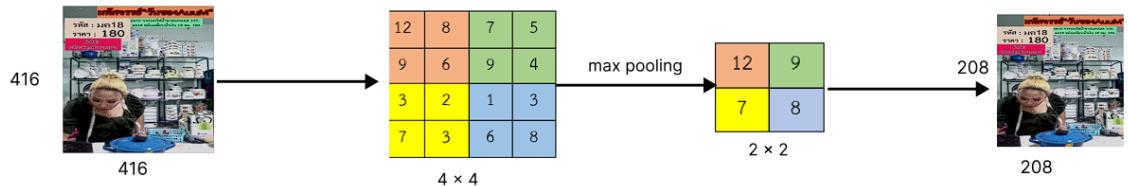
2	-1
5	-3
2	-0.1
5	-0.3

ภาพประกอบที่ 3.14 การคำนวณ ReLU Leaky



ภาพประกอบที่ 3.15 กราฟของ ReLU Leaky

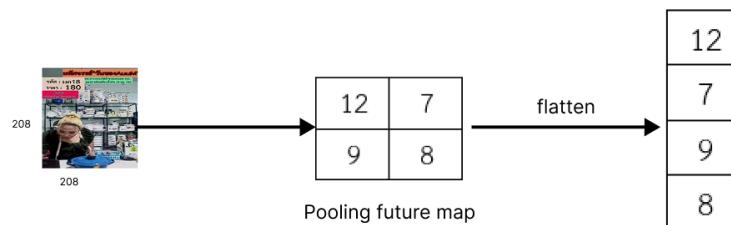
หลังจากการทำ ReLU เพื่อทำให้ค่าเป็นบวกแล้วจะทำการ Pooling layer เพื่อทำการการสกัดหรือลดขนาดของข้อมูล เพื่อเอาส่วนที่สำคัญที่สุดของข้อมูลโดยทั่วไปมักจะเลือกใช้ max pooling หรือ average pooling



ภาพประกอบที่ 3.16 Pooling layer

### 3.3.4 Pooling layer

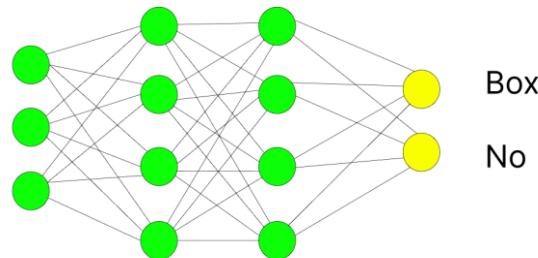
การทำ Pooling layer คือการสกัดหรือลดขนาดของข้อมูล future map ที่ได้จากการทำ Convolution เพื่อเอาส่วนที่สำคัญที่สุดของข้อมูล และเพิ่มประสิทธิภาพการประมวลผลให้รวดเร็วยิ่งขึ้น โดย Max Pooling layer คือการสกัดเอาเฉพาะค่าสูงสุดของ Matrix เก็บไว้ใน Output เช่นจากภาพ Max Pooling layer ที่มีขนาด 2x2



ภาพประกอบที่ 3.17 การทำ Flatten

### 3.3.5 Flatten

จะทำการแปลงข้อมูลจากชั้น Pooling layer ที่ได้มา ให้กลายเป็น vector โดยที่ข้อมูลทุกตัวจะถูกวางต่อกันเป็น列เดียวกัน โดยที่ไม่มีการเปลี่ยนแปลงค่าข้อมูล เช่น ถ้ามี matrix ขนาด  $2 \times 2$  แล้วทำการ flatten จะได้ vector ขนาด  $4$  ( $2 \times 2 = 4$ ) โดยที่ข้อมูลจะเรียงตามลำดับของ列และคอลัมน์ของเมทริกซ์เดิม ก่อนจะนำไปทำในขั้นตอนของ Fully Connected Layer

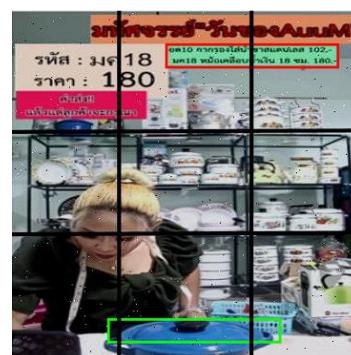


ภาพประกอบที่ 3.18 Fully Connected Layer

### 3.3.6 Fully Connected Layer

Fully Connected Layer คือการรวมผลลัพธ์ของตัวแปร input จาก Layer ก่อนหน้าทั้งหมด และคุณด้วยน้ำหนัก (weight) ของแต่ละโนนด (neuron) ใน Fully-Connected Layer นี้ โดยที่แต่ละโนนดจะมี weight และ bias ของตัวเองที่แตกต่างกันไป การคูณนี้จะทำให้ผลลัพธ์เป็นเวกเตอร์หรือเมทริกซ์ขนาดเล็กกว่า input และเมทริกซ์นี้จะถูกส่งต่อไปยัง Layer ถัดไป

ส่วนที่ 2 เป็นส่วนที่ทำหน้าที่ Detect Object โดยใช้การสร้าง bounding box โดยการคำนวณพิกัดของ object ที่ Convolution Layer เพื่อทวนยความน่าจะเป็นของแต่ละ Object Class และ bounding box coordinates



ภาพประกอบที่ 3.19 รูปภาพ input ที่ใส่ grid

การกำหนดข้อมูลที่เราต้องส่งให้ไปใน API จะมีขั้นตอนดังต่อไปนี้ เราจะต้องส่งข้อมูลที่ว่าด้วยผลเฉลยแล้วไปยังโมเดลเพื่อฝึกฝน และจะแบ่งภาพออกเป็นตารางขนาด  $3 \times 3$  และมีทั้งหมด 2 คลาสที่ต้องการให้วัดถูก ถูกจัดประเภท ซึ่งใน 2 คลาสจะมี 'คลาส' box' ที่เป็นจุดที่เป็นกรอบของข้อความและคลาส 'go' เป็นจุดที่ไม่สนใจ ดังนั้นสำหรับแต่ละเซลล์ใน grid จะมี ค่าผลเฉลย y เจ็ดค่า ดังต่อไปนี้

ตารางที่ 5 ค่า parameter ใน ค่าผลเฉลย y ทั้ง 7

y	pc
	bx
	by
	bh
	bw
	c1
	c2

pc คือ ค่าที่กำหนดว่าวัดถูกมีอยู่ในตารางหรือไม่

bx คือ ค่าที่ระบุตำแหน่งในแกน x ของ Bounding Box ของวัดถูกเมื่อมีวัดถูกภายในภาพ

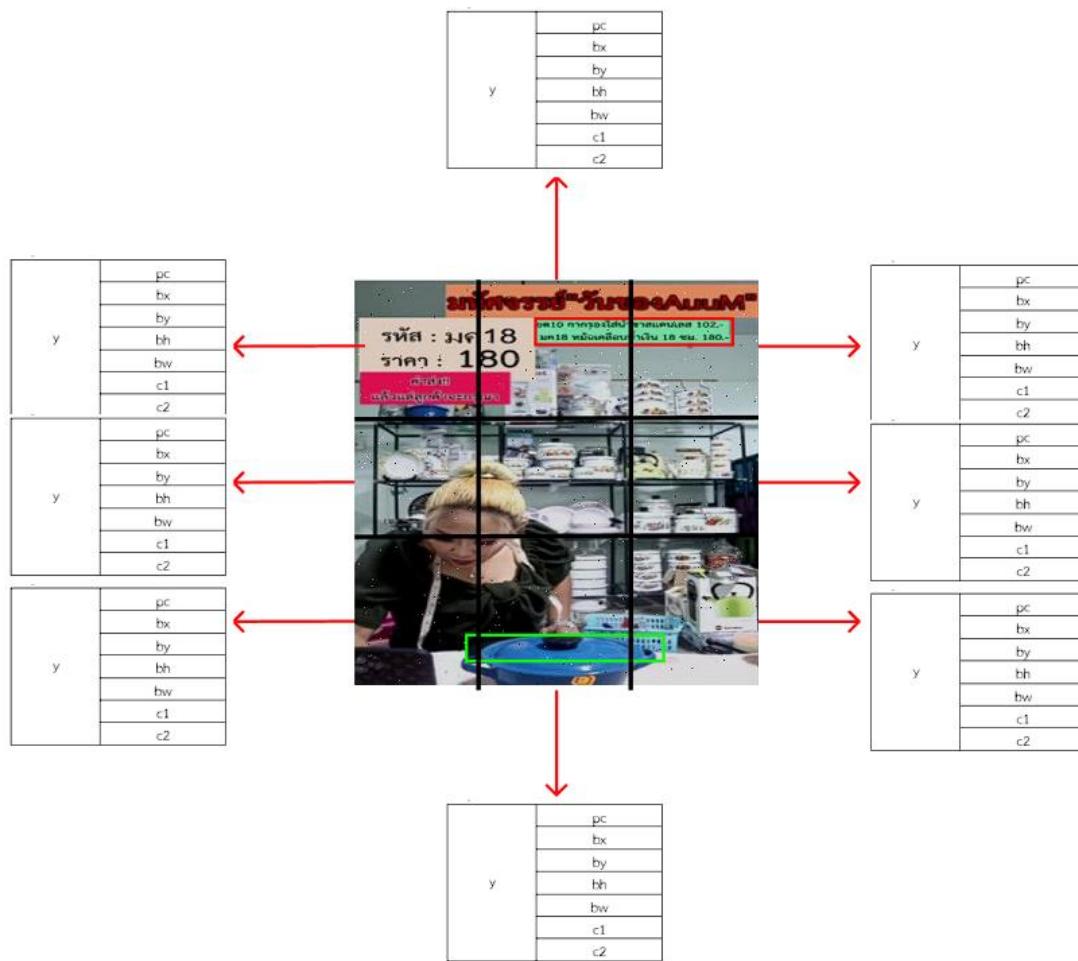
by คือ ค่าที่ระบุตำแหน่งในแกน y ของ Bounding Box ของวัดถูกเมื่อมีวัดถูกภายในภาพ

