

การพัฒนาโปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จาก
อุบัติเหตุ
PROTOTYPE PROGRAM FOR ANALYZING CAR DAMAGE
FROM ACCIDENT



ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานปีการศึกษา 2565 อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROTOTYPE PROGRAM FOR ANALYZING CAR DAMAGE
FROM ACCIDENT



SUDAPHON SRIMMOKDA
HATAIKAN SAWATNATEE

A SPECIAL PROBLEM SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR
THE DEGREE OF BACHELOR OF SCIENCE PROGRAM IN COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE, SCHOOL OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
ACADEMIC YEAR 2022

หัวข้อปัญหาพิเศษ	การพัฒนาโปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ Prototype program for analyzing car damage from accident	
ชื่อนักศึกษา	นางสาวสุดาพร ศรีมุกดา	รหัสนักศึกษา 62050239
	นางสาวหทัยกาญจน์ สวัสดิ์นที	รหัสนักศึกษา 62050242
ปริญญา	วิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)	
ภาควิชา	วิทยาการคอมพิวเตอร์	
ปีการศึกษา	2565	
อาจารย์ที่ปรึกษา	ผศ.กฤษฎา บุศรา	

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ประจำปีการศึกษา 2565

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.ดร.วิสันต์ ตั้งวงษ์เจริญ	
ประธานกรรมการ	
ผศ.ดร.วรางคณา กิมปาน	
กรรมการ	
ผศ.กฤษฎา บุศรา	
กรรมการและอาจารย์ที่ปรึกษา	

ลิขสิทธิ์ของคณะวิทยาศาสตร์
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	การพัฒนาโปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ Prototype program for analyzing car damage from accident	
ชื่อนักศึกษา	นางสาวสุศดาพร ศรีมุกดา รหัสนักศึกษา 62050239 นางสาวหทัยกาญจน์ สวัสดิ์นที รหัสนักศึกษา 62050242	
ปริญญา	วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)	
ภาควิชา	วิทยาการคอมพิวเตอร์	
คณะ	วิทยาศาสตร์	
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)	
ปีการศึกษา	2565	
อาจารย์ที่ปรึกษา	ผศ.กฤษฎา บุศรา	

บทคัดย่อ

งานวิจัยนี้ นำเสนอการพัฒนาโปรแกรมต้นแบบสำหรับการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุเพื่อเป็นโปรแกรมเสริมการทำงาน ด้วยวิธีการโครงข่ายประสาทเทียมแบบคอนโวลูชัน ในการตรวจจับวัตถุและการแบ่งกลุ่มความหมาย โดยใช้งานผ่านเว็บแอปพลิเคชัน ซึ่งสามารถระบุชื่อประเภทของส่วนพื้นที่รถยนต์จากรูปภาพ ส่วนพื้นที่ของรถยนต์ที่มีร่องรอยความเสียหาย อีกทั้งระบุลักษณะหรือประเภทของร่องรอยความเสียหาย และแนะนำการเป็นผู้ชนจากความเสียหายของรถที่กระทำและรถที่ถูกกระทำ ซึ่งผลการทดสอบทั้งสี่ชุดข้อมูลพบว่าแบบจำลองที่มีประสิทธิภาพแม่นยำที่สุดในการแบ่งส่วนพื้นที่ของรถยนต์คือ HTC ResNet50 การตรวจจับร่องรอยความเสียหายคือ GCNet ResNet101 การแบ่งประเภทร่องรอยความเสียหายคือ HTC ResNet50 การแบ่งประเภทการเป็นผู้ชนจากร่องรอยความเสียหายคือ HRNet ResNet101 จากนั้นจึงได้นำแบบจำลอง HTC ResNet50 และ GCNet ResNet101 มาฝึกอบรมในชุดข้อมูลการแบ่งส่วนพื้นที่และความเสียหาย พบว่า GCNet ResNet101 มีค่า mAP 56.40% ในการตรวจจับวัตถุ และค่า mAP 53.20% ในการแบ่งส่วนภาพ และมีค่า Accuracy 98.10% ค่า Precision 84.76% ค่า Recall 84.76% ซึ่งมากกว่า HTC ResNet50 ดังนั้นจึงเลือกใช้ GCNet ResNet101 ในการพัฒนาเว็บแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
คำสำคัญ : โครงข่ายประสาทเทียมแบบคอนโวลูชัน การตรวจจับวัตถุ การแบ่งกลุ่มความหมาย

Title	Prototype program for analyzing car damage from accident	
Students	Miss Sudaphon Srimookda	Student ID 62050239
	Miss Hataikan Sawatnatee	Student ID 62050242
Degree	Bachelor of Science (Computer Science)	
Department	Computer Science	
School	Science	
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)	
Academic Year	2022	
Advisor	Asst. Prof.Krudsada Budsara	

Abstract

This research presents the development of a prototype software for analyzing car damage from accidents, as an auxiliary program. It utilizes the convolutional neural network approach for Object Detection and Semantic Segmentation, enabling it to identify the types of car regions from images. It can also identify the damaged areas of the car and provide information about the nature or type of the damage, which can be linked to the severity of the damage. Additionally, it recommends being a collision from the damage of the vehicle that was committed and the vehicle that was struck. The results of the four dataset tests showed that the most accurate model for car region segmentation was HTC ResNet50, for damage detection was GCNet ResNet101, for damage classification was HTC ResNet50, and for determining the responsible party was HRNet. Based on the training conducted on the dataset for car region segmentation and damage detection, the HTC ResNet50 and GCNet ResNet101 models were used. The evaluation results showed that GCNet ResNet101 achieved an mAP of 56.40% for object detection and 53.20% for image segmentation. It also demonstrated an accuracy of 98.10%, precision of 84.76%, and recall of 84.76%. These performance metrics were higher than those of HTC ResNet50. Therefore, GCNet ResNet101 was selected as the preferred model for the development of the web application.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าการใช้...
Keywords : convolutional neural network, object detection, semantic segmentation

กิตติกรรมประกาศ

งานวิจัยเรื่องการพัฒนาโปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ เพื่อการสำเร็จการศึกษาของนักศึกษาระดับปริญญาตรีสามารถดำเนินการจนประสบความสำเร็จลุล่วงไปได้ด้วยดี เนื่องจากข้าพเจ้าได้รับความอนุเคราะห์และสนับสนุนเป็นอย่างดียิ่งจาก ผู้ช่วยศาสตราจารย์ กฤษฎา บุศรา ที่กรุณาให้ความช่วยเหลือ ดูแลเอาใจใส่ แนะนำ ชี้แนะข้อบกพร่องและร่วมแก้ปัญหา ติดตามความก้าวหน้าของงานวิจัย รวมทั้งฝึกฝนให้ข้าพเจ้ามีทักษะในด้านการคิด การอ่าน การเขียน และการนำเสนอผลงานวิชาการ ซึ่งเป็นประโยชน์อย่างมากในการพัฒนาตนเอง จนกระทั่งการวิจัยครั้งนี้สำเร็จเรียบร้อยด้วยดี ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้

ขอขอบคุณเพื่อนร่วมงานทุกคนที่อำนวยความสะดวกและช่วยเหลือในการทำวิจัยครั้งนี้

ขอกราบขอบพระคุณบิดามารดาและครอบครัวของข้าพเจ้าที่ได้ให้ชีวิตและโอกาสทางการศึกษา คอยเป็นกำลังใจและให้ความหวังใฝ่เสมอมา ตลอดจนคุณครูและอาจารย์ทุกท่านที่กรุณาประสิทธิ์ประสาทวิชาความรู้อันเป็นประโยชน์แก่ข้าพเจ้า

สุดท้ายนี้ผู้วิจัยหวังว่างานวิจัยฉบับนี้คงเป็นประโยชน์สำหรับ หน่วยงานที่เกี่ยวข้องและผู้สนใจศึกษาต่อไป

สุดาพร ศรีมุกดา
หทัยกาญจน์ สวัสดิ์นที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ช
บทที่ 1 บทนา	1
1.1 ความเป็นมาและความสคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของงานวิจัย.....	1
1.3 ขอบเขตของงานวิจัย.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	3
2.1 การเรียนรู้เชิงลึก (Deep learning).....	3
2.2 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks).....	4
2.3 การประยุกต์ใช้การเรียนรู้เชิงลึกในคอมพิวเตอร์วิทัศน์.....	7
2.4 Microsoft COCO (Common Objects in Context).....	8
2.5 OpenMMLap.....	9
2.6 เมธอดที่ใช้ในการทดลอง.....	11
2.7 การประเมินประสิทธิภาพ (Evaluate model).....	14
2.8 งานวิจัยที่เกี่ยวข้อง.....	17
บทที่ 3 วิธีการดาเนินงานวิจัย	21
3.1 ขั้นตอนการดาเนินงาน.....	21
3.2 การศึกษาและสวจข้อมูลเบื้องต้น.....	22
3.2.1 การเก็บรวบรวมตัวอย่างข้อมูลรูปภาพรถยนต์.....	22
3.2.2 การเตรียมข้อมูลสห้การฝึกสอน.....	23
3.3 การพัฒนาแบบจาสงโดยใช้การเรียนรู้ของเครื่อง.....	26
3.3.1 การพัฒนาแบบจำลองโดยใช้การเรียนรู้ของเครื่อง.....	26
3.3.2 การทดสอบประสิทธิภาพของแบบจาสง.....	27

3.4 การพัฒนาส่วนติดต่อกับผู้ใช้	28
3.4.1 โครงสร้างของโปรแกรมเบื้องต้น	28
3.4.2 การออกแบบฐานข้อมูลของโปรแกรม	28
บทที่ 4 ผลการวิจัยและการอภิปรายผล	30
4.1 การทดสอบประสิทธิภาพของแบบจ าสง	30
4.1.1 ประสิทธิภาพของแบบจ าสงในการแบ่งส่วนพื้นที่ของรถยนต์	30
4.1.2 ประสิทธิภาพของแบบจ าสงในการตรวจจ้งร่องรอยความเสียหาย	32
4.1.3 ประสิทธิภาพของแบบจ าสงในการแบ่งประเภทร่องรอยความเสียหาย	33
4.1.4 ประสิทธิภาพของแบบจ าสงในการแบ่งประเภทการเป็นผู้ชนจากร่องรอย	35
4.1.5 ประสิทธิภาพของแบบจ าสงส าหรับการแบ่งส่วนพื้นที่รถและความเสียหาย	37
4.1.6 Confusion Matrix ของแบบจ าสงส าหรับการแบ่งส่วนพื้นที่รถและความเสียหาย	38
4.1.7 ส่วนติดต่อกับผู้ใช้ของเว็บแอปพลิเคชัน	44
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	48
5.1 สรุปผลการวิจัย	48
5.2 ขอจ กัด	48
5.3 ข้อเสนอแนะ	49
เอกสารอ้างอิง	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 3.1 การกระจายตัวของชุดข้อมูลการแบ่งส่วนพื้นที่ของรถยนต์.....	24
ตารางที่ 3.2 แบบจ าสงที่ใช้ทดลองในงานวิจัย	27
ตารางที่ 3.3 อธิบายตาราง Images ในฐานข้อมูล	29
ตารางที่ 3.4 อธิบายตาราง Models ในฐานข้อมูล	29
ตารางที่ 3.5 อธิบายตาราง Predictions ในฐานข้อมูล	29
ตารางที่ 4.1 ประสิทธิภาพจากการฝึกอบรมแบบจ าสงการแบ่งส่วนพื้นที่ของรถยนต์	30
ตารางที่ 4.2 ประสิทธิภาพจากการฝึกอบรมแบบจ าสงการตรวจจ ับร่องรอยความเสียหาย	32
ตารางที่ 4.3 ประสิทธิภาพจากการฝึกอบรมแบบจ าสงการแบ่งประเภทร่องรอยความเสียหาย.....	34
ตารางที่ 4.4 ประสิทธิภาพจากการฝึกอบรมแบบจ าสงการแบ่งประเภทการเป็นผู้ชน.....	35
ตารางที่ 4.5 ประสิทธิภาพจากการฝึกอบรมแบบจ าสงในแต่ละชุดข้อมูล	36
ตารางที่ 4.6 ประสิทธิภาพจากการฝึกอบรมแบบจ าสงส หรับการแบ่งส่วนพื้นที่ร อด ความเสียหาย 37	37
ตารางที่ 4.7 ประสิทธิภาพแบบจ าสง HTC ResNet50 จาก Confusion Matrix	39
ตารางที่ 4.8 ประสิทธิภาพแบบจ าสง GCNet ResNet101 จาก Confusion Matrix	42
ตารางที่ 4.9 กรณีตัวอย่างการทดลองใช้ GCNet ResNet101	393

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
รูปที่ 2.1 machine learning เทียบกับ deep learning.....	4
รูปที่ 2.2 การคำนวณ cross-correlation.....	5
รูปที่ 2.3 Max pooling ที่มีหน้าต่าง pooling window รูปร่าง 2 x 2.....	5
รูปที่ 2.4 padding two-dimensional cross-correlation.....	6
รูปที่ 2.5 cross-correlation strides 3 และ 2 ส าหรับความสูงและความกว้าง	6
รูปที่ 2.6 Convolutional Neural Networks	7
รูปที่ 2.7 ตัวอย่างการประยุกต์ใช้การเรียนรู้เชิงลึกในคอมพิวเตอร์วิทัศน์.....	8
รูปที่ 2.8 ตัวอย่างโครงสร้าง อธิบายประกอบภาพ Object Detection	9
รูปที่ 2.9 Mask R-CNN framework ส าหรับ instance segmentation	12
รูปที่ 2.10 Hybrid Task Cascade.....	12
รูปที่ 2.11 High-Resolution Net (HRNet) อย่างง่าย.....	13
รูปที่ 2.12 Cascade Mask R-CNN	14
รูปที่ 2.13 ตาราง Confusion Matrix.....	15
รูปที่ 2.14 detection evaluation metrics ที่พัฒนาโดย COCO	17
รูปที่ 3.1 ขั้นตอนการด าเนินงาน	22
รูปที่ 3.2 ตัวอย่างรูปภาพรถยนต์จาก Coco Car Damage Detection Dataset	23
รูปที่ 3.3 ตัวอย่างรูปภาพรถยนต์จาก Stanford Cars Dataset.....	23
รูปที่ 3.4 ตัวอย่างการแบ่งส่วนพื้นที่ของรถยนต์	24
รูปที่ 3.5 ตัวอย่างชุดข้อมูลการแบ่งส่วนพื้นที่ของรถยนต์ (Test set).....	25
รูปที่ 3.6 ตัวอย่างชุดข้อมูลร่องรอยความเสียหาย (Test set).....	25
รูปที่ 3.7 รูปโครงสร้างของฐานข้อมูล	29
รูปที่ 4.1 ตัวอย่างรูปภาพผลการท นายของแบบจ าสงการแบ่งส่วนพื้นที่ของรถยนต์.....	31
รูปที่ 4.2 ตัวอย่างรูปภาพผลการท นายของแบบจ าสงการแบ่งส่วนพื้นที่ของรถยนต์.....	31
รูปที่ 4.3 ตัวอย่างรูปภาพผลการท นายของแบบจ าสงการแบ่งส่วนพื้นที่ของรถยนต์.....	32
รูปที่ 4.4 ตัวอย่างรูปภาพผลการท นายของแบบจ าสงการตรวจจ ับร่องรอยความเสียหาย	33
รูปที่ 4.5 ตัวอย่างรูปภาพผลการท นายของแบบจ าสงการแบ่งประเภทร่องรอยความเสียหาย	34
รูปที่ 4.6 ตัวอย่างรูปภาพผลการท นายของแบบจ าสงการเป็นผู้ชนจากร่องรอยความเสียหาย	36
รูปที่ 4.7 Confusion Matrix HTC ResNet50 ส าหรับการแบ่งส่วนพื้นที่รถและความเสียหาย	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถเผยแพร่ไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.8 Confusion Matrix GCNet ResNet101 สำหรับการแบ่งส่วนพื้นที่รถและความเสียหาย 40

