การจำแนกเกรดรังนกแอ่นกินรังด้วย YOLOv8n

พงศ์ภัค เกษมสวัสดิ์

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
บัณฑิตศึกษา สาขาเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย-ญี่ปุ่น
ปีการศึกษา 2568

CLASSIFYING SWALLOW NESTS BY GRADE USING YOLOV8N

Pongpak Kasemsawat

A Term Paper Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Information Technology
Graduate Studies
Thai-Nichi Institute of Technology
Academic Year 2025

หัวข้อสารนิพนธ์	การจำแนกเกรดรังนกแอ่นกินรังด้วย YOLOv8n
โดย	พงศ์ภัค เกษมสวัสดิ์
สาขาวิชา	เทคโนโลยีสารสนเทศ
อาจารย์ที่ปรึกษาสารนิพนธ์	ดร.ศรายุทธ นนท์ศิริ

บัณฑิตศึกษา สถาบันเทคโนโลยีไทย-ญี่ปุ่น อนุมัติให้นับสารนิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....รองอธิการบดีฝ่ายวิชาการ
(รองศาสตราจารย์ ดร.วรากร ศรีเชวงทรัพย์)
วันที่.....เดือน.....พ.ศ.....

คณะกรรมการสอบสารนิพนธ์

.....ประธานกรรมการ
(ดร.ประมุข บุญเสี่ยง)

.....กรรมการ
(ว่าที่ร้อยตรี ดร.พิชิตชัย คำอินทร์)

.....อาจารย์ที่ปรึกษาสารนิพนธ์
(ดร.ศรายุทธ นนท์ศิริ)

พงศ์ภัค เกษมสวัสดิ์ : การจำแนกเกรดรังนกแอ่นกินรังด้วย YOLOv8n อาจารย์ที่ปรึกษา : ดร.ศรายุทธ นนท์ศิริ, 43 หน้า.

งานวิจัยนี้มุ่งเน้นการใช้โมเดล YOLOv8n สำหรับการจำแนกเกรดรังนกแอ่นกินรัง โดยอิงจากชุดข้อมูลที่ปรับแต่งเองซึ่งประกอบด้วย 3 คลาส ได้แก่ เกรดเอ, เกรดบี และ รังมุม การศึกษานี้ประเมินประสิทธิภาพของโมเดลในด้าน Precision, Recall และ F1 Score พร้อมทั้งปรับค่า Hyperparameters เพื่อเพิ่มความแม่นยำในการตรวจจับ โมเดล YOLOv8n สามารถทำค่า F1 Score ได้สูงถึง 0.99 ที่ค่า Confidence Threshold 0.442 และมีค่า mAP@0.5 และ mAP@0.5:0.95 เท่ากับ 0.995 และ 0.779 ตามลำดับ ผลการทดลองแสดงถึงความสามารถในการตรวจจับที่มีประสิทธิภาพสำหรับคลาส เกรดเอ และเกรดบี ในขณะที่คลาส รังมุม มีอัตราการทำนายผิดพลาดสูงกว่า ซึ่งชี้ให้เห็นว่าเรายังสามารถปรับปรุงแก้ไขให้ดีขึ้นกว่านี้ได้ กระบวนการฝึกแสดงถึงการ Convergence ที่มีประสิทธิภาพ ด้วยการลดค่าความสูญเสียอย่างต่อเนื่องและเวลาในการประมวลผลเฉลี่ย 12 ms ทำให้โมเดลนี้เหมาะสำหรับการใช้งานแบบเรียลไทม์ ผลการวิจัยนี้ชี้ให้เห็นถึงความมีประสิทธิภาพและความน่าเชื่อถือของ YOLOv8n สำหรับการจำแนกเกรดรังนกแอ่นกินรัง

บัณฑิตศึกษา ลายมือชื่อนักศึกษา .....

สาขาวิชา เทคโนโลยีสารสนเทศ ลายมือชื่ออาจารย์ที่ปรึกษา .....

ปีการศึกษา 2568

PONGPAK KASEMSAWAT : CLASSIFYING SWALLOW NESTS BY GRADE USING YOLOV8N.
ADVISOR : DR.SARAYUT NONSIRI, 43 PP.

This research focuses on using the YOLOv8n object detection model for classifying the grades of edible bird's nests, based on a custom dataset containing three classes: grade A, grade B, and corner. The study evaluates the model's performance in terms of precision, recall, and F1 score while optimizing hyperparameters for improved detection accuracy. The YOLOv8n model achieved an F1 score of 0.99 at an optimal confidence threshold of 0.442, with mAP@0.5 and mAP@0.5:0.95 values of 0.995 and 0.779, respectively. The results show strong detection performance for grade A and grade B, while the corner class exhibited higher misclassification rates, indicating room for improvement. The training process demonstrated effective convergence, with consistent reductions in loss values and an average inference time of 12ms, making the model suitable for real-time applications. The findings underscore YOLOv8n efficiency and reliability for classifying edible bird's nest grades.

Graduate Studies Student's Signature.....
Field of Study Information Technology Advisor's Signature.....
Academic Year 2025

กิตติกรรมประกาศ

ในการจัดทำสารนิพนธ์ฉบับนี้ได้รับความกรุณาและการสนับสนุนจากหลายภาคส่วน ผู้วิจัยขอขอบพระคุณ ดร.ศรายุทธ นนท์ศิริ อาจารย์ที่ปรึกษาสารนิพนธ์ ซึ่งได้ให้คำแนะนำ ตรวจแก้ไข และชี้แนะแนวทางการทำวิจัยอย่างละเอียดตลอดระยะเวลา การสนับสนุนทั้งด้านองค์ความรู้ซึ่งเป็นส่วนสำคัญอย่างยิ่งที่ทำให้งานวิจัยฉบับนี้สำเร็จได้

ผู้วิจัยขอขอบคุณ สโรชา อิ่มประภา เพื่อนร่วมการศึกษา ผู้ให้คำปรึกษาและข้อเสนอแนะด้านรูปเล่ม ตลอดจนเป็นผู้ชักชวนให้เข้ามาศึกษาต่อในหลักสูตรนี้ อีกทั้งยังร่วมแลกเปลี่ยนความรู้และช่วยเหลือกันในการศึกษาอย่างต่อเนื่องจนทำให้การเรียนและการทำสารนิพนธ์สำเร็จลุล่วงไปด้วยดี

สุดท้ายนี้ ผู้วิจัยขอขอบคุณครอบครัว เพื่อนร่วมงาน และผู้เกี่ยวข้องทุกท่านที่ให้การสนับสนุนในทุกด้าน ทำให้ผู้วิจัยสามารถดำเนินงานวิจัยฉบับนี้จนสำเร็จสมบูรณ์

พงศ์ภัค เกษมสวัสดิ์

สารบัญ

สารบัญ (ต่อ)

สารบัญตาราง

สารบัญรูป

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

รังนกแอ่นกินรังเป็นหนึ่งในสมุนไพรจีนที่มีมูลค่าสูงในปัจจุบัน [1] โดยผลิตภัณฑ์นี้ได้จากน้ำลายของนกแอ่นกินรัง มีชื่อทางวิทยาศาสตร์ว่า Aerodramus Fuciphagus [2] รังนกที่มีคุณภาพสูงสุดจะมาจากการเก็บในถ้ำ ซึ่งรังจะมีขนาดใหญ่ เนื้อหนา และมีสารอาหารมากกว่า ในขณะที่รังจากฟาร์มหรือ “คอนโดนก” จะมีคุณภาพรองลงมา

สถานการณ์ปัจจุบันมีความต้องการรังนกเพิ่มขึ้นอย่างมาก โดยเฉพาะจากประเทศจีนและฮ่องกง การผลักดันของกระทรวงเกษตรและสหกรณ์ร่วมกับสมาคมการค้าผู้ผลิตและค้ารังนกแอ่น (ประเทศไทย) ทำให้ราคารังนกปรับตัวสูงขึ้นมากกว่า 30,000 บาทต่อกิโลกรัม เกษตรกรไทยหันมาทำฟาร์มเลี้ยงนกแอ่นเพิ่มขึ้นทั่วทุกภูมิภาค ปัจจุบันมีฟาร์มรังนกประมาณ 10,000 แห่งทั่วประเทศ โดยจำนวนประชากรนกแอ่นเพิ่มขึ้นเป็น 5 ล้านตัวจากอดีตที่มีเพียงเล็กน้อย

จากข้อมูลการวิจัยที่ได้รับการสนับสนุนจากสำนักงานกองทุนสนับสนุนการวิจัย (สกว.) ในปี พ.ศ. 2562 พบว่า ในพื้นที่จังหวัดนครศรีธรรมราช ปัตตานี นราธิวาส และสมุทรสาคร มีบ้านรังนกหรือคอนโดนกจำนวน 17,720 หลัง และมีปริมาณนกเพิ่มขึ้นกว่า 9.6 ล้านตัว แนวโน้มในอนาคตคาดว่าจะมีจำนวนเพิ่มขึ้นอย่างต่อเนื่อง ส่งผลให้ปริมาณผลผลิตรังนกแอ่นที่ทำรังในสิ่งปลูกสร้างเติบโตตามไปด้วย

อย่างไรก็ตาม การทำฟาร์มนกแอ่นในเขตชุมชนต้องเผชิญกับข้อบังคับทางกฎหมายเกี่ยวกับการเลี้ยงนกแอ่นในอาคาร ตามพระราชบัญญัติสงวนและคุ้มครองสัตว์ป่า พ.ศ. 2535 และกฎกระทรวงปี พ.ศ. 2546 ซึ่งให้นกแอ่นกินรังเป็นสัตว์ป่าคุ้มครอง [3] บุคคลที่เก็บหรือครอบครองรังนกแอ่นต้องได้รับอนุญาตตามพระราชบัญญัติอากรรังนกแอ่น พ.ศ. 2540 นอกจากนี้ ยังมีข้อบังคับตามพระราชบัญญัติการสาธารณสุข พ.ศ. 2535 ในการควบคุมเรื่องผลกระทบต่อชุมชน เช่น กลิ่น เสียง และเชื้อโรคจากการเลี้ยงนกในอาคาร

ความต้องการรังนกจากตลาดต่างประเทศที่เพิ่มขึ้นนี้ ทำให้เกิดโอกาสสำหรับการพัฒนาและเพิ่มประสิทธิภาพการผลิต ในงานวิจัยนี้ จึงมีการนำเสนอแนวทางการใช้เทคโนโลยีในการคัดแยกเกรดรังนกอย่างมีประสิทธิภาพ โดยเฉพาะการใช้โมเดลการเรียนรู้เชิงลึก (Deep Learning) อย่าง YOLO ซึ่งเป็นหนึ่งในเทคโนโลยีการตรวจจับวัตถุที่มีความแม่นยำและรวดเร็ว

เทคโนโลยี YOLO มีศักยภาพในการช่วยจำแนกคุณภาพรังนก โดยสามารถตรวจจับขนาดและคุณลักษณะเฉพาะอื่นๆ ของรังนกได้อย่างแม่นยำ โมเดลนี้สามารถประมวลผลภาพรังนก และคัดแยกตามเกรดต่าง ๆ ได้อัตโนมัติ ซึ่งจะช่วยลดเวลาและความผิดพลาดในการคัดแยกด้วยคน อีกทั้งยังสามารถประยุกต์ใช้กับการผลิตในระดับอุตสาหกรรมเพื่อเพิ่มความสามารถในการแข่งขันในตลาด

การนำเทคโนโลยีนี้เข้ามาใช้ในกระบวนการผลิตรังนกแอ่นกินรัง จะไม่เพียงแค่เพิ่มประสิทธิภาพในด้านการคัดแยกเกรดเท่านั้น แต่ยังช่วยเพิ่มความเชื่อมั่นในคุณภาพของผลิตภัณฑ์รังนกไทยในตลาดโลกอีกด้วย

1.2 วัตถุประสงค์ของการวิจัย

เพื่อศึกษา วิเคราะห์ และ จำแนกเกรดรังนกแอ่นกินรังด้วย YOLOv8n

1.3 ขอบเขตของการวิจัย

ออกแบบและเตรียมชุดข้อมูลที่ใช้สำหรับการเรียนรู้ของโปรแกรม ด้วย YOLOV8n

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 สามารถคัดแยกเกรดรังนกแอ่นกินรังได้ ด้วย YOLOv8n

1.4.2 สามารถลดระยะเวลาในการคัดแยกเกรดรังนกแอ่นกินรังได้และเพิ่มความแม่นยำในการคัดแยกเกรดรังนกแอ่นกินรัง

บทที่ 2
แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

ในการศึกษาเรื่องเทคโนโลยีตรวจจับวัตถุ เพื่อคัดแยกเกรดรังนกแอ่นทางผู้ดำเนินงานวิจัยได้สืบค้นข้อมูล และ ทฤษฎีและงานวิจัยที่เกี่ยวข้อง เพื่อเป็นแนวทางในการศึกษา โดยมี หัวข้อดังต่อไปนี้

2.1 การคัดเกรดรังนกแอ่นกินรัง

2.2 ปัญญาประดิษฐ์ (Artificial Intelligence)

2.3 การเรียนรู้ของเครื่อง (Machine Learning)

2.4 การเรียนรู้เชิงลึก (Deep Learning)

2.5 โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Network)

2.6 การตรวจจับวัตถุ (Object Detection)

2.7 LabelImg

2.8 YOLO (You Only Look Once)

2.9 Transfer Learning

2.10 COCO Dataset

2.11 การประมวลผลภาพ (Image Processing)

2.12 การประมวลผลภาพและวิเคราะห์แบบดิจิทัล (Digital Image Processing)

2.13 คอมพิวเตอร์วิทัศน์ (Computer Vision)

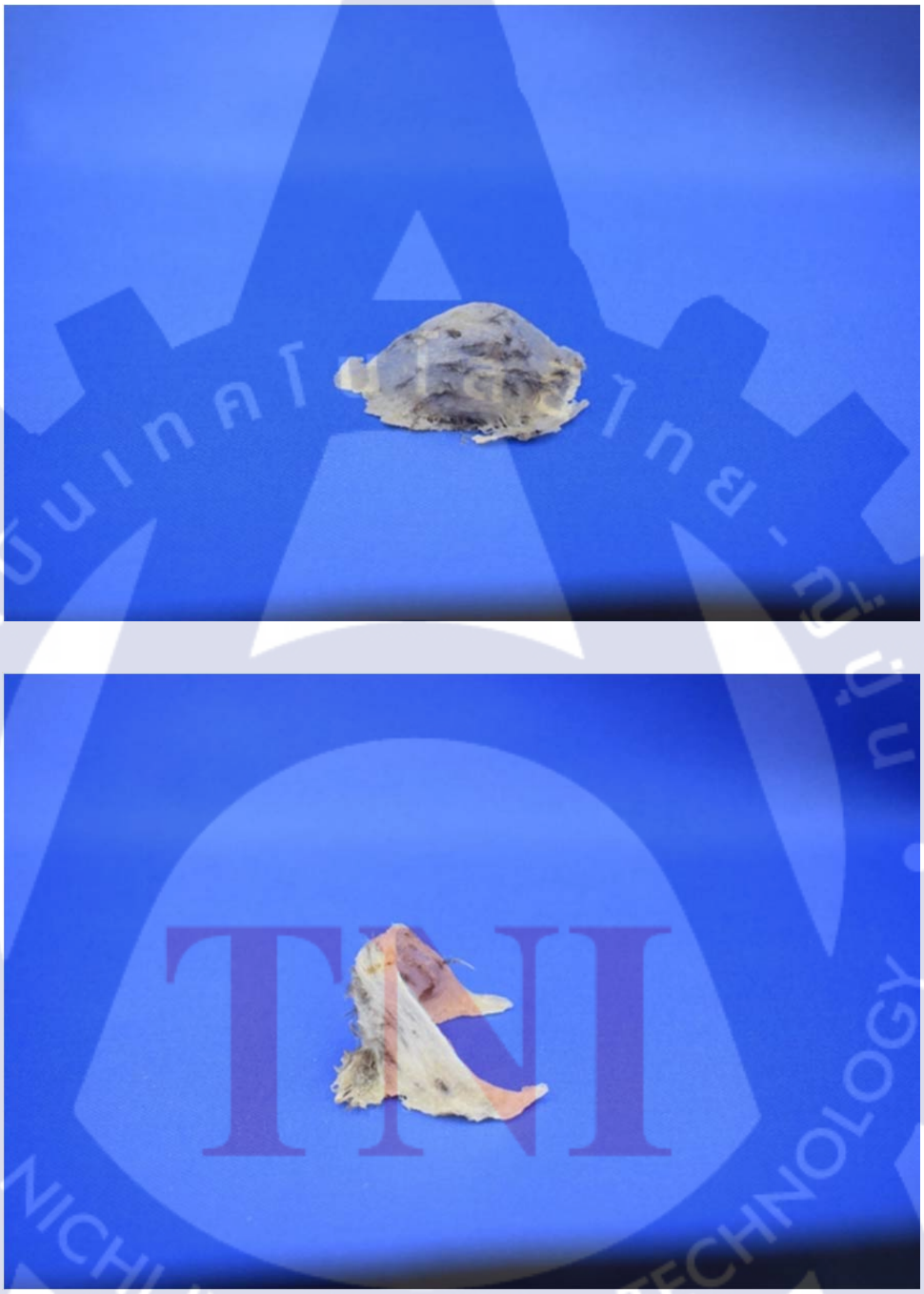
2.14 ชุดข้อมูล Dataset

2.1 การคัดเกรดรังนกแอ่นกินรัง

2.1.1 รังนกเกรดเอ คือรังนกที่มีรูปทรงสมบูรณ์ตามธรรมชาติมากที่สุด มีความต้องการในตลาดสูง และมีลักษณะสีขาวอมเหลือง