bh คือ ค่าที่ระบุตำแหน่งความสูงของ Bounding Box ของวัดถูกเมื่อมีวัดถูกภายในภาพ

bw คือ ค่าที่ระบุตำแหน่งความกว้างของ Bounding Box ของวัดถูกเมื่อมีวัดถูกภายในภาพ

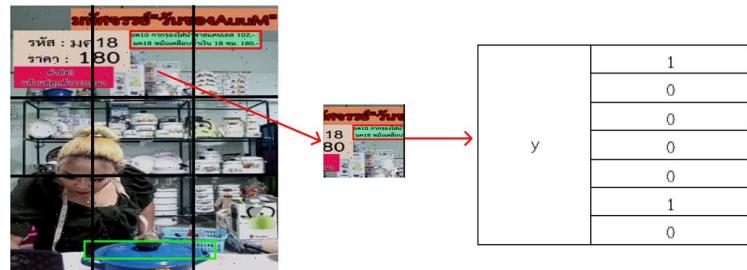
c1 คือ คลาสของวัดถูกที่เจอนอกภาพที่เจอด้วยคลาสนี้จะมีชื่อว่า 'box' ถ้าเจอวัดถูกจะตั้งกล่าวค่าจะเป็น 1 ถ้าไม่ใช่จะเป็น 0

c2 คือ คลาสของวัดถูกที่เจอนอกภาพที่เจอด้วยคลาสนี้จะมีชื่อว่า 'go' ถ้าเจอวัดถูกจะตั้งกล่าวค่าจะเป็น 1 ถ้าไม่ใช่จะเป็น 0



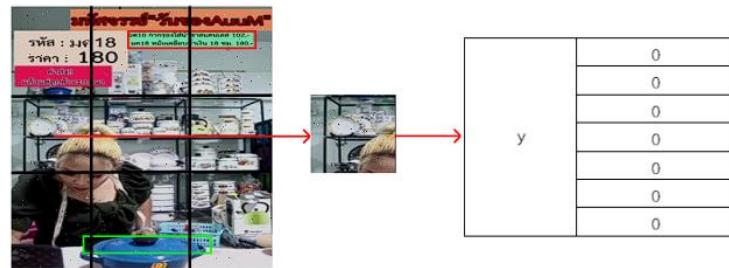
ภาพประกอบที่ 3.20 ค่าผลเฉลย y ในแต่ละ grid cell

การกำหนด input ของรูปภาพสำหรับ grid นี้ YOLO จะตัดสินใจว่ามีวัตถุอยู่ใน grid จริง หรือไม่โดย YOLO จะใช้จุดกึ่งกลางของวัตถุและวัตถุเหล่านั้นจะถูกกำหนดให้กับ grid ที่มีจุดกึ่งกลางของวัตถุนั้น มี ค่าผลเฉลย y สำหรับ grid ซึ่งนั่นๆ ถ้าเจอวัตถุในภาพ



ภาพประกอบที่ 3.21 ตัวอย่าง grid ที่มีวัตถุในภาพและการแทนค่า

ในกรณีที่เจอวัตถุใน grid ในตารางนี้  $pc = 1$  และในส่วน  $bx$ ,  $by$ ,  $bh$ ,  $bw$  จะถูกคำนวณเทียบกับเซลล์ grid ที่ทำการคำนวณอยู่ในภายหลัง และเนื่องจากเจอกรอบข้อความในรูปภาพ ค่า  $c1 = 1$  และ  $c2 = 0$



ภาพประกอบที่ 3.22 ตัวอย่าง grid ที่ไม่มีวัตถุในภาพและการแทนค่า

ในกรณีที่ไม่เจอวัตถุใน grid ในตารางนี้  $pc = 0$  เมื่อมีค่า  $pc = 0$  เท่ากับว่าไม่มีวัตถุที่สนใจในภาพเราจะไม่แทนค่าต่อใน grid cell นี้



ภาพประกอบที่ 3.23 จุดกึ่งกลางของ Bounding Box

### 3.3.7 การแทนค่าความสูงและความกว้างของ Bounding Box

Bounding Box คือการ วัดกล่องรอบวัตถุ ดังที่ได้กล่าวไว้ก่อนหน้านี้โดย  $bx$ ,  $by$ ,  $bh$ ,  $bw$  จะถูกคำนวณเทียบกับเซลล์ grid ที่ทำการคำนวณอยู่ พิจารณา grid ตรงที่มีกรอบของตัวอักษร  $bx$  คือพิกัด  $x$  ของจุดกึ่งกลางของวัตถุที่อยู่ใน grid นี้ ในกรณีนี้  $bx = 0.9$  โดยจะวัดจากจุดกึ่งกลาง  $by$  คือพิกัด  $y$  ของจุดกึ่งกลางของวัตถุที่อยู่ใน grid นี้ ในกรณีนี้  $bx = 0.4$  โดยจะวัดจากจุดกึ่งกลาง  $bh$  คืออัตราส่วนของความสูงของ Bounding Box กับความสูงของเซลล์ grid ที่เกี่ยวข้องซึ่งในกรณีของรูปภาพในตัวอย่างจะมีค่าคือประมาณ  $bh = 0.3$  โดยประมาณ  $bw$  คืออัตราส่วนของความกว้างของ Bounding Box กับความกว้างของเซลล์ grid ที่เกี่ยวข้องซึ่งในกรณีของรูปภาพในตัวอย่างจะมีค่าคือประมาณ  $bw = 1.5$  โดยประมาณ โดยค่า  $y$  สำหรับ grid นี้จะมีค่าดังนี้

#### ตารางที่ 6 การแทนค่า $bx$ , $by$ , $bh$ , $bw$

y	1
	0.9
	0.4
	0.3
	1.5
	1
	0

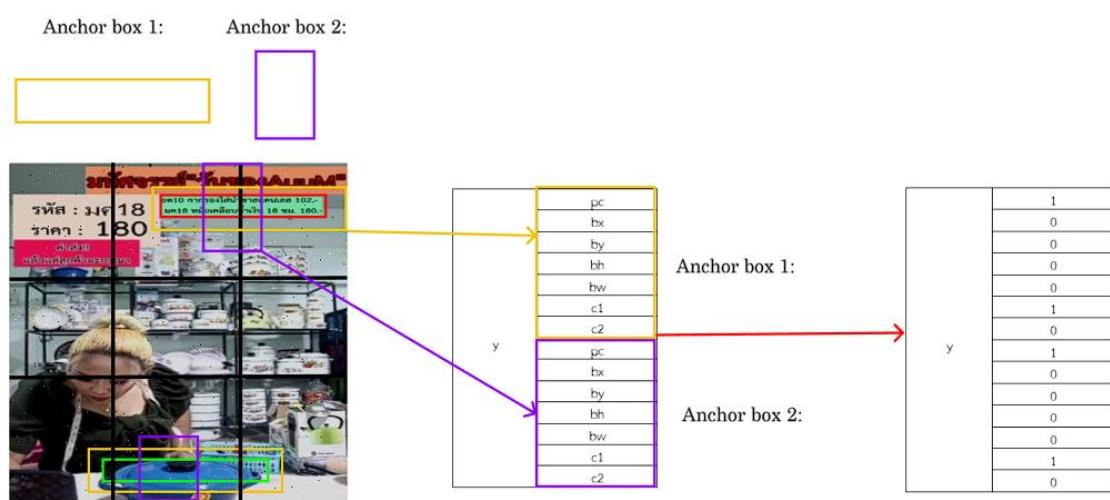
$bx$  และ  $by$  จะอยู่ในช่วงระหว่าง 0 ถึง 1 เสมอเนื่องจากจุดกึ่งกลางจะอยู่ภายใน grid เสมอ ขณะที่  $bh$  และ  $bw$  สามารถมากกว่า 1 ในกรณีที่ Bounding Box มากกว่าขนาดของ grid

### 3.3.8 การทำ anchor box

เป็นการสร้าง anchor box ที่มีรูปร่างต่างกันเพื่อเพิ่มความแม่นยำในการตรวจจับของโมเดลโดยถ้ามี anchor box 2 กล่อง ค่า label y โดยที่ค่า parameter 7 แต่แรกเป็นของ anchor box ที่ 1 และอีก 7 แต่ที่เหลือเป็นของ anchor box ที่ 2 โดยจะเปลี่ยนแปลงไปดังนี้

ตารางที่ 7 parameter ของการทำ anchor box

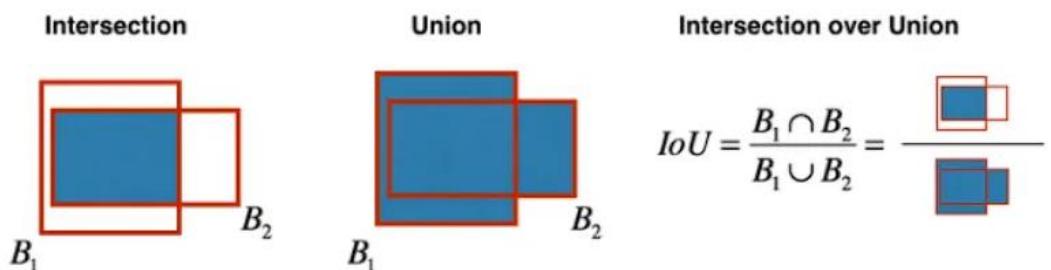
y	pc
	bx
	by
	bh
	bw
	c1
	c2
	pc
	bx
	by
	bh
	bw