รูปที่ 4.9 หน้าจอหลักของแอปพลิเคชัน 45

รูปที่ 4.10 หน้าจอแสดงผลการตรวจจับของแบบจำลอง 46

รูปที่ 4.11 หน้าจอแสดงข้อมูลตารางในฐานข้อมูล..... 47



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันปัญญาประดิษฐ์ (Artificial Intelligence) ได้เข้ามามีบทบาทสำคัญอย่างมากในหลายด้าน ซึ่งยังอยู่ในช่วงเริ่มต้นของการพัฒนาความสามารถ โดยได้มีการพัฒนาอัลกอริทึมโดยการสอนให้ระบบคอมพิวเตอร์ทำการเรียนรู้ได้ด้วยตนเองโดยใช้ข้อมูล (Machine Learning) และพัฒนาสู่อัลกอริทึมแบบระบบเรียนรู้เชิงลึก (Deep learning) ที่สามารถทำงานได้มีประสิทธิภาพมากขึ้น โดยทั้ง 3 เทคโนโลยีนี้มีความสำคัญอย่างยิ่งต่อสาขาวิทยาศาสตร์ข้อมูล (Data Science)

เมื่อเกิดอุบัติเหตุทางรถยนต์ บริษัทประกันภัยรถยนต์จะต้องมีการเก็บข้อมูลของอุบัติเหตุที่เกิดขึ้น เพื่อนำมาประมวลผลและวิเคราะห์สรุปผล ซึ่งได้มีการเก็บภาพถ่ายเพื่อวิเคราะห์สิ่งที่เกิดขึ้น ตัวอย่างเช่น ร่องรอยของการเกิดความเสียหายจากอุบัติเหตุรถยนต์ เนื่องจากการทำงานของบุคลากรต้องใช้เวลาในการวิเคราะห์และอาจเกิดข้อผิดพลาดขึ้นได้ เพื่อลดความเสี่ยงจึงได้มีการพัฒนาโปรแกรมต้นแบบเพื่อรองรับการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุมาใช้ในการวิเคราะห์และทำนายผลลัพธ์ ซึ่งโปรแกรมได้มีการประยุกต์ใช้อัลกอริทึมแบบระบบเรียนรู้เชิงลึก (Deep learning) คือวิธีการเรียนรู้แบบอัตโนมัติด้วยการเลียนแบบการทำงานของโครงข่ายประสาทของมนุษย์ (Neurons) โดยนำระบบโครงข่ายประสาท (Neural Network) มาซ้อนกันหลายชั้น (Layer) และทำการเรียนรู้ข้อมูลตัวอย่าง ซึ่งข้อมูลดังกล่าวจะถูกนำไปใช้ในการตรวจจับรูปแบบ (Pattern) หรือจัดหมวดหมู่ข้อมูล (Classify the Data) โดยโปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ จะมีการตรวจจับรูปแบบในรูปภาพเพื่อวิเคราะห์และจำแนกตำแหน่งความเสียหาย วิเคราะห์ร่องรอยความเสียหายรถยนต์ และทำนายผลลัพธ์แนะนำการเป็นผู้ชนจากความเสียหายของรถที่กระทำและรถที่ถูกกระทำ

ทางผู้จัดทำจึงมีความสนใจในการพัฒนาโปรแกรมต้นแบบสำหรับการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ โดยนำมาใช้เป็นโปรแกรมเสริมการทำงานของการทำงานปกติจากบุคคลหรือบุคลากรที่รับผิดชอบในบริษัทประกันภัยรถยนต์

1.2 วัตถุประสงค์ของงานวิจัย

- 1) เพื่อศึกษาการทำงานและการพัฒนาโปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ
- 2) เพื่อพัฒนาโปรแกรมต้นแบบสำหรับการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ
- 3) เพื่อเป็นโปรแกรมเสริมการทำงานในการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ

1.3 ขอบเขตของงานวิจัย

- 1) ศึกษาขอบเขตการทำงานการวิเคราะห์ความเสียหายจากอุบัติเหตุรถยนต์ประเภทสี่ล้อ
- 2) วิเคราะห์ด้วยโมเดลการเรียนรู้เชิงลึก (Deep learning)
- 3) พัฒนาโปรแกรมต้นแบบสำหรับการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ โดยครอบคลุมการทำงาน
 - 3.1) การจัดการรูปภาพความเสียหายของรถยนต์ที่เกิดจากอุบัติเหตุของทั้งผู้เสียหายและคู่กรณี
 - 3.2) วิเคราะห์และจำแนกตำแหน่งความเสียหายของรถยนต์จากรูปภาพ
 - 3.3) วิเคราะห์ร่องรอยความเสียหายในลักษณะต่าง ๆ จากรูปภาพ
 - 3.4) วิเคราะห์และแนะนำการเป็นผู้ชนจากความเสียหายของรถที่กระทำและรถที่ถูกกระทำจากรูปภาพ

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ได้ศึกษาเรียนรู้การทำงานและการพัฒนาโปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ
- 2) ได้พัฒนาโปรแกรมต้นแบบสำหรับการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ
- 3) ได้โปรแกรมเสริมการทำงานในการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

งานวิจัยนี้จัดทำเพื่อศึกษาการทำงานและการพัฒนาโปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ เพื่อนำไปพัฒนาเป็นโปรแกรมต้นแบบและโปรแกรมเสริมการทำงาน โดยผู้วิจัยได้ทำการศึกษาเอกสารและงานวิจัยต่าง ๆ ที่เกี่ยวข้องและมีความสำคัญต่องานวิจัยนี้เพื่อเป็นแนวทางเข้าสู่กระบวนการวิจัย โดยมีเนื้อหาสาระที่เกี่ยวข้องดังต่อไปนี้

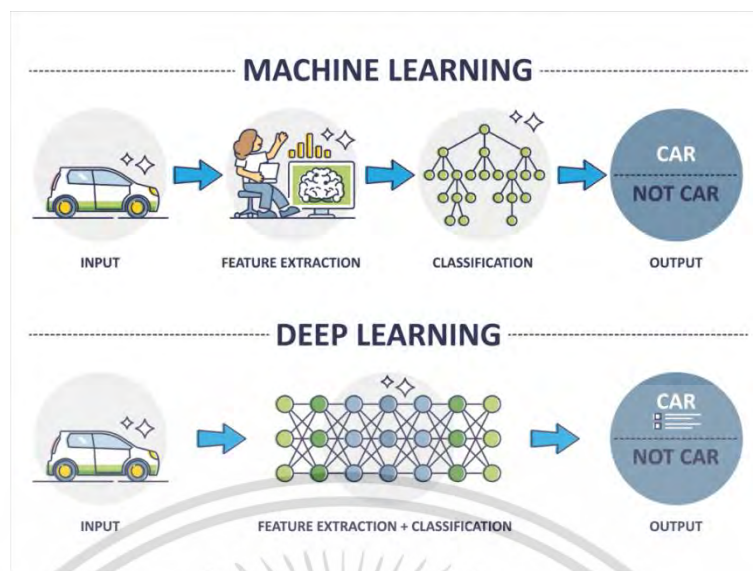
2.1 การเรียนรู้เชิงลึก (Deep learning)

ในช่วงหลายปีที่ผ่านมาพบว่าวิธีการเรียนรู้เชิงลึก (deep learning) มีที่มาจากการใช้ layer ของ neural network หลาย ๆ อันมาต่อกัน deep learning มีประสิทธิภาพเหนือกว่าเทคนิคการเรียนรู้ของเครื่อง (machine learning) ที่ล้ำสมัยก่อนหน้านี้ในหลายสาขา กล่าวคือ deep learning algorithm ให้ความแม่นยำมากกว่า machine learning algorithm ในด้านของจำแนกภาพ (image classification) มากถึง 41%

การเรียนรู้เชิงลึกช่วยให้โมเดลการคำนวณของเลเยอร์การประมวลผลหลายชั้นสามารถเรียนรู้และแสดงข้อมูลด้วยนามธรรมหลายระดับ ซึ่งเลียนแบบวิธีที่สมองรับรู้และเข้าใจข้อมูลหลายรูปแบบ ดังนั้นจึงสามารถจับภาพโครงสร้างที่ซับซ้อนของข้อมูลขนาดใหญ่ การเรียนรู้เชิงลึกเป็นกลุ่มวิธีการที่หลากหลาย ซึ่งครอบคลุมโครงข่ายประสาทเทียม (neural networks) แบบจำลองความน่าจะเป็นแบบลำดับชั้น และอัลกอริทึมการเรียนรู้คุณลักษณะที่ไม่ได้รับการดูแลและควบคุมที่หลากหลาย ความสนใจที่เพิ่มขึ้นอย่างรวดเร็วในวิธีการเรียนรู้เชิงลึกเป็นผลมาจากข้อเท็จจริงที่ว่าวิธีการดังกล่าวมีประสิทธิภาพเหนือกว่าเทคนิคที่ล้ำสมัยก่อนหน้านี้ในหลาย ๆ งาน เช่นเดียวกับข้อมูลที่ซับซ้อนมากมายจากแหล่งข้อมูลต่าง ๆ (เช่น ภาพ เสียง การแพทย์ สังคม และเซ็นเซอร์) ท่ามกลางปัจจัยที่สนับสนุนการเรียนรู้เชิงลึกอย่างมากคือการมีชุดข้อมูลขนาดใหญ่ คุณภาพสูง พร้อมกับการเสริมศักยภาพของการประมวลผลของหน่วยประมวลผลกราฟิกประสิทธิภาพสูง (GPU) แบบขนาน ซึ่งทำให้เปลี่ยนจาก CPU ไปเป็น GPU การฝึกอบรมจึงช่วยให้สามารถเร่งความเร็วอย่างมากในการฝึกแบบจำลองเชิงลึก ซึ่งสร้างความต้องการทรัพยากรภายในจำนวนมากและมีค่าใช้จ่ายสูง

การเรียนรู้เชิงลึกได้กระตุ้นให้เกิดความก้าวหน้าอย่างมากในปัญหาคอมพิวเตอร์วิทัศน์ที่หลากหลาย เช่น การตรวจจับวัตถุ การแบ่งกลุ่มความหมาย Convolutional Neural Networks (CNNs) ถือเป็นหนึ่งในแบบจำลองการเรียนรู้เชิงลึกที่ได้รับความนิยมมากที่สุดที่เกี่ยวข้องกับการ

เอกสารนี้เป็นทรัพย์สินทางปัญญาของสถาบันเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 machine learning เทียบกับ deep learning

2.2 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks)

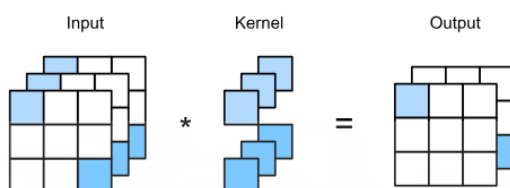
Convolutional Neural Networks (CNNs) เป็นตระกูลของโครงข่ายประสาทเทียมหลายชั้น (multi-layer neural networks) CNN คือ neural network หลายเลเยอร์ที่มีโครงสร้างเฉพาะตัว โดยถูกออกแบบมาเพื่อเพิ่มความสามารถในการสกัดเอา feature ที่มีความซับซ้อนมากยิ่งขึ้นจากข้อมูล ที่ออกแบบมาโดยเฉพาะ สำหรับใช้กับข้อมูลสองมิติ เช่น รูปภาพและวิดีโอ

สถาปัตยกรรมของ CNN หรือ ConvNet คล้ายคลึงกับรูปแบบการเชื่อมต่อของเซลล์ประสาทในสมองมนุษย์ และได้รับแรงบันดาลใจจากการจัดระเบียบของ Visual Cortex เซลล์ประสาทแต่ละเซลล์จะตอบสนองต่อสิ่งเร้าเฉพาะในพื้นที่จำกัดของลานสายตาที่เรียกว่า Receptive Field คอลเล็กชันของฟิลต์ดังกล่าวทับซ้อนกันเพื่อครอบคลุมพื้นที่ภาพทั้งหมด

CNN ประกอบด้วยชั้นประสาทหลัก 3 ประเภท ได้แก่ (i) ชั้นคอนโวลูชันนัล (Convolutional layer) (ii) ชั้นการรวมเข้าด้วยกัน (Pooling layer) และ (iii) ชั้นที่เชื่อมต่ออย่างสมบูรณ์ (Fully-Connected layer) เลเยอร์แต่ละประเภทมีบทบาทที่แตกต่างกัน CNN ประสบความสำเร็จอย่างมากในการประยุกต์ใช้การมองเห็นด้วยคอมพิวเตอร์ เช่น การจดจำใบหน้า การตรวจจบบั้ววัตถุ การเพิ่มพลังการมองเห็นในวิทยาการหุ่นยนต์ และรถยนต์ที่ขับเคลื่อนด้วยตนเอง

(i) **Convolutional layer** เป็นองค์ประกอบหลักของ CNN และเป็นพื้นที่ที่การคำนวณส่วนใหญ่เกิดขึ้น ต้องใช้ส่วนประกอบบางอย่าง ได้แก่ ข้อมูลเข้า (input data) ตัวกรอง (filter) และฟังก์ชันลักษณะ (feature map) เพื่อรวมภาพทั้งหมดรวมถึงแมปคุณลักษณะระดับกลาง เพื่อสร้าง feature map ต่าง ๆ หลังจากการดำเนินการบิดแต่ละครั้ง CNN จะใช้การแปลง Rectified Linear Unit (ReLU) กับ feature map โดยแนะนำความไม่เชิงเส้น (non linear) ให้กับโมเดล เนื่องจากข้อ

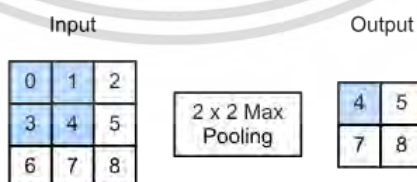
ได้เปรียบของการดำเนินการบิด งานหลายชิ้นได้เสนอให้ใช้แทนเลเยอร์ที่เชื่อมต่อกันอย่างสมบูรณ์ (Fully Connected layer) เพื่อให้ได้รับเวลาการเรียนรู้ที่เร็วขึ้น รูปที่ 2 ตัวอย่างการคำนวณ cross-correlation ใช้ 1×1 convolution kernel อินพุต 3 channels and เอาต์พุต 2 channels อินพุตและเอาต์พุตมีความสูงและความกว้างเท่ากัน



รูปที่ 2.2 การคำนวณ cross-correlation

(ii) **Pooling layer** มีหน้าที่รับผิดชอบในการลดมิติเชิงพื้นที่ (width x height) ลดจำนวนพารามิเตอร์ ในอินพุตเช่นเดียวกับ Convolutional layer แต่จะไม่ส่งผลกระทบต่อมิติความลึกของปริมาตรข้อมูล การดำเนินการที่ดำเนินการโดยเลเยอร์นี้เรียกอีกอย่างว่าการสุ่มตัวอย่างย่อย (subsampling) หรือการลดสุ่มตัวอย่าง (downsampling) แม้ว่าการลดขนาดทำให้สูญเสียข้อมูลไปพร้อม ๆ กัน อย่างไรก็ตาม การสูญเสียดังกล่าวเป็นประโยชน์สำหรับ CNN เนื่องจากขนาดที่ลดลงช่วยลดความซับซ้อน ปรับปรุงประสิทธิภาพ และจำกัดความเสี่ยงของการทำงานเกินพอดี (overfitting) อีกด้วย โดย pooling มีสองประเภทหลัก ได้แก่