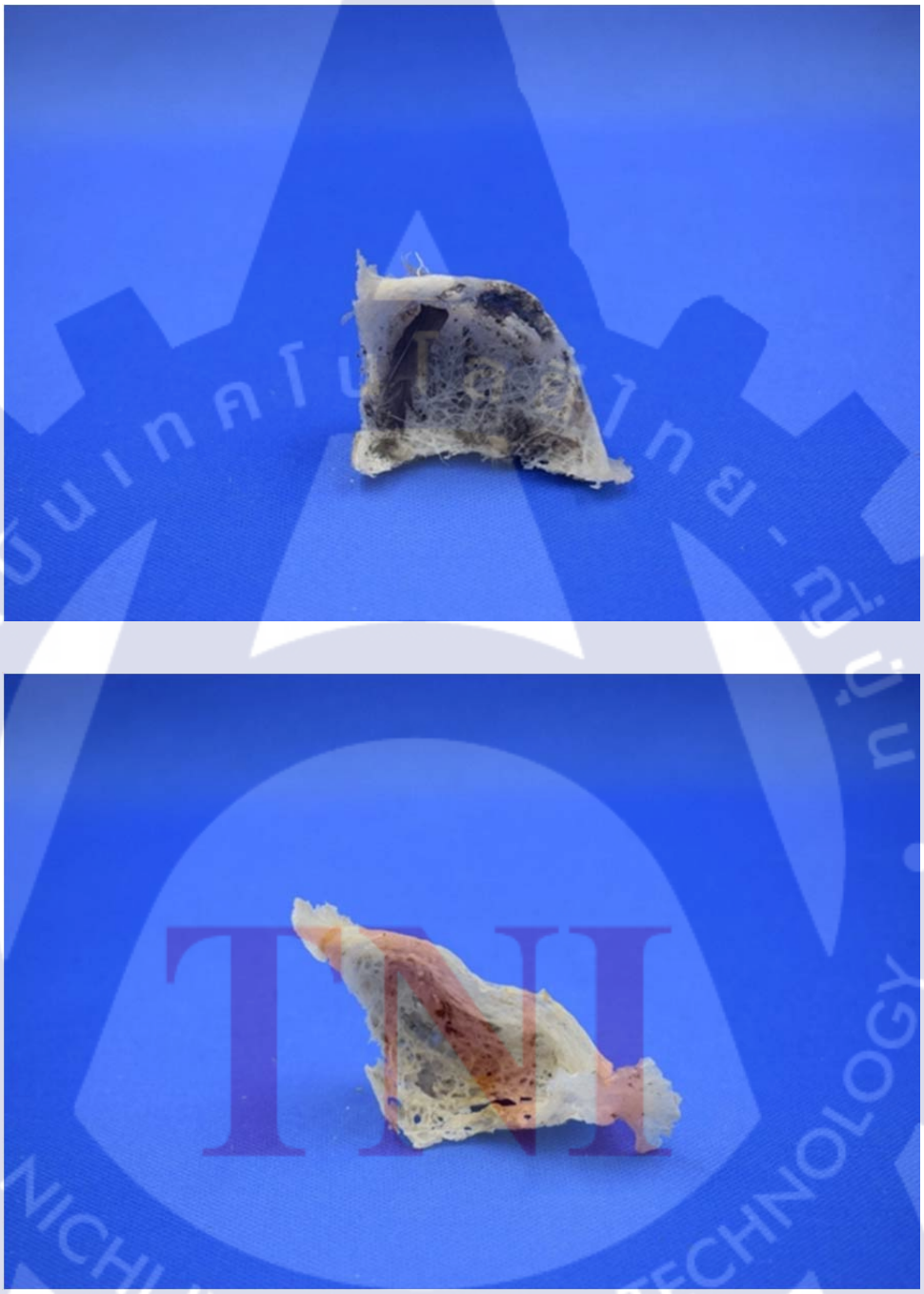
รูปที่ 2.1 รังนก เกรดเอ

2.1.2 รังนกเกรดบี คือรังที่มีรูปทรงเป็นไปตามธรรมชาติของรังนกแอ่นกินรังโดยจะมีคุณภาพรองจากเกรดเอ



รูปที่ 2.2 รังนก เกรดบี

2.1.3 รังมุม คือรังนกที่สร้างบริเวณรอยต่อของมุมไม้ที่ตัดกัน รังประเภทนี้มักไม่เป็นที่นิยมในตลาดและมีมูลค่าต่ำกว่ารังนกรูปทรงปกติ



รูปที่ 2.3 รังมุม

2.2 ปัญญาประดิษฐ์ (Artificial Intelligence)

ปัญญาประดิษฐ์ (Artificial Intelligence: AI) หมายถึง การออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์และระบบที่มีความสามารถในการทำงานในลักษณะที่คล้ายคลึงกับมนุษย์ ไม่ว่าจะเป็นการเรียนรู้ การตัดสินใจ การแก้ปัญหา รวมถึงการประมวลผลข้อมูลภาพและเสียง ซึ่ง AI เป็นเทคโนโลยีที่มีการนำไปประยุกต์ใช้อย่างกว้างขวางในหลายสาขา เช่น การพัฒนาเกมคอมพิวเตอร์ ระบบค้นหาข้อมูล การวิเคราะห์ข้อมูลภาพและเสียง และงานอื่น ๆ ที่ต้องการการตัดสินใจเชิงซับซ้อน

หนึ่งในกระบวนการสำคัญที่ใช้ในปัญญาประดิษฐ์ (Artificial Intelligence: AI) คือการเรียนรู้เชิงลึก (Deep Learning) ซึ่งเป็นการสร้างแบบจำลองการเรียนรู้ผ่านเครือข่ายประสาทเทียม (Artificial Neural Networks) โมเดลนี้จะทำการเรียนรู้จากข้อมูลที่ถูกป้อนเข้ามา (Input) และประมวลผลข้อมูลเหล่านั้นเพื่อให้ได้ผลลัพธ์ที่ถูกต้อง (Output) ผ่านกระบวนการปรับแต่งพารามิเตอร์หรือค่าคงที่ในโมเดลเพื่อให้สามารถทำงานได้อย่างมีประสิทธิภาพมากขึ้น การปรับแต่งพารามิเตอร์นี้ช่วยให้ระบบสามารถเรียนรู้และทำความเข้าใจข้อมูลได้ดีขึ้น เมื่อได้รับข้อมูลเพิ่มเติม

ปัญญาประดิษฐ์ (Artificial Intelligence: AI) มีบทบาทสำคัญในการพัฒนานวัตกรรมใหม่ๆ ที่ช่วยแก้ไขปัญหาที่ซับซ้อนและเพิ่มประสิทธิภาพการทำงาน เช่น ในการวิเคราะห์ภาพทางการแพทย์เพื่อการวินิจฉัยโรค การพัฒนารถยนต์ไร้คนขับ การจัดการระบบค้นหาข้อมูลในโลกออนไลน์ และการสร้างผู้ช่วยเสมือน (Virtual Assistants) ที่สามารถตอบสนองความต้องการของผู้ใช้งานในลักษณะอัตโนมัติ

ปัญญาประดิษฐ์(Artificial Intelligence: AI) โดยหลักๆ จะแบ่งออกเป็น 3 ประเภท

2.2.1 Artificial Narrow Intelligence (ANI)

ระบบระบบปัญญาประดิษฐ์ที่ถูกออกแบบและเข้าใจได้เฉพาะในงานที่มีขอบเขตที่จำกัดหรือถูกกำหนดไว้ล่วงหน้า โดย ANI ไม่สามารถทำงานอื่นๆ นอกเหนือจากขอบเขตที่กำหนดไว้ได้ ดังนั้น Artificial Narrow Intelligence (ANI) มักถูกนำไปใช้งานในงานหรือกิจกรรมที่มีความซ้ำซาก เช่น การเตรียมข้อมูลในธุรกิจ การค้นหาข้อมูลในระบบ หรือการทำงานในสายอุตสาหกรรม แต่ยังไม่สามารถเข้าใจและแก้ไขปัญหาที่ซับซ้อนหรือมีความยืดหยุ่นมากๆ ได้เช่นกัน ยกตัวอย่างเช่น ระบบตอบโต้อัตโนมัติของธนาคารที่สามารถตอบคำถามเกี่ยวกับบัญชีธนาคารของลูกค้าได้ แต่ไม่สามารถทำงานอื่นที่ไม่ได้ระบุไว้ล่วงหน้าได้

2.2.2 Artificial General Intelligence (AGI)

คือ การพัฒนาเทคโนโลยีปัญญาประดิษฐ์ (AI) ที่สามารถทำงานและแก้ไขปัญหาได้หลากหลายแบบในระดับที่เทียบเท่ากับความสามารถของมนุษย์ โดย Artificial General Intelligence (AGI) จะต้องมีความสามารถในการเรียนรู้และปรับปรุงตนเองได้เอง มีความสามารถในการประเมินและ

ปรับตัวเพื่อเหมาะสมกับสถานการณ์และงานที่กำลังจะทำ รวมถึงสามารถเข้าใจและสื่อสารด้วยภาษาธรรมชาติได้ อย่างไรก็ตาม AGI ยังไม่ได้ถูกพัฒนาขึ้นมาในปัจจุบัน แต่เป็นวิสัยที่ถูกต้องว่าถ้ามีการพัฒนาให้สำเร็จ จะมีผลกระทบต่อการดำเนินชีวิตและเทคโนโลยีต่างๆ อย่างมากมายในอนาคต

2.2.3 Artificial Super Intelligence (ASI)

คือ ระดับของเทคโนโลยีปัญญาประดิษฐ์ (AI) ที่มีความสามารถในการคิดและแก้ไขปัญหาอย่างมีประสิทธิภาพมากกว่าความสามารถของมนุษย์ โดย ASI สามารถเรียนรู้และปรับปรุงตนเองได้โดยอิสระและมีความสามารถในการทำงานที่หลากหลายและซับซ้อนเกินกว่าความสามารถของมนุษย์จะทำได้ โดย ASI มักถูกพูดถึงเป็นการพัฒนา AI ที่เป็นเป้าหมายสูงสุดของการวิจัยด้านปัญญาประดิษฐ์ ถ้า ASI ถูกพัฒนาขึ้นมาจริง มันอาจจะมีความสามารถในการคิดและสร้างสิ่งใหม่ได้อย่างอิสระ ซึ่งอาจมีผลกระทบต่อการดำเนินชีวิตและสังคมอย่างมากในอนาคต

2.3 การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง (Machine Learning: ML) เป็นส่วนหนึ่งของปัญญาประดิษฐ์ (Artificial Intelligence: AI) ที่มุ่งเน้นการใช้ชุดอัลกอริทึมที่ผ่านการฝึกฝนบนชุดข้อมูล (Dataset) เพื่อสร้างโมเดล (Model) ที่ช่วยให้เครื่องจักรสามารถทำงานได้จากที่เดิมทีต้องใช้มนุษย์ เช่น การทำนายความผันผวนของราคา การวิเคราะห์ข้อมูล และการจำแนกภาพ นอกจากนี้การเรียนรู้ของเครื่อง (Machine Learning: ML) ยังเน้นให้เครื่องจักรสามารถเรียนรู้จากข้อมูล เข้าใจรูปแบบ และตัดสินใจได้ด้วยการมีส่วนร่วมน้อยลงจากมนุษย์

ธุรกิจหรืออุตสาหกรรมที่นำเทคโนโลยีนี้มาใช้สามารถสร้างความได้เปรียบในการแข่งขัน ซึ่งช่วยเพิ่มช่องว่างระหว่างผู้นำและผู้ตามในตลาด ธุรกิจสามารถลดเวลาในการวิเคราะห์ข้อมูลและลดต้นทุนแรงงานได้อย่างมีนัยสำคัญ

หลักการทำงานของการเรียนรู้ของเครื่อง การเรียนรู้ของเครื่องมีหลักการที่คล้ายกับการเรียนรู้ของมนุษย์ ซึ่งต้องการการเรียนรู้จากประสบการณ์ ตัวอย่างเช่น การสอนเด็กให้แยกความแตกต่างระหว่างดินสอและปากกา เราต้องเริ่มจากการสอนให้เด็กทราบว่าดินสอและปากกามีลักษณะอย่างไรเพื่อให้เด็กสามารถเรียนรู้และแยกความแตกต่างระหว่างสิ่งของสองอย่างนี้ได้

รูปแบบการเรียนรู้หลัก 3 ปะเภท การเรียนรู้ของเครื่องสามารถแบ่งออกเป็น 3 กลุ่มใหญ่ตามลักษณะของข้อมูลที่ป้อนเข้าไป ดังนี้

การเรียนรู้แบบมีผู้สอน (Supervised Learning) คือ กระบวนการที่คอมพิวเตอร์เรียนรู้จากข้อมูลที่มี “คำตอบ” หรือ “ฉลาก” กำกับอยู่แล้ว ตัวอย่างเช่น ถ้าเราสอนให้คอมพิวเตอร์แยกแยะ

ระหว่าง “แมว” กับ “สุนัข” เราจะให้มันดูรูปแมวและสุนัขจำนวนมาก พร้อมบอกว่ารูปไหนคืออะไร เมื่อมันเห็นรูปใหม่ มันจะสามารถทำนายได้ว่ารูปนั้นเป็นแมวหรือสุนัขโดยอ้างอิงจากสิ่งที่เคยเรียนรู้

การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) ในทางตรงกันข้าม การเรียนรู้แบบไม่มีผู้สอนคือกระบวนการที่ระบบประมวลผลจากข้อมูลที่ไม่มีการระบุคำตอบล่วงหน้า คอมพิวเตอร์จะต้องค้นหารูปแบบหรือความสัมพันธ์เองตัวอย่างเช่น หากนำเข้าข้อมูลพฤติกรรมการซื้อของลูกค้าโดยไม่มีการระบุว่าใครเป็นลูกค้ากลุ่มไหน ระบบอาจค้นพบเองว่าลูกค้ากลุ่มนี้มักซื้อสินค้าประเภทเดียวกันแล้วจัดกลุ่มลูกค้าให้โดยอัตโนมัติ ซึ่งมีประโยชน์ในงานตลาดและแนะนำสินค้าได้ดีขึ้น

การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning) การเรียนรู้แบบเสริมกำลังเป็นกระบวนการที่ตัวแทน (Agent) ทำการเรียนรู้ผ่านการปฏิสัมพันธ์กับสิ่งแวดล้อม โดยจะได้รับรางวัล (Reward) หรือบทลงโทษ (Penalty) ขึ้นอยู่กับผลลัพธ์ที่ได้รับ วิธีนี้ใช้กระบวนการลองผิดลองถูกเพื่อพัฒนาแนวทางการตัดสินใจให้เหมาะสมที่สุด ตัวอย่างที่เห็นได้ชัดคือ AlphaGo ซึ่งเป็นปัญญาประดิษฐ์ที่เรียนรู้กลยุทธ์การเล่นหมากล้อมโดยวิเคราะห์การตอบโต้ของคู่แข่งและปรับปรุงการตัดสินใจให้มีประสิทธิภาพมากขึ้น

ประโยชน์ของการเรียนรู้ของเครื่อง เทคโนโลยีการเรียนรู้ของเครื่องมีการนำไปประยุกต์ใช้ในหลายด้าน เช่น ระบบนำทางอัจฉริยะ Google Maps ใช้ Machine Learning เพื่อคำนวณเส้นทาง ระยะทาง และเวลาเดินทางโดยประมาณ การแปลภาษาอัตโนมัติ Google Translate ใช้เทคนิคการเรียนรู้ของเครื่องเพื่อเพิ่มความแม่นยำในการแปลข้อความ การรู้จำเสียงพูดแอปพลิเคชัน LINE ใช้เทคโนโลยีแปลงเสียงเป็นข้อความ (Speech-to-Text) เพื่อช่วยให้ผู้ใช้สามารถสื่อสารได้สะดวกยิ่งขึ้น

2.4 การเรียนรู้เชิงลึก (Deep Learning)

การเรียนรู้เชิงลึก (Deep Learning) เป็นสาขาหนึ่งของการเรียนรู้ของเครื่อง (Machine Learning) ที่เน้นการสร้างโมเดลปัญญาประดิษฐ์ที่มีโครงสร้างซับซ้อนและมีหลายชั้น เพื่อให้สามารถเรียนรู้และทำนายผลลัพธ์จากข้อมูลที่มีปริมาณมากและซับซ้อนได้อย่างแม่นยำ โดยใช้เครือข่ายประสาทเทียม (Neural Networks) เป็นโครงสร้างหลัก