ภาพประกอบที่ 3.24 การแทนค่าใน anchor box และการรวมของ anchor box

### 3.3.9 Intersection over Union

Intersection over Union คือ วิธีการวัดความเหมือนหมายถึงการเปรียบเทียบระหว่างกรอบสี่เหลี่ยมที่รอบตัวของวัตถุ (bounding box) ที่ระบุโดยโมเดลกับกรอบสี่เหลี่ยมที่ถูกต้อง (ground truth bounding box) โดยการคำนวณ IoU จะได้ผลลัพธ์ออกมาเป็นเลขระหว่าง 0 ถึง 1 โดยจะมีค่ามากเมื่อกรอบสี่เหลี่ยมที่ระบุโดยโมเดลมีการครอบคลุมวัตถุที่ถูกต้องมากขึ้น



ภาพประกอบที่ 3.25 IoU

ตัวอย่างการหาค่า IoU

box 1										
1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1

box 2										
1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1

1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

ภาพประกอบที่ 3.26 ตัวอย่างการคำนวณค่า IoU

โดยภาพ box 1 คือ ภาพ Matrix แรก และภาพ box 2 คือ ภาพ Matrix ที่สอง และในภาพ box 1 union box 2 คือสีเขียว ส่วน box 1 intersection box 2 คือสีเขียว สีแดงคือทายขาดโดยสมการดังนี้

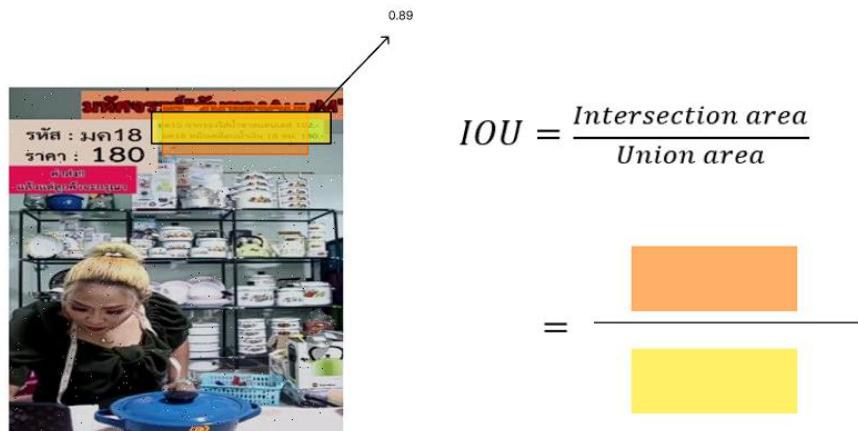
$$\text{Union area} = \text{box1} + \text{box2} - \text{intersection area}$$

$$\text{Union area} = 54 + 54 - 51 = 57$$

โดยค่า IOU จะหาได้จากการสมการดังนี้

$$\text{IOU} = \frac{\text{Intersection area}}{\text{Union area}}$$

$$\text{IOU} = \frac{51}{57} = 0.89$$



ภาพประกอบที่ 3.27 ตัวอย่างของ ค่า IOU ในรูปภาพ

### 3.3.10 Non-Max Suppression

Non-Max Suppression เป็นเทคนิคหนึ่งที่ใช้ในการลดจำนวนของ bounding boxes หรือ detections ที่ซ้อนทับกันในการตรวจจับวัตถุ ซึ่งเป็นส่วนสำคัญในการประมวลผลภาพการฝึกฝนข้อมูล วิธีการทำงานของ Non-Max Suppression คือ การเลือก bounding box ที่มีค่า confidence score สูงสุด และกำจัด bounding box อื่น ๆ ที่ซ้อนทับอยู่ในพื้นที่เดียวกัน โดยมีขั้นตอนดังนี้

1. จัดเรียง bounding box ตามค่า confidence score จากมากไปน้อย
2. เลือก bounding box ที่มีค่า confidence score สูงสุด และเก็บไว้
3. ลบ bounding box ที่มีค่า IOU หรือ threshold น้อยกว่าหรือเท่ากับค่าที่กำหนด
4. ทำขั้นตอนที่ 2 และ 3 จนกว่าจะไม่มี bounding box ที่เหลือให้ตรวจสอบ

ตัวอย่างเมื่อกำหนดหนนค่า IoU ให้มากกว่าหรือเท่ากับ 0.5

ก่อองที่	confidence score
1	0.9
2	0.6
3	0.75



ก่อองที่	confidence score
1	0.9

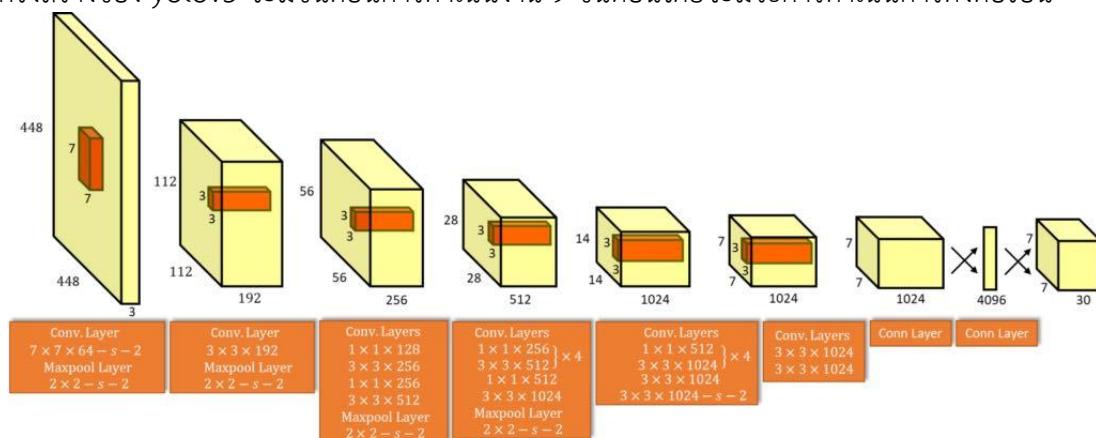
ภาพประกอบที่ 3.28 ตัวอย่างการทำ Non-Max Suppression



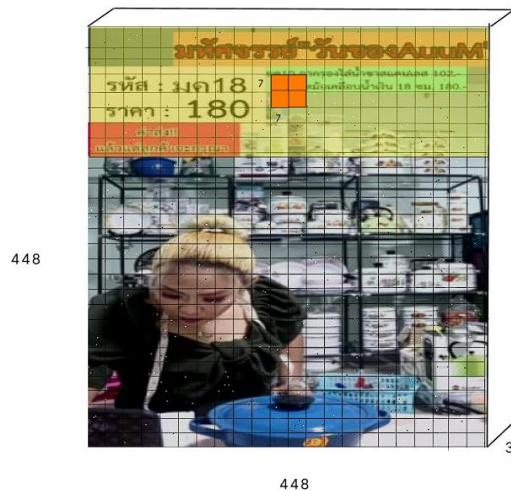
ภาพประกอบที่ 3.29 ตัวอย่างการทำ Non-Max Suppression

### 3.3.11 โครงสร้างของ Yolov5

ในโครงสร้างของ yolov5 นี้เป็นการแสดงรูปแบบของรูปภาพเมื่อผ่าน layer ต่างๆ ในโครงสร้างของ yolov5 จะมีขั้นตอนการดำเนินงาน 9 ขั้นตอนโดยจะมีวิธีการดำเนินการดังต่อไปนี้



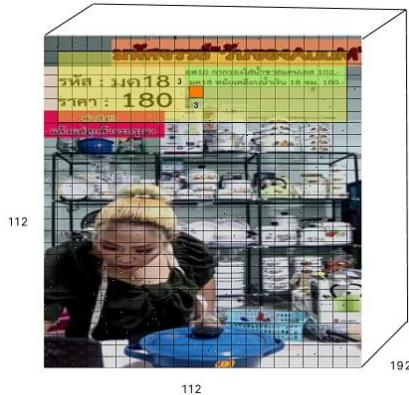
ภาพประกอบที่ 3.30 Layer ของ Yolov5



448

ภาพประกอบที่ 3.31 รูปภาพผลลัพท์ในขั้นตอนที่ 1

ขั้นตอนที่ 1 จะทำการรับรูปภาพขนาด  $448 \times 448$  pixels เข้ามาและทำการ ผ่านการทำ Convolution ที่มี kernel ขนาด  $7 \times 7$  และมี feature map ใหม่โดยใช้ filters จำนวน 64 ตัวและ kernel จะเคลื่อนที่ทีละ 2 ช่องและทำการผ่าน Maxpool Layer เพื่อทำการลดขนาดของ feature map โดย Maxpool Layer จะมีขนาด  $2 \times 2$  และจะทำการขยับไปทีละ 2 ช่อง

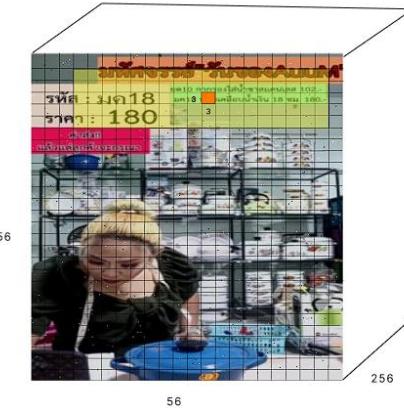


112

192

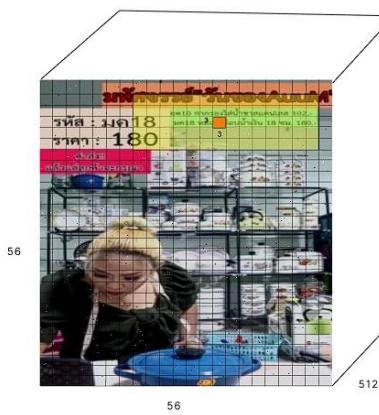
ภาพประกอบที่ 3.32 รูปภาพผลลัพท์ในขั้นตอนที่ 2