Max pooling ขณะที่ตัวกรองเคลื่อนผ่านอินพุต ตัวกรองจะเลือกพิกเซลที่มีค่าสูงสุดเพื่อส่งไปยังอาร์เรย์เอาต์พุต นอกจากนี้ วิธีการนี้มีแนวโน้มที่จะใช้บ่อยกว่า ตัวอย่างในรูป 2. Max pooling ที่มีหน้าต่าง pooling window รูปร่าง 2×2 ส่วนที่แรงงาเป็นองค์ประกอบเอาต์พุตแรก เช่นเดียวกับองค์ประกอบเทนเซอร์อินพุตที่ใช้สำหรับการคำนวณเอาต์พุต เอาต์พุตเทนเซอร์ในรูปมีความสูง 2 และความกว้าง 2 องค์ประกอบทั้งสี่ได้มาจากค่าสูงสุดในแต่ละ pooling window
 $\max(0,1,3,4) = 4$, $\max(1,2,4,5) = 5$, $\max(3,4,6,7) = 7$, $\max(4,5,7,8) = 8$

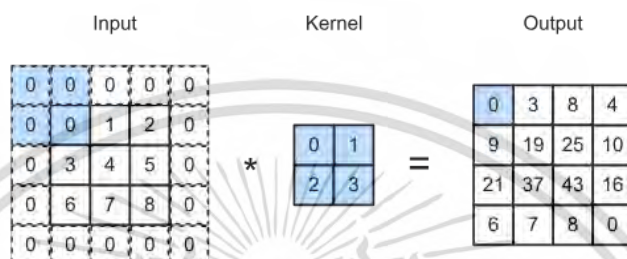


รูปที่ 2.3 Max pooling ที่มีหน้าต่าง pooling window รูปร่าง 2×2

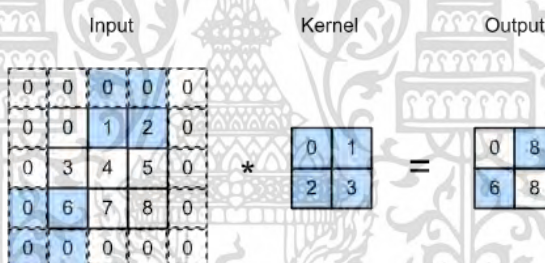
Average pooling ขณะที่ตัวกรองเคลื่อนผ่านอินพุต จะคำนวณค่าเฉลี่ยภายในฟิลต์รับเอกสารนี้สัญญาณเพื่อส่งไปยังอาร์เรย์เอาต์พุตเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้ ยังมีรูปแบบอื่น ๆ อีกจำนวนมากของ Pooling layer ในเอกสาร ซึ่งแต่ละรูปแบบได้รับแรงบันดาลใจจากแรงจูงใจที่แตกต่างกันและตอบสนองความต้องการที่แตกต่างกัน ตัวอย่างเช่น การรวมเลเยอร์แบบสุ่ม (stochastic pooling) การรวมพีระมิดเชิงพื้นที่ (spatial pyramid pooling) และ def-pooling

Pooling layer จะเปลี่ยนรูปร่างเอาต์พุต ซึ่งสามารถปรับการท างานเพื่อให้ได้รูปร่างเอาต์พุตที่ต้องการได้โดยการเสริม (padding) อินพุตและปรับระยะก้าว (strides)



รูปที่ 2.4 padding two-dimensional cross-correlation

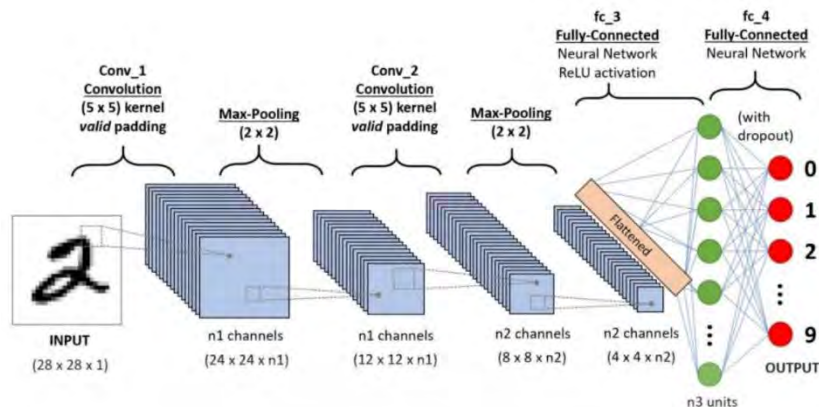


รูปที่ 2.5 cross-correlation strides 3 และ 2 สำหรับความสูงและความกว้าง

(iii) **Fully Connected layer** ค่าพิกเซลของภาพอินพุตไม่ได้เชื่อมต่อกันโดยตรงกับเลเยอร์เอาต์พุตในเลเยอร์ที่เชื่อมต่อบางส่วน การให้เหตุผลระดับสูงในโครงข่ายประสาทเทียมจะดำเนินการผ่าน Fully Connected layer ในแต่ละโหนดในเลเยอร์เอาต์พุตจะเชื่อมต่อกันโดยตรงกับโหนดในเลเยอร์ก่อนหน้า เลเยอร์นี้จะแปลง feature map 2 มิติเป็นเวกเตอร์ 1 มิติ เวกเตอร์ที่ได้มาอาจถูกส่งต่อไปยังหมวดหมู่จำนวนหนึ่งสำหรับการจำแนกประเภท หรืออาจถูกพิจารณาว่าเป็นเวกเตอร์คุณลักษณะ (feature vector) ส สำหรับการประมวลผลเพิ่มเติม

เลเยอร์นี้ทำหน้าที่จัดประเภทตามคุณสมบัติที่แยกผ่านเลเยอร์ก่อนหน้าและตัวกรองต่างๆ ในขณะที่เลเยอร์ Convolutional และ Pooling มักจะใช้ฟังก์ชัน ReLU เลเยอร์ FC มักจะใช้ประโยชน์จากฟังก์ชันการเปิดใช้งาน softmax เพื่อจำแนกอินพุตอย่างเหมาะสม สร้างความน่าจะเป็น

เอกสารนี้ตั้งแต่ 0 ถึง 1 เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 Convolutional Neural Networks

2.3 การประยุกต์ใช้การเรียนรู้เชิงลึกในคอมพิวเตอร์วิทัศน์

2.3.1 การจำแนกภาพ (Image classification)

การจำแนกภาพ (Image classification) เป็นงานพื้นฐานที่พยายามทำความเข้าใจภาพรวมทั้งหมด เป้าหมายคือการจำแนกรูปภาพโดยกำหนดให้กับป้ายกำกับเฉพาะ โดยทั่วไป การจัดประเภทรูปภาพหมายถึงรูปภาพที่มีวัตถุเพียงชิ้นเดียวปรากฏขึ้นและได้รับการวิเคราะห์ ในทางตรงกันข้าม การตรวจจับวัตถุ (object detection) นั้นเกี่ยวข้องกับทั้งการจัดหมวดหมู่และการระบุตำแหน่ง (localization) และใช้เพื่อวิเคราะห์กรณีที่มีวัตถุหลายชิ้นอาจมีอยู่ในรูปภาพที่เหมือนจริงมากขึ้น

2.3.2 การตรวจจับวัตถุ (Object detection)

การตรวจจับวัตถุ (Object detection) เป็นงานคอมพิวเตอร์วิทัศน์ที่เป้าหมายคือการตรวจจับและค้นหาวัดที่สนใจในภาพหรือวิดีโอ งานเกี่ยวข้องกับการระบุตำแหน่งและขอบเขตของวัตถุในภาพ และจำแนกวัตถุออกเป็นประเภทต่าง ๆ

วิธีการที่ทันสมัยสามารถแบ่งออกเป็นสองประเภทหลัก ได้แก่ วิธีการแบบขั้นตอนเดียวเป็นการจัดลำดับความสำคัญของความเร็วการอนุมาน และโมเดลตัวอย่าง ได้แก่ YOLO, SSD และ RetinaNet และวิธีการแบบสองขั้นตอน เป็นจัดลำดับความสำคัญของความแม่นยำในการตรวจจับ และโมเดลตัวอย่าง ได้แก่ Faster R-CNN, Mask R-CNN และ Cascade R-CNN เกณฑ์มาตรฐานที่ได้รับความนิยมมากที่สุดคือชุดข้อมูล MSCOCO โดยทั่วไปแล้วโมเดลจะได้รับการประเมินตามตัวชี้วัดความแม่นยำเฉลี่ย (Mean Average Precision metric)

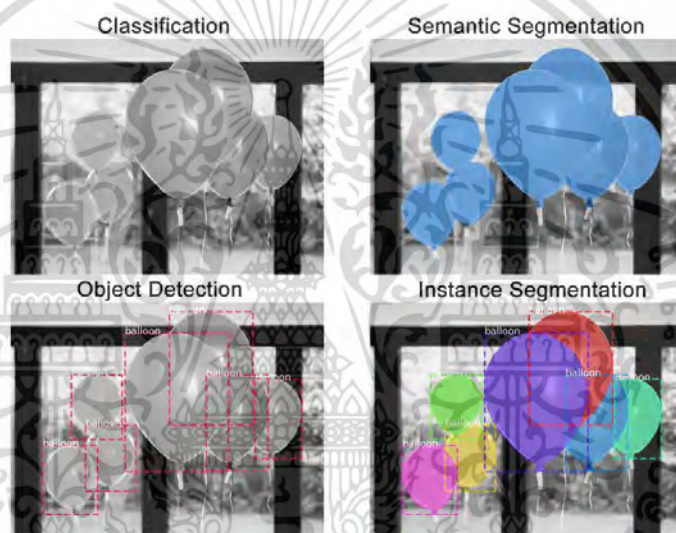
2.3.3 การแบ่งกลุ่มความหมาย (Semantic segmentation)

การแบ่งข้อมูลภาพ (Semantic segmentation) เป็นงานคอมพิวเตอร์วิทัศน์ที่มีเป้าหมายเพื่อจัดหมวดหมู่แต่ละพิกเซลในภาพเป็นคลาสหรือวัตถุ เป้าหมายคือการสร้างแผนที่ (map) แบ่งส่วนไม่วากกรณีใดๆทั้งสิ้น อีกทั้งยังมีเหตุผลและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พิกเซลที่หนาแน่นของรูปภาพ โดยที่แต่ละพิกเซลถูกกำหนดให้กับคลาสหรือวัตถุเฉพาะ ตัวอย่างการวัดประสิทธิภาพสำหรับงานนี้คือ Cityscapes, PASCAL VOC และ ADE20K แบบจำลองมักจะได้รับการประเมินด้วยเมตริก Mean Intersection-Over-Union (Mean IoU) และ Pixel Accuracy metrics

2.3.4 การแบ่งส่วนอินสแตนซ์ (Instance segmentation)

การแบ่งส่วนอินสแตนซ์ (Instance segmentation) เป็นงานคอมพิวเตอร์วิทัศน์ที่เกี่ยวข้องกับการระบุและแยกวัตถุแต่ละชิ้นภายในภาพ รวมถึงการตรวจจับขอบเขตของวัตถุแต่ละชิ้นและกำหนดป้ายกำกับเฉพาะให้กับแต่ละวัตถุ เป้าหมายของการแบ่งส่วนอินสแตนซ์คือการสร้างแผนที่แบ่งส่วนตามพิกเซลของรูปภาพ โดยที่แต่ละพิกเซลถูกกำหนดให้กับอินสแตนซ์ของวัตถุเฉพาะ



รูปที่ 2.7 ตัวอย่างการประยุกต์ใช้การเรียนรู้เชิงลึกในคอมพิวเตอร์วิทัศน์

2.4 Microsoft COCO (Common Objects in Context)

COCO (Common Objects in Context) เป็นชุดข้อมูลที่ใช้ในการส่งเสริมการวิจัยและพัฒนาด้านประมวลผลภาพคอมพิวเตอร์ ซึ่งประกอบด้วยรูปภาพและข้อมูลเชิงลึกสำหรับการตรวจจับวัตถุ การตรวจจับจุดสนใจ การแยกพื้นที่ของสิ่งของ และงานอื่น ๆ

COCO format เป็นรูปแบบการจัดเก็บข้อมูลใน COCO ซึ่งใช้ JSON (JavaScript Object Notation) เพื่อเก็บข้อมูลป็น รูปแบบนี้ใช้ในการเก็บข้อมูลเชิงลึกเกี่ยวกับวัตถุและป้ายกำกับของวัตถุ ในรูปภาพ รวมถึงข้อมูลอื่น ๆ เช่น พิกัดของกล่องคำบรรยาย (caption) และข้อมูลเพิ่มเติมเกี่ยวกับ

หมวดหมู่ของวัตถุ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น COCO และ COCO format คือ COCO format ใช้ในการเก็บข้อมูลและโครงสร้างข้อมูลในรูปแบบที่เหมาะสมสำหรับการใช้งานกับชุดข้อมูล COCO และ COCO API ถูกสร้างขึ้นมาเพื่อให้สามารถเข้าถึงและจัดการข้อมูลในรูปแบบ COCO format ได้ง่ายขึ้น

```

annotation{
  "id"           : int,
  "image_id"     : int,
  "category_id"  : int,
  "segmentation" : RLE or [polygon],
  "area"         : float,
  "bbox"         : [x,y,width,height],
  "iscrowd"      : 0 or 1,
}

categories[ {
  "id"           : int,
  "name"         : str,
  "supercategory": str,
} ]

```

รูปที่ 2.8 ตัวอย่างโครงสร้าง คอธิบายประกอบภาพObject Detection

2.5 OpenMMLap

OpenMMLab เป็นชุดเครื่องมือและเฟรมเวิร์ก (framework) ที่เน้นการพัฒนาและวิจัยด้านการประมวลผลภาพและการมองเห็นของคอมพิวเตอร์ (computer vision) ซึ่งถูกพัฒนาขึ้นโดยทีมงาน OpenMMLab ภายใต้การดูแลและสนับสนุนของ OpenAI

OpenMMLab เป็นโครงการโอเพนซอร์ส (open-source) และมุ่งเน้นในการสร้างเครื่องมือที่มีประสิทธิภาพสูงและมีความยืดหยุ่นสูงสำหรับงานด้านการประมวลผลภาพ ทั้งในเชิงวิจัยและการพัฒนาแอปพลิเคชันที่เกี่ยวข้องกับการมองเห็นคอมพิวเตอร์ เช่น การตรวจจับวัตถุ การติดตามวัตถุ การแยกแยะวัตถุ การปรับปรุงภาพ การสร้างภาพเคลื่อนไหว และอื่น ๆ

MMDetection เป็นหนึ่งในโครงการย่อยของ OpenMMLab ซึ่งเป็นชุดเครื่องมือและเฟรมเวิร์ก (framework) ที่เน้นการตรวจจับวัตถุในภาพ (object detection) โดยใช้เทคนิคการประมวลผลภาพและการมองเห็นของคอมพิวเตอร์ (computer vision) เพื่อระบุและติดตามวัตถุในภาพหรือวิดีโอ MMDetection ประกอบด้วย 7 ส่วน ได้แก่