เครือข่ายประสาทเทียมในการเรียนรู้เชิงลึกประกอบด้วยหลายชั้น (Layers) ซึ่งแต่ละชั้นทำหน้าที่ประมวลผลข้อมูลแล้วส่งต่อผลลัพธ์ไปยังชั้นถัดไป ชั้นเหล่านี้ทำงานร่วมกันเพื่อค้นหารูปแบบที่ซับซ้อนในข้อมูล ตัวอย่างเช่น [4] โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Networks: CNNs) มักถูกใช้ในการประมวลผลข้อมูลภาพ ในขณะที่โครงข่ายประสาทแบบลำดับ (Recurrent Neural Networks: RNNs) มักถูกนำไปใช้กับข้อมูลที่เป็นลำดับ เช่น ข้อความหรือเสียง

การเรียนรู้เชิงลึก [5] ได้รับความสนใจเพิ่มขึ้นในช่วงสองถึงสามทศวรรษที่ผ่านมา เนื่องจากสามารถจัดการข้อมูลที่ซับซ้อน เช่น การจำแนกภาพ การรู้จำเสียง การแปลภาษา และการตอบคำถามได้อย่างมีประสิทธิภาพ นอกจากนี้ ยังมีการประยุกต์ใช้อย่างแพร่หลายในหลายอุตสาหกรรม เช่น ใน

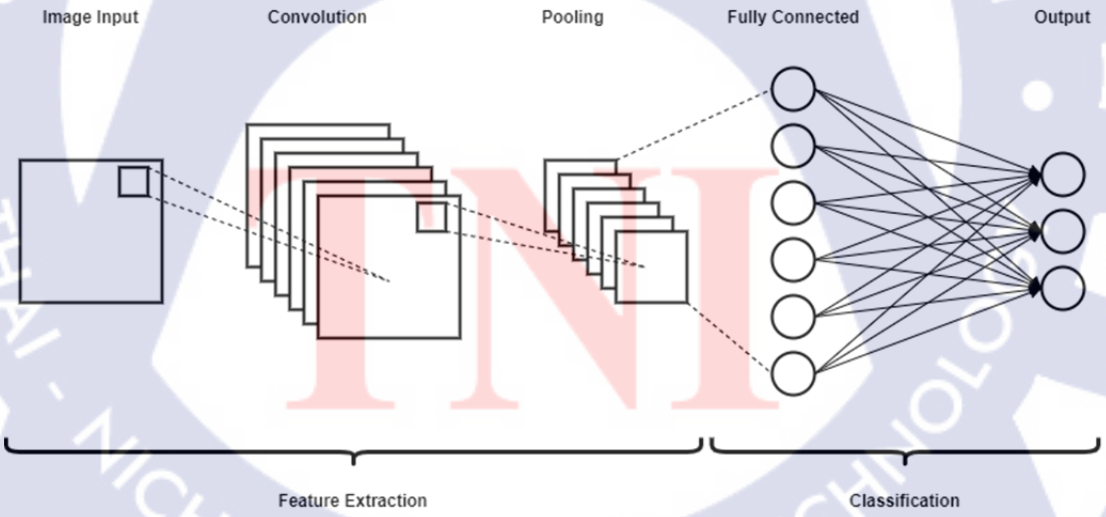
เทคโนโลยีรถยนต์ขับเคลื่อนอัตโนมัติ และการวิเคราะห์ภาพทางการแพทย์ ซึ่งใช้เพื่อช่วยในการวินิจฉัยโรค [6], [7]

2.5 โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Network)

เป็นโมเดลปัญญาประดิษฐ์ที่ถูกออกแบบมาเพื่อการประมวลผลภาพและวิดีโอ โดย CNN มีโครงสร้างที่มองภาพเป็นเมทริกซ์และใช้เทคนิคการคอนโวลูชัน (Convolution) เพื่อสกัดลักษณะเด่น (Features) จากภาพเพื่อนำไปใช้ในการจำแนกหรือทำนาย โดยมีชั้นโครงข่าย (Layers) หลายชั้นซึ่งประกอบด้วยชั้นคอนโวลูชัน (Convolution Layers) ชั้นพูลลิ่ง (Pooling Layers) และชั้นเชื่อมโยง (Fully Connected Layers)

ในชั้นคอนโวลูชัน จะนำเมทริกซ์และ Kernel หรือ Filter มาทำการคอนโวลูชันกัน เพื่อสกัดลักษณะเด่นออกมา โดยที่ Kernel จะเป็นเมทริกซ์ขนาดเล็กที่ใช้สำหรับการคำนวณค่าในการคอนโวลูชัน จากนั้นใช้ชั้นพูลลิ่งเพื่อลดขนาดของภาพลงและยังสกัดลักษณะเด่นเพิ่มเติมอีก จนกว่าจะได้ลักษณะเด่นที่เพียงพอต่อการจำแนกหรือทำนาย

ในชั้นเชื่อมโยง จะนำลักษณะเด่นที่สกัดได้จากชั้นคอนโวลูชันและชั้นพูลลิ่งมาเชื่อมโยงกัน เพื่อให้โมเดลสามารถจำแนกหรือทำนายได้โดยอัตโนมัติ และในท้ายที่สุดจะมีชั้นออกผลลัพธ์ (Output Layer) ที่จะแสดงผลลัพธ์การจำแนก



รูปที่ 2.4 Convolutional Neural Network (CNN)

2.5.1 เลเยอร์คอนโวลูชัน (Convolutional Layers)

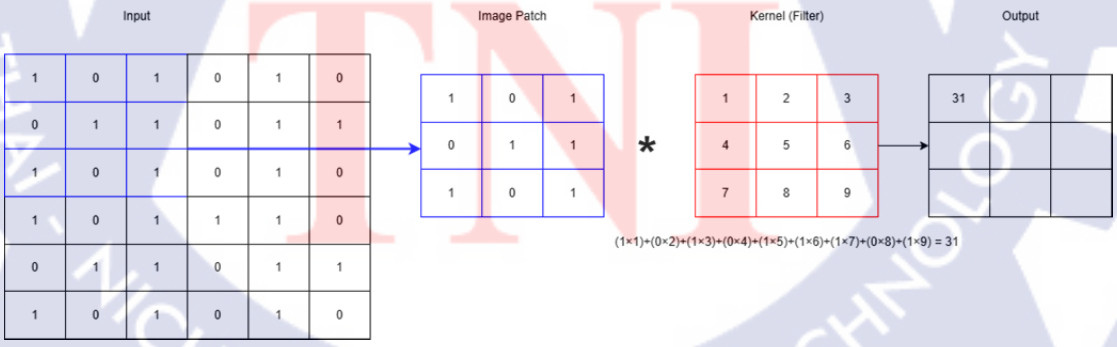
เป็นส่วนสำคัญของโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks CNNs) ที่ถูกพัฒนาขึ้นเพื่อการประมวลผลข้อมูลเชิงพื้นที่ เช่น ภาพและสัญญาณเสียง โครงข่ายนี้ได้รับแรงบันดาลใจจากโครงสร้างการมองเห็นในสมองของมนุษย์โดยเลเยอร์คอนโวลูชัน (Convolutional Layers) [8] ทำหน้าที่ดึงลักษณะเฉพาะ (Features) จากข้อมูลอินพุตผ่านการใช้ตัวกรอง (Filters) หรือ Kernel และหน้าที่สำคัญคือการลดขนาดของข้อมูล [9] เพื่อดึงคุณลักษณะจากพื้นที่ในอินพุตช่วยลดความซับซ้อนของข้อมูลโดยไม่สูญเสียลักษณะสำคัญและสามารถจับลักษณะซับซ้อนได้ดี โดยมีหลักการดังนี้

2.5.1.1 Kernel คือ เมทริกซ์ขนาดเล็กที่เลื่อนผ่านข้อมูลอินพุตเพื่อคำนวณข้อมูลคุณลักษณะเฉพาะในพื้นที่ย่อย ตัวอย่างเช่น หากข้อมูลอินพุตมีขนาด 32×32 พิกเซล และ Kernel มีขนาด 3×3 การคอนโวลูชันจะสร้าง Feature Map ที่แสดงลักษณะเฉพาะในอินพุต

2.5.1.2 Stride คือ กำหนดระยะการเลื่อนของ Kernel ในแต่ละครั้ง ตัวอย่างเช่น Stride 1 หมายถึงการเลื่อน Kernel หนึ่งพิกเซลต่อครั้ง หากเพิ่ม Stride เป็น 2 Kernel จะเลื่อนสองพิกเซลต่อครั้ง ซึ่งช่วยลดขนาดของ Output feature map

2.5.1.3 Padding คือ ใช้เพื่อรักษาขนาดของผลลัพธ์ให้เท่ากับขนาดของอินพุต (Same Padding) หรือเพื่อเพิ่มความสามารถในการจับลักษณะบริเวณขอบ (Valid padding)

2.5.1.4 Activation Function คือ ฟังก์ชันเช่น ReLU (Rectified Linear Unit) ถูกนำมาใช้ในแต่ละตำแหน่งของ feature map เพื่อเพิ่มความไม่เชิงเส้น (Non-linearity) ให้ระบบสามารถจำแนกข้อมูลที่ซับซ้อนได้ดีขึ้น



รูปที่ 2.5 ตัวอย่างการทำงานของ Convolution Layer

2.5.2 Max Pooling Layers

เป็นองค์ประกอบสำคัญในโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks, CNNs) ที่ช่วยลดขนาดข้อมูล (dimensionality reduction) และปรับปรุงประสิทธิภาพในการดึงลักษณะเฉพาะจากข้อมูล โดยทำงานผ่านกระบวนการเลือกค่ามากที่สุดในแต่ละส่วนย่อยของข้อมูลอินพุต (Input) [10]

2.6 การตรวจจับวัตถุ (Object Detection)

คือ กระบวนการที่ใช้เทคโนโลยีการเรียนรู้ของเครื่อง เพื่อตรวจจับวัตถุในภาพหรือวิดีโอ โดยการวิเคราะห์ภาพหรือวิดีโอ เพื่อหาวัตถุที่ต้องการตรวจจับและสามารถแยกแยะจากวัตถุอื่น ที่อยู่ในภาพหรือวิดีโอได้ ซึ่งการตรวจจับวัตถุนั้นเป็นส่วนสำคัญของการทำ Computer Vision โดยจะใช้ Deep Learning และ Convolutional Neural Networks (CNNs) เป็นวิธีการเรียนรู้เพื่อให้เครื่องสามารถรู้จำและระบุวัตถุได้ในภาพหรือวิดีโอ การตรวจจับวัตถุนี้มีการใช้งานอย่างแพร่หลายในงานวิจัยและการพัฒนาแอปพลิเคชันต่างๆ เช่น ระบบประชาสัมพันธ์ต่างๆ, ระบบควบคุมการจราจร, หุ่นยนต์ และโครงการการศึกษาด้านการแยกแยะภาพโรคร้ายแรงต่างๆ ในสิ่งมีชีวิต เป็นต้น

2.7 LabelImg

LabelImg ถูกพัฒนาโดย Tzutalin [11] เขียนขึ้นด้วย Python โดยมีการแสดงผลข้อมูลในรูปแบบ GUI ที่ช่วยให้ผู้ใช้สามารถทำงานได้สะดวกและมีการเผยแพร่เป็นโอเพ่นซอร์สบน GitHub เป็นเครื่องมือโอเพ่นซอร์สสำหรับการทำ Data Annotation โดยเฉพาะการทำ Bounding Box สำหรับภาพถ่ายหรือรูปภาพ ซึ่งถูกนำไปใช้กันอย่างแพร่หลายในการสร้างชุดข้อมูลสำหรับการเรียนรู้เชิงลึก (Deep Learning) ในด้าน Computer Vision เช่น การตรวจจับวัตถุ (Object Detection) และการจำแนกประเภทภาพ (Image Classification)

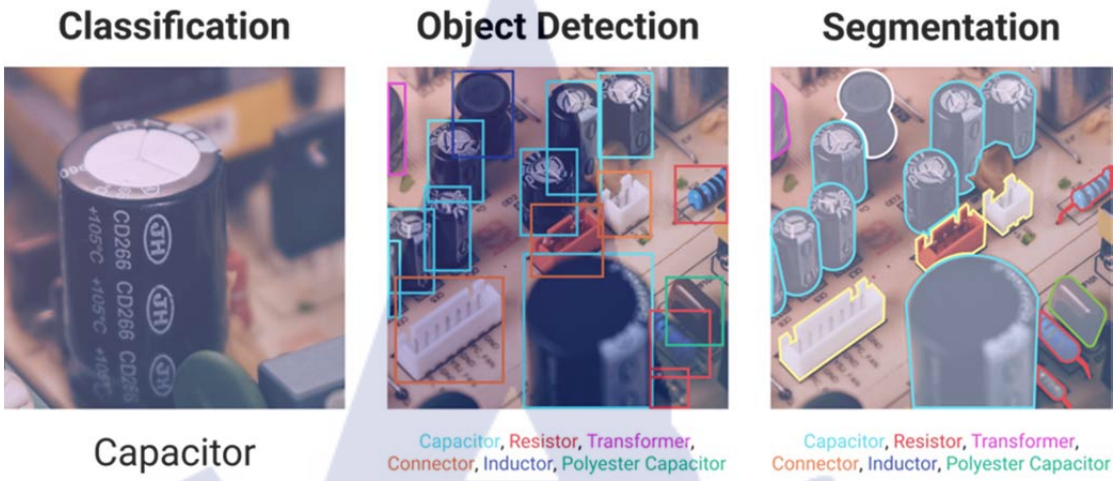
รูปที่ 2.6 โปรแกรม LabelImg

2.8 YOLO (You Only Look Once)

YOLOv8 (You Only Look Once version 8) [12] เป็นโมเดลการตรวจจับวัตถุที่ถูกพัฒนาขึ้นโดยอาศัยโครงข่าย Neural Network แบบ Convolutional Neural Network (CNN) [13] และเทคนิค Deep Learning อื่นๆ เพื่อใช้ในการตรวจจับวัตถุในภาพหรือวิดีโอ โมเดล YOLOv8 นี้ถูกพัฒนาโดย Alexey Bochkovskiy และอัพเดตล่าสุดเมื่อปี 2021 โดยใช้วิธีการอัพเดตแบบ Incremental Improvements ในการปรับปรุงประสิทธิภาพการตรวจจับวัตถุ ทำให้ YOLOv8 มีความแม่นยำและประสิทธิภาพที่ดีกว่า YOLOv7 ที่เป็นเวอร์ชันก่อนหน้านี้

สำหรับงานวิจัย เราสามารถใช้ YOLOv8 เพื่อตรวจจับวัตถุในภาพหรือวิดีโอได้ และจะสามารถนำมาใช้ในงานวิจัยด้าน Computer Vision เช่น การตรวจจับวัตถุในรูปภาพทางการแพทย์, การตรวจจับวัตถุในรูปภาพทางวิทยาศาสตร์ การตรวจสอบความปลอดภัยในระบบวงจรปิด การตรวจจับวัตถุบนโครงสร้างพื้นฐาน การตรวจจับการเคลื่อนไหวของวัตถุ และอื่นๆ ในสาขา Computer Vision ได้ถูกต้องและมีประสิทธิภาพดี [14]-[16]

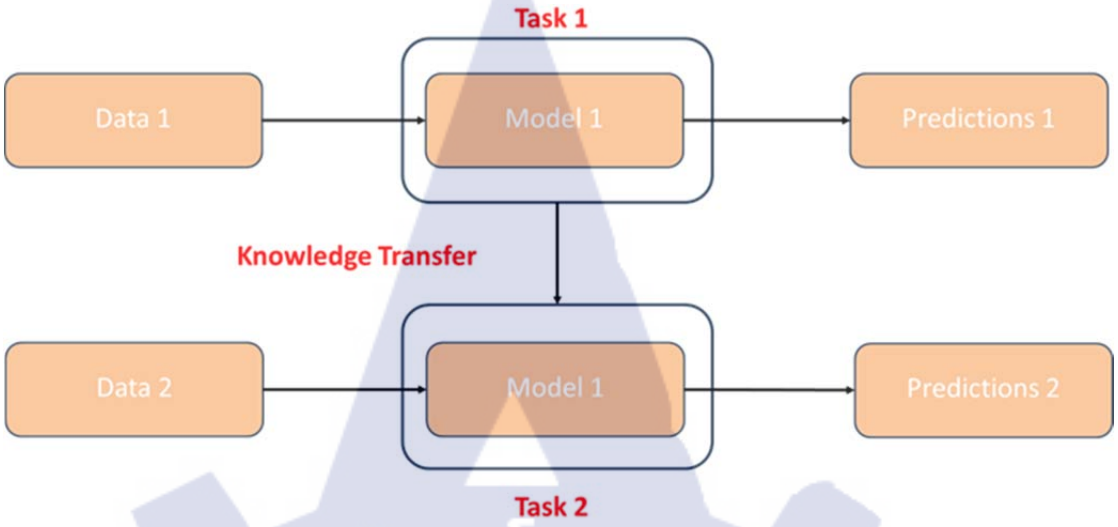
สำหรับ YOLOv8 จะเป็นเวอร์ชันล่าสุดในด้านการทำงานเกี่ยวกับ การตรวจจับวัตถุแบบเรียลไทม์ (Object Detection) การแบ่งหมวดหมู่(Segmentation) หรือ การแยกประเภท(Classification) [17]-[20]



รูปที่ 2.7 Classification, Object Detection, Segmentation [21]