ขั้นตอนที่ 2 จะทำการนำ feature map เดิมเข้ามาและทำการ ผ่านการทำ Convolution ที่มี kernel ขนาด  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 129 ตัวและ kernel จะเคลื่อนที่ทีละ 1 ช่องและทำการผ่าน Maxpool Layer เพื่อทำการลดขนาดของ feature map โดย Maxpool Layer จะมีขนาด  $2 \times 2$  และจะทำการขยับไปทีละ 2 ช่อง



ภาพประกอบที่ 3.33 รูปภาพผลลัพธ์ในขั้นตอนที่ 3

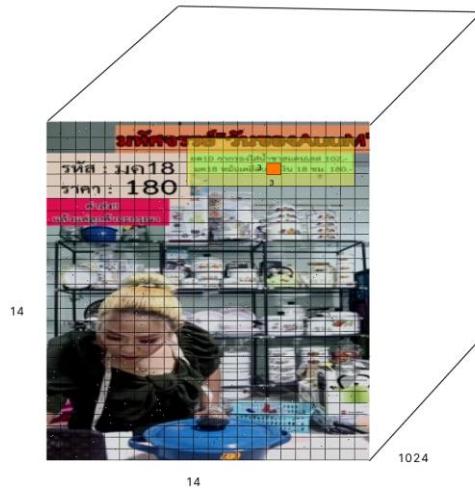
ขั้นตอนที่ 3 จะนำรูปภาพเข้ามาผ่านการทำ Convolution 4 ชั้น โดย  
 ชั้นที่ 1 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 128  
 ชั้นที่ 2 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 256  
 ชั้นที่ 3 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 256  
 ชั้นที่ 4 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 512  
 และ kernel ของชั้นที่ 1 - 4 จะเคลื่อนที่ละ 1 ช่องและทำการผ่าน Maxpool Layer เพื่อทำการลด  
 ขนาดของ feature map โดย Maxpool Layer จะมีขนาด  $2 \times 2$  และจะทำการขยับไปทีละ 2 ช่อง



ภาพประกอบที่ 3.34 รูปภาพผลลัพธ์ในขั้นตอนที่ 4

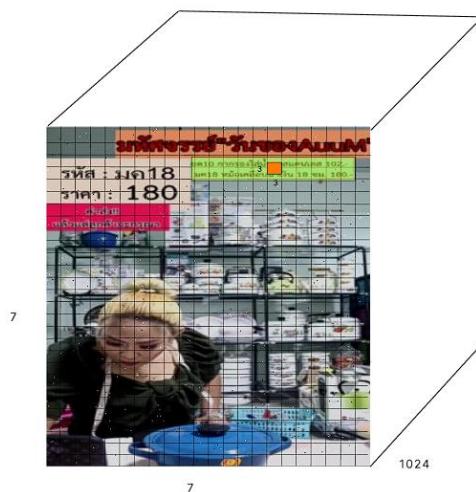
ขั้นตอนที่ 4 จะนำรูปภาพเข้ามาผ่านการทำ Convolution 10 ชั้น โดย  
 ชั้นที่ 1 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 256  
 ชั้นที่ 2 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 512  
 ชั้นที่ 3 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 256  
 ชั้นที่ 4 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 512  
 ชั้นที่ 5 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 256  
 ชั้นที่ 6 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 512

ขั้นที่ 7 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 256  
 ขั้นที่ 8 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 512  
 ขั้นที่ 9 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 512  
 ขั้นที่ 10 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 1024  
 และ kernel ของขั้นที่ 1 – 10 จะเคลื่อนที่ทีละ 1 ช่องและทำการผ่าน Maxpool Layer เพื่อทำการลด  
 ขนาดของ feature map โดย Maxpool Layer จะมีขนาด  $2 \times 2$  และจะทำการขยับไปทีละ 2 ช่อง



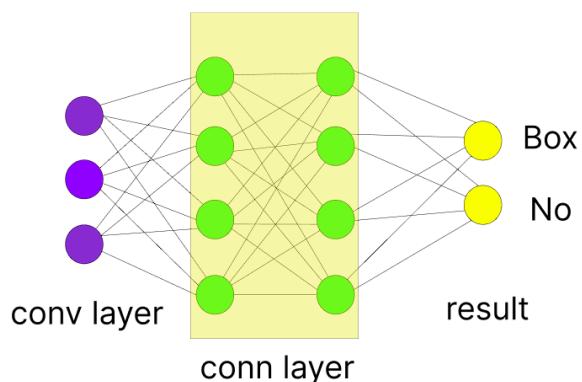
### ภาพประกอบที่ 3.35 รูปภาพผลลัพธ์ในขั้นตอนที่ 5

ขั้นตอนที่ 5 จะนำรูปภาพเข้ามาผ่านการทำ Convolution 10 ขั้น โดย  
 ขั้นที่ 1 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 512  
 ขั้นที่ 2 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 1024  
 ขั้นที่ 3 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 512  
 ขั้นที่ 4 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 1024  
 ขั้นที่ 5 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 512  
 ขั้นที่ 6 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 1024  
 ขั้นที่ 7 จะเป็นแบบ  $1 \times 1$  และมี feature map ใหม่โดยใช้ filters จำนวน 512  
 ขั้นที่ 8 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 1024  
 ขั้นที่ 9 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 1024  
 และ kernel ของขั้นที่ 1 - 9 จะเคลื่อนที่ทีละ 1 ช่องและทำการผ่าน  
 ขั้นที่ 10 โดยจะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 1024  
 และ kernel ของช่องที่ 10 จะเคลื่อนที่ทีละ 2 ช่อง



ภาพประกอบที่ 3.36 รูปภาพผลลัพธ์ในขั้นตอนที่ 6

ขั้นตอนที่ 6 จะนำรูปภาพเข้ามาผ่านการทำ Convolution 2 ชั้น โดยชั้นที่ 1 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 1024 ชั้นที่ 2 จะเป็นแบบ  $3 \times 3$  และมี feature map ใหม่โดยใช้ filters จำนวน 1024 และ kernel ของช่องที่ 1 – 2 จะเคลื่อนที่ทีละ 1 ช่องและทำการผ่าน Maxpool Layer เพื่อทำการลดขนาดของ feature map โดย Maxpool Layer จะมีขนาด  $2 \times 2$  และจะทำการขยับไปทีละ 2 ช่อง



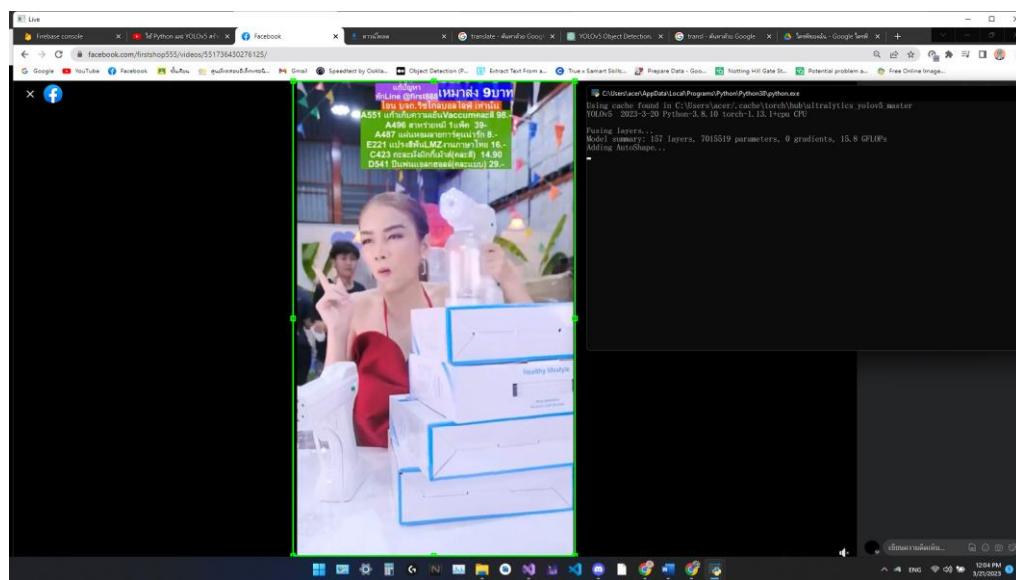
ภาพประกอบที่ 3.37 connection layer

ขั้นตอนที่ 7 และ 8 จะผ่านการ connection layer โดยจะทำการเชื่อมต่อชั้น convolutional layers เข้าด้วยกันเพื่อปรับปรุงความแม่นยำของโมเดล โดยที่แต่ละชั้น convolutional layers จะมี feature maps ที่มีขนาดและลักษณะการเข้ารหัสที่แตกต่างกัน การเชื่อมต่อหลาย feature maps นี้จะช่วยให้โมเดลมีความสามารถในการจำแนกวัตถุที่ซับซ้อนและติดตามได้ดีขึ้น