- 1) apis ให้ API ระดับสูงสำหรับการอนุมานโมเดลช่วยให้ผู้ใช้สามารถโหลดโมเดลที่ถูกฝึกสอนมาก่อนหน้านี้และใช้งานได้ง่าย โดยใช้เครื่องมือหรือไลบรารีที่มีอินเทอร์เฟซระดับสูง ซึ่งจะมีฟังก์ชันและเมทอดที่ช่วยในกระบวนการประมวลผลข้อมูลของโมเดล เช่น การโหลดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทดสอบ (inference data) และวัตถุ (inference objects) การประมวลผลสำหรับการทำนาย (inference prediction) และวิธีการดำเนินการอื่น ๆ ที่เกี่ยวข้องกับการใช้งานโมเดล
- 2) structures จัดเตรียมโครงสร้างข้อมูล เช่น bbox, mask และ DetDataSample
 - 3) datasets รวบรวมชุดข้อมูลต่างๆ สำหรับการตรวจจับวัตถุ การแบ่งส่วนอินสแตนซ์ และการแบ่งส่วนแบบพาโนปิก
 - 4) models เป็นส่วนสำคัญที่สุดสำหรับเครื่องตรวจจับและประกอบด้วยส่วนประกอบต่างๆ ของเครื่องตรวจจับ
 - o detectors คือการกำหนดคลาสโมเดลการตรวจจับทั้งหมด
 - o data_preprocessors ใช้สำหรับประมวลผลข้อมูลอินพุตของโมเดลล่วงหน้า
 - o backbones เป็นส่วนหนึ่งในโมเดลเชิงลึกที่มีบทบาทสำคัญในการทำนายและคัดแยกลักษณะของภาพ
 - o necks Necks เป็นส่วนหนึ่งในโมเดลเชิงลึกที่มีบทบาทสำคัญในการรวมและรวบรวมข้อมูลที่ได้จาก backbones
 - o dense_heads คือหัวตรวจจับต่างๆ ที่ทำการคาดคะเนแบบหนาแน่น
 - o roi_heads คือหัวตรวจจับต่างๆ ที่คาดการณ์จาก Rols
 - o seg_heads คือเอาข้อมูลผ่านการรวมรวบรวมและเตรียมกับข้อมูลที่ได้จากขั้นตอนก่อนหน้า ในการแยกแยะและกำหนดส่วนของภาพที่แตกต่างกัน
 - o losses คือการสูญเสียประกอบด้วยฟังก์ชันการสูญเสียต่างๆ
 - o task_modules คือจัดเตรียมโมดูลสำหรับงานตรวจหา เช่น ตัวกำหนด ตัวสุมตัวอย่าง ตัวเข้ารหัสกล่อง และตัวสร้างก่อนหน้า
 - o layers คือชั้นการประมวลผลสามารถเป็นอย่างต่อเนื่องหรือซ้อนทับกันได้เพื่อสร้างโครงสร้างที่ซับซ้อนและมีความสามารถทางการเรียนรู้ที่สูงขึ้น
 - 5) engine เป็นส่วนประกอบสำหรับรันไทม์คอมพิวเตอร์
 - 6) evaluation คือตัวชี้วัดที่แตกต่างกันสำหรับการประเมินประสิทธิภาพของแบบจำลอง
 - 7) visualization เป็นการแสดงผลการตรวจจับ

MMCV เป็นไลบรารีที่พัฒนาโดย OpenMMLab เพื่อสนับสนุนการพัฒนาโมเดลทางคอมพิวเตอร์ที่เกี่ยวข้องกับปัญหาวิทยาศาสตร์การมองเห็น (computer vision) และการประมวลผลภาพ มีเป้าหมายในการทำให้ง่ายต่อการพัฒนาโมเดลด้วยการจัดการและปรับแต่งเครื่องมือต่างๆ ที่ใช้ในการพัฒนาและการทดสอบโมเดล โดยรวมไปถึงฟังก์ชันที่มีประสิทธิภาพสำหรับการประมวลผลภาพเช่นการอ่านและเขียนภาพจากและไปยังไฟล์ที่ได้รับการรับรอง การปรับขนาดภาพ การจัดการกับสี การสร้างตัวอย่างสำหรับการฝึกสอน และอื่น ๆ มีการสนับสนุนสำหรับการพัฒนาโมเดลแบบหลายเฟรมเวิร์ก (framework) อย่างเช่น PyTorch, TensorFlow, MxNet ซึ่งทำให้เป็นที่นิยม

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในวงกว้างสำหรับผู้พัฒนาโมเดลทางคอมพิวเตอร์ในสาขาวิทยาศาสตร์การมองเห็นและการประมวลผลภาพ MMCV มีการรองรับโครงการวิจัยต่าง ๆ รวมถึง MMDetection อีกด้วย

MMEEngine (OpenMMLab Model Management Engine) เป็นส่วนหนึ่งของ OpenMMLab ที่ใช้ในการจัดการและบริหารจัดการโมเดลทางคอมพิวเตอร์ที่ถูกพัฒนาขึ้นเพื่อ งานวิจัยและพัฒนาด้านการมองเห็นคอมพิวเตอร์ (computer vision) และการประมวลผลภาพ

MMEEngine เป็นส่วนสำคัญในการจัดการทรัพยากรที่เกี่ยวข้องกับโมเดล เช่น การจัดการกับข้อมูล ต้นฉบับ (dataset) การเตรียมข้อมูล (data preprocessing) การเลือกและเตรียมโมเดล (model selection and preparation) การฝึกสอนโมเดล (model training) และการประเมินผลลัพธ์ (evaluation)

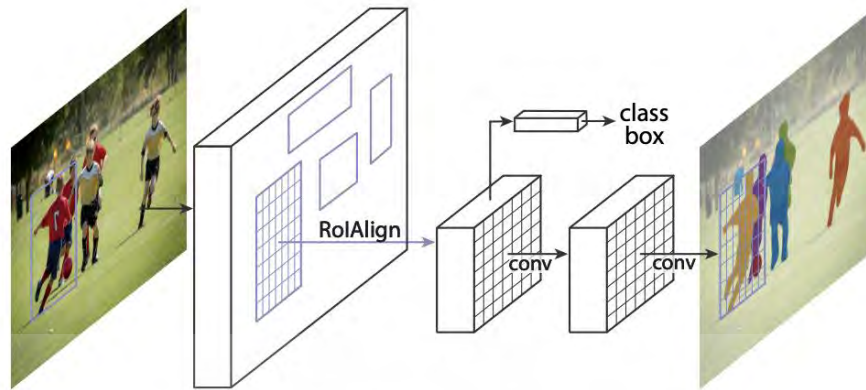
MMEEngine มีเครื่องมือและฟังก์ชันที่ช่วยในกระบวนการทำงานเหล่านี้ เช่น ระบบการ กำหนดค่า (configuration system) ที่ใช้ในการกำหนดค่าพารามิเตอร์ของโมเดล การจัดการกับ ตัวอย่างข้อมูล (data loader) เพื่อการโหลดและจัดรูปแบบข้อมูลต้นฉบับ ฟังก์ชันสำหรับฝึกสอน โมเดล (trainer functions) เพื่อการปรับแต่งและการจัดการขั้นตอนในการฝึกสอน และฟังก์ชัน สำหรับการประเมินผลลัพธ์ (evaluator functions) เพื่อการวัดประสิทธิภาพและความแม่นยำของ โมเดลที่ฝึกสอนแล้ว

2.6 เมธอดที่ใช้ในการทดลอง

2.6.1 Mask R-CNN

Mask R-CNN คือเฟรมเวิร์ก (framework) ที่ใช้สำหรับการแบ่งส่วนวัตถุในภาพ (instance segmentation) พัฒนาขึ้นจากแบบจำลอง Faster R-CNN โดยเพิ่มส่วนการสร้างแมสก์ (mask) สำหรับแต่ละวัตถุที่ตรวจจับที่สร้างขึ้นพร้อมกับ branch ที่มีอยู่แล้วสำหรับการรู้จักกล่องขอบเขตของ วัตถุ (bounding box) วิธีการนี้สามารถตรวจจับวัตถุในภาพได้อย่างมีประสิทธิภาพพร้อมกับสร้าง แมสก์การแบ่งส่วนที่มีคุณภาพสูงสำหรับแต่ละอินสแตนซ์ Mask R-CNN เป็นวิธีการที่เรียบง่ายในการ ฝึกและเพิ่มโอเวอร์เฮดเพียงเล็กน้อยเมื่อเทียบกับ Faster R-CNN ซึ่งสามารถทำงานได้ที่อัตราเฟรมสูง ถึง 5 เฟรมต่อวินาที วิธีการนี้ยังสามารถนำไปใช้กับงานอื่น ๆ เช่นการประมาณการทำมือของมนุษย์ ในกรอบการทำงานเดียวกัน และได้รับผลลัพธ์ที่ดีในการแข่งขัน COCO เช่น การแบ่งส่วนอินสแตนซ์ การตรวจจับวัตถุด้วยกล่องขอบเขต และการตรวจจับจุดสำคัญบุคคล โดย Mask R-CNN มีผลลัพธ์ที่ดีกว่าแบบจำลองเดียวที่มีอยู่ในงานต่าง ๆ และเป็นผู้ชนะในการแข่งขัน COCO ปี 2016

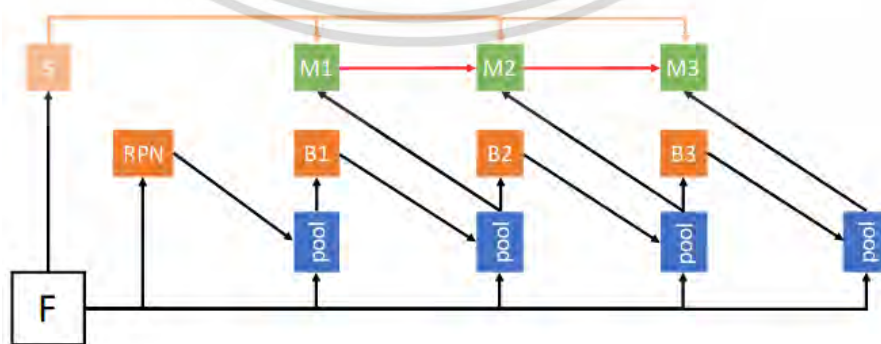
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 Mask R-CNN framework สำหรับ instance segmentation

2.6.2 Hybrid Task Cascade (HTC)

Hybrid Task Cascade (HTC) คือเฟรมเวิร์ก (framework) ที่ออกแบบมาเพื่อแก้ปัญหาการตรวจจับวัตถุและการกำหนดตำแหน่งของวัตถุเชิงตัวอย่าง โดยวิธีการใช้ Cascade R-CNN และ Mask R-CNN ร่วมกัน ใช้สำหรับการแบ่งส่วนวัตถุในภาพ (instance segmentation) วิธีการทำงานของ HTC คือการปรับปรุงระบบแบบ cascade ในการทำงานของการตรวจจับวัตถุและการแบ่งส่วนวัตถุให้เป็นระบบที่เชื่อมโยงกันและทำงานพร้อมกันในหลายขั้นตอน นอกจากนี้ HTC ยังเพิ่มส่วนของการใช้สถาปัตยกรรม Fully Convolutional Network (FCN) เพื่อให้มีความเข้าใจบริบทเชิงพื้นที่ (spatial context) ซึ่งช่วยในการแยกวัตถุที่ยากและพื้นหลังที่ซับซ้อน โดยรวม HTC สามารถเรียนรู้ลักษณะของวัตถุให้แยกแยะได้อย่างต่อเนื่องและรวมลักษณะที่เสริมกันในแต่ละขั้นตอน เฟรมเวิร์กนี้สามารถเรียนรู้คุณสมบัติการแบ่งแยกได้มากขึ้นเรื่อย ๆ ในขณะที่รวมคุณสมบัติเสริมเข้าด้วยกันในแต่ละขั้นตอน อีกทั้งระบบโดยรวมได้รับคะแนน Mask AP 48.6 ใน test-challenge split จึงได้รับอันดับที่ 1 ในการแข่งขัน COCO ปี 2018



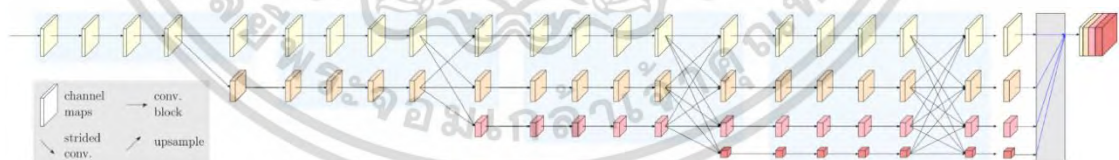
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ส่วนตัวเท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.3 Global Context Network (GCNet)

Global Context Network (GCNet) พัฒนามาจาก Non-Local Network (NLNet) เป็นแบบจำลองที่ถูกออกแบบมาเพื่อการจัดโมเดลบริบททั่วโลก (global context modeling) ในงานการรู้จำต่าง ๆ เครื่องข่ายนี้มีเป้าหมายในการเพิ่มความเข้าใจและการรับรู้เกี่ยวกับ global context ของภาพให้มีน้ำหนักเบาและสามารถจับความสัมพันธ์ระหว่างวัตถุและพื้นที่ในภาพได้อย่างมีประสิทธิภาพ ซึ่งเป็นการพิจารณาข้อมูลที่เกี่ยวข้องกับวัตถุที่เราสนใจในภาพโดยไม่จำกัดเพียงตำแหน่งเดียว โดย GCNet ใช้โครงสร้างบล็อกบริบททั่วโลก (Global Context block) ซึ่งช่วยในการสร้างความเข้าใจเกี่ยวกับความสัมพันธ์ระยะไกลในภาพ ผ่านการรวมข้อมูลจากตำแหน่งต่าง ๆ ภายในภาพเพื่อให้เครือข่ายสามารถเรียนรู้และจัดการ global context ได้อย่างมีประสิทธิภาพ ด้วยคุณสมบัตินี้ GCNet มักจะมีประสิทธิภาพที่ดีกว่าโมเดลอื่นในการรับรู้และการทำนายในงานต่าง ๆ เกี่ยวกับการรู้จำและการจัดกลุ่มวัตถุในภาพ

2.6.4 High-Resolution Net (HRNet)

High-Resolution Net (HRNet) เป็นแบบจำลองที่ออกแบบมาเพื่อการรับรู้ที่มีความละเอียดสูงในงานต่าง ๆ เช่นการรู้จำตำแหน่งของมนุษย์ (Human Pose Estimation) หรือการตรวจจับวัตถุ (Object Detection) HRNet มีความเฉพาะเจาะจงในการรักษาความละเอียดสูงในระหว่างกระบวนการที่ไม่ต้องลดระดับละเอียดของข้อมูล ในการออกแบบ HRNet ใช้ Cascade Connection เพื่อรักษารายละเอียดที่มีประสิทธิภาพสูง โดยในแต่ละระดับของเครือข่ายจะรับข้อมูลในระดับละเอียดที่สูงและใช้ข้อมูลจากระดับก่อนหน้าในการปรับปรุงและเพิ่มความแม่นยำของการรับรู้ในแต่ละระดับ ด้วยคุณสมบัตินี้ HRNet มักจะมีประสิทธิภาพที่ดีในการรับรู้และการตรวจจับที่มีความละเอียดสูงในงานต่าง ๆ และเป็นที่ยอมรับในการศึกษาและการประยุกต์ใช้งาน

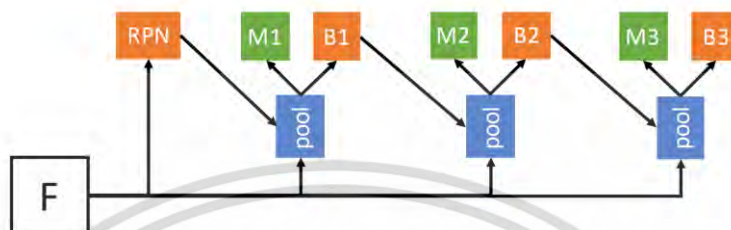


รูปที่ 2.11 High-Resolution Net (HRNet) อย่างง่าย

2.6.5 Cascade Mask R-CNN

Cascade Mask R-CNN เป็นการรวมแบบจำลองการตรวจจับวัตถุแบบ Cascade R-CNN และการกำหนดตำแหน่งของวัตถุแบบ Mask R-CNN เข้าด้วยกัน โดยเพิ่มชั้นการตรวจจับที่ฝึกฝนด้วยการกำหนดค่าความแน่นอน (IoU) ที่เพิ่มขึ้นในแต่ละขั้นตอน ในกระบวนการฝึกฝน Cascade Mask R-CNN ทุกชั้นการตรวจจับถูกฝึกฝนตามลำดับ โดยใช้ผลลัพธ์ของชั้นการตรวจจับในขั้นตอนก่อนหน้า