2.9Transfer Learning

Transfer Learning เป็นเทคนิคในของการเรียนรู้เชิงลึก (Deep Learning) ที่นำความรู้ที่ได้จากการฝึกฝนโมเดลในชุดข้อมูลหนึ่ง (Source Domain) มาใช้กับงานในชุดข้อมูลอื่น (Target Domain) ที่มีความเกี่ยวข้องกัน วิธีนี้ช่วยลดเวลาการฝึกโมเดลและทรัพยากรที่จำเป็นโดยเฉพาะในงานที่มีชุดข้อมูลขนาดเล็ก Transfer Learning ใน YOLOv8 จะใช้หลักการเดียวกับการถ่ายโอนความรู้ (Knowledge Transfer) จากโมเดลที่ผ่านการฝึกฝนในชุดข้อมูลขนาดใหญ่ เช่น COCO Dataset ไปยังงานเฉพาะเจาะจง (Domain-Specific Tasks) เช่น การตรวจจับวัตถุในงานทางการแพทย์, การเกษตร หรือในอุตสาหกรรมเฉพาะโดยกระบวนการจะเริ่มต้นจากโมเดลที่ผ่านการฝึกฝน (Pretrained Model) YOLOv8 มีโมเดล Pretrained ที่ถูกฝึกด้วย COCO Dataset ซึ่งครอบคลุมวัตถุ 80 ประเภทโดยโมเดลเหล่านี้เรียนรู้ฟีเจอร์พื้นฐาน เช่น ขอบภาพ รูปทรง และโครงสร้างทั่วไป ผ่านการปรับแต่งโมเดล (Fine-tuning) สำหรับงานใหม่ โมเดลที่ผ่านการฝึกสามารถปรับแต่งด้วยชุดข้อมูลเฉพาะของงานเป้าหมาย เช่น การปรับน้ำหนัก (Weights) ในเลเยอร์หลัง เพื่อปรับให้เหมาะสมกับคลาสใหม่ในชุดข้อมูลเฉพาะ หรือการเพิ่ม/ลดจำนวนคลาส (Classes) ใน YOLOv8 สามารถกำหนดคลาสเฉพาะของงาน เช่น จาก 80 คลาสเป็น 5 คลาส การฝึกฝนเพิ่มเติม ฝึกฝนโมเดลต่อด้วยชุดข้อมูลเป้าหมายเพื่อให้โมเดลสามารถตรวจจับวัตถุหรือทำงานได้อย่างแม่นยำยิ่งขึ้น [22]



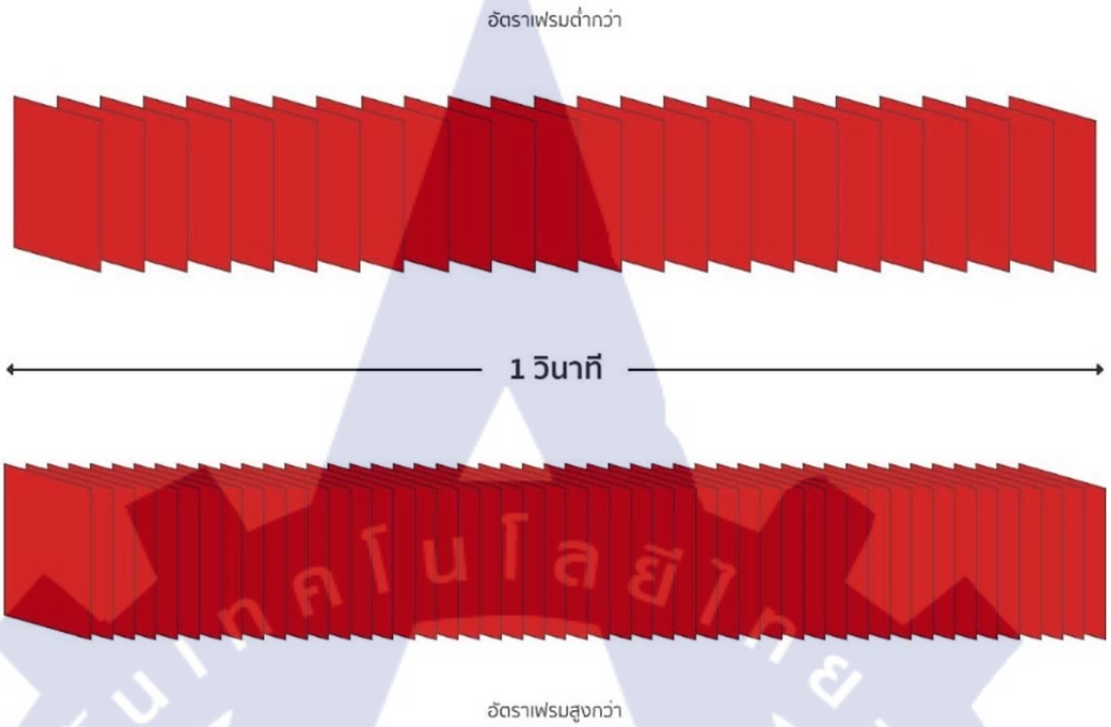
รูปที่ 2.8 Transfer Learning

2.10 COCO Dataset

COCO Dataset (Common Objects in Context) [23] เป็นชุดข้อมูลมาตรฐานที่ใช้ในการฝึกฝนและประเมินผลโมเดลในงาน Computer Vision โดยเฉพาะงานที่เกี่ยวกับการตรวจจับวัตถุ (Object Detection), การแบ่งส่วนภาพ (Image Segmentation), และการแยกประเภทภาพ (Image Classification) ชุดข้อมูลนี้พัฒนาโดย Microsoft และได้รับความนิยมอย่างแพร่หลายในวงการวิจัย เนื่องจากมีความหลากหลายของวัตถุและการกำหนดฉลากที่ครอบคลุม โดยประเภทของข้อมูลใน COCO Dataset มีประมาณ 330,000 ภาพโดยมีทั้งหมดประมาณ 80 คลาสเช่น คน รถ ต้นไม้ เฟอร์นิเจอร์ โดยข้อมูลจะจัดเก็บในรูปแบบ JSON ซึ่งบรรจุข้อมูลเมตาดาต้า (Metadata) เช่น ชื่อไฟล์ ตำแหน่งกรอบวัตถุ และข้อมูลฉลาก โดย YOLO (You Only Look Once) ใช้ COCO Dataset เป็นหนึ่งในชุดข้อมูลหลักในการฝึกโมเดล (Training) และประเมินผล (Evaluation) โดยโมเดล YOLO เช่น YOLOv8 จะใช้ข้อมูลจาก COCO เพื่อเรียนรู้ฟีเจอร์ทั่วไปของวัตถุ เช่น ขอบเขต รูปร่าง และบริบทของวัตถุในภาพ

2.11 การประมวลผลภาพและวิเคราะห์แบบดิจิทัล (Digital Image Processing)

การประยุกต์ใช้งานการประมวลผลสัญญาณและเก็บข้อมูลจะเป็นแบบ 2 มิติ โดยจะเก็บค่าแบบเลขฐาน 2 (Binary) อยู่ในแมทริกซ์ (Matrix) หลังจากนั้นได้พัฒนามาเป็นการใช้ตัวเลขจำนวนจริงอยู่ในรูปแบบ อาร์เรย์ (Array) โดยจะเป็นเทคนิคและอัลกอริทึมที่ใช้ประมวลผลภาพที่อยู่ในรูปแบบดิจิทัล (Digital) หรือ ภาพดิจิทัล (Digital Image) รวมถึงวิดีโอ (Video) ซึ่งจะเป็นชุดของภาพนิ่งหลายๆ เฟรมต่อกันไปตามเวลา



รูปที่ 2.9 Digital Image Processing [24]

ในการวิเคราะห์และการประมวลผลภาพดิจิทัล การทำงานในแต่ละขั้นตอนของงานวิจัยนี้จะเริ่มด้วยการรับภาพเข้ามาในระบบการประมวลผลก่อน การแบ่งส่วนภาพหรือการตรวจจับวัตถุที่ต้องการในภาพ การหาคุณลักษณะเฉพาะของวัตถุ การวิเคราะห์ข้อมูลภาพโดยนำคุณลักษณะเฉพาะมาวิเคราะห์ผ่านตัวจำแนก (Classifier) ในแต่ละขั้นตอนมีรายละเอียดดังนี้

การรับภาพเข้ามาในระบบ การรับภาพเข้ามาในระบบประมวลผลดิจิทัลจะแบ่งออกเป็น 2 ประเภท

1. Bitmap คือ ภาพที่ประกอบขึ้นจากจุดเล็กๆ หรือเรียกว่า พิกเซล (Pixel) โดยจุดเหล่านี้เราไม่สามารถมองเห็นได้ด้วยตาเปล่า

2. Vector คือ ภาพที่เกิดขึ้นโดยการคำนวณทางคณิตศาสตร์มีสีและตำแหน่งที่แน่นอน โดยภาพจะเป็นอิสระต่อกันและเมื่อขยายภาพความละเอียดจะไม่ลดลง

รูปที่ 2.10 Vector and Bitmap [25]

2.11.1 ขั้นตอนก่อนเริ่มกระบวนการ

คือ การนำข้อมูลมาเตรียมเป็นในส่วนของไบนารี (Binary) หรือ ภาพสีเทา (Grey Scale) หรือภาพสี (Color Image)

การแบ่งส่วนภาพ หรือ การตรวจจับวัตถุที่ต้องการในภาพ คือ การแยกวัตถุที่สำคัญหรือที่เราต้องการออกจากภาพพื้นหลัง หลักการของการแยกวัตถุออกจากภาพมี 2 วิธี คือ การหาเส้นขอบ และ เทคนิคเทรชโฮลด์ เนื่องจากมีงานบางประเภทต้องแยกประเภทตัวอักษรสีต่างๆ ออกจากพื้นหลัง ส่วนการแยกส่วนของภาพนั้นไม่ได้มีแค่การหาเส้นขอบ และ เทคนิคเทรชโฮลด์ ยังมีการหาจุดในภาพ หาเส้นในภาพ หรือการหาพื้นที่

การหาเส้นขอบของวัตถุ จะแสดงโครงร่างวัตถุภายในภาพ ซึ่งจะประกอบไปด้วยข้อมูลสำคัญที่ต้องการใช้งาน โดยเส้นขอบจะเกิดจากการเปลี่ยนแปลงความเข้มระดับเทา (Grey Scale) จากต่ำไปสูง หรือ ความไม่ต่อเนื่องของพิกเซล (Pixel) ที่อยู่ติดกัน

รูปที่ 2.11 Threshold Filter

เทรชโฮลด์ [26] เป็นเทคนิคประมวลภาพอย่างง่าย จะใช้ค่าระดับความเข้มเทา (Grey Scale) ค่าหนึ่ง เป็นตัวกำหนดการแยกแยะ ส่วนของภาพเพื่อให้ได้ผลลัพธ์เป็นแบบไบนารี ที่มีค่าความเข้ม 2 ระดับ คือ ขาวและดำ ซึ่งการกำหนดค่าของ เทรชโฮลด์ ถ้ากำหนดไม่เหมาะสมอาจจะทำให้ส่วนของภาพนั้นขาดหาย หรือสิ่งแปลกปลอมประปนอยู่ (Noise)

2.11.2 การหาคุณลักษณะเฉพาะของวัตถุ

จะดูขอบเขตเป็นของวัตถุ ได้แก่ ทฤษฎีเส้นโค้งของวัตถุ (Curvature) ความยาวเส้นรอบวงของวัตถุ (Perimeter) วิธีตรวจจับของโซเบล (Sobel Detection) วิธีตรวจจับของแคนนี่ (Canny Detection) ทฤษฎีการตัดต่อหรือแต่งเติมขอบภาพ (Erosion and Dilation) เป็นต้น และจะดูพื้นที่ของวัตถุ ได้แก่ พื้นที่ของวัตถุ (Area), ค่าความกระชับของวัตถุ (Compactness) และอัตราส่วนของพื้นที่วัตถุ (Aspect Ratio) ในกระบวนจำแนกวัตถุนั้นอาจใช้การตรวจหาคุณลักษณะพื้นฐาน หลายๆ วิธีมาผสมกันเพื่อให้เกิดเป็นคุณลักษณะใหม่เพื่อให้การจำแนกนั้นถูกต้องสูงสุด

ค่าความกระชับ (Compactness) คือ อัตราส่วนของพื้นที่ต่อพื้นที่ของวงกลมเส้นรอบวงเดียว (Perimeter) หรือกล่าวได้อีกอย่างคือความเป็นกลุ่มก้อนของพิกเซลในพื้นที่หนึ่งวงกลม มีสูตรคำนวณดังนี้

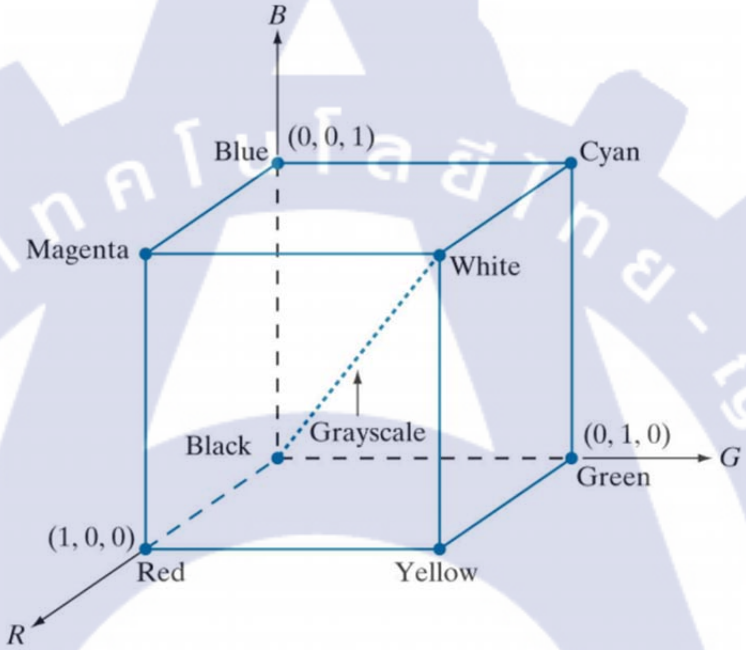
อัตราส่วนของภาพ (Aspect Ratio) คือ อัตราส่วนของภาพที่มีพื้นที่วัตถุ ซึ่งได้กล่าวไว้ข้างต้นว่าภาพดิจิทัลนั้นถูกเก็บอยู่ในรูปแบบของอาร์เรย์สองมิติ กว้าง x สูง หรือ m x n และอัตราส่วนของภาพนั้นแบ่งได้เป็นสามลักษณะ คือ Landscape เป็นภาพลักษณะแนวนอนเป็นภาพลักษณะสี่เหลี่ยมจัตุรัส และ Portrait เป็นภาพลักษณะแนวตั้ง (ดังภาพที่ 2.16 ข.) ดังนั้นการหาอัตราส่วนของภาพจะใช้วิธีเปรียบเทียบระหว่าง m: n เช่น ภาพของวัตถุมีขนาดกว้าง(m) 200 พิกเซล: ขนาดสูง (n) 150 พิกเซล จะได้ค่าอัตราส่วนเป็น 4:3 เป็นต้น

การสกัดโครงร่างวัตถุด้วยวิธีโครงกระดูก (Skeletonization) เป็นเทคนิคการประมวลผลภาพเพื่อลดทอนองค์ประกอบของวัตถุให้เหลือเพียงเส้นแกนกลาง กระบวนการนี้อาศัยเทคนิคที่เรียกว่าการทำให้ภาพบาง (Thinning) ซึ่งเป็นการตรวจสอบพิกเซลบริเวณขอบของวัตถุแบบวนซ้ำ (Iterative Process) เพื่อพิจารณาและกำจัดพิกเซลที่ไม่ส่งผลกระทบต่อความเชื่อมโยงของโครงสร้าง (Topology) และรูปร่างทางเรขาคณิตของวัตถุต้นฉบับ

2.11.3 คุณลักษณะเกี่ยวกับสี

แบบจำลองสีเป็นแบบจำลองที่ใช้กำหนดค่าสีต่างๆ ให้เป็นมาตรฐานเดียวกัน ซึ่งแบบจำลองสีแต่ละแบบนั้นมีคุณสมบัติและความสามารถในการใช้งานแตกต่างกันไปในแบบจำลองของสีแต่ละรูปแบบจะมีแม่สีหลักแตกต่างกันไป และใช้วิธีการผสมผสานให้เกิดสีอื่นๆ ขึ้นมา แบบจำลองสีที่คุ้นเคย

และพบเห็นได้มาก เช่น แบบจำลองสีอาร์จีบี (RGB Model) แบบจำลองอาร์จีบีมีแม่สีหลักสามสี (Primary Color) ได้แก่ สีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) ซึ่งเป็นสีที่เกิดจากการรวมกันของแสง (Additive Color) โดยจุดกำเนิดของแต่ละแม่สีจะมีค่าเป็นศูนย์ถึงหนึ่ง ซึ่งแต่ละตำแหน่งจะแสดงถึงค่าความเข้มของสีของสีแต่ละสี โดยที่ (R,G,B) = (0,0,0) คือค่าของสีดำ และ (R,G,B) = (1,1,1) คือค่าของสีขาว ส่วนการแสดงค่าระดับเทาในแบบจำลองสีอาร์จีบีจะเป็นเส้นทะแยงมุมจากตำแหน่ง (0,0,0) ถึง (1,1,1) เป็นเส้นทแยงมุม