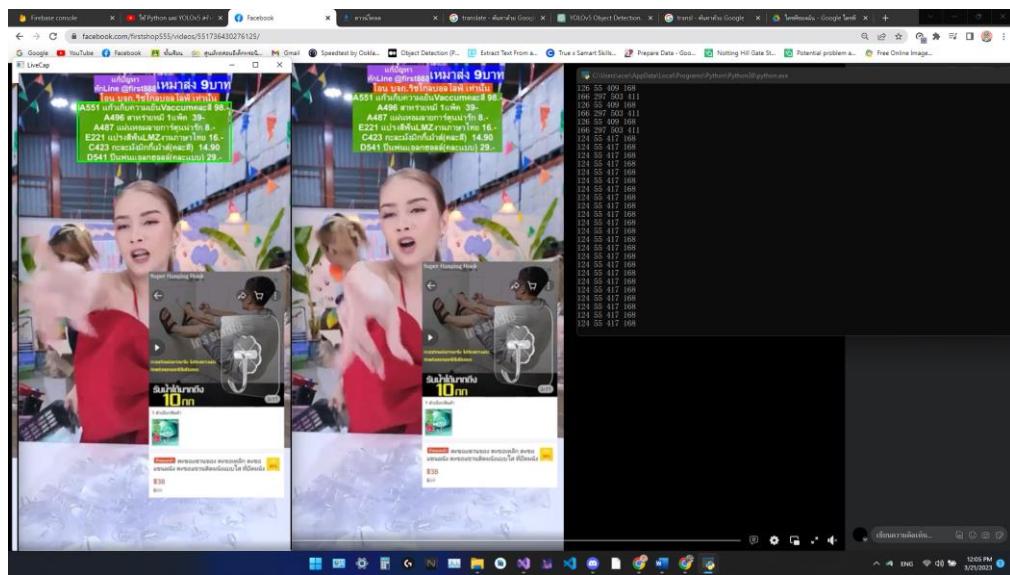


### ภาพประกอบที่ 3.38 ขั้นตอนการทำ NMS และแสดงผลลัพท์

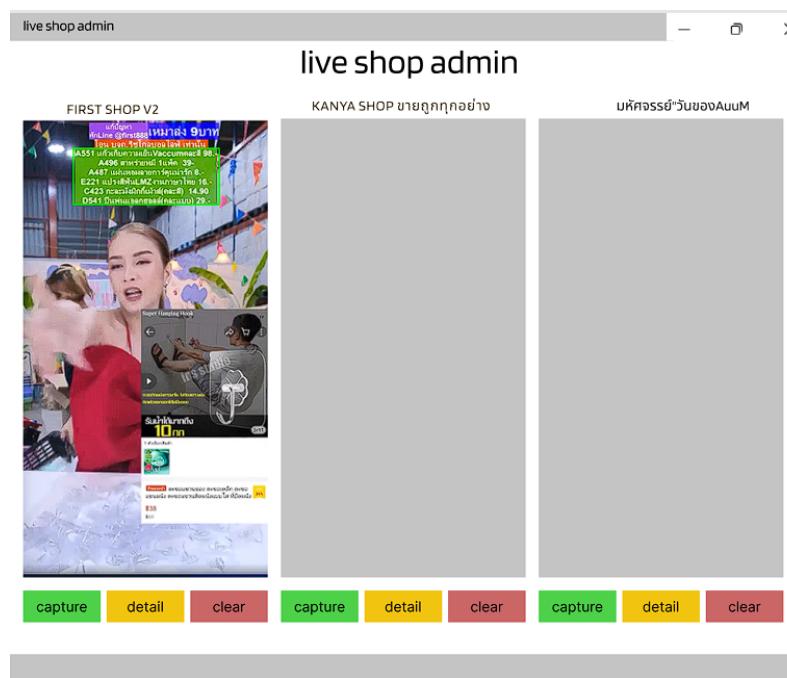
ขั้นตอนที่ 9 จะเป็น การทำ Non-Max Suppression เพื่อการลดจำนวนของ bounding boxes ที่ซ้อนทับกันในการตรวจจับวัตถุ ที่มีความน่าจะเป็นสูงที่สุด โดยทำการคัดลอก bounding box ที่มี IoU (Intersection over Union) มากกว่าหรือเท่ากับ threshold ที่กำหนด และเลือก bounding box ที่มีคะแนนความเชื่อมั่น (confidence score) สูงที่สุดเพื่อใช้เป็นผลลัพธ์สุดท้ายของโมเดล สุดท้าย จึงแสดงผลลัพธ์ออกมาในรูปแบบของ bounding box ที่ว่ารอบ object และคำแนะนำว่า object นั้นมีชื่อว่าอะไร (class label) พร้อมกับคะแนนความเชื่อมั่นของการตรวจจับ (confidence score)



### ภาพประกอบที่ 3.39 การทำงานในส่วนของการเลือกส่วนที่ต้องการ



ภาพประกอบที่ 3.40 ผลลัพท์ในการ detect และการเลือกพื้นที่ที่ต้องการ



ภาพประกอบที่ 3.41 ผลลัพท์ในการ detect ใน desktop application

### 3.4 การตัดบรรทัดด้วยเทคนิค projection profile

ในส่วนของงานในส่วนนี้จะเป็นการนำรูปภาพจากที่ทำการ detect มาได้มาทำการแปลงเป็นภาพ binary และทำการใช้เทคนิค horizontal projection profile มาทำการแบ่งบรรทัดเพื่อทำให้สินค้าแยกขึ้นกันอย่างชัดเจน

#### 3.4.1 การเตรียมภาพ

พ23 เตาแม่เหล็กไฟฟ้า 555.-  
พ24 หม้อหุงต้ม สีขาว 1 ลิตร 345.-  
พ25 ถ้วยมัลลาย 9 ซ. 59.-  
พ26 ไฟสองกบ คละสี 49.-

ภาพประกอบที่ 3.42 ตัวอย่างภาพนำเข้า

การเตรียมพร้อมภาพ(Pre-Processing) เป็นขั้นตอนการนำเข้าภาพเพื่อให้พร้อมต่อการนำไปประมวลผลต่อไป โดยประกอบด้วย ขั้นตอนการแปลงเป็นภาพระดับเทา และ การแปลงให้เป็นใบหน้า ขาว-ดำ และตัดรูปภาพให้มีเฉพาะตรงส่วนที่มีข้อความอยู่ โดยขั้นตอนต่อไปนี้

#### 3.4.2 แปลงภาพสี RGB ไปเป็นภาพระดับเทา

ขั้นตอนการแปลงภาพทำได้โดยอาศัยค่าของ RGB ที่อยู่ในแต่ละพิกเซลของภาพต้นฉบับมาแปลงเป็นภาพระดับเทา โดยช่องสี R จะมีค่าอยู่ที่ 0 – 255 ช่องสี G มีค่าอยู่ที่ 0-255 และช่องสี B จะมีค่าอยู่ที่ 0-255

		0		
	255	0	0	
	0			

		0		
	0	0	250	
		250		

		0		
	255	0	0	
	0			

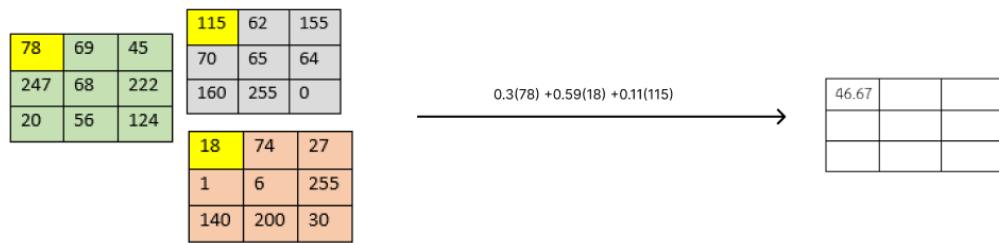
ภาพประกอบที่ 3.43 ตารางค่าสี R G B ตามลำดับ

วิธีการถ้านำพิกเซลตำแหน่งที่(3,4)จากตารางมาแปลงเป็นภาพระดับเทาสามารถทำได้ดังนี้  
จากสูตร  $Gray = 0.3R + -0.59 G + 0.11B$

แทนค่า  $Gray = 0.3(255)+0.59(0)+0.11(255)$

จะได้ค่า  $Gray = 76.5 + 0 + 28.5 = 104.55$

โดยค่าคำตอบของ  $Gray$  จะถูกแทนลงในตำแหน่ง (3,4)ของภาพระดับเทา ตัวอย่างการแปลงภาพสี RGB เป็นภาพระดับเทา ของภาพที่มีขนาด  $3 \times 3$  พิกเซล



ภาพประกอบที่ 3.44 การแปลงภาพสี RGB ไปเป็นระดับเทา

46.67	71.18	46.48
82.39	31.09	224.09
106.2	162.85	54.9

ภาพประกอบที่ 3.45 ตัวอย่างผลลัพธ์หลังจากการแปลงเป็นภาพระดับเทา



ภาพประกอบที่ 3.46 ตัวอย่างหลังจากการแปลงเป็นภาพระดับเทา

### 3.4.3 การแปลงภาพระดับเทา ไปเป็นภาพขาวดำ (Binary image)

เป็นการแปลงค่าสีจากระดับเทาที่มีค่าอยู่ระหว่าง 0 – 255 ให้เป็นภาพแบบใบหนาริที่มีค่า 0 และ 255 โดยจะหาค่าขีดแบ่ง (Threshold) ที่เหมาะสม โดยใช้ Otsu จะคำนวณหาความแปรปรวนระหว่างกลุ่มเพื่อหาค่าขีดแบ่งที่ดีที่สุดในการแยกตัว象ออกจากพื้นหลังออกจากพื้นหลัง โดยพิจารณาจากค่า  $t$  ที่เป็นไปได้ ทั้งหมดโดย โดยมีวิธีการคำนวณดังนี้