เป็นชุดฝึกฝนสำหรับขั้นตอนถัดไป กระบวนการนี้ช่วยปรับปรุงความแม่นยำของการตรวจจับเรื่อยๆ และลดความผิดพลาดที่เกิดจากความแน่นอนที่สูงขึ้น ทำให้ได้ผลลัพธ์การตรวจจับที่มีคุณภาพสูงขึ้น นอกจากนี้ Cascade Mask R-CNN ยังสามารถรวมการตรวจจับวัตถุและการกำหนดตำแหน่งของวัตถุเชิงตัวอย่าง (instance segmentation) ได้อีกด้วย



รูปที่ 2.12 Cascade Mask R-CNN

2.6.6 Residual Networks (ResNet)

Residual Networks (ResNet) คือแบบจำลองเครือข่ายประสาทเทียมที่มีความลึกสูงและได้รับความนิยมมากในการตรวจจับวัตถุ แบบจำลอง ResNet นำเสนอกรอบการเรียนรู้แบบ residual learning ที่ช่วยให้การฝึกฝนแบบเครือข่ายที่มีความลึกสูงกว่างานวิจัยก่อนหน้านี้ง่ายขึ้น โดยที่แต่ละชั้นในเครือข่ายถูกตั้งค่าใหม่ให้เป็นการเรียนรู้ฟังก์ชัน residual โดยอ้างอิงถึงข้อมูลเข้าของชั้นก่อนหน้า ซึ่งช่วยลดปัญหาการสูญเสียค่าเกรเดียนต์ในกระบวนการฝึกฝนและช่วยเพิ่มความแม่นยำของแบบจำลองจากความลึกที่เพิ่มขึ้นได้ แบบจำลอง ResNet ได้รับการประเมินบนชุดข้อมูล ImageNet โดยมีความลึกสูงสุดถึง 152 ชั้น ซึ่งเป็นความลึก 8 เท่าของแบบจำลอง VGG แต่มีความซับซ้อนต่ำกว่า และได้รับรางวัลอันดับที่ 1 ในงานแข่งขัน ILSVRC 2015 ในหมวดหมู่การจำแนกภาพ

2.7 การประเมินประสิทธิภาพ (Evaluate model)

การประเมินประสิทธิภาพโมเดลเป็นกระบวนการของการใช้เมตริกการประเมินต่าง ๆ เพื่อทำความเข้าใจประสิทธิภาพของโมเดลตลอดจนจุดแข็งและจุดอ่อน การประเมินโมเดลเป็นสิ่งสำคัญในการประเมินประสิทธิภาพของโมเดลในระหว่างขั้นตอนการวิจัยเบื้องต้น และยังมีบทบาทในการตรวจสอบโมเดลอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

รูปที่ 2.13 ตาราง Confusion Matrix

Confusion Matrix (เมทริกซ์ความสับสน) คือเมทริกซ์ที่สรุปประสิทธิภาพของโมเดลแมชชีน เลิร์นนิ่งในชุดข้อมูลทดสอบ มักใช้เพื่อวัดประสิทธิภาพของแบบจำลองการจำแนกประเภทและ Confusion Matrix จะแสดงผลลัพธ์ของการจำแนกหรือการทำนายเป็นตารางที่ประกอบด้วยแนวตั้ง และแนวนอนที่แทนสถานะของข้อมูลจริง (actual class) และสถานะของการทำนาย (predicted class) ภายในตาราง โดยประกอบด้วยค่าต่างๆ เช่น True Positive (TP), False Positive (FP), False Negative (FN), และ True Negative (TN)

True positives (TP) คือ ข้อมูลที่ทำนายตรงกับข้อมูลจริงในคลาสที่กำลังพิจารณา ในกรณีทำนายว่าจริง และสิ่งที่เกิดขึ้น ก็คือ จริง

True negatives (TN) คือ ข้อมูลที่ทำนายตรงกับข้อมูลจริงในคลาสที่ไม่ได้พิจารณา ในกรณีทำนายว่า ไม่จริง และสิ่งที่เกิดขึ้น ก็คือ ไม่จริง

False positives (FP) คือ ข้อมูลที่ทำนายผิดเป็นคลาสที่กำลังพิจารณา คือทำนายว่า จริง แต่สิ่งที่เกิดขึ้น คือ ไม่จริง

False negatives (FN) คือ ข้อมูลที่ทำนายผิดเป็นคลาสที่ไม่ได้พิจารณา คือทำนายว่าไม่จริง แต่สิ่งที่เกิดขึ้น คือ จริง

การใช้ Confusion Matrix มาคำนวณ การประเมินประสิทธิภาพของการทำนายในรูปแบบค่าต่าง ๆ ได้หลายค่า ได้แก่

ความถูกต้อง (Accuracy) คือ ความถูกต้องที่เราทำนายได้ตรงกับสิ่งที่เกิดขึ้นจริง

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความแม่นยำ (Precision) คือ จำนวน TP หารด้วยจำนวนผลลัพธ์ positive ที่โมเดลตรวจพบ ความแม่นยำเป็นตัววัดการประเมินที่ถูกต้องเมื่อต้องการมั่นใจในการตรวจจับ

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

ความไว (Sensitivity หรือ Recall) เป็นตัวเลือกที่ถูกต้องของเมตริกการประเมินเมื่อต้องการเก็บผลบวกให้ได้มากที่สุด ความไวบ่งชี้สัดส่วนที่ตัวอย่างที่เป็น TP ได้รับการจำแนกอย่างถูกต้อง

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

F1 score คือ ค่าเฉลี่ยแบบ harmonic mean ระหว่าง precision และ recall จุดประสงค์ของการสร้าง F1 ขึ้นมา คือ เพื่อเป็น single metric ที่วัดความสามารถของโมเดล

$$\text{f1score} = \frac{2(\text{recall} \times \text{precision})}{(\text{recall} + \text{precision})}$$

ค่าเฉลี่ยความแม่นยำ (mAP) เป็นเมตริกสำหรับการประเมินแบบจลของการตรวจจับวัตถุ เช่น Fast R-CNN, YOLO และ Mask R-CNN mAP คือค่าเฉลี่ยของความแม่นยำเฉลี่ย (AP) ของแต่ละคลาส mAP ถูกกำหนดโดยการคำนวณ AP สำหรับแต่ละคลาส จากนั้นหาค่าเฉลี่ยของ AP เหล่านี้ mAP คำนึงถึงทั้ง FP และ FN และสะท้อนถึงการแลกเปลี่ยนระหว่าง Precision และ Recall

$$\text{mAP} = \frac{1}{n} \sum_{k=1}^{k=n} \text{AP}_k$$

COCO Evaluator

COCO Evaluator เป็นเครื่องมือที่ใช้ในการวัดประสิทธิภาพของระบบตรวจจับวัตถุ (object detection) และระบบติดตามวัตถุ (object tracking) ในงานคอมพิวเตอร์วิทัศน์ (computer vision) โดยใช้ค่าวัดหลักคือ COCO metrics ซึ่งเป็นเกณฑ์ที่กำหนดโดยภาควิชาวิจัยและนักพัฒนา COCO dataset ซึ่งมักถูกใช้ในงานวิจัยและพัฒนาระบบที่เกี่ยวข้องกับการตรวจจับวัตถุและติดตามวัตถุ เพื่อให้สามารถวัดและเปรียบเทียบประสิทธิภาพของระบบได้อย่างเป็นทางการ COCO Evaluator ใช้ COCO metrics เพื่อคำนวณค่า precision, recall, average precision (AP) และ mean average precision (mAP) ที่เป็นตัววัดประสิทธิภาพของระบบตรวจจับหรือติดตามวัตถุ ซึ่งเอกสารนี้จะให้คำแนะนำตามความแม่นยำในการระบุตำแหน่งและหมวดหมู่ของวัตถุในภาพใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมตริกการประเมินการตรวจหา (detection evaluation metrics) ที่พัฒนาโดย COCO นั้น แบ่งเป็นค่า Average Precision (AP) 6 เมตริก และค่า Average Recall (AR) 6 เมตริก โดยทั้ง 12 เมตริกนี้ใช้สำหรับระบุลักษณะการท างานของเครื่องตรวจจับวัตถุบน COCO

AP และ AR จะเฉลี่ยผ่านค่า Intersection over Union (IoU) หลายค่า ใช้เกณฑ์ 10 IoU ที่ .50:.05:.95 โดย AP เป็นค่าเฉลี่ยของทุกหมวดหมู่ หรือเรียกว่า ค่าเฉลี่ยความเที่ยงตรง (mAP) ส่วน AR คือการเรียกคืนสูงสุดตามจำนวนการตรวจจับที่แน่นอนต่อภาพ เฉลี่ยตามหมวดหมู่และ IoU โดยเมตริกการประเมินสำหรับการตรวจจับด้วยกล่องขอบ (bounding boxes) และมาสก์การแบ่งส่วน (segmentation masks) จะเหมือนกันทุกประการ ยกเว้นการคำนวณ IoU

Average Precision (AP):	
AP	% AP at IoU=.50:.05:.95 (primary challenge metric)
$AP^{IoU=.50}$	% AP at IoU=.50 (PASCAL VOC metric)
$AP^{IoU=.75}$	% AP at IoU=.75 (strict metric)
AP Across Scales:	
AP^{small}	% AP for small objects: area < 32^2
AP^{medium}	% AP for medium objects: $32^2 < \text{area} < 96^2$
AP^{large}	% AP for large objects: area > 96^2
Average Recall (AR):	
$AR^{max=1}$	% AR given 1 detection per image
$AR^{max=10}$	% AR given 10 detections per image
$AR^{max=100}$	% AR given 100 detections per image
AR Across Scales:	
AR^{small}	% AR for small objects: area < 32^2
AR^{medium}	% AR for medium objects: $32^2 < \text{area} < 96^2$
AR^{large}	% AR for large objects: area > 96^2

รูปที่ 2.14 detection evaluation metrics ที่พัฒนาโดย COCO

2.8 งานวิจัยที่เกี่ยวข้อง

จากงานวิจัยของ Mahboub Parhizkar, Majid Amirfakhrian (2565) เรื่อง Car detection and damage segmentation in the real scene using a deep learning approach งานวิจัยนี้ได้พัฒนาวิธีการตรวจจับความเสียหายของรถยนต์โดยอัตโนมัติโดยอ้างอิงจาก cascade Convolutional Neural Network (CNN) โดยเป้าหมายหลักของการวิจัยนี้คือขั้นแรกในการระบุตัวรถยนต์ในภาพถ่ายที่ถ่ายหลังจากเกิดอุบัติเหตุ จากนั้นจึงระบุตำแหน่งของความเสียหายใด ๆ เช่น ไฟและกระจกที่แตก การบิดเบี้ยวที่ด้านหน้า ด้านหลัง และด้านข้างเหนือพื้นผิวด้านนอกของรถ สิ่งสำคัญคือต้องกล่าวว่าความสำเร็จของเทคนิคที่เลือกขึ้นอยู่กับปัจจัยสำคัญบางประการ เช่น การเปลี่ยนแปลงของพื้นที่ที่เสียหาย คุณภาพของภาพ จำนวนภาพการฝึกและการแบ่งประเภทข้อมูล เป็นต้น วิธีการที่น่าเสนอนี้ใช้เทคนิคการจำแนกเชิงจุดภาพ (pixel-based) โดยใช้ CNNs ที่แตกต่างกันสอซุด เพื่อกำหนดพื้นที่ความเสียหายจากบริเวณภายนอกของรถยนต์จากภาพที่ได้มา ผลจากการไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทดลองแสดงให้เห็นว่าวิธีการที่ได้นำเสนอให้ประสิทธิภาพสูงเมื่อเปรียบเทียบกับวิธีการขั้นสูงอื่น ๆ ที่มีในปัจจุบัน

จากงานวิจัยของ Atharva Shirode, Tejas Rathod, Parth Wanjari, Aparna Halbe (2565) เรื่อง Car Damage Detection and Assessment Using CNN งานวิจัยนี้ได้ทำการเปรียบเทียบ 4 อัลกอริทึม ได้แก่ VGG16, VGG19, Resnet50, Inception V3 และได้แนะนำวิธีการสองแบบที่มีความแม่นยำมากที่สุดจากการทดลอง ซึ่งเป็นของ Convolutional Neural Network (CNN) ประกอบด้วย VGG16 ใช้เพื่อตรวจจับความเสียหายบนรถ ตำแหน่งของความเสียหายและความรุนแรง เนื่องจากมีประสิทธิภาพในแง่ของความสามารถในการตรวจจับวัตถุ (รถยนต์) และการจำแนกประเภท (ความรุนแรงและตำแหน่ง) และในส่วนของที่สองที่ใช้งานคือ Mask R-CNN ได้ถูกนำมาใช้เพื่อระบุความเสียหายของรถ รูปภาพถูกใส่คำอธิบายประกอบโดยใช้เครื่องมือใส่คำอธิบายประกอบ และสร้างไฟล์ JSON สำหรับป้ายกำกับหรือเฉลยและฝึกสำหรับรูปภาพที่มีคำอธิบายประกอบ ไฟล์ JSON นี้ใช้เพื่อฝึกโมเดล R-CNN โดยทั้งสองโมเดลให้แนวคิดที่เป็นธรรมชาติเกี่ยวกับความเสียหายที่เกิดขึ้นกับรถยนต์ ซึ่งช่วยให้บริษัทประกันภัยสามารถดำเนินการเคลมประกันต่อไปได้โดยไม่เสียเวลาและทรัพยากรในการตรวจสอบด้วยตนเอง

จากงานวิจัยของ Atharva Shirode, Tejas Rathod, Parth Wanjari, Aparna Halbe (2564) เรื่อง Car Damage Identification and Categorization Using Various Transfer Learning Models งานวิจัยนี้ได้ทำการรวบรวมรูปภาพจากทั้งการค้นหาโดย Google และที่เก็บ GitHub จากนั้นได้ฝึกฝนโดยใช้ชุดข้อมูลเพื่อคาดการณ์ความเสียหายเปรียบเทียบรูปแบบการเรียนรู้การถ่ายโอนที่มีอยู่ใน Keras ซึ่งได้ตัดสินใจทดลองกับ 6 อัลกอริทึม ได้แก่ Inception V3, Xception, VGG16, VGG19, ResNet50 และ MobileNet และได้พบว่า MobileNet มีความแม่นยำมากกว่าและความเร็วในการฝึกก็น้อยกว่าเมื่อเทียบกับรุ่นอื่น ๆ

จากงานวิจัยของ Kitsuchart Pasupa, Phongsathorn Kittiworapanya, Napasin Hongngern & Kuntpong Woraratpanya (2564) เรื่อง Evaluation of deep learning algorithms for semantic segmentation of car parts งานวิจัยนี้ได้ประเมินและเปรียบเทียบ อัลกอริทึมการเรียนรู้เชิงลึก 5 แบบสำหรับการแบ่งส่วนความหมายของชิ้นส่วนรถยนต์ อัลกอริทึมอ้างอิงพื้นฐานคือ Mask R-CNN และอัลกอริทึมอื่นๆ คือ HTC, CBNet, PANet และ GCNet และพบว่า HTC ที่มี ResNet-50 เป็นอัลกอริทึมที่ดีที่สุดสำหรับการแบ่งส่วนอินสแตนซ์ (Instance Segmentation) ของรถยนต์ประเภทต่าง ๆ รองลงมาคือ Mask R-CNN และ GCNet ทั้งในการตรวจจับวัตถุและการแบ่งกลุ่มความหมายในสภาพอากาศปกติ นอกจากนี้ยังได้ประเมินความแข็งแกร่ง (robust) ของอัลกอริทึมในสภาพแวดล้อมจริงและสภาพแสง โดยจำลองสภาพที่จะเกิดขึ้นในสนาม เมื่อถ่ายภาพโดยใช้สมาร์ทโฟน GCNet เป็นรุ่นที่แข็งแกร่งที่สุด เนื่องจากได้รับประสิทธิภาพ