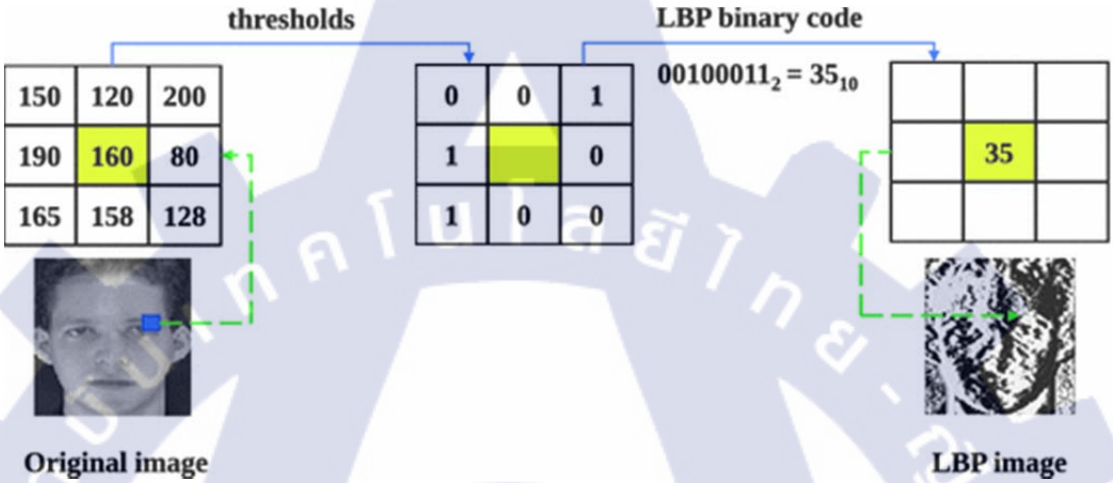


รูปที่ 2.12 RGB

การประมวลภาพสี (Color Image Processing) บนแบบจำลองสี RGB แบบจำลองสี RGB ในข้อมูลภาพนั้นจะประกอบไปด้วยแม่สีหลักสามสีซ้อนกันอยู่ ถ้าเราแยกชั้น (Layer) ของสีแต่ละสีแล้วภาพจะถูกประมวลผลของแต่ละชั้นสีได้ การหาคุณลักษณะเกี่ยวกับพื้นผิวของวัตถุ

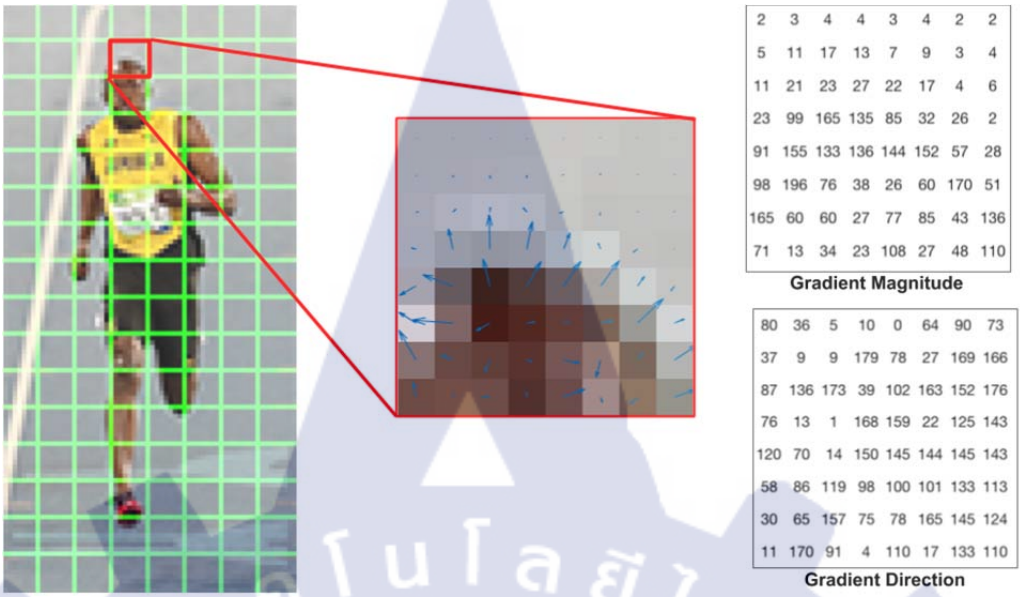
โลคอลไบนารีแพทเทิร์น (Local Binary Pattern: LBP) เป็นเทคนิคที่ใช้ในการวิเคราะห์พื้นผิวของภาพโดยอาศัยความสัมพันธ์ระหว่างค่าความเข้มของพิกเซลใกล้เคียงกัน ภายในพื้นที่ขนาด 3×3 พิกเซล จะเลือกพิกเซลตรงกลางเป็นจุดอ้างอิง แล้วนำค่าความเข้มของพิกเซลโดยรอบทั้ง 8 จุด มาเปรียบเทียบกับพิกเซลศูนย์กลาง หากค่าความเข้มของพิกเซลโดยรอบมากกว่าหรือเท่ากับค่าของพิกเซลศูนย์กลาง จะกำหนดค่าเป็น 1 ในทางกลับกัน หากค่าความเข้มของพิกเซลโดยรอบน้อยกว่าค่าของพิกเซลศูนย์กลาง จะกำหนดค่าเป็น 0 เมื่อได้ผลลัพธ์จากทุกพิกเซลโดยรอบแล้ว จะนำค่าทั้ง 8 บิตที่ได้มาเรียงกันเป็นเลข

ฐานสอง (Binary Pattern) ซึ่งสามารถแปลงเป็นค่าทศนิยมเพื่อให้ง่ายต่อการนำไปใช้งานต่อไป ค่าทศนิยมที่ได้จากทุกพิกเซลในภาพสามารถนำมาสร้าง ฮิสโตแกรม ซึ่งเป็นตัวแทนของลักษณะพื้นผิวของภาพ เทคนิคนี้มีประสิทธิภาพในการดึงคุณลักษณะของพื้นผิวออกมาและสามารถนำไปประยุกต์ใช้ในงานด้านการรู้จำรูปแบบ (Pattern Recognition) และการประมวลผลภาพ (Image Processing) ได้อย่างมีประสิทธิภาพ



รูปที่ 2.13 LBP การหาค่าความถี่ของทิศทางตามค่าเกรเดียนท์ (Histograms of Oriented Gradients: HOG) [27]

เป็นวิธีที่คิดค้นขึ้นมาเพื่อการตรวจจับมนุษย์ ด้วยการที่ดึงลักษณะเฉพาะของภาพในการกระจายตัวอยู่บนพื้นหลัง (Background) โดยที่ใช้การกระจายตัวของความเข้มเกรเดียนท์ หรือทิศทางของเส้นขอบวัตถุ การทำงานของการหาค่าความถี่ของทิศทางตามค่าเกรเดียนท์จะแบ่งภาพออกเป็นในพื้นที่เป็นส่วนเล็กๆ และเรียกพื้นที่นั้นว่า เซลล์ โดยที่แต่ละเซลล์จะรวบรวมฮีสโตแกรมของทิศทางเกรเดียนท์หรือทิศทางของขอบภายในเซลล์ การรวมกันของฮีสโตแกรมนั้นจะแสดงถึงคุณลักษณะของวัตถุ เพื่อเพิ่มประสิทธิภาพของความถูกต้อง สามารถนำเอาฮีสโตแกรมมานอร์มอลไลซ์ ด้วยวิธีการคำนวณตัวชี้วัดค่าความเข้มจากโอเวอร์แลปของเซลล์ภายในบล็อกเพื่อลดผลกระทบการจากการเปลี่ยนแปลงของแสงเงา



รูปที่ 2.14 Histograms of Oriented Gradients [28]

2.12 คอมพิวเตอร์วิทัศน์ (Computer Vision)

เป็นสาขาหนึ่งของการประมวลผลภาพดิจิทัล ซึ่งเน้นการพัฒนาโปรแกรมและอัลกอริทึมเพื่อให้คอมพิวเตอร์สามารถเข้าใจและวิเคราะห์ภาพดิจิทัลได้เหมือนมนุษย์ โดยโดยทั่วไปแล้ว คอมพิวเตอร์จะใช้เทคนิคการประมวลผลภาพ การแยกแยะวัตถุ เช่นการตรวจจับวัตถุ เช่น ใบหน้า รถยนต์ หรือเครื่องบิน การวิเคราะห์ภาพ เช่น การจำแนกวัตถุ การติดตามวัตถุ หรือการค้นหาวัตถุ รวมถึงการปรับปรุงภาพ เช่น การเพิ่มความคมชัด การลบบทบรรยายหรือแต่งภาพ โดยมีการนำไปใช้ในหลายอุตสาหกรรม เช่น การสแกนเอกสาร ระบบกล้องวงจรปิด ระบบช่วยการขับรถ และระบบรักษาความปลอดภัย ฯลฯ

2.13 งานวิจัยที่เกี่ยวข้อง

ตารางที่ 2.1 งานวิจัยที่เกี่ยวข้อง

ชื่อหัวข้องานวิจัย	ปี	วัตถุประสงค์	เทคนิค	ชุดข้อมูล	การวัดประสิทธิภาพ
1. Enhanced Chest CT Detection of Pulmonary Nodules Based on YOLOv8 [29]	2024	พัฒนา YOLOv8 เพื่อปรับปรุงการตรวจจับก้อนเนื้อปอดที่เล็กในภาพ CT	YOLOv8	2 Dataset (LIDC-IDRI, LUNA16), 2 Classes	mAP@0.5: LIDC-IDRI = 76.9%, LUNA16 = 77.5%, Accuracy = 92.3%

ตารางที่ 2.1 งานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่อหัวข้องานวิจัย	ปี	วัตถุประสงค์	เทคนิค	ชุดข้อมูล	การวัดประสิทธิภาพ
2. Bird Species Recognition Using YOLOv8 [30]	2024	จำแนกชนิดนกเพื่ออนุรักษ์ที่อยู่อาศัย	YOLOv8	1 Dataset, 15 Classes	mAP@0.5: 94.3%, Accuracy = 88.7%
3. MI-YOLO: Improved Traffic Sign Detection Based on YOLOv8 [31]	2024	ตรวจจับป้ายจราจรในสภาพแวดล้อมการจราจรที่ซับซ้อน	YOLOv8	1 Dataset (TT100K), 100 Classes	mAP@0.5: 83.8%, Accuracy = 90.1%
4. FGOD-YOLOv8: Fine-Grained Object Detection [32]	2024	พัฒนา YOLOv8 สำหรับวัตถุขนาดเล็กในเกษตรกรรม	YOLOv8	1 Dataset (Crops and Weeds), 4 Classes	mAP@0.5: 87.2%, Accuracy = 91.5%
5. RLGS-YOLO: Metro Station Passenger Detection [33]	2024	ตรวจจับผู้โดยสารในสถานีรถไฟใต้ดิน	YOLOv8	1 Dataset (Metro Passenger), 3 Classes	mAP@0.5: 89.4%, Accuracy = 93.0%

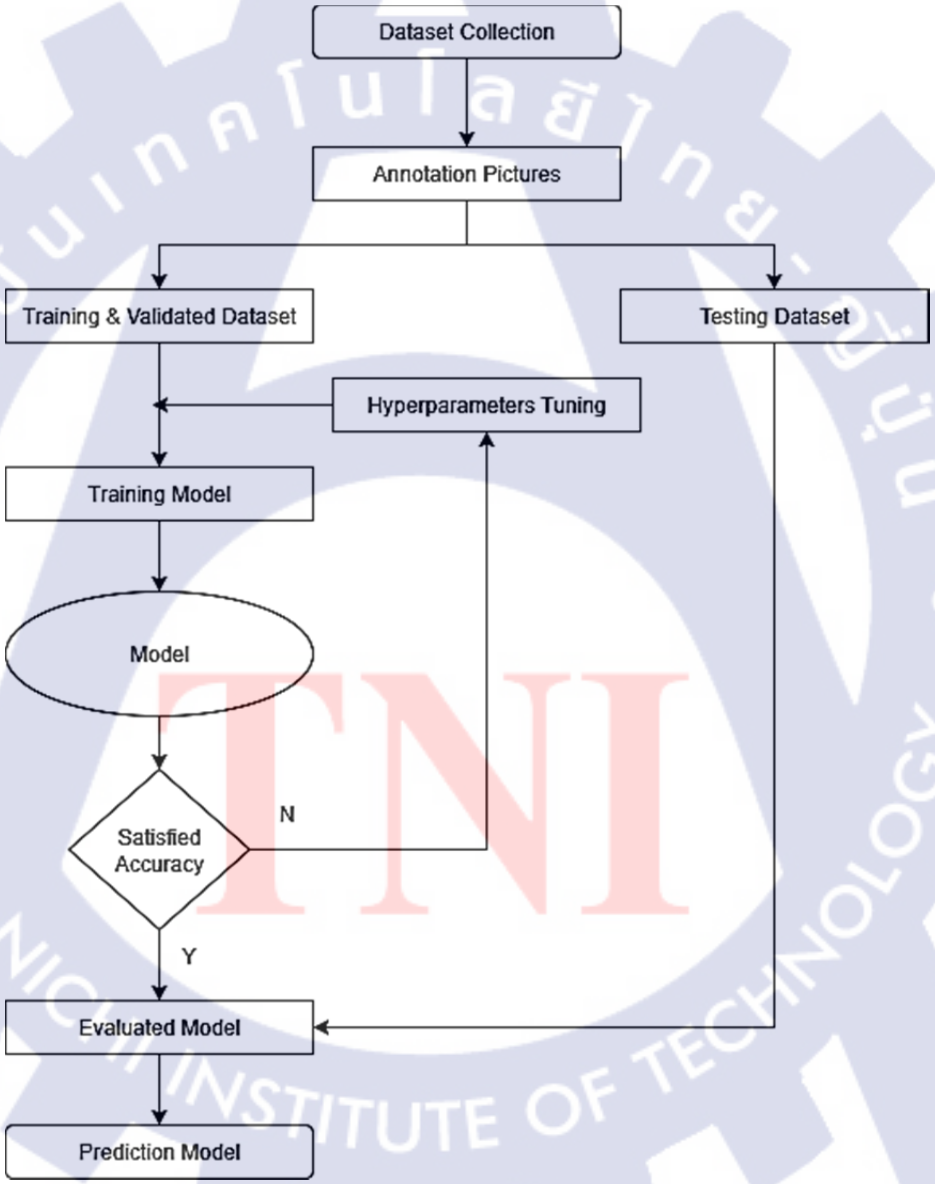
2.14 ชุดข้อมูล (Dataset)

ชุดข้อมูล คือกลุ่มของข้อมูลที่เกี่ยวข้องกันและถูกเก็บรวบรวมมาเพื่อใช้ในการสร้างและทดสอบโมเดลและอัลกอริทึมต่างๆ ในการเรียนรู้ของเครื่องจักร (Machine Learning) และปัญญาประดิษฐ์ (Artificial Intelligence) โดย Dataset จะประกอบไปด้วยข้อมูลต่างๆ เช่น ภาพดิจิทัล วิดีโอ ข้อความ และอื่นๆ โดย Dataset ที่มีขนาดใหญ่และครอบคลุมด้านคุณภาพและความหลากหลายจะช่วยให้โมเดลและอัลกอริทึมในการเรียนรู้ได้มีประสิทธิภาพและมีความแม่นยำในการทำนายข้อมูลได้ดียิ่งขึ้น [34] โดย Dataset สามารถเป็นได้ทั้งข้อมูลที่เก็บรวบรวมมาจากแหล่งต่างๆ หรือเป็นข้อมูลที่สร้างขึ้นโดยเฉพาะสำหรับการทดสอบโมเดลและอัลกอริทึม ในบางกรณี Dataset อาจเป็นส่วนสำคัญที่ช่วยให้โมเดลและอัลกอริทึมมีความแม่นยำและได้ผลลัพธ์ที่ดี

บทที่ 3
วิธีดำเนินงานวิจัย

จากการศึกษาทฤษฎีและงานวิจัย ที่เกี่ยวข้องกับเรื่องการพัฒนาการจำแนกเกรดรังนกแอ่นกินรังด้วย YOLOv8n ผู้ดำเนินงานวิจัยได้นำความรู้ที่ได้รับมาดำเนินตามขั้นตอนดังนี้

3.1 แผนผังการทำงาน



รูปที่ 3.1 แผนผังการทำงาน

3.2 รวบรวมและเตรียมข้อมูล

ในงานวิจัยนี้ การทดลองได้ใช้ชุดข้อมูลภาพถ่ายที่เก็บรวบรวมมาทั้งหมดโดยแบ่งเป็นคลาสหรือประเภทของรังนกได้ 3 คลาสได้แก่ เกรดเอ เกรดบี และรังมุม โดยสรุปชุดข้อมูลอย่างย่อตามตารางที่ 3.1

ตารางที่ 3.1 ชุดข้อมูลที่ใช้สำหรับทดลองจำแนกประเภทรังนก

Dataset	Training	Validating	Testing	Total
เกรดเอ	297	85	42	424
เกรดบี	249	71	36	355
รังมุม	148	42	21	148