ขั้นตอนที่ 1 กำหนดค่า  $L$  ให้เท่ากับ 4 และ  $n_i = \{29, 21, 37, 18\}$

ขั้นตอนที่ 2 คำนวณหาค่าความน่าจะเป็น ดังสมการ

$$p_i = \frac{n_i}{N}, p_i \geq 0, \sum_{i=0}^{L-1} p_i = 1 \quad (3.2)$$

โดยที่  $p_i$  คือ จำนวนจุดภาพที่ระดับความเข้มที่  $i$

$N$  คือจำนวนพิกเซล ณ ระดับที่  $i$

N คือผลรวมของพิกเซลทั้งหมด ในแต่ละระดับ

L คือ ระดับความเข้มของภาพระดับสีเทา

- คำนวณความน่าจะเป็น  $p_i$

$$p_0 = \frac{29}{105} = 0.2761$$

$$p_1 = \frac{21}{105} = 0.2$$

$$p_2 = \frac{37}{105} = 0.3524$$

$$p_3 = \frac{18}{105} = 0.1714$$

ขั้นตอนที่ 3 คำนวณหาค่า  $\omega_0$  และ  $\omega_1$  โดยค่า  $\omega_0$  คำนวณได้ตามสมการดังนี้

$$\omega_0 = \sum_{i=1}^K P_i = \omega(k) \quad (3.3)$$

โดยที่  $\omega_0$  คือ ความน่าจะเป็นของกลุ่มที่ 1

$$k = 1$$

$$p = 0.2761$$

$$\omega_0 = \sum_{i=1}^K P_i = 0.2761$$

$$\omega_1 = \sum_{i=1}^L P_i \quad (3.4)$$

โดยที่  $\omega_1$  คือ ความน่าจะเป็นของกลุ่มที่ 2

$$k = 1$$

$$p = \{0.2 + 0.3524 + 0.1714\}$$

$$\omega_1 = \sum_{i=1}^k P_i = 0.7238$$

- คำนวณหาค่า  $\omega_0$  และ  $\omega_1$

$$\omega_0 = \{0.2761, 0.3524, 0.1714\}$$

$$\omega_1 = \{0.7238, 0.3524, 0.1714\}$$

ขั้นตอนที่ 4 คำนวณหาค่าเฉลี่ย ดังสมการ

$$\mu_0 = \frac{\mu(k)}{\omega(k)} \quad (3.5)$$

โดยที่  $\mu_0$  คือ ค่าเฉลี่ยของกลุ่มที่ 1

$$\mu_k \sum_{i=1}^k i * p_i \text{ คือผลรวมของ } i * p_i \text{ ตั้งแต่ } i = k + 1 \text{ ถึง } L$$

$$\mu_1 = \sum_{i=k+1}^L \frac{i * p_i}{\omega_1} \quad (3.6)$$

โดยที่  $\mu_1$  คือ ค่าเฉลี่ยของกลุ่มที่ 2

$$\mu_1 = \sum_{i=k+1}^L \frac{i * p_i}{\omega_1} \text{ คือผลรวมของ } \frac{i * p_i}{\omega_1} \text{ ตั้งแต่ } i = k + 1 \text{ ถึง } L$$

- คำนวณหาค่า  $\mu_0$   $\mu_1$  และ  $\mu_k$  แทนค่าในสูตร กำหนด  $k = 3$

$$\mu_k = (1 * 0.2761) = 0.2761$$

$$\mu_0 = (1 * 0.2761) + (2 * 0.2) + (3 * 3524) = 1.7333$$

$$\mu_1 = \left( \frac{2*0.2}{0.7238} \right) + \left( \frac{3*0.324}{0.7238} \right) + \left( \frac{4*0.1714}{0.7238} \right) = 2.9604$$

- คำนวณหาค่า  $\mu_0$   $\mu_1$  และ  $\mu_k$  ทุกตำแหน่ง กำหนด  $k = 1$

$$u_k = \{0.2761, 0.4, 1.0572\}$$

$$u_0 = \{1, 0.8402, 1.2760\}$$

$$u_1 = \{0.5526, 2.0132, 2.9604\}$$

ขั้นตอนที่ 5 คำนวณหาค่าความแปรปรวน ดังสมการ

$$\sigma_B^2 = \omega_0 \omega_1 (\mu_0 - \mu_1)^2 \quad (3.7)$$

โดยที่  $\sigma_B^2$  คือค่าความแปรปรวนของกลุ่มตั้งแต่รอบที่ 1-255

$$u_1 = \sum_{i=k+1}^L \frac{i*p_i}{\omega_1} \quad \text{คือผลรวมของ } \frac{i*p_i}{\omega_1} \quad \text{ตั้งแต่ } i = k + 1 \text{ ถึง } L$$

- คำนวณหาค่า  $\sigma_B^2$  จะได้

$$\sigma_B^2 = \{0.04, 0.6545, 2.3492\}$$

ขั้นตอนที่ 6 คำนวณหาค่าความแปรปรวนสูงสุด ดังสมการ

$$\sigma_B^2(k^*) = \max_{1 \leq k < L} \sigma_B^2(k) \quad (3.8)$$

โดยที่  $\sigma_B^2(k^*)$  คือ ค่าความแปรปรวนสูงสุดของกลุ่มตั้งแต่รอบที่  $k = 1-255$

- คำนวณ หาค่า  $\sigma_B^2(k^*)$  จะได้

$$\sigma_B^2(k^*) = 2.3492$$

โดยจะเลือกค่า  $k$  ที่ให้ค่าความแปรปรวนสูงสุด คือ 2 ที่เท่ากับ 2.3492 กำหนดให้เป็นค่า

Threshold จะได้  $t=2$  จะได้ผลลัพธ์เป็นภาพ binary

พ23 เตาแม่เหล็กไฟฟ้า 555.-  
พ24 หม้อหุงต้ม สีขาว 1 ลิตร 345.-  
พ25 ถ้วยม้าลาย 9 ช. 59.-  
พ26 ไฟสองกบ คละสี 49.-

พ23 เตาแม่เหล็กไฟฟ้า 555.-  
พ24 หม้อหุงต้ม สีขาว 1 ลิตร 345.-  
พ25 ถ้วยม้าลาย 9 ช. 59.-  
พ26 ไฟสองกบ คละสี 49.-

ภาพประกอบที่ 3.47 ตัวอย่างการแปลงภาพระดับ เท่า เป็นภาพ Binary

#### 3.4.4 การวัดเส้นในจุดที่เป็นที่ว่าง

ในขั้นตอนนี้จะทำการวัดเส้นในส่วนที่เป็นช่องว่างระหว่างข้อความเพื่อแบ่งสินค้าออกเป็นแต่ละชิ้นโดยทำการหาพิกัด  $(x,y)$  เริ่มต้นของภาพและໄลเช็คไปทีลิสพิกเซลจากซ้ายไปขวาและเมื่อเจอจุดที่เป็นช่องว่างจะนำมาทำการวัดเส้นแบ่งไว้โดยมีรายละเอียดและขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 เริ่มสแกนรูปภาพจากซ้ายไปขวาจนกว่าจะพบพิกเซลวัตถุ โดยหากค่าพิกเซลที่พบมีค่ามากกว่าหรือเท่า 1 แปลว่าพิกเซลที่พบมีข้อความอยู่

0	0	0	0	0
1	0	0	0	0
1	1	1	1	1
1	0	0	1	0
0	0	0	0	0
1	1	1	1	1
0	0	0	0	0

ภาพประกอบที่ 3.48 ตัวอย่างสแกนรูปภาพจากซ้ายไปขวา

ขั้นตอนที่ 2 หลังจากได้จุดที่น่าจะเป็นขอบข้อความแล้วจะทำการมาร์คเส้น ใน index ที่ร่างแล้วใส่เลข 5 ลงไปแทน และเริ่มสแกนไปยังแผลด้านไป โดยในตำแหน่งที่เป็นเลข 5 จะกำหนดให้เป็นสีแดง และจะทำการวาดเส้นในตำแหน่งที่เป็นเลข 5 เพื่อทำการระบุว่าจุดตรงนี้คือช่องว่างระหว่างบรรทัด

0	0	0	0	0
1	0	0	0	0
1	1	1	1	1
1	0	0	1	0
5	5	5	5	5
1	1	1	1	1
5	5	5	5	5

ภาพประกอบที่ 3.49 ตัวอย่างการมาร์คเส้น

ฟ23 เตาแม่เหล็กไฟฟ้า 555.-
ฟ24 หม้อหงต้ม สีขาว 1 ลิตร 345.-
ฟ25 ถ้วยม้าลาย 9 ซ. 59.-
ฟ26 ไฟส่องกบ คละสี 49.-

ภาพประกอบที่ 3.50 ผลลัพท์ในการวางแผนแยกบรรทัด

### 3.5 ขั้นตอนการเรียกใช้ Tesseract OCR

ในกระบวนการ Optical Character Recognition (OCR) นั้นจำเป็นจะต้องมีข้อมูลักษณะ (Feature) ของตัวอักษรนั้นๆ ก่อน เพื่อนำมาประมวลผลที่ยึดเคียงกับข้อมูลที่ได้จากการ ข้อมูล Feature ที่ได้มาจากการ (Train) ซึ่งค่อนข้างมีความซับซ้อนในการพัฒนา ดังนั้นในโครงการนี้จะใช้เครื่องมือที่ช่วยลดความยุ่งยาก ที่มีเช่นว่า Tesseract OCR เพื่อนำมาเปรียบเทียบผลลัพธ์ที่ได้

วิธีการดำเนินการ การสกัดตัวหนังสือภาษาไทยและภาษาอังกฤษ โดยขั้นตอนแรกนำภาพเข้าไปจากนั้นเขียน Code เพื่อทดสอบการทำงานและตรวจสอบผลลัพธ์

#### ตารางที่ 8 การแปลงรูปเป็นตัวอักษรด้วย tesserract

```
import pytesseract as tess
from PIL import Image
tess.pytesseract.tesseract_cmd = r'D:\ocr\tesseract.exe'
image = Image.open('D:\ocr\kan2.png')
text = tess.image_to_string(image, lang='tha+eng')
print(text)
```