เอกสารนี้สูงสุดในภาพรวมและในสภาวะจริงงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากงานวิจัยของ Daniel Widjojo, Endang Setyati, Yosi Kristian (2564) เรื่อง Integrated Deep Learning System for Car Damage Detection and Classification Using Deep Transfer Learning งานวิจัยนี้มีวัตถุประสงค์เพื่อเร่งกระบวนการเคลมประกันภัยรถยนต์โดยจัดทำระบบการเรียนรู้เชิงลึกแบบบูรณาการ สามารถแบ่งกลุ่มความเสียหายของรถยนต์ ตรวจสอบชิ้นส่วนที่เสียหายและจำแนกระดับความรุนแรงของความเสียหายของรถยนต์ ชุดข้อมูลประกอบด้วย 3 ส่วน ได้แก่ ชุดข้อมูลการแบ่งส่วนที่เสียหาย ชุดข้อมูลการแบ่งส่วนชิ้นส่วนรถยนต์ และชุดข้อมูลการจัดประเภทความเสียหายของรถยนต์ แบบจำลองที่ใช้ในบทความนี้ได้รับการฝึกฝนโดยใช้วิธีการเรียนรู้การถ่ายโอนเชิงลึกเพื่อประหยัดเวลาและทรัพยากร แบบจำลองสำหรับงานการแบ่งกลุ่มคือ Mask R-CNN ในขณะที่งานการจัดหมวดหมู่ได้เปรียบเทียบกับ EfficientNet และ MobileNetV2 มีการพยายามเพิ่มคะแนน F1 โดยเพิ่ม CNN แบบง่ายที่เชื่อมติดกันและใช้ผลลัพธ์จากการแบ่งส่วนความเสียหายเป็นอินพุตเพิ่มเติมสำหรับแบบจำลองการจัดประเภทการทดลองแสดงให้เห็นว่า Mask R-CNN สามารถตรวจจับชิ้นส่วนรถยนต์ที่เสียหายได้ การปรับเปลี่ยน CNN อย่างง่ายที่เชื่อมโยงกันบนโมเดลการจัดหมวดหมู่สามารถเพิ่มโมเดลให้มีความแตกต่างโดยเฉลี่ยสูงถึง 9% สำหรับคะแนน F1 โมเดลการจัดหมวดหมู่ที่ดีที่สุดคือ MobileNetV2 ซึ่งใช้การแก้ไข CNN อย่างง่ายด้วยคะแนน F1 สูงถึง 91%

จากงานวิจัยของ Qinghui Zhang, Xianing Chang, Shanfeng Bian (2563) เรื่อง Vehicle-Damage-Detection Segmentation Algorithm Based on Improved Mask RCNN งานวิจัยนี้มีจุดประสงค์เพื่อแก้ไขปัญหาการชดเชยอุบัติเหตุจราจรอย่างรวดเร็ว ได้สร้างอัลกอริทึมการแบ่งส่วนการตรวจจับความเสียหายของยานพาหนะตามการเรียนรู้การถ่ายโอนและ Mask RCNN ขั้นแรก การทดลองจะเก็บภาพความเสียหายของรถยนต์สำหรับการประมวลผลล่วงหน้า และใช้ Labelme เพื่อสร้างฉลากชุดข้อมูล ซึ่งแบ่งออกเป็นชุดฝึกอบรมและชุดทดสอบ มีการปรับ residual network (ResNet) ให้เหมาะสม และการแยกคุณลักษณะจะดำเนินการร่วมกับ Feature Pyramid Network (FPN) ผลลัพธ์ของการฝึกอบรมชุดข้อมูลที่สร้างขึ้นเองโดยเฉพาะและการทดสอบแสดงให้เห็นว่า Mask RCNN ที่ปรับปรุงแล้วมีค่า Average Precision (AP) ที่ดีกว่า ความแม่นยำในการตรวจจับและความแม่นยำในการ mask

จากงานวิจัยของ Ranjodh Singh, Meghna P. Ayyar, Tata Venkata Sri Pavan (2562) เรื่อง Automating Car Insurance Claims Using Deep Learning Techniques งานวิจัยนี้ได้จัดทำระบบถ่ายภาพรถที่เสียหายเป็นข้อมูลเข้าและให้ข้อมูลที่เกี่ยวข้อง เช่น ชิ้นส่วนที่เสียหาย และประเมินขอบเขตของความเสียหาย (ไม่มีความเสียหาย เล็กน้อยหรือรุนแรง) ในแต่ละส่วน เพื่อทำหน้าที่ในการประมาณการค่าใช้จ่ายในการซ่อมแซมซึ่งจะใช้ในการตัดสินใจจำนวนเงินค่าสินไหมทดแทน ได้ทดลองกับโมเดลการแบ่งกลุ่มอินสแตนซ์ที่เป็นที่นิยม เช่น Mask R-CNN, PANet และกลุ่มของทั้งสองนี้พร้อมกับการเรียนรู้การถ่ายโอนที่ใช้เครือข่าย VGG16 เพื่อทำงานที่แตกต่างกันของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแปลและตรวจจับชิ้นส่วนและความเสียหายต่าง ๆ ที่พบในรถ นอกจากนี้ ระบบที่นำเสนออย่างได้คะแนน mAP ที่ดีสำหรับการแปลชิ้นส่วนและการแปลความเสียหาย (0.38 และ 0.40 ตามลำดับ)

จากงานวิจัยของ Kalpesh Patil, Mandar Kulkarni, Anand Sriraman, Shirish Karande (2560) เรื่อง Deep Learning Based Car Damage Classification งานวิจัยนี้ได้ทำการสร้างชุดข้อมูลโดยรวบรวมรูปภาพจากเว็บและใส่คำอธิบายประกอบด้วยตนเอง และทดลองด้วยเทคนิคการเรียนรู้เชิงลึกหลายอย่าง เช่น การฝึกอบรม CNN จากการเริ่มต้นแบบสุ่ม (random initialization) การฝึกอบรมล่วงหน้าโดยใช้ Convolution Autoencoder ตามด้วยการปรับแต่งอย่างละเอียด ภายใต้การดูแลและการถ่ายโอนการเรียนรู้ (transfer learning) ซึ่งพบว่าการเรียนรู้การถ่ายโอนทำงานได้ดีที่สุด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินงานวิจัย

งานวิจัยนี้จัดทำเพื่อศึกษาการทำงานและการพัฒนาโปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุ เพื่อนำไปพัฒนาเป็นโปรแกรมต้นแบบและโปรแกรมเสริมการทำงาน ซึ่งในการดำเนินงานวิจัยมีลำดับดังนี้

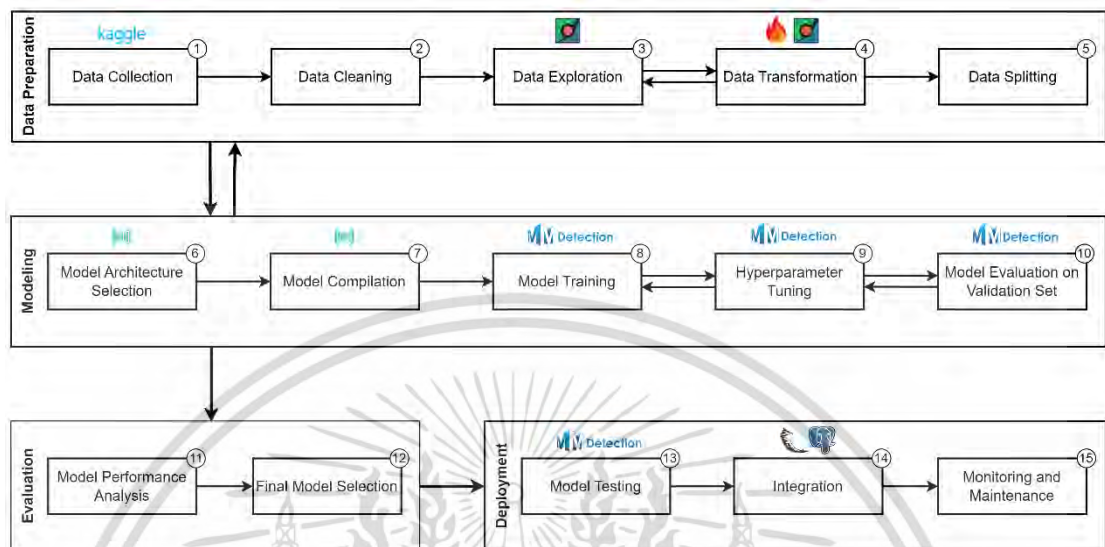
- 3.1 การออกแบบการทดลอง
- 3.2 การศึกษาและสำรวจข้อมูลเบื้องต้น
 - 3.2.1 การเก็บรวบรวมตัวอย่างข้อมูลภาพรถยนต์
 - 3.2.2 การเตรียมข้อมูลสำหรับการฝึกสอน
- 3.3 การพัฒนาแบบจำลองโดยใช้การเรียนรู้ของเครื่อง
 - 3.3.1 การพัฒนาแบบจำลองโดยใช้การเรียนรู้ของเครื่อง
 - 3.3.2 การทดสอบประสิทธิภาพของแบบจำลอง
- 3.4 การพัฒนาส่วนติดต่อกับผู้ใช้
 - 3.4.1 โครงสร้างของโปรแกรมเบื้องต้น
 - 3.4.2 การออกแบบฐานข้อมูลของโปรแกรม

3.1 ขั้นตอนการดำเนินงาน

ในงานวิจัยครั้งนี้เราได้ทำการรวบรวมรูปภาพจากแหล่งต่าง ๆ ในอินเทอร์เน็ต จากนั้นก็เลือกรูปภาพที่เหมาะสมกับการนำมาใช้ เมื่อได้ภาพที่ต้องการแล้วจึงนำมาแปลงให้เป็นไฟล์สกุล jpg แล้วสร้างป้ายกำกับ แปลงให้อยู่ใน coco format โดยใช้ datatorch และแบ่งข้อมูลออกเป็น train set, valid set และ test set เพื่อนำไปใช้ในการฝึกฝนและวัดประสิทธิภาพของแบบจำลอง โดยแบบจำลองที่นำมาใช้ในครั้งนี้เป็นแบบจำลองจากโครงการ OpenMMLab ที่มีความสามารถในการตรวจจับวัตถุและแบ่งส่วนความหมาย ซึ่งจะเลือกแบบจำลองที่น่าสนใจจากข้อมูลของวิจัยอื่นที่ได้ศึกษาและแบบจำลองที่มี mAP สูง ที่ใช้ฝึกฝนร่วมกับชุดข้อมูลที่เราสร้างมาได้ จากนั้นเราจะทำการฝึกฝนแบบจำลองที่เลือกมา โดยจะมีการปรับค่าปรับพารามิเตอร์ เมตริกการประเมิน อัตราการเรียนรู้ (learning rate) ขนาดแบทช์ (batch size) เพื่อให้ได้ผลการประเมินที่ดีขึ้น โดยในอัตราการเรียนรู้ 0.1, 0.01, 0.001, 0.0025 (default) และพบว่า 0.0025 ได้ผลการประเมินใน valid set ดีที่สุด เมื่อได้ผลการประเมินแต่ละแบบจำลองและเราได้นำมาเปรียบเทียบโดยการนำมาจัดอันดับ (rank) และเปรียบเทียบผลการทำนายที่ได้จากการ mask ภาพใน test set หาแบบจำลองที่เหมาะสมที่สุดเพื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำมาใช้ในส่วนของการพัฒนาเว็บแอปพลิเคชันต่อไป โดยจะทำขั้นตอนเหล่านี้ในทุก ๆ การทดลองในฟังก์ชันข้อมูล ซึ่งสามารถดูรายละเอียดเพิ่มเติมในหัวข้อถัด ๆ ไป



รูปที่ 3.1 ขั้นตอนการดำเนินงาน

3.2 การศึกษาและสำรวจข้อมูลเบื้องต้น

3.2.1 การเก็บรวบรวมตัวอย่างข้อมูลรูปภาพรถยนต์

รูปภาพที่นำมาเป็นชุดข้อมูลเพื่อใช้ในการพัฒนาและทดลองแบบจำลองนั้น เป็นภาพรถยนต์ที่ไม่มีความเสียหายและมีความเสียหายในลักษณะต่าง ๆ จากทั้ง Kaggle และช่องทางอื่นในอินเทอร์เน็ตที่อนุญาตให้นำมาใช้งานได้ เพื่อความสะดวกในการจัดการข้อมูลได้มีการใช้ชุดข้อมูลจาก COCO (Common Objects in Context) ของ Microsoft ซึ่งเป็นชุดข้อมูลรูปภาพที่ออกแบบมาสำหรับการตรวจจับวัตถุ การแบ่งส่วนวัตถุ และการสร้างคำบรรยาย โดยชุดข้อมูลที่ได้จะเป็นรูปภาพและ COCO format ที่มีการจัดเก็บคำอธิบายประกอบในรูปแบบ JSON

ในส่วนของชุดข้อมูลที่ใช้ฝึกแบบจำลองตรวจจับร่องรอยความเสียหายรถยนต์ จะนำภาพรถยนต์ที่มีความเสียหายทั้งหมด 268 ภาพ โดยแบ่งเป็นภาพจาก Coco Car Damage Detection Dataset ซึ่งเราได้ทำการเลือกภาพรถยนต์มา 8 ภาพ และภาพจากแหล่งอื่นอีก 260 ภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 ตัวอย่างรูปภาพรถยนต์จาก Coco Car Damage Detection Dataset

ในส่วนของคุณข้อมูลที่ใช้ฝึกแบบจำลองแบ่งส่วนพื้นที่ของรถยนต์ จะนำภาพรถยนต์ทั้งที่มีความเสียหายและไม่มีความเสียหายทั้งหมด 540 ภาพ โดยแบ่งเป็นภาพจาก Kaggle Dataset ซึ่งเราได้ทำการเลือกภาพรถยนต์มา 302 ภาพ ซึ่งมีรถยนต์ทั้งหมด 38 คันในหลาย ๆ มุมมอง จากนั้นเพื่อเพิ่มความหลากหลายของภาพและปรับปรุงประสิทธิภาพให้แบบจำลองจึงได้เพิ่มภาพรถยนต์ที่มีรูปร่างใกล้เคียงกับชุดข้อมูลก่อนหน้าที่ถูกถ่ายจากสภาพแวดล้อมด้านนอกจาก Stanford Cars Dataset ทั้งหมด 100 ภาพ และใช้ภาพบางส่วนจาก Dataset ภาพรถยนต์ที่เสียหาย 138 ภาพ