3.3 การกำหนดคำอธิบายรูปภาพสำหรับสร้างชุดข้อมูลฝึก (Annotation Pictures)

หลังจากรวบรวมข้อมูลแล้ว ขั้นตอนถัดมาคือการทำ Annotation หรือการกำหนดกรอบวัตถุในแต่ละภาพ พร้อมกับระบุชื่อคลาสที่สอดคล้องกัน โดยในงานวิจัยนี้ใช้เครื่องมือที่ชื่อว่า Labellmg ในการกำหนดกรอบของวัตถุหรือรังนกในภาพที่นำมาใช้สำหรับฝึกตัวโมเดล ความแม่นยำและความถูกต้องของการทำ Annotation มีความสำคัญอย่างมาก เพราะส่งผลโดยตรงต่อประสิทธิภาพของโมเดล

รูปที่ 3.2 การทำ Annotation

3.4 การแบ่งชุดข้อมูลการฝึก การตรวจสอบความถูกต้อง และการทดสอบ

ในการสร้างแบบจำลองการตรวจจับวัตถุ ข้อมูลจะถูกแบ่งออกเป็นสามชุดย่อย ได้แก่ ชุดข้อมูลการฝึก (Training Set) ชุดข้อมูลตรวจสอบความถูกต้อง (Validation Set) และชุดข้อมูลทดสอบ (Test Set) ชุดข้อมูลการฝึกเป็นชุดข้อมูลที่ใช้ในการฝึกแบบจำลอง ในแต่ละ Epoch แบบจำลองจะถูกฝึกซ้ำๆ กับข้อมูลในชุดการฝึกนี้

ชุดข้อมูลตรวจสอบความถูกต้องเป็นชุดข้อมูลที่แยกจากชุดข้อมูลฝึก ใช้สำหรับตรวจสอบความถูกต้องของแบบจำลองในระหว่างการฝึก โดยขั้นตอนการตรวจสอบนี้ให้ข้อมูลที่ใช้ในการปรับค่าพารามิเตอร์ต่างๆ (Hyperparameters) รายละเอียดการตั้งค่าพารามิเตอร์เฉพาะที่ใช้แสดงในตารางที่ 3.2

ตารางที่ 3.2 การตั้งค่าไฮเปอร์พารามิเตอร์ที่สำคัญที่ใช้สำหรับการฝึกโมเดลในการจำแนก รวมถึงอัลกอริธึมการปรับค่า (Optimizer) และคุณสมบัติของข้อมูลอินพุตที่จำเป็น (Hyperparameters Configuration)

Hyperparameters	
Parameters	**Value**
Optimizer	Adam
Color	RGB
Input Image size	640*640
Epochs	100
Batch size	16

จากตารางที่ 3.2 แสดงการกำหนดค่าตัวแปรไฮเปอร์พารามิเตอร์ต่างๆ ที่จำเป็นสำหรับการฝึกตัวจำแนกด้วยวิธีการเรียนรู้เชิงลึกซึ่งประกอบไป ด้วยตัวปรับค่าหรือ Optimizer กำหนดเป็น Adam ถูกนำมาใช้เนื่องจากมีคุณสมบัติในการปรับอัตราการเรียนรู้อัตโนมัติ ซึ่งช่วยเพิ่มประสิทธิภาพการลู่เข้าสู่ผลลัพธ์ ภาพอินพุตถูกปรับขนาดเป็น 640 × 640 พิกเซลและแปลงเป็นโหมดสี RGB เพื่อให้ข้อมูลมีความสม่ำเสมอและช่วยให้โมเดลสามารถประมวลผลคุณสมบัติสีได้อย่างมีประสิทธิภาพ โมเดลได้รับการฝึกทั้งหมด 100 รอบ (Epochs) โดยมี Batch Size เท่ากับ 16 เพื่อสร้างความสมดุลระหว่างประสิทธิภาพการคำนวณและความสามารถในการจำแนกที่หลากหลายของโมเดล

ในการศึกษาครั้งนี้ ชุดข้อมูลทั้งหมดจะถูกแบ่งออกเป็นชุดฝึก ชุดตรวจสอบความถูกต้อง และชุดทดสอบในอัตราส่วน 70:20:10

3.5 ทดสอบโมเดลการตรวจจับวัตถุ

เมื่อทำการสร้างโมเดลเสร็จแล้ว เราจำเป็นต้องนำโมเดลการตรวจจับวัตถุที่ได้มาทำการทดสอบกับชุดข้อมูลทดสอบ เพื่อประเมินผลความแม่นยำของโมเดลการตรวจจับวัตถุด้วยการวัด Precision, Recall, F1-score หรืออื่น ๆ ตามที่ต้องการ หลังจากนั้นปรับแต่งโมเดลการตรวจจับวัตถุและทำการเทรนโมเดลใหม่ตามความเหมาะสมเพื่อเพิ่มความแม่นยำยิ่งขึ้น

3.5.1 การวัดประสิทธิภาพโมเดลการตรวจจับวัตถุ

ในการตรวจจับวัตถุ (Object Detection) การประเมินประสิทธิภาพของโมเดลมีความสำคัญอย่างมาก Precision, Recall และ F1 Score เป็นตัวชี้วัดพื้นฐานที่ถูกใช้ในการวัดความถูกต้องและความน่าเชื่อถือของโมเดล โดยเฉพาะในกรณีที่ต้องคำนึงถึงความสมดุลระหว่างการตรวจจับวัตถุอย่างถูกต้องและการลดข้อผิดพลาดจากการตรวจจับข้อผิดพลาด

3.5.2 Precision

คือสัดส่วนของวัตถุที่โมเดลตรวจจับได้ถูกต้อง (True Positives) ต่อวัตถุที่โมเดลตรวจจับได้ทั้งหมด (True Positives + False Positives) ค่านี้ตอบคำถามว่า “เมื่อโมเดลทำนายวัตถุขึ้นมา โมเดลทำนายได้ถูกต้องบ่อยแค่ไหน?” โดยมีสมการในการคำนวณค่า Precision ดังนี้

$$\text{Precision} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Positive (FP)}}$$

ค่า Precision ที่สูงแสดงว่าโมเดลสามารถหลีกเลี่ยง False Positives (การตรวจจับข้อผิดพลาด) ได้ดี ค่านี้มีความสำคัญในกรณีที่ False Positives สร้างผลกระทบที่ร้ายแรง เช่น การตรวจจับโรคในภาพถ่ายทางการแพทย์ หรือระบบรักษาความปลอดภัย

3.5.3 Recall

Recall คือสัดส่วนของวัตถุที่โมเดลตรวจจับได้ถูกต้อง (True Positives) ต่อวัตถุทั้งหมดที่มีอยู่จริงในชุดข้อมูล (True Positives + False Negatives) ค่านี้ตอบคำถามว่า “โมเดลสามารถตรวจจับวัตถุทั้งหมดในข้อมูลได้มากแค่ไหน” โดยเราสามารถคำนวณค่า Recall ได้จากสมการดังต่อไปนี้

$$\text{Recall} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Negative (FN)}}$$

ค่า Recall ที่สูงแสดงว่าโมเดลสามารถตรวจจับวัตถุทั้งหมดได้ดี โดยลด False Negatives (การตรวจจับที่ผิดพลาด) ให้น้อยที่สุดค่านี้มีความสำคัญในสถานการณ์ที่การพลาดการตรวจจับวัตถุเป็นเรื่องร้ายแรง เช่น การตรวจจับคนเดินเท้าในรถยนต์ไร้คนขับ

3.5.4 Mean Average Precision (mAP)

ค่าที่ใช้วัดความสามารถของโมเดลในการตรวจจับวัตถุ โดยคำนวณจากค่า Average Precision (AP) ของแต่ละคลาส แล้วนำมาหาค่าเฉลี่ย โดยเมื่อโมเดลทำนายวัตถุในภาพโมเดลสามารถทำนายได้แม่นยำแค่ไหน โดยสูตรคำนวณค่า mAP คือ

$$\text{mAP} = \left(\frac{1}{N}\right)\sum_{i=1} AP_i$$

N = จำนวนคลาสทั้งหมด

AP_i = ค่า Average Precision ของคลาส i

ค่า mAP ที่สูงบ่งบอกว่าโมเดลสามารถตรวจจับวัตถุได้แม่นยำและเสถียร หากค่า mAP ต่ำ อาจแสดงว่าโมเดลยังมีปัญหาในการจำแนกวัตถุ หรือมี False Positives และ False Negatives ในระดับสูง ซึ่งอาจส่งผลต่อความน่าเชื่อถือของโมเดล โดยเฉพาะในงานที่ต้องการความแม่นยำสูง

3.5.5 F1 Score

F1 Score คือค่าเฉลี่ยฮาร์โมนิก (Harmonic Mean) ระหว่างค่า Precision และค่า Recall สาเหตุที่ต้องใช้ค่าเฉลี่ยฮาร์โมนิกแทนค่าเฉลี่ยเลขคณิต (Arithmetic Mean) เนื่องจากในงานตรวจจับวัตถุ หากโมเดลมีค่า Precision สูงมากแต่ค่า Recall ต่ำมาก หรือในทางกลับกัน ค่า Precision ต่ำมากแต่ค่า Recall สูงมาก ค่าเฉลี่ยเลขคณิตอาจจะยังดูสูงเกินความเป็นจริง แต่ค่าเฉลี่ยฮาร์โมนิกจะดึงค่า F1 Score ให้ต่ำลงอย่างมีนัยสำคัญเพื่อเตือนให้ผู้ดำเนินงานวิจัยว่าโมเดลยังไม่มีประสิทธิภาพเพียงพอในด้านใดด้านหนึ่ง

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

3.5.6 Intersection over Union (IoU)

Intersection over Union (IoU) เป็นพื้นฐานในการตัดสินว่าการตรวจจับนั้น ถูกต้องหรือไม่ค่า IoU คืออัตราส่วนของพื้นที่ทับซ้อน (Overlap) ต่อพื้นที่รวม (Union) ของกรอบทำนายและกรอบจริง

$$IOU = \frac{Intersection\ Area}{Union\ Area}$$

บทที่ 4
ผลการวิเคราะห์ข้อมูล

การทดลองนี้ใช้โมเดล YOLOv8n ซึ่งเป็นโมเดลตรวจจับวัตถุ โดยได้ทำการฝึกและทดสอบบนชุดข้อมูลที่ได้รับการปรับแต่งให้เหมาะสม ซึ่งมี 3 คลาส ได้แก่ เกรดเอ, เกรดบี, และรังมุม ชุดข้อมูลถูกแบ่งเป็นชุดสำหรับการฝึก (70%) ชุดสำหรับการตรวจสอบความถูกต้อง (20%) และชุดสำหรับทดสอบ (10%) เพื่อให้มีการกระจายตัวของข้อมูลแต่ละคลาสอย่างสมดุล นอกจากนี้ โมเดลยังใช้ Pre-trained Weights เพื่อเร่งการเรียนรู้และเพิ่มความแม่นยำในการตรวจจับวัตถุ

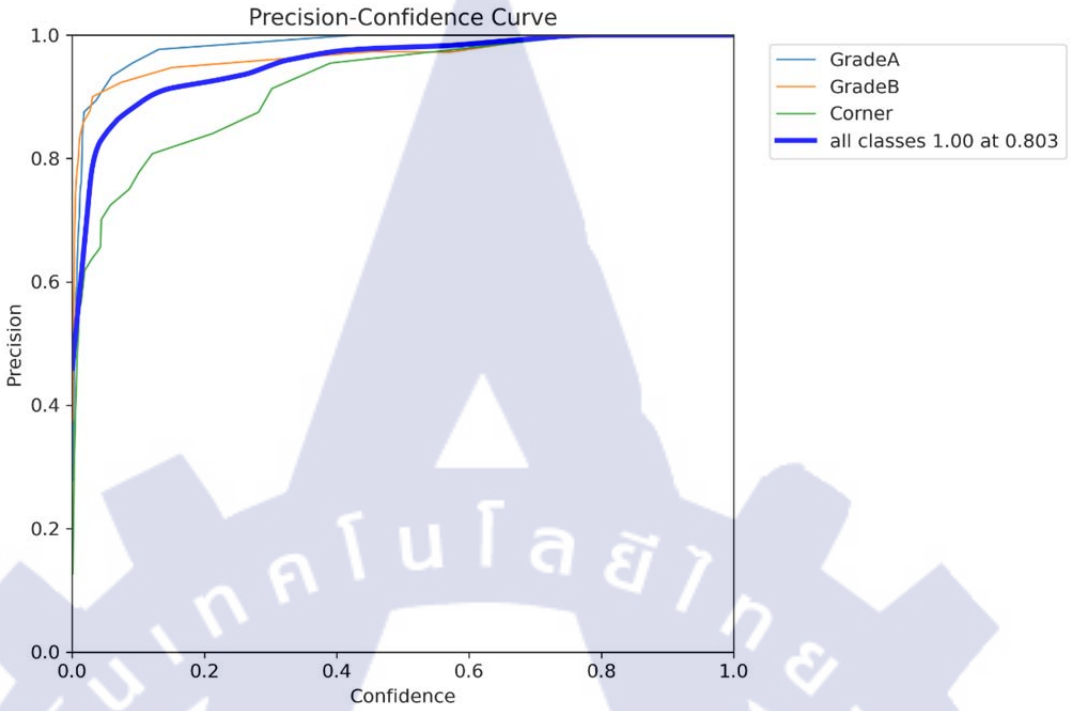
4.1 Experimental Environment

ในการทดลองครั้งนี้ ใช้โมเดล YOLO เวอร์ชัน 8 ร่วมกับภาษาโปรแกรม Python เวอร์ชัน 3.10.12 ภายใต้เฟรมเวิร์ก PyTorch เวอร์ชัน 2.1.0 และใช้งาน CUDA เวอร์ชัน 12.1 เพื่อรองรับการประมวลผลแบบขนานบนหน่วยประมวลผลกราฟิก (GPU) รุ่น NVIDIA A100-SXM4 ขนาดหน่วยความจำ 40 GB

ระบบปฏิบัติการที่ใช้คือ Ubuntu 22.04.3 LTS (Jammy Jellyfish) ภายใต้สภาพแวดล้อม Google Colab Pro+ ซึ่งให้ทรัพยากรการประมวลผลเป็นหน่วยประมวลผลกลาง (CPU) Intel Xeon ความเร็ว 2.00 GHz แบบ 8 คอร์ หน่วยความจำหลัก (RAM) ขนาด 51 GB และพื้นที่เก็บข้อมูล 202 GB รายละเอียดของสภาพแวดล้อมที่ใช้ในการฝึกแบบจำลองสรุปไว้ในตารางที่ 4.1

ตารางที่ 4.1 Experimental Environment

Cloud-hosted	Google Colab Pro+
Algorithm	YOLO Version 8n
CPU	Intel (R) Xeon (R) CPU @ 2.00GHz (8-core)
GPU	NVIDIA A100-SXM4-40GB GPU with 40514 MB of VRAM
RAM	51 GB
Data Storage	202 GB
Operating System	Linux Ubuntu 22.04.2 LTS (Jammy Jellyfish)
Software Environment	Python 3.10.12, Torch 2.1.0, CUDA 12.1



รูปที่ 4.1 Precision-Confidence Curve

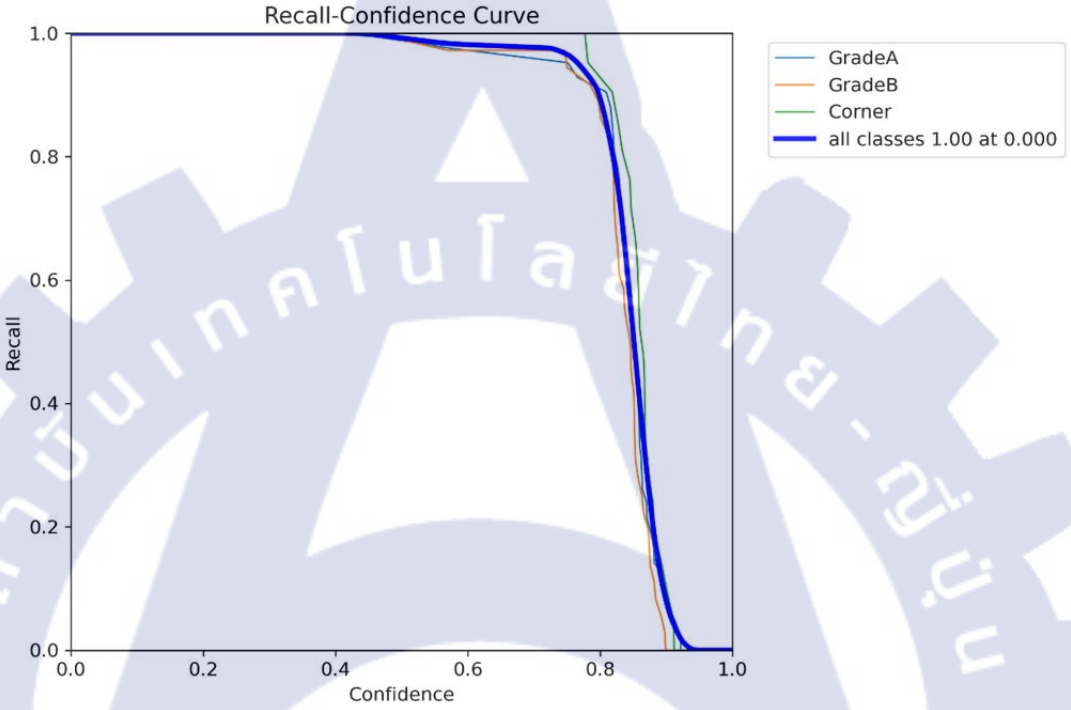
4.2 Precision-Confidence Curve

รูปที่ 4.1 แสดงกราฟ Precision-Confidence curve ซึ่งใช้ในการวิเคราะห์ความสัมพันธ์ระหว่าง Precision และระดับ Confidence ของแบบจำลองในแต่ละคลาส ได้แก่ เกรดเอ, เกรดบี และรังมุม รวมถึงค่าเฉลี่ยของทุกคลาส