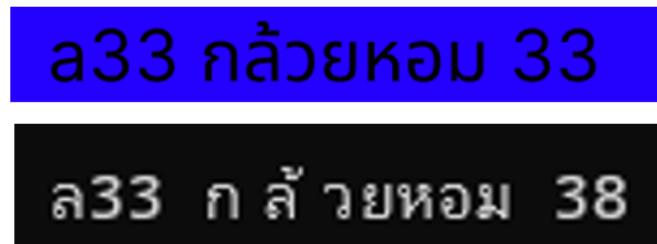
บรรทัดที่ 1 import flies tesseract

บรรทัดที่ 2 import image เพื่อใช้ในการเพิ่มรูปภาพเข้ามา

บรรทัดที่ 3 คือการอ่าน path files ของ tesseract

บรรทัดที่ 4 การอ่าน files รูปภาพแล้วก็ปะไว้ในตัวแปลง image เพื่อนำไปประมวลผล

บรรทัดที่ 5 การเอารูปภาพมาแปลงเป็น text จะแปลงเป็นภาษาไทยและภาษาอังกฤษ

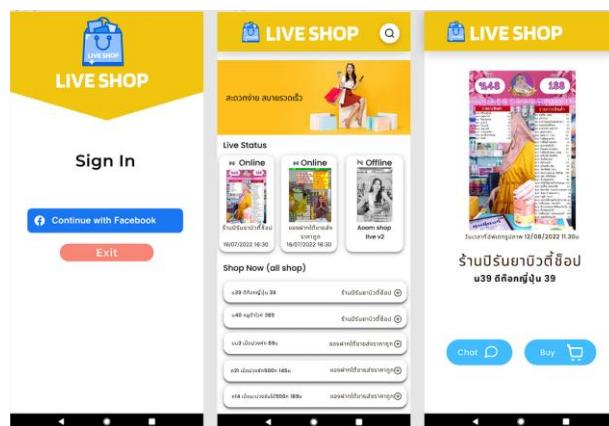


ภาพประกอบที่ 3.51 ตัวอย่างของผลลัพธ์ของการแปลงที่ผิดพลาด

การใช้ Training Dataset ในการสกัดตัวอักษรไทยและอังกฤษจากภาพ จะพบว่าสามารถสกัดข้อความออกมาได้ถูกบางคำ เพื่อป้องกันข้อผิดพลาดโครงการนี้จึงจะสรุปภาพให้ผู้ใช้ด้วย เพื่อลดข้อผิดพลาดลง

### 3.6 Mobile application

ในส่วนของ mobile application จะใช้ Flutter ในการสร้าง application ที่ทำการล็อกอินด้วย facebookได้และเชื่อมต่อกับ firebase และดึงข้อมูลจาก firebase มาแสดงผลได้และเพื่อลดความผิดพลาดในการแสดงผลข้อมูลจะมีรูปภาพของร้านค้าในขนาดที่ทำการไลฟ์สดขึ้นมาในหน้ารายละเอียดสินค้าด้วย

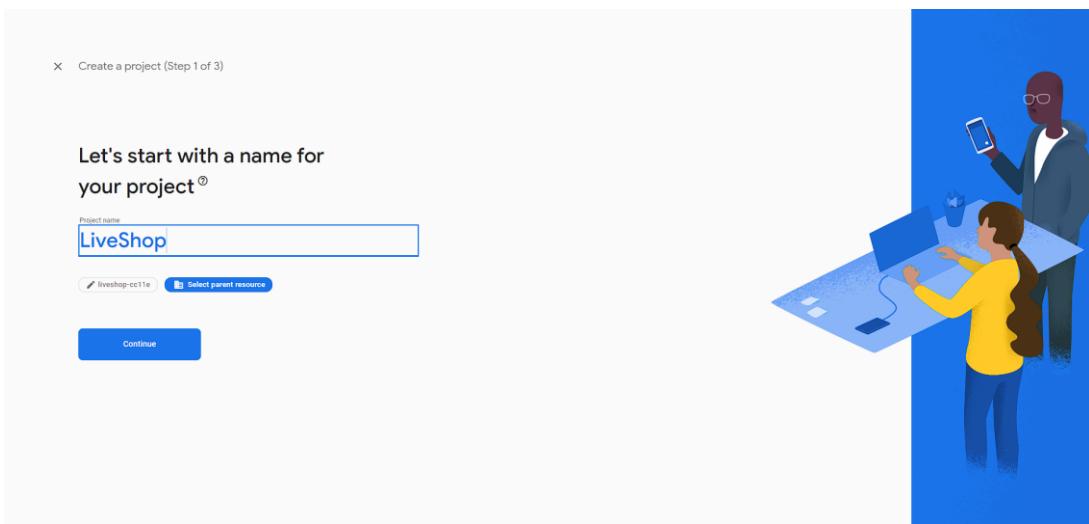


ภาพประกอบที่ 3.52 ตัวอย่างหน้าตา UI mobile application

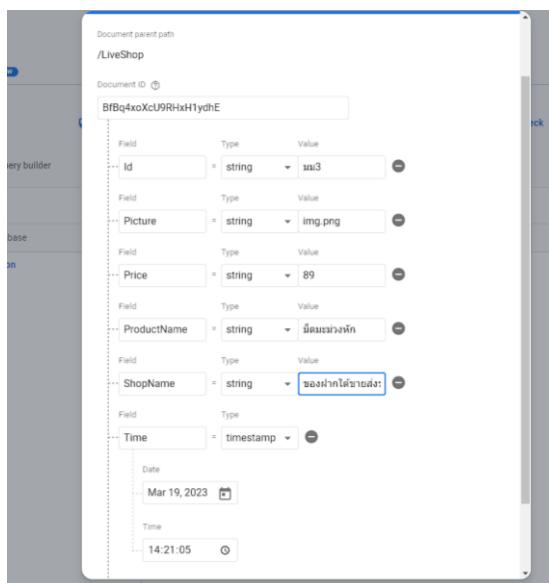
### 3.7 การจัดเก็บข้อมูล

ในส่วนของการจัดการข้อมูลจะใช้ Cloud Firestore จัดเก็บข้อมูล โดยใช้ฐานข้อมูล NoSQL ที่โฮสต์บนคลาวด์ Cloud Firestore โดยโครงสร้างจะมี 3 ส่วนคือ

1. Collection เป็น Folder ที่ไว้เก็บเอกสาร และมีชื่อบอกว่าเก็บเอกสารเกี่ยวกับอะไร
2. Document เป็นกระดาษไว้สำหรับเก็บข้อมูล และมีชื่อบอกว่าเก็บข้อมูลเกี่ยวกับอะไร
3. Data เป็นที่เก็บข้อมูล



ภาพประกอบที่ 3.53 การสร้างโปรเจ็ค firebase



ภาพประกอบที่ 3.54 การสร้าง Document ใน firebase

โดยข้อมูลที่ทำการส่งไปเก็บในฐานข้อมูลจะประกอบไปด้วย

- ชื่อร้านค้า
- รหัสสินค้า
- ชื่อสินค้า
- ราคาสินค้า
- รูปภาพในการไฟล์สตด
- วันเวลาที่ capture รูปภาพ

LiveShop	⋮	BfBq4xoXcU9RHxH1ydhE	⋮
+ Add document		+ Start collection	
BfBq4xoXcU9RHxH1ydhE	>	+ Add field	
		Id: "مم3" Picture: "img.png" Price: "89" ProductName: "มีดคมะม่วงหัก" ShopName: "ของฝากได้ขายส่งราคาถูก" Time: March 19, 2023 at 2:21:05 PM UTC+7	

ภาพประกอบที่ 3.55 ตัวอย่างข้อมูลและประเภทของข้อมูล

### 3.8 วัดประสิทธิภาพ

#### 1. การวัดประสิทธิภาพการ Detect วัดด้วย Mean Average Precision (mAP)

ค่าเฉลี่ยความแม่นยำเฉลี่ย Mean Average Precision (mAP) เป็นเมตริกที่ใช้ในการประเมินแบบจำลองการตรวจจับวัตถุ เช่น Fast R-CNN, YOLO, Mask R-CNN เป็นต้น ค่าเฉลี่ยของค่าความแม่นยำเฉลี่ย (AP) จะคำนวณจากค่าการเรียกคืนตั้งแต่ 0 ถึง 1 mAP คำนวณได้จากค่าดังนี้

Confusion Matrix ถือเป็นเครื่องมือสำคัญในการประเมินผลลัพธ์ของการทำนาย หรือ Prediction ที่ทำนายจาก Mode ที่เราสร้างขึ้น ในMachine learning โดยมีไอเดียจากการวัดว่า สิ่งที่เราคิด (Model ทำนาย) กับ สิ่งที่เกิดขึ้นจริง มีสัดส่วนเป็นอย่างไร

		Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)	
	False Negatives (FNs)	True Negatives (TNs)	
Predicted Negative (0)			

ภาพประกอบที่ 3.56 Confusion Matrix

True Positive (TP)=สิ่งที่ทำนายตรงกับสิ่งที่เกิดขึ้นจริงในกรณีทำนายว่าจริงและสิ่งที่เกิดขึ้นก็คือ จริง

True Negative (TN)=สิ่งที่ทำนายตรงกับสิ่งที่เกิดขึ้น ในกรณีทำนายว่าไม่จริงและสิ่งที่เกิดขึ้นก็คือไม่จริง  
 False Positive (FP)=สิ่งที่ทำนายไม่ตรงกับสิ่งที่เกิดขึ้น คือทำนายว่า จริง แต่สิ่งที่เกิดขึ้น คือ ไม่จริง  
 False Negative (FN)=สิ่งที่ทำนายไม่ตรงกับที่เกิดขึ้นจริง คือทำนายว่าไม่จริง แต่สิ่งที่เกิดขึ้น คือ จริง

Intersection over Union(IoU) เป็นวิธีทางสถิติที่ใช้วัดความสอดคล้องของข้อมูลสองชุด โดยมีข้อมูลสองเซ็ตคือ P ซึ่งแทนเซตของพิกเซลในกรอบที่ไม่เดลเลือกมา และ G คือเซตของพิกเซลในกรอบที่เป็นเฉลย ใช้ P แทนคำว่า Predicted ส่วน G คือ Ground truth

$$IoU(P, G) = \frac{|P \cap G|}{|P \cup G|}$$

ภาพประกอบที่ 3.57 สมการหาค่า IoU

1	1	1	1	0	0	0	0	0	0
0	0	1	1	1	0	0	0	0	0
0	0	1	1	1	0	1	1	0	0
0	0	1	1	1	1	1	1	1	1
0	0	1	1	1	1	0	0	0	1
0	0	1	1	0	0	0	0	0	1