รูปที่ 3.3 ตัวอย่างรูปภาพรถยนต์จาก Stanford Cars Dataset

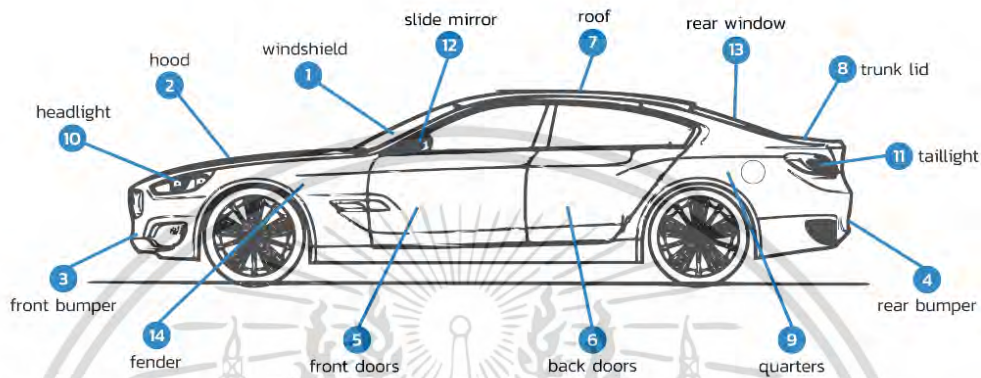
3.2.2 การเตรียมข้อมูลสำหรับการฝึกสอน

หลังจากที่ได้รวบรวมรูปภาพรถยนต์ซึ่งยังเป็นข้อมูลดิบ เพื่อนำมาใช้เป็นชุดข้อมูลแล้วต้องมีการเตรียมข้อมูลก่อนจึงจะสามารถนำเข้าฝึกอบรมได้ โดยรูปภาพที่นำมาใช้จะถูกแปลงเป็นรูปแบบไฟล์ JPG จากนั้นเราจะใส่คำอธิบายประกอบรูปภาพให้อยู่ใน COCO format สำหรับแบบจำลองโดยใช้ Dataloader ซึ่งเป็นเครื่องมือใส่คำอธิบายประกอบข้อมูลการเรียนรู้ของเครื่อง

ชุดข้อมูลที่ใช้ในการแบ่งส่วนพื้นที่ของรถยนต์ได้แบ่งรูปภาพที่ได้ออกเป็นสามชุด ได้แก่ ชุด

รูปภาพสำหรับการฝึกอบรม (Train set) 432 ภาพ คิดเป็น 80% ชุดรูปภาพสำหรับการทดสอบ (Test set) 54 ภาพ คิดเป็น 10% และชุดรูปภาพสำหรับการตรวจสอบความถูกต้อง (Validation set) 54 ภาพ คิดเป็น 10% และชุดรูปภาพสำหรับการตรวจสอบความถูกต้อง (Validation set) 54 ภาพ คิดเป็น 10% และชุดรูปภาพสำหรับการตรวจสอบความถูกต้อง (Validation set) 54 ภาพ คิดเป็น 10%

set) 54 ภาพ คิดเป็น 10% โดยได้ทำการแบ่งส่วนพื้นที่รถยนต์เป็น 14 ส่วน ได้แก่ กระจกหน้ารถ (windshield), กระจังหน้ารถ (hood), กันชนหน้า (front bumper), กันชนหลัง (rear bumper), ประตูหน้า (front doors), ประตูหลัง (back doors), หลังคา (roof), ฝากระจังหลัง (trunk lid), บังโคลนหลังหรือแผงที่เปิดล้อหลัง (quarters), ไฟหน้ารถ (headlight), ไฟท้ายรถ (taillight), กระจกมองข้าง (slide mirror), กระจกหลัง (rear window), บังโคลนหน้าหรือแผงที่เปิดล้อหน้า (fender)

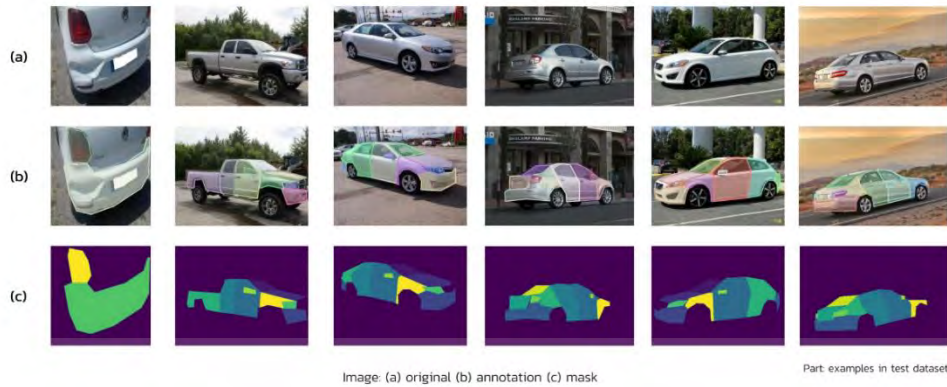


รูปที่ 3.4 ตัวอย่างการแบ่งส่วนพื้นที่ของรถยนต์

ตารางที่ 3.1 การกระจายตัวของชุดข้อมูลการแบ่งส่วนพื้นที่ของรถยนต์

Classes	Count (train)	Count (valid)	Count (test)	Total
windshield	190	28	25	243
hood	218	35	28	281
front bumper	279	40	37	356
rear bumper	303	35	42	380
front doors	355	47	47	449
back doors	289	37	36	362
roof	165	26	23	214
trunk lid	175	19	25	219
quarters	331	44	42	417
headlight	216	35	31	282
taillight	257	29	37	323
slide mirror	401	51	52	504
rear window	129	14	20	163
fender	326	46	44	416
Total	3634	486	489	4609

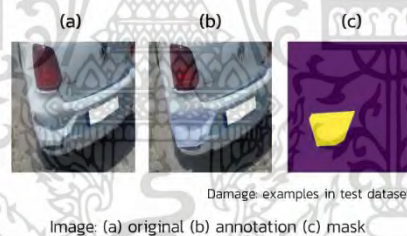
เอกสารนี้เป็นเอกสารที่งานวิจัยสำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 ตัวอย่างชุดข้อมูลการแบ่งส่วนพื้นที่ของรถยนต์ (Test set)

ชุดข้อมูลร่องรอยความเสียหายได้แบ่งรูปภาพที่ได้ออกเป็นสามชุดย่อย ได้แก่ ชุดรูปภาพสำหรับการฝึกอบรม (Train set) 192 ภาพ คิดเป็นร้อยละ 72 ชุดรูปภาพสำหรับการทดสอบ (Test set) 38 ภาพ คิดเป็นร้อยละ 14 และชุดรูปภาพสำหรับการตรวจสอบความถูกต้อง (Validation set) 38 ภาพ คิดเป็นร้อยละ 14 ในส่วนของชุดข้อมูลความเสียหายเรานำมาทำเป็น 3 ชุดข้อมูล ได้แก่

1) ชุดข้อมูลร่องรอยความเสียหายเพื่อตรวจจับร่องรอยความเสียหายเท่านั้น



รูปที่ 3.6 ตัวอย่างชุดข้อมูลร่องรอยความเสียหาย (Test set)

2) ชุดข้อมูลประเภทร่องรอยความเสียหายเพื่อตรวจจับและแยกประเภทร่องรอยความเสียหาย โดยแบ่งเป็นสี่ประเภท ได้แก่

1) รอยขีดข่วน (scratch) เป็นความเสียหายที่เล็กน้อยที่สุด อาจเป็นรอยขีดเล็ก ๆ ที่ผิวรถ ไม่มีผลกระทบต่อความสามารถในการใช้งานของรถ

2) รอยบุ๋มขนาดเล็ก (minor dent) เป็นความเสียหายที่มีความกว้างและความลึกเล็กน้อย โดยอาจเกิดจากการกระแทกหรือกระแทกบริเวณที่น้อยกว่ารอยบุ๋มขนาดใหญ่ ส่วนใหญ่รอยบุ๋มขนาดเล็กไม่ได้เป็นปัญหาที่ส่งผลต่อการใช้งาน

3) รอยบุ๋มขนาดใหญ่ (major dent) เป็นความเสียหายที่มีความกว้างและความลึกมากกว่า รอยบุ๋มขนาดเล็ก อาจเกิดจากการกระแทกหรือกระแทกบริเวณที่กว้างและลึก อาจทำให้เกิดไม่ว่ากรณีใดๆ การบิดเบี้ยวหรือการเปลี่ยนรูป อย่างไรก็ตามบุ๋มขนาดใหญ่ยังสามารถซ่อมแซมได้ การนำไปใช้

4) รอยแตก (cracked) เป็นความเสียหายที่รุนแรงที่สุด โดยเกิดจากการแตกหักหรือแยกออกเป็นส่วน ๆ อาจเป็นผลจากแรงกระแทกหรือการบิดเบี้ยวที่รุนแรงมาก การแตกรรนี้อาจทำให้สิ่งของเสียหายอย่างมากและไม่สามารถซ่อมแซมได้โดยง่าย

3) ชุดข้อมูลประเภทการเป็นผู้ชนจากรอยความเสียหาย โดยแบ่งเป็นสามประเภท ได้แก่

1) รถที่กระทำหรือรถคันที่ไปชน (action vehicle) รถที่เป็นผู้กระทำในเหตุการณ์การชนซึ่งอาจทำให้เกิดรอยได้หลายประเภทขึ้นอยู่กับสภาพความแข็งแรงและทันทานของรถ จากวิจรณ์ญานของของผู้จัดทำชุดข้อมูลจะมีการพิจารณาทั้งจุดของรถที่มีร่องรอยความเสียหายและลักษณะการเกิดรอยความยาว ความกว้าง ความลึก และรอยอื่นที่เกิดร่วมด้วย โดยส่วนใหญ่แล้วรอยที่เกิดกับรถที่กระทำมักเกิดในส่วนด้านหน้าของรถ ส่วนมากเป็นรอยแตก และรอยมีความกว้างลึกไม่มากเท่ารถที่ถูกกระทำ

2) รถที่ถูกกระทำหรือรถคันที่ถูกชน (victim vehicle) รถที่ถูกกระทำในเหตุการณ์การชนก็อาจทำให้เกิดรอยได้หลายประเภท จากวิจรณ์ญานของของผู้จัดทำชุดข้อมูลจะมีการพิจารณาเช่นเดียวกับรถที่กระทำ โดยส่วนใหญ่แล้วรอยที่เกิดกับรถที่ถูกกระทำมักเกิดในส่วนด้านข้างของรถ มีความกว้าง ความลึกมากกว่ารถที่กระทำ

3) รถเฉี่ยว (scrape vehicle) เป็นการเฉี่ยวกันระหว่างรถหรือวัตถุบางอย่างที่ทำให้เกิดรอยขีดข่วน (scratch) เป็นความเสียหายที่เล็กน้อยที่สุด

3.3 การพัฒนาแบบจำลองโดยใช้การเรียนรู้ของเครื่อง

3.3.1 การพัฒนาแบบจำลองโดยใช้การเรียนรู้ของเครื่อง

ในขั้นตอนการพัฒนาแบบจำลองโดยใช้การเรียนรู้ของเครื่อง เราใช้แบบจำลองที่มาจาก MMDetection ซึ่งเป็นไลบรารีโอเพนซอร์สที่พัฒนาขึ้นบนเฟรมเวิร์ค PyTorch และเป็นส่วนหนึ่งของโครงการ OpenMMLab เพื่อทดสอบความสามารถของแบบจำลองในงานวิจัยของเราทั้งหมด 8 แบบจำลองที่มีโครงสร้าง Model, Context, Backbone เป็นโครงสร้างหลักของโมเดลที่ใช้ในการแยกแยะภาพ แบบจำลอง (Model) เป็นส่วนที่ประกอบด้วยอัลกอริทึมและกระบวนการทางคณิตศาสตร์ที่ใช้ในการเรียนรู้และทำนายข้อมูล บริบท (Context) เป็นส่วนที่ระบุรายละเอียดเกี่ยวกับบริบทข้อมูลที่ล้อมรอบวัตถุหรือพิกเซลในรูปภาพ และ Backbone เป็นส่วนที่มีหน้าที่สร้างโครงสร้างพื้นฐานของแบบจำลอง ซึ่งรับภาพเข้ามาและสกัดคุณลักษณะหลักของภาพ เพื่อให้แบบจำลองสามารถเรียนรู้และทำนายผลลัพธ์ได้ดีขึ้นในงานแยกแยะภาพ

เพื่อการทดสอบและวิจัยที่มีความแม่นยำมากที่สุดในการตรวจจับวัตถุ (Object Detection) และการแบ่งส่วนภาพ (Semantic Segmentation) จากเครื่องมือ MMDetection เราได้เลือกใช้แบบจำลองที่มีค่าความแม่นยำของ Mask AP (Mask Average Precision) ที่สูงในชุดข้อมูลที่ใช้ในการทดสอบ นอกจากนี้เรายังทดลองใช้แบบจำลองอื่น ๆ ที่น่าสนใจโดยพิจารณาจากผลลัพธ์ของ

งานวิจัยที่เคยศึกษามาก่อนหน้านี้ เพื่อให้ได้ผลลัพธ์ที่มีความแม่นยำและเป็น ประโยชน์สูงสุดใน งานวิจัยของเรา ซึ่งแบบจ าสงที่เลือกใช้มีดังนี้

ตารางที่ 3.2 แบบจ าสงที่ใช้ทดลองในงานวิจัย

No	Model	Context	Backbone
1	Mask R-CNN		ResNet50
2			ResNet101
3	Mask R-CNN	GCB	ResNet50
4			ResNet101
5	Mask R-CNN		HRNetV2p-w40
6	HTC		ResNet50
7	Cascade Mask R-CNN		ResNet50
8			ResNet101

(<https://paperswithcode.com/lib/mmdetection>)

3.3.2 การทดสอบประสิทธิภาพของแบบจ าสง

หลังจากที่ได้ฝึกแบบจ าสงโดยใช้ชุดข้อมูลที่เตรียมไว้ก่อนหน้านี้ จากนั้นได้ท การทดสอบเพื่อ วัดประสิทธิภาพของแบบจ าสงในการตรวจจับวัตถุ (Object Detection) และการแบ่งส่วนภาพ (Semantic Segmentation) โดยใช้ COCO evaluator เป็นเครื่องมือที่ใช้ในการวัดและประเมิน ประสิทธิภาพของแบบจ าสงในงานตรวจจับวัตถุและการแบ่งส่วนภาพ โดยใช้ชุดข้อมูล COCO (Common Objects in Context) dataset เป็นชุดข้อมูลทดสอบ

COCO evaluator จะทำการคำนวณค่า AP (Average Precision) สำหรับแต่ละวัตถุใน COCO dataset โดยใช้ IoU (Intersection over Union) เป็นเกณฑ์การตรวจจับ โดย AP จะถูก ค านวณในแต่ละหมวดหมู่ (categories) ของวัตถุและค่าเฉลี่ยทั้งหมดจะถูกนามารวมกันเป็นค่า mAP (mean Average Precision) ซึ่งเป็นตัวบ่งชี้สำคัญในการประเมินและเปรียบเทียบประสิทธิภาพของ แบบจ าสง และมีการคำนวณ AP โดยพิจารณาหลาย ๆ ระดับของการตรวจจับ (scales) ช่วยให้ ได้รับภาพรวมของความแม่นยำของแบบจ าสงในการตรวจจับวัตถุที่ครอบคลุมหลาย ๆ ขนาดและ สัดส่วนของวัตถุ

Average Precision (AP)

AP AP at IoU=.50:.05:.95 (primary challenge metric)

$AP^{IoU=.50}$ AP at IoU=.50 (PASCAL VOC metric)

$AP^{IoU=.75}$ AP at IoU=.75 (strict metric)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิใช่ผู้จัดทำให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นหากมีเหตุจำเป็นพิเศษอื่นใด ซึ่งต้องขออนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AP Across Scales

AP^{small}	AP for small objects: $\text{area} < 32^2$
AP^{medium}	AP for medium objects: $32^2 < \text{area} < 96^2$
AP^{large}	AP for large objects: $\text{area} > 96^2$