จากผลการวิเคราะห์ พบว่าแบบจำลองสามารถทำนายคลาส เกรดเอ และเกรดบี ได้อย่างมีประสิทธิภาพสูง โดยมีค่า Precision มากกว่า 0.9 ในช่วง Confidence เกือบทั้งหมด ขณะที่คลาส รังมุม มีค่า Precision ต่ำกว่าอย่างชัดเจน โดยเฉพาะในช่วง Confidence ที่ต่ำกว่า 0.2 ซึ่งสะท้อนถึงข้อจำกัดของแบบจำลองในการจำแนกตัวอย่างที่อยู่ในคลาสดังกล่าว

เมื่อพิจารณาผลรวมทุกคลาส พบว่าเส้นโค้งเฉลี่ย (สีน้ำเงินเข้ม) แสดงให้เห็นว่าแบบจำลองมีค่า Precision เฉลี่ยสูงกว่า 0.8 ตั้งแต่ช่วง Confidence มากกว่า 0.3 ขึ้นไป และที่ค่า Confidence เท่ากับ 0.803 แบบจำลองสามารถทำให้ค่า Precision เท่ากับ 1.00 ได้ นั่นหมายความว่า หากกำหนด Threshold ที่ระดับความมั่นใจดังกล่าว โมเดลจะสามารถทำนายได้โดยปราศจากความผิดพลาด อย่างไรก็ตาม การเลือก Threshold ที่สูงเกินไปอาจส่งผลให้ค่า Recall ลดลง เนื่องจากโมเดลจะยอมรับเฉพาะตัวอย่างที่มี Confidence สูงมากเท่านั้น

สรุปได้ว่า แบบจำลองมีประสิทธิภาพสูงในการจำแนกตัวอย่างในคลาส เกรดเอ และเกรดบี ในขณะที่คลาส รังมุม ยังต้องการการปรับปรุงเพิ่มเติม ทั้งนี้ การเลือก Threshold ที่เหมาะสมควรพิจารณาสมดุลระหว่าง Precision และ Recall โดยในการใช้งานจริง อาจเลือก Threshold ที่ช่วง 0.4-0.6 เพื่อคงประสิทธิภาพของการทำนายไว้ในระดับสูง



รูปที่ 4.2 Recall-Confidence Curve

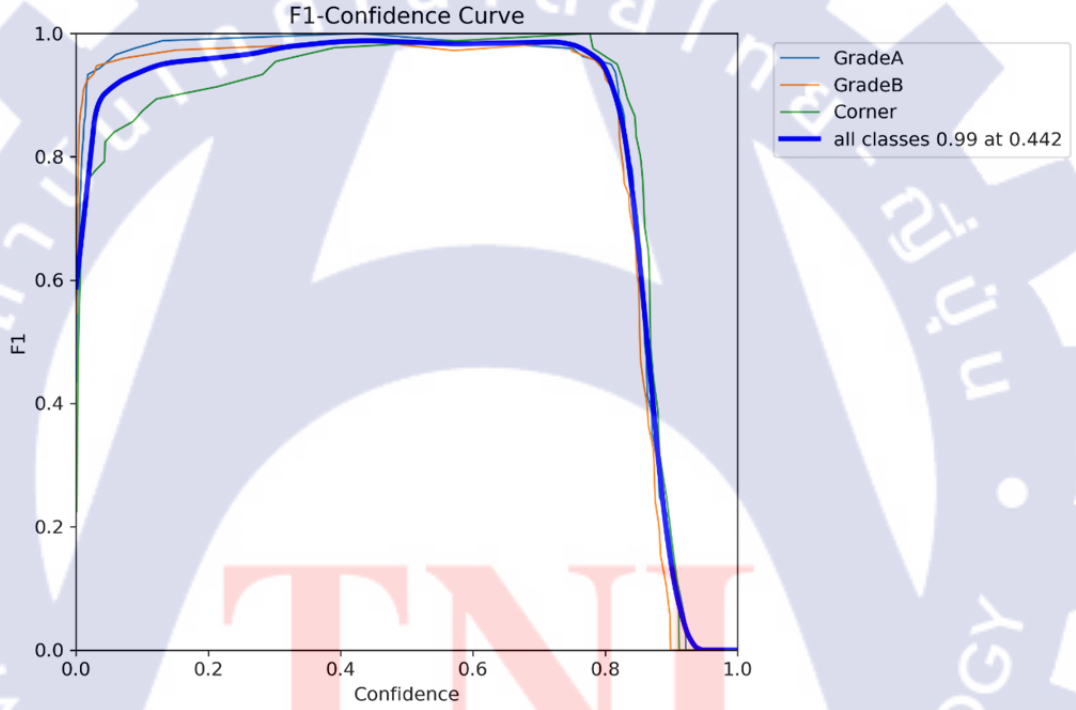
4.3 Recall-Confidence Curve

รูปที่ 4.2 แสดงกราฟ Recall-Confidence Curve ซึ่งใช้ในการวิเคราะห์ความสัมพันธ์ระหว่าง Recall และระดับ Confidence ของแบบจำลองในแต่ละคลาส ได้แก่ เกรดเอ, เกรดบี และรังมุม รวมถึงค่าเฉลี่ยของทุกคลาส

จากผลการวิเคราะห์ พบว่าแบบจำลองสามารถรักษาค่า Recall ให้อยู่ในระดับสูง (ใกล้ 1.0) ได้เมื่อ Confidence อยู่ในช่วง 0.0-0.7 โดยเส้นโค้งของทุกคลาสเกาะกลุ่มใกล้เคียงกัน ซึ่งสะท้อนว่าแบบจำลองมีความสามารถในการครอบคลุมตัวอย่าง ได้ดีในช่วงค่า Confidence ที่ไม่สูงมากนัก อย่างไรก็ตาม เมื่อค่า Confidence เกินกว่า 0.8 ค่า Recall ลดลงอย่างรวดเร็วในทุกคลาส โดยเฉพาะเมื่อค่า Confidence ใกล้ 1.0 ที่ค่า Recall ลดลงเหลือใกล้ศูนย์

เมื่อพิจารณาผลรวมทุกคลาส (เส้นสีน้ำเงินเข้ม) พบว่าแบบจำลองมีค่า Recall เท่ากับ 1.00 ที่ Confidence เท่ากับ 0.000 ซึ่งหมายความว่าหากยอมรับการทำนายทั้งหมดโดยไม่ตั้ง Threshold โมเดลจะสามารถตรวจจับตัวอย่างได้ครบทุกตัว อย่างไรก็ตาม ในการใช้งานจริง การเลือก Threshold ที่ต่ำเกินไปจะทำให้เกิดความเสี่ยงที่ค่า Precision ลดลง เพราะโมเดลอาจยอมรับการทำนายที่ไม่แม่นยำเข้ามามากเกินไป

แบบจำลองมีความสามารถในการรักษาค่า Recall ให้อยู่ในระดับสูงเมื่อใช้ Confidence ต่ำ แต่จะสูญเสียความครอบคลุมอย่างมากหากเพิ่ม Threshold เกินกว่า 0.8 ดังนั้น การเลือก Threshold ที่เหมาะสมควรพิจารณาร่วมกับผลการวิเคราะห์จาก Precision-Confidence Curve เพื่อสร้างสมดุลระหว่าง Precision และ Recall ให้เหมาะสมกับลักษณะการใช้งานจริง



รูปที่ 4.3 F1-Confidence Curve

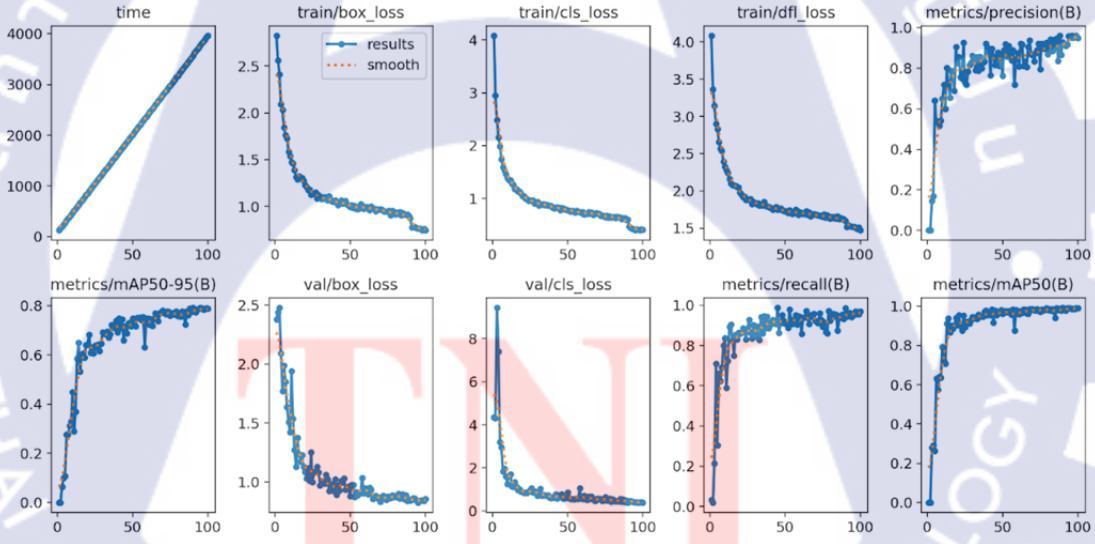
4.4 F1-Confidence Curve

รูปที่ 4.3 แสดงกราฟ F1–Confidence Curve ซึ่งใช้ในการวิเคราะห์ความสัมพันธ์ระหว่างค่า F1-score และระดับ Confidence ของแบบจำลองในแต่ละคลาส ได้แก่ เกรดเอ, เกรดบี และรังมุม รวมถึงค่าเฉลี่ยของทุกคลาส

จากผลการวิเคราะห์ พบว่าแบบจำลองสามารถรักษาค่า F1-score ให้อยู่ในระดับสูง (มากกว่า 0.9) ได้ในช่วง Confidence ที่กว้าง ตั้งแต่ประมาณ 0.2-0.8 ซึ่งสะท้อนว่าแบบจำลองมีความสมดุลที่ดีระหว่าง Precision และ Recall ในช่วงดังกล่าว โดยเฉพาะที่ค่า Confidence เท่ากับ 0.442 แบบจำลองสามารถทำให้ค่า F1-score เฉลี่ยของทุกคลาส เท่ากับ 0.99 ได้ ซึ่งถือว่าเป็นผลลัพธ์ที่มีประสิทธิภาพสูงมาก

เมื่อพิจารณาในเชิงคลาส พบว่า เกรดเอ และ เกรดบี มีค่า F1-score ที่สูงและคงที่มากกว่ารังมุม ในช่วง Confidence ต่ำ (<0.2) อย่างไรก็ตาม เมื่อ Confidence อยู่ในช่วง 0.3-0.7 ค่า F1-score ของคลาสรังมุมเพิ่มขึ้นจนใกล้เคียงกับสองคลาสแรก และมีแนวโน้มลดลงอย่างรวดเร็วเมื่อ Confidence เกินกว่า 0.8

กราฟ F1-Confidence Curve แสดงให้เห็นถึงประสิทธิภาพโดยรวมของแบบจำลองที่สามารถรักษาความสมดุลระหว่าง Precision และ Recall ได้ดี โดย Threshold ที่เหมาะสมอาจอยู่ในช่วงประมาณ 0.4-0.6 ซึ่งช่วยให้ได้ค่า F1-score ที่สูงและคงที่ พร้อมกับรองรับการประยุกต์ใช้งานจริงที่ต้องการทั้งความแม่นยำและความครอบคลุม



รูปที่ 4.4 ข้อมูลการวัดผลที่สำคัญระหว่างการฝึกและการทดสอบโมเดล YOLOv8n สำหรับงานจำแนกรังนก

4.5 การวัดประสิทธิภาพของโมเดล

รูปที่ 4.4 แสดงข้อมูลการวัดผลที่สำคัญระหว่างการฝึกและการทดสอบโมเดล YOLOv8n สำหรับงานจำแนกรังนก โดยใช้จำนวนรอบการฝึก (Epoch) เป็นแกน X และตัวชี้วัดด้านประสิทธิภาพ (Metrics) และค่าความสูญเสีย (Loss) เป็นแกน Y ซึ่งสามารถอธิบายได้ดังนี้

ผลการฝึกแสดงให้เห็นว่า ค่า train/box_loss, train/cls_loss, และ train/dfl_loss มีแนวโน้มลดลงอย่างต่อเนื่องตลอดกระบวนการฝึก เช่นเดียวกับค่า val/box_loss และ val/cls_loss ที่ลดลงอย่างชัดเจนในชุดข้อมูล Validation สะท้อนว่าแบบจำลองมีการเรียนรู้ที่มีเสถียรภาพและสามารถ Generalize ไปยังข้อมูลที่ไม่เคยเห็นมาก่อนได้โดยไม่เกิด Overfitting อย่างมีนัยสำคัญ

ในขณะที่ตัวชี้วัดด้านประสิทธิภาพ เช่น Precision, Recall และ mAP มีแนวโน้มเพิ่มขึ้นอย่างต่อเนื่อง โดยค่า Precision และ Recall ของแบบจำลองสามารถรักษาระดับให้อยู่ใกล้ 0.9 ได้ หลังจากจำนวนรอบการฝึกมากกว่า 50 epoch ขึ้นไป ส่วนค่า mAP50 มีค่าเกือบเท่ากับ 1.0 และค่า mAP50-95 อยู่ในช่วงประมาณ 0.75–0.8 ซึ่งถือว่าเป็นระดับที่สูงและแสดงถึงคุณภาพโดยรวมของการตรวจจับวัตถุได้อย่างแม่นยำ

เมื่อพิจารณาร่วมกับผลจากกราฟ Precision-Confidence Curve, Recall-Confidence Curve และ F1-Confidence Curve พบว่าแบบจำลองสามารถรักษาสมดุลระหว่าง Precision และ Recall ได้ดี โดยเฉพาะเมื่อกำหนด Threshold ของ Confidence ในช่วงประมาณ 0.4-0.6 ซึ่งช่วยให้ได้ค่า F1-score ที่สูงและคงที่

ตารางที่ 4.2 Yolov8n Result

Yolov8n				
Classes	**Precision**	**Recall**	**mAP50**	**mAP50-95**
GradeA	1	0.998	0.995	0.786
GradeB	0.972	1	0.994	0.730
Corner	0.961	1	0.995	0.819
All	0.978	0.999	0.995	0.779

จากตารางที่ 4.2 แสดงผลการประเมินแบบจำลอง Yolov8n โดยใช้ตัวชี้วัดด้านประสิทธิภาพ ได้แก่ Precision, Recall, mAP50 และ mAP50-95 สำหรับแต่ละคลาส (เกรดเอ, เกรดบี และรังมุม) รวมถึงค่าเฉลี่ยทั้งหมด

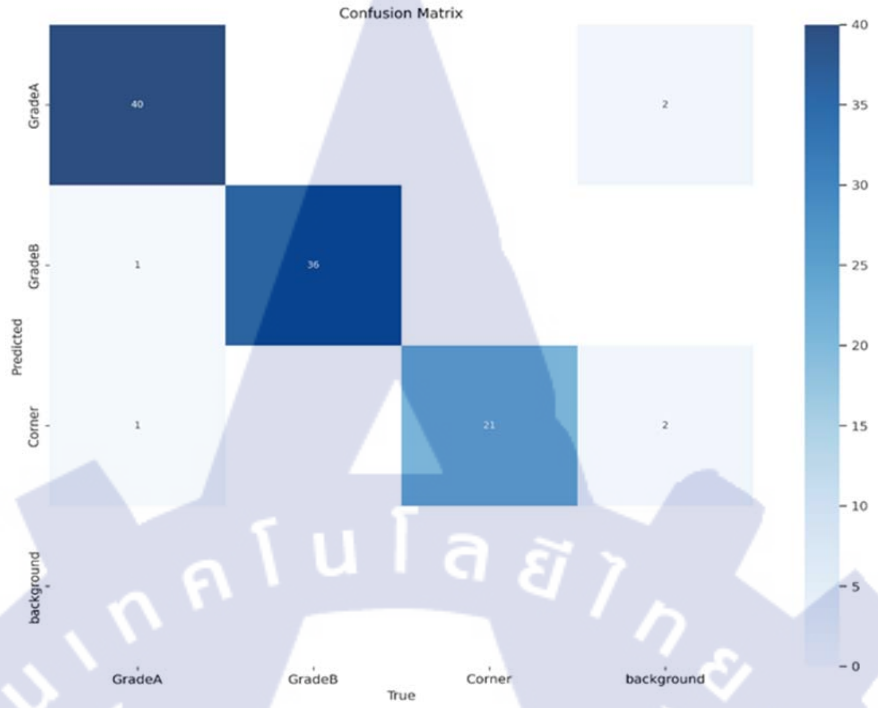
ผลการประเมินพบว่า คลาส เกรดเอ มีค่า Precision เท่ากับ 1.000 และค่า Recall เท่ากับ 0.998 ซึ่งแสดงให้เห็นถึงความสามารถของแบบจำลองในการทำนายได้อย่างถูกต้องแทบทั้งหมด ขณะที่คลาส เกรดบี มีค่า Precision เท่ากับ 0.972 และค่า Recall เท่ากับ 1.000 สะท้อนว่าแบบจำลองสามารถตรวจจับตัวอย่างได้ครบถ้วน (Recall สูงสุด) แม้ว่าจะมีความถูกต้อง (Precision) ต่ำกว่าคลาสเกรดเอเล็กน้อย สำหรับคลาส รังมุม แบบจำลองมีค่า Precision เท่ากับ 0.961 และค่า Recall เท่ากับ 1.000 ซึ่งแม้จะมี Precision ต่ำกว่าคลาสอื่น แต่ยังคงมีประสิทธิภาพสูงในการครอบคลุมการตรวจจับ