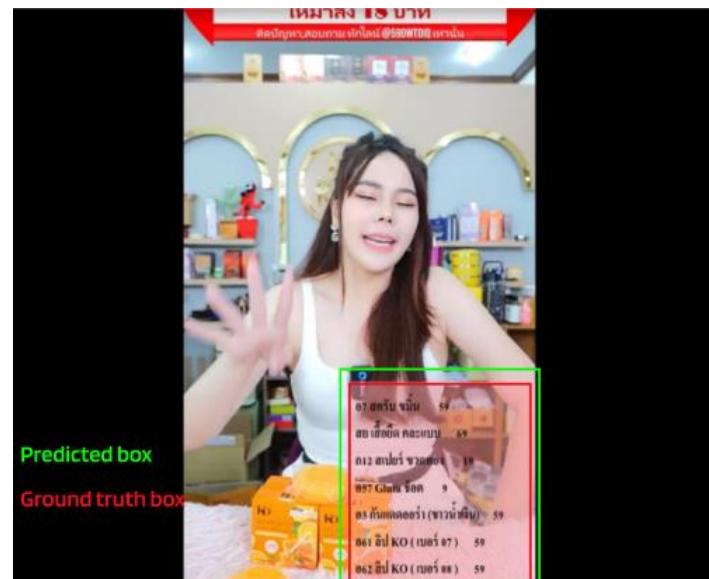
0	0	1	1	0	0	0	0	0	0
0	0	1	1	1	1	0	1	1	0
0	0	1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1	1
0	0	1	1	1	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0

1	1	1	1	0	0	0	0	0	0
0	0	1	1	1	0	1	1	0	0
0	0	1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1	1
0	0	1	1	1	1	0	0	0	1
0	0	1	1	0	0	0	0	0	1

ภาพประกอบที่ 3.58 ตัวอย่างการหาค่า IoU

โดยภาพ P คือ ภาพ Matrix แรก และภาพ G คือ ภาพ Matrix ที่สอง และในภาพ P union G คือสีเทา ส่วน P intersection G คือสีเขียว สีเหลืองคือทายเกิน สีแดงคือทายขาด โดยสามารถดังนี้

$$\begin{aligned} \text{union} &= 54 + 54 - 31 = 77 \\ \text{IOU} &= 31/77 = 0.4 \end{aligned}$$



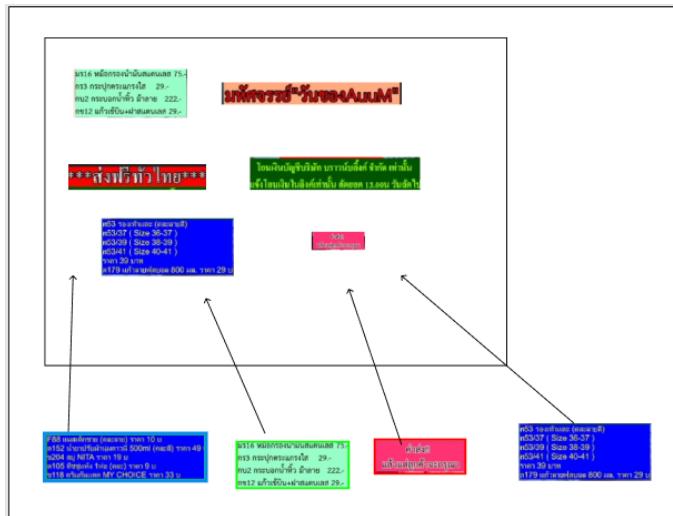
ภาพประกอบที่ 3.59 Predicted box และ Ground truth box

Precision คือ ค่าความแม่นยำจะวัดว่าสามารถค้นหาผลบวกที่แท้จริง (TP) จากการคาดการณ์ เชิงบวกทั้งหมดได้ดีเพียงใด (TP+FP)

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

ภาพประกอบที่ 3.60 การหาค่า Precision

Recall คือ จำนวนที่ถูกต้องจำนวนของ GroundTruth ทั้งหมด ตัวอย่าง Model เราหาย มาเป็นลักษณะนี้ เราจะให้ Model เราพยายามแค่ ครอบข้อความ หมายถึง เอาแต่ข้อความ มาให้เรา เอาอย่างอื่นมาให้ถือว่าผิด

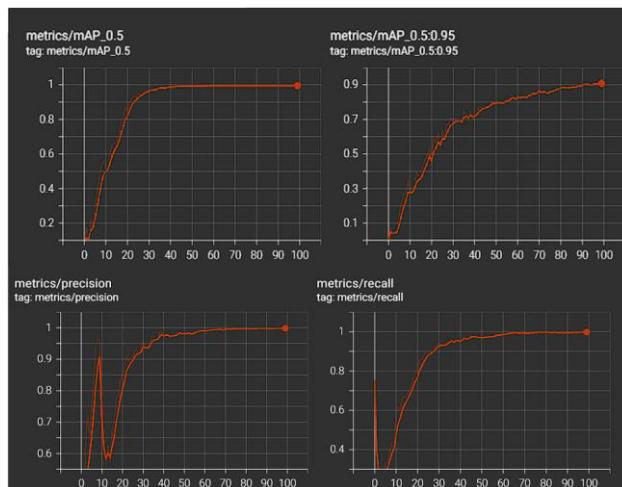


ภาพประกอบที่ 3.61 ตัวอย่างผลลัพธ์ในการทำนาย

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

ภาพประกอบที่ 3.62 การหาค่า recall

จาก model ของเราทำการเลือกภาพมาให้เราดังภาพที่อยู่นอกกรอบ จะเห็นว่า ได้รูปกรอบ ข้อความ ได้มา 3 รูป ส่วน สีเหลี่ยม รูป ลองคำนวณ Precision และ Recall ดูจากตัวอย่างจะได้ 3 TP และ 1 FP ดังนั้น precision คือ  $3/4 = 0.75$  และ recall คือ  $3/6 = 0.5$



ภาพประกอบที่ 3.63 ภาพการเปลี่ยนแปลงค่า mAP precision และ recall ของ model 2. การวัดประสิทธิภาพการ OCR วัดด้วย CER( Character Error Rate)

ค่า CER จะวัดเป็นเปอร์เซ็นต์ สังเกตว่า ยิ่งค่า CER เยอะ ประสิทธิภาพของโมเดลก็จะยิ่งแย่ค่าความเหมาะสมของค่า CER ที่เหมาะสมสำหรับงานที่กำหนดไว้คือค่า CER ต้องน้อยกว่า 10% โดยค่า CER สามารถหาค่าได้ดังนี้

$$CER = \frac{S + D + I}{N}$$

#### ภาพประกอบที่ 3.64 สมการ CER

I (inserted words) คือ จำนวนตัวอักษรที่ถูกแทรกขึ้นมาจากการซื้อความเดิม

D (deleted words) คือ จำนวนตัวอักษรที่หายไปจากการซื้อความเดิม

S (substituted words) คือ จำนวนตัวอักษรที่ถูกแทนที่ไปจากการซื้อความเดิม

N คือ จำนวนตัวอักษรทั้งหมด

A037 เค้กบุบสกสกี้หอบ 33.-	original words
A037 เค้ <span style="background-color: red;">บ</span> บสกสกี้หอบ 33.-	deleted words
A037 เค้กบุบสกสกี้หอบ 33.-	inserted words
A037 เค้กบุบสกสกี้หอบ <span style="background-color: green;">ก</span> 33.-	substituted words

#### ภาพประกอบที่ 3.65 ตัวอย่างตัวแปรในสมการ CER

original words: A037 เค้กบุบสกสกี้หอบ 33.-

OCR result: A037 เค้บบสกสกี้หอบก 33.-

$$CER = (1+1+0)/28$$

#### ภาพประกอบที่ 3.66 ตัวอย่างการหาค่า CER

จาก ภาพประกอบที่ 3.66 ตัวอย่างการหาค่า CER เมื่อนำค่าจากผลลัพท์จากการ OCR มาแทนค่าในตัวแปรทั้ง 4 จะมีผลลัพท์ดังนี้

S (substituted words) จะมีค่าเท่ากับ 1

D (deleted words) จะมีค่าเท่ากับ 1

I (inserted words) จะมีค่าเท่ากับ 0

N จะมีค่าเท่ากับ 28

เมื่อแทนค่าทั้งหมดลงในสมการ  $CER = (S+D+I)/N$  จะได้  $(1+1+0)/28 = 0.07$

เมื่อนำมาคิดเป็นเปอร์เซ็นต์จะได้ค่า CER = 7%

## อ้างอิง

- [1] Hizoka, "Flutter," [Online]. Available: <https://shorturl.asia/zPKJZ>. [Accessed 15 07 2022].
- [2] J. Saengow, "Firebase," [Online]. Available: <https://shorturl.asia/g6H0J>. [Accessed 07 16 2022].
- [3] N. Jesadapatrakul, "Image Processing," [Online]. Available: <https://shorturl.asia/4LirK>. [Accessed 10 06 2022].
- [4] NUMKINGSTON, "Optical Character Recognition," [Online]. Available: <https://shorturl.asia/nmq7h>. [Accessed 12 07 2022].
- [5] S. KANOKTIPSATHARPORN. [Online]. Available: <https://shorturl.asia/twFLQ>. [Accessed 14 07 2022].
- [6] S. D. R. G. a. A. F. J. Redmon, "ieeexplore," [Online]. Available: <https://ieeexplore.ieee.org/document/7780460>. [Accessed 14 07 2022].
- [7] H. Z. a. Z. N. F. Zhou, "Safety Helmet Detection Based on YOLOv5," [Online]. Available: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9362711>. [Accessed 07 15 2022].
- [8] I. Kunakorntum, "medium," [Online]. Available: <http://surl.li/ctfuc>. [Accessed 15 07 2022].
- [9] E. BARRETT, "fortune," [Online]. Available: <https://fortune.com/2018/10/28/in-china-facial-recognition-tech-is-watching-you/>. [Accessed 14 07 2022].