ในเชิง mAP พบว่า ค่า mAP50 ของทุกคลาสอยู่ที่ประมาณ 0.994-0.995 แสดงว่าแบบจำลองสามารถตรวจจับวัตถุได้แม่นยำมากเมื่อใช้เกณฑ์ IoU 0.5 ส่วนค่า mAP50–95 มีค่าอยู่ระหว่าง 0.730-0.819 โดยคลาส รังมุม มีค่า mAP50-95 สูงสุด (0.819) ขณะที่คลาส เกรดบี มีค่าต่ำที่สุด (0.730) สะท้อนว่าเมื่อใช้เกณฑ์ IoU ที่เข้มงวดขึ้น (0.5-0.95) แบบจำลองยังคงมีความแม่นยำที่แตกต่างกันในแต่ละคลาส

เมื่อพิจารณาผลรวมทุกคลาส พบว่าแบบจำลอง Yolov8n สามารถทำงานได้อย่างมีประสิทธิภาพสูง โดยมีค่า Precision เท่ากับ 0.978, Recall เท่ากับ 0.999, mAP50 เท่ากับ 0.995 และ mAP50-95 เท่ากับ 0.779 ซึ่งสอดคล้องกับผลการวิเคราะห์จากกราฟ Precision-Confidence, Recall-Confidence และ F1-Confidence Curve ที่ยืนยันว่าแบบจำลองมีความสามารถในการรักษาสมดุลระหว่าง Precision และ Recall ได้อย่างมีประสิทธิภาพ

4.6 Confusion Matrix

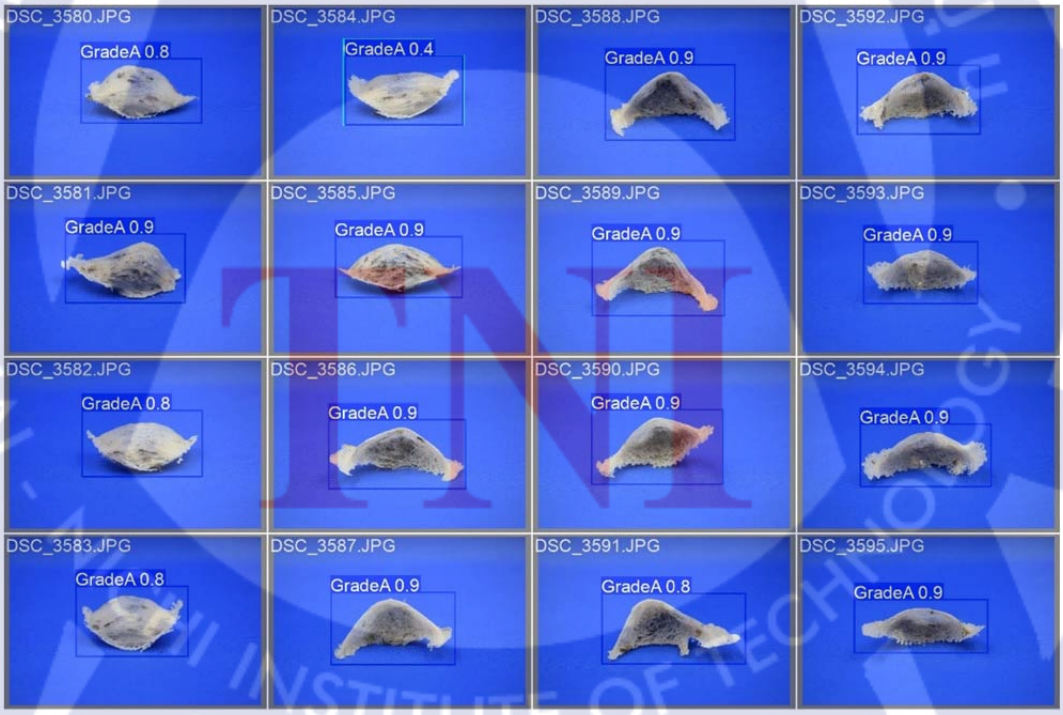
จากรูปที่ 4.5 แสดง Confusion Matrix แสดงให้เห็นว่าโมเดล YOLOv8n สามารถตรวจจับวัตถุในคลาส เกรดเอ, เกรดบี, รังมุม และ Background ได้อย่างแม่นยำในภาพรวม โดยในคลาสเกรดเอ โมเดลทำนายถูกต้อง 40 ตัวอย่างพลาดไปทำนายว่าเป็นเกรดบี และรังมุม อย่างละ 1 ตัวอย่าง และ Background 2 ตัวอย่าง เกรดบีโมเดลทำนายถูกต้อง 36 ตัวอย่างมีพลาดไปทำนายเป็นเกรดเอ 1 ตัวอย่างรังมุมโมเดลทำนายถูกต้อง 21 ตัวอย่างแต่พลาดไปเป็นเกรดเอ 1 ตัวอย่างและ background 2 ตัวอย่างโดยโมเดลนั้นทำนายรังมุมเป็นเกรดเอ และโมเดลยังสามารถหลีกเลี่ยงการตรวจจับวัตถุที่ไม่มีอยู่จริงในคลาส Background ได้อย่างมีประสิทธิภาพ โดยรวมแล้ว โมเดลมีประสิทธิภาพสูงในการตรวจจับคลาส เกรดเอ และ เกรดบี แต่ยังมีโอกาสพัฒนาในส่วนของคลาส รังมุม ซึ่งอาจเกิดจากลักษณะคลาสที่ซ้อนทับกันหรือข้อมูลสำหรับการฝึกที่ยังไม่เพียงพอสำหรับคลาสนี้ และรูปภาพที่ 4.6, 4.7 และ 4.8 ตัวอย่างรูปภาพจากการทดลองการใช้โมเดลสำหรับจำแนกรังนกที่ทำการฝึกโดยค่าที่แสดงจะเป็นค่า Confidence Score



รูปที่ 4.5 แสดง Confusion Matrix

รูปที่ 4.6 ตัวอย่างรูปภาพจากการทดลองการใช้โมเดลสำหรับจำแนกรังนกที่ทำการฝึก ชุดที่ 1

รูปที่ 4.7 ตัวอย่างรูปภาพจากการทดลองการใช้โมเดลสำหรับจำแนกรังนกที่ทำการฝึก ชุดที่ 2



รูปที่ 4.8 ตัวอย่างรูปภาพจากการทดลองการใช้โมเดลสำหรับจำแนกรังนกที่ทำการฝึก ชุดที่ 3

บทที่ 5
สรุปผลการวิจัย

5.1 สรุปผล

งานวิจัยนี้แสดงให้เห็นว่า YOLOv8n มีประสิทธิภาพสูงในการตรวจจับวัตถุในชุดข้อมูลที่ปรับแต่งเอง โดยเฉพาะในคลาส เกรดเอ และ เกรดบี ซึ่งมีค่า Precision และ Recall สูง และค่า F1 Score สูงสุดที่ 0.995 ที่ค่า Confidence Threshold 0.442

5.2 อภิปรายผล

จากรูปกราฟความสัมพันธ์ Precision-Confidence Curve และ Recall-Confidence Curve แสดงถึงประสิทธิภาพของโมเดลในการลด False Positives และ False Negatives อย่างไรก็ตาม คลาส รังมุม มีอัตราการตรวจจับผิดพลาดที่สูงกว่า ซึ่งสะท้อนถึงข้อจำกัดของโมเดลในการแยกความแตกต่างของลักษณะวัตถุที่ซ้อนทับกันหรือการขาดความหลากหลายของข้อมูลในคลาสนี้ อัตราการลดลงของค่าความสูญเสีย (Loss Rate) ที่มีความสม่ำเสมอในระหว่างการฝึก แสดงให้เห็นถึงกระบวนการเรียนรู้ที่ถูกต้องและการ Convergence ของโมเดล นอกจากนี้ ตัวชี้วัดสุดท้าย เช่น ค่า mAP@0.5 ที่ 0.995 และค่า mAP@0.5:0.95 ที่ 0.779 แสดงถึงความน่าเชื่อถือของโมเดลสำหรับงานตรวจจับวัตถุ

5.3 ข้อเสนอแนะ

งานวิจัยในอนาคตควรเน้นไปที่การพัฒนาประสิทธิภาพของโมเดลในคลาส รังมุม โดยการเพิ่มความหลากหลายของข้อมูลผ่านเทคนิค Data Augmentation ขั้นสูง และเพิ่มตัวอย่างที่ซับซ้อนในชุดข้อมูลฝึก การปรับค่า Hyperparameters เช่น ค่า Confidence Threshold และ Learning Rate อาจช่วยให้โมเดลสามารถสร้างสมดุลระหว่าง Precision และ Recall ได้ดียิ่งขึ้น นอกจากนี้ การสำรวจเทคนิคอื่นๆ เช่น Ensemble หรือการผสมผสานกลไก อาจช่วยเพิ่มความแม่นยำและความทนทานของการตรวจจับโดยเฉพาะสำหรับวัตถุที่ซับซ้อนหรือมีลักษณะซ้อนทับกันได้ดียิ่งขึ้น

บรรณานุกรม

บรรณานุกรม

[1] M. S. Lee et al., “The rapid and sensitive detection of edible bird’s nest (aerodramus fuciphagus) in processed food by a loop-mediated isothermal amplification assay,” *J. Food Drug Anal.,* vol. 27, no. 1, pp. 154-163, January 2019.

[2] C. T. Guo et al., “Edible bird’s nest extract inhibits influenza virus infection,” *International Journal of Physical Distribution & Logistics Management,* vol. 70, no. 3, pp. 140-146, July 2006.

[3] นิรัญ เลขสุข, *“ฟาร์มนกแอ่นกินรังในประเทศไทย,”* [Online]. Available : https://elibrary.tsri.or.th/project_content.asp?PJID=RDG6010056. [Accessed : 19 พฤษภาคม 2567].

[4] N. D. Nath et al., “Deep learning for site safety: Real-time detection of personal protective equipment,” *Automat Constr,* vol. 112, no. 3, pp. 103,085, April 2020.

[5] A. M. Vukicevic et al., “Generic compliance of industrial PPE by using deep learning techniques,” *Journal of Retailing,* vol. 148, no. 5, pp. 105,646, April 2022.

[6] J. Zhang et al., “Deep learning system assisted detection and localization of lumbar spondylolisthesis,” *Frontiers in Bioengineering and Biotechnology,* vol. 11, no. 7, pp. 119-210, July 2023.

[7] H. Jeon et al., “Assessment of the object detection ability of interproximal caries on primary teeth in periapical radiographs using deep learning algorithms,” *Journal of the Korean Academy of Pediatric Dentistry,”* vol. 50, no. 3, pp. 263-276, August 2023.

[8] Y. LeCun et al., “Deep learning,” *International Journal of Educational Technology in Higher Education,* vol. 521, no. 13, pp. 436-444, May 2015.

[9] A. Krizhevsky et al., “ImageNet classification with deep convolutional neural networks,” *Advances in Neural Information Processing Systems,* NIPS 2012, Lake Tahoe, Nevada, USA, December 3-6, 2012, pp. 1,097-1,105.

[10] K. He et al., “Deep residual learning for image recognition,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* CVPR 2016, Las Vegas, Nevada, USA, June 27-30, 2016, pp. 770-778.

[11] Tzutalin, *"LabelImg : A Graphical Image Annotation Tool and Label Object Bounding Boxes in Images,"* [Online]. Available : https://github.com/tzutalin/labelImg. [Accessed : May 19, 2024].

[12] J. Redmon et al., "You only look once : Unified, real-time object detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, pp. 779-788.

[13] S. Ren et al., "Faster R-CNN : Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 39, no. 6, pp. 1,137-1,149, June 2017.

[14] I. Song and S. Kim, "AVILNet : A new pliable network with a novel metric for small-object segmentation and detection in infrared images," *Remote Sens,* vol. 13, no. 4, pp. 555, February 2021.

[15] X. Huang and Y. Zhang, "ScanGuard-YOLO : Enhancing x-ray prohibited item detection with significant performance gains," *Sensors,* vol. 24, no. 1, pp. 180, January 2024.

[16] T. Nilmanee et al., "The classification of VARK learning style of computer engineering students of rajamangala university of technology phra nakhon," *Journal of Yala Rajabhat University,* vol. 11, no. 2, pp. 145-157, May 2016.

[17] J. Redmon and A. Farhadi, "YOLOv3 : An incremental improvement," *KKU Engineering Journal,* vol. 34, no. 3, pp. 267-274, April 2018.

[18] R. Yang et al., "DWCA-YOLOv5 : An improved single shot detector for safety helmet detection," *Journal of Yala Rajabhat University,* vol. 11, no. 2, pp. 145-157, September 2021.

[19] A. Bochkovskiy et al., "YOLOv4 : Optimal speed and accuracy of object detection," *International Journal of Behavioral Analytics,* vol. 2, no. 3, pp. 1-12, April 2020.

[20] J. Redmon and A. Farhadi, "YOLOv3 : An incremental improvement," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops,* CVPRW 2018, Salt Lake City, UT, USA, June 18-22, 2018, pp. 1-6.

[21] Ultralytics, *"YOLOv8 - State-of-the-Art YOLO Models,"* [Online]. Available : https://www.ultralytics.com/yolov8. [Accessed : April 20, 2024].

[22] G. Jocher et al., *"YOLO by Ultralytics,"* [Online]. Available : https://github.com/ultralytics/yolov8. [Accessed : April 20, 2024].

[23] T. Y. Lin et al., “Microsoft COCO : Common objects in context,” *Proceedings of the European Conference on Computer Vision,* ECCV 2014, Zurich, Switzerland, September 6-12, 2014, pp. 740-755.

[24] วุฒิพงศ์ นิ่มอ่อน, *“อัตราเฟรมคืออะไร,”* [Online]. Available : https://www.dozzdiy.com/อัตราเฟรมคืออะไร/. [Accessed : 20 เมษายน 2567].

[25] LogovectorUK, *“What is the Difference Between Bitmap and Vector?,”* [Online]. Available: https://logovector.co.uk/blogs/news/what-is-the-difference-between-bitmap-and-vector. [Accessed : April 20, 2024].

[26] Dragonfly, *“Thresholding Filters,”* [Online]. Available : https://www.theobjects.com/dragonfly/dfhelp/20241/Content/Image%20Filtering/Thresholding%2015Filters.htm. [Accessed : April 20, 2024].

[27] H. P. Truong and Y. G. Kim, *Enhanced Line Local Binary Patterns (EL-LBP) : An Efficient Image Representation for Face Recognition,* New Jersey : Pearson Prentice Hall, 2018.

[28] S. Mallick, *“Histogram of Oriented Gradients Explained Using OpenCV,”* [Online]. Available : https://learnopencv.com/histogram-of-oriented-gradients/. [Accessed : April 20, 2024].

[29] Y. Chai and X. Zhang, “Enhanced chest CT detection of pulmonary nodules based on YOLOv8,” *Engineering Letters,* vol. 32, no. 12, pp. 2,221-2,231, December 2024.

[30] L. H. V. Reddy et al., *“Bird Species Recognition Using YOLOv8,”* [Online]. Available : https://www.researchgate.net/. [Accessed : April 20, 2024].

[31] S. Wang and Y. Xu, “MI-YOLO : Improved traffic sign detection based on YOLOv8,” *Engineering Letters,* vol. 32, no. 12, pp. 130-140, December 2024.

[32] Z. Li et al., “FGOD-YOLOv8 : Fine-grained object detection for crops and weeds,” *IEEE Signal Processing Letters,* vol. 32, no. 5, pp. 791-795, December 2024.

[33] Y. Qin et al., “RLGS-YOLO : Metro station passenger detection,” *Engineering Research Express,* vol. 2, no. 1, pp. 267-274, March 2024.

[34] S. Varma and R. Simon, “Bias in error estimation when using cross-validation for model selection,” *BMC Bioinformatics,* vol. 7, no.2, pp. 91, February 2006.