3.4 การพัฒนาส่วนติดต่อกับผู้ใช้

3.4.1 โครงสร้างของโปรแกรมเบื้องต้น

เมื่อได้ทำการทดสอบเพื่อหาแบบจำลองที่มีประสิทธิภาพที่ดีที่สุด เราจะนำเสนอแบบจำลองที่ได้มาใช้ในการพัฒนาเว็บไซต์โดยใช้ Flask ซึ่งเป็นเฟรมเวิร์กที่ช่วยในการพัฒนาเว็บแอปพลิเคชันด้วยภาษา Python โดยมีความยืดหยุ่นสูงและง่ายต่อการใช้งาน เพื่อสร้างเว็บไซต์ที่สามารถรับข้อมูลรูปภาพจากผู้ใช้และนำไปใช้กับแบบจำลองที่มีประสิทธิภาพที่ได้ทดสอบมาแล้ว

ผู้ใช้งานสามารถอัปโหลดรูปภาพเข้าสู่ระบบ และเว็บไซต์จะใช้แบบจำลองที่ได้ถูกฝึกสอนเพื่อตรวจจับวัตถุหรือแบ่งส่วนภาพตามที่ต้องการ ผลลัพธ์ที่ได้จะถูกแสดงผลกลับมาให้ผู้ใช้งานเห็นผ่านทางเว็บไซต์ โดยจะมีการเก็บผลลัพธ์ที่ได้จากการทำนายได้ดังนี้

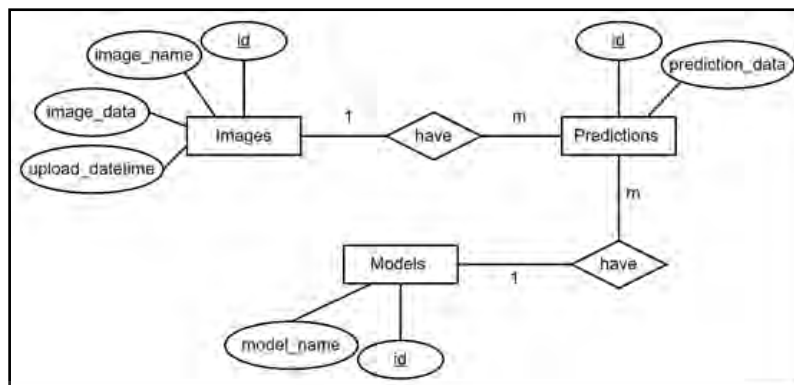
- 1) เก็บภาพต้นฉบับและภาพที่ผ่านกระบวนการ mask เพื่อใช้ในการแสดงผลหรือวิเคราะห์เพิ่มเติมในภายหลังเก็บเป็นไฟล์ภาพในระบบหรือเก็บข้อมูลเกี่ยวกับภาพในฐานข้อมูลโดยใช้ BLOB (Binary Large Object)
- 2) เก็บผลลัพธ์ที่ได้จากการทำนายเป็นข้อมูล JSON ในฐานข้อมูล PostgreSQL โดยใช้ JSONB (JSON Binary) หรือฟิลด์ข้อมูลชนิด TEXT เพื่อเก็บข้อความ JSON และอาจมีการดัดแปลงข้อมูล JSON เพื่อจัดเก็บให้อยู่ในรูปแบบที่เหมาะสมกับการวิเคราะห์หรือการนำไปใช้งานอื่น ๆ ต่อไป

เมื่อผู้ใช้อัปโหลดรูปภาพเข้าสู่ระบบแล้ว จะดึงข้อมูลภาพและผลลัพธ์ที่เกี่ยวข้องออกมาจากฐานข้อมูล PostgreSQL เพื่อนำไปแสดงผล

3.4.2 การออกแบบฐานข้อมูลของโปรแกรม

โปรแกรมวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุจะมีการเก็บข้อมูลภาพต้นฉบับที่ได้จากอินพุตหน้าเว็บไซต์ และเก็บข้อมูลเอาต์พุตที่ได้แก่ ภาพที่ถูกมาสก์ และ ไฟล์ JSON ที่เป็นผลจากการประมวลผลของแบบจำลอง ซึ่งในแต่ละภาพต้นฉบับก็อาจมีได้มากกว่าผลการทำนายขึ้นอยู่กับผู้ใช้งานว่าจะเลือกใช้งานและกดบันทึกผลการทำนายหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 รูปโครงสร้างของฐานข้อมูล

ตารางที่ 3.3 อธิบายตาราง Images ในฐานข้อมูล

ลำดับที่	ชื่อคอลัมภ์	คำอธิบาย	ชนิดข้อมูล	กฎข้อบังคับ
1	id	รหัสของรูปภาพต้นฉบับ	INTEGER	PK
2	image_name	ชื่อของรูปภาพ	VARCHAR(100)	
3	image_data	ข้อมูลรูปภาพในรูปแบบไบนารี	BLOB (BYTEA)	
4	upload_datetime	วันที่และเวลาที่อัปโหลด	TIMESTAMP	

ตารางที่ 3.4 อธิบายตาราง Models ในฐานข้อมูล

ลำดับที่	ชื่อคอลัมภ์	คำอธิบาย	ชนิดข้อมูล	กฎข้อบังคับ
1	id	รหัสของรูปภาพต้นฉบับ	INTEGER	PK
2	model_name	ชื่อแบบจำลอง	VARCHAR(100)	

ตารางที่ 3.5 อธิบายตาราง Predictions ในฐานข้อมูล

ลำดับที่	ชื่อคอลัมภ์	คำอธิบาย	ชนิดข้อมูล	กฎข้อบังคับ
1	id	รหัสของการทำนาย	INTEGER	PK
2	image_id	รหัสของรูปภาพต้นฉบับ	INTEGER	FK
3	image_pred	รูปภาพผลลัพธ์ในรูปแบบไบนารี	BLOB (BYTEA)	
4	prediction_data	ข้อมูลผลลัพธ์จากการทำนายในรูปแบบ JSON	JSONB	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการวิจัยและการอภิปรายผล

ในบทนี้ได้กล่าวถึงผลที่ได้จากการทดสอบการตรวจจับวัตถุ (Object Detection) และการแบ่งส่วนภาพ (Semantic Segmentation) โดยผลการทดสอบแบ่งออกเป็น 4 ส่วน ที่ได้จากแต่ละชุดข้อมูล ประกอบด้วย ผลการทดสอบของชุดข้อมูลการแบ่งส่วนพื้นที่ของรถยนต์ ผลการทดสอบของชุดข้อมูลร่องรอยความเสียหาย ผลการทดสอบของชุดข้อมูลประเภทร่องรอยความเสียหาย ผลการทดสอบของชุดข้อมูลประเภทการเป็นผู้ชนจากร่องรอยความเสียหาย

4.1 การทดสอบประสิทธิภาพของแบบจำลอง

4.1.1 ประสิทธิภาพของแบบจำลองในการแบ่งส่วนพื้นที่ของรถยนต์

การทดสอบประสิทธิภาพการทางานของแบบจำลองสำหรับการแบ่งส่วนพื้นที่ของรถยนต์ ซึ่งปรับค่าตัวแปรอัตราการเรียนรู้ที่ 0.0025 และรอบการเรียนรู้ 40 รอบ ซึ่งรอบการเรียนรู้ที่ 38 และ 40 เป็นรอบที่มีผลเฉลี่ย mAP สูงที่สุดเมื่อเทียบกับรอบอื่น ๆ ในแต่ละแบบจำลอง โดยแบบจำลองที่ 3 มีค่าเฉลี่ย accuracy ในรอบสุดท้ายของการฝึกที่ 40 อยู่ที่ 97.83% แบบจำลองที่ 5 มีค่า accuracy อยู่ที่ 96.31%

ตารางที่ 4.1 ประสิทธิภาพจากการฝึกอบรมแบบจำลองการแบ่งส่วนพื้นที่ของรถยนต์

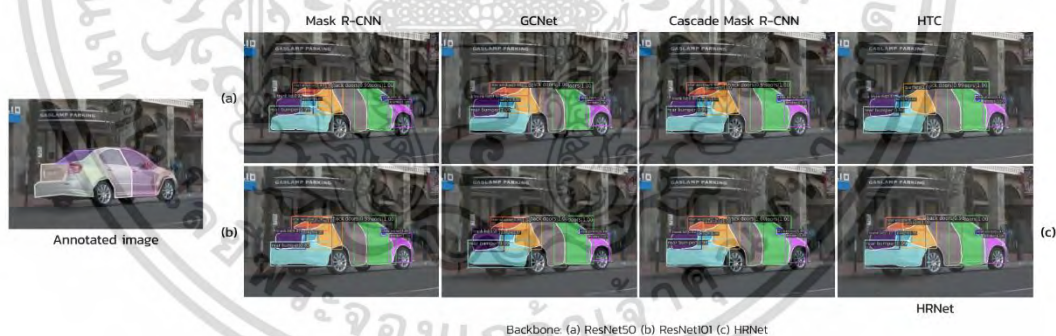
No	Model	Backbone	object detection						semantic segmentation						AVG
			mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	
1	Mask R-	ResNet50	62.70	92.50	72.50	34.30	62.90	79.30	59.70	92.50	67.90	31.90	59.70	77.60	69.94
2	CNN	ResNet101	64.20	93.90	72.60	40.40	64.90	79.90	60.30	92.40	67.50	28.60	60.40	79.80	70.71
3	HTC	ResNet50	65.60	93.20	72.80	48.90	65.20	83.60	61.40	93.10	68.50	32.60	61.40	79.70	72.03
4	GCNet	ResNet50	62.40	92.00	71.00	33.40	63.40	78.10	58.60	91.20	64.50	27.20	59.50	77.30	68.61
5		ResNet101	64.90	92.90	75.50	39.60	64.90	82.60	61.40	92.90	66.40	31.00	61.50	80.20	71.55
6	HRNet	HRNet	64.50	93.10	74.10	38.40	64.70	80.30	61.60	92.80	66.50	31.30	62.00	80.30	71.10
7	Cascade	ResNet50	65.90	92.90	74.60	39.00	66.60	84.60	60.80	91.30	68.20	28.80	61.20	79.40	71.69
8	Mask R-CNN	ResNet101	65.50	92.30	74.20	40.30	66.00	82.80	60.80	91.60	68.90	30.30	61.10	79.70	71.58

จากการฝึกอบรมแบบจำลองสำหรับการแบ่งส่วนพื้นที่ของรถยนต์ในตาราง 4.1 เป็นข้อมูลประสิทธิภาพของแต่ละแบบจำลอง เมื่อคำนวณค่าเฉลี่ย (average) พร้อมให้ค่าน้ำหนัก (weight) ตัวแปร mAP เท่ากับ 3, AP^{IoU=50} เท่ากับ 2, AP^{IoU=75} เท่ากับ 4, AP^{Small} เท่ากับ 1, AP^{Medium} เท่ากับ 1, AP^{Large} เท่ากับ 3 เนื่องจากเราให้ความสำคัญโดยรวมในการตรวจจับวัตถุที่มีขนาดใหญ่และไม่มีการใกล้เคียงกับพื้นที่ของข้อมูลที่เป็นจริง (Ground truth) จากตารางแสดงให้เห็นว่า HTC ResNet50

(แบบจ ลองที่ 3) เป็นแบบจ ลองที่มีประสิทธิภาพดีที่สุดในอันดับที่ 1 ในการประเมินผลลัพธ์โดยมีค่า mAP ที่ 65.60% ในการตรวจจับวัตถุ และค่า mAP ที่ 83.60% ในการแบ่งส่วนภาพ ตามมาด้วย Cascade Mask R-CNN ResNet50 (แบบจ ลองที่ 7) ที่มีประสิทธิภาพดีเป็นอันดับที่ 2 และ Cascade Mask R-CNN ResNet10 (แบบจ ลองที่ 8) เป็นอันดับที่ 3 อย่างไรก็ตาม GCNet ResNet50 (แบบจ ลองที่ 4) เป็นแบบจ ลองที่มีประสิทธิภาพดีที่สุดในการฝึกอบรม โดยมีค่า mAP ที่ 62.40% ในการตรวจจับวัตถุ และค่า mAP ที่ 78.10% ในการแบ่งส่วนภาพ ซึ่งอยู่ในอันดับสุดท้ายในการประเมินผลลัพธ์



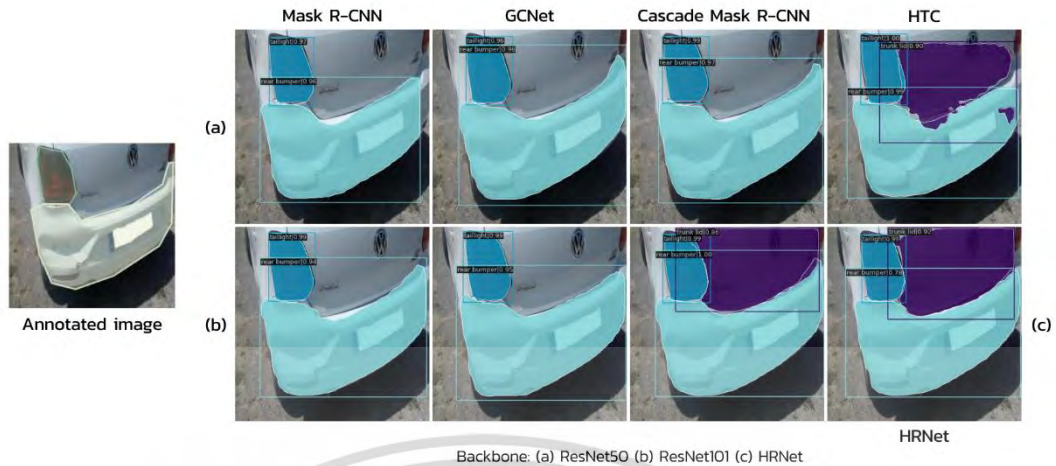
รูปที่ 4.1 ตัวอย่างรูปภาพผลการท านายของแบบจ ลองการแบ่งส่วนพื้นที่ของรถยนต์



รูปที่ 4.2 ตัวอย่างรูปภาพผลการท านายของแบบจ ลองการแบ่งส่วนพื้นที่ของรถยนต์

จากรูปที่ 4.1 และ 4.2 ในการแบ่งส่วนพื้นที่ของรถจะเห็นได้ว่าแต่ละแบบจ ลองมีผลการท านายในการ mask ภาพ ที่ใกล้เคียงกันมาก มีการท านายส่วนพื้นที่ของรถตาม Annotated image ซึ่งเป็นผลลัพธ์ที่คาดหวัง ยกเว้น HTC ResNet50 ในรูปที่ 4.2 ที่ขาดไป 1 ส่วน คือ กระจกหลัง (rear window)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 ตัวอย่างรูปภาพผลการทนายของแบบจำลองการแบ่งส่วนพื้นที่ของรถยนต์

จากรูปที่ 4.3 ในการแบ่งส่วนพื้นที่ของรถจะเห็นว่าแต่ละแบบจำลองมีผลการทนายในการ mask ภาพ ที่ใกล้เคียงกันมาก มีการทนายส่วนพื้นที่ของรถตาม Annotated image ซึ่งเป็นผลลัพธ์ที่คาดหวัง ยกเว้น HTC ResNet50 , HRNet, Cascade Mask R-CNN ResNet101 ที่มีการทนายเพิ่มมาในส่วนของฝากระโปรงหลัง (trunk lid) ซึ่งถือว่าทนายถูกต้อง

4.1.2 ประสิทธิภาพของแบบจำลองในการตรวจจับร่องรอยความเสียหาย

การทดสอบประสิทธิภาพการทนายงานของแบบจำลองสำหรับการตรวจจับร่องรอยความเสียหาย ซึ่งปรับค่าตัวแปรอัตราการเรียนรู้ที่ 0.0025 และรอบการเรียนรู้ที่ 60 รอบ โดยแบบจำลองที่ 5 มีค่า accuracy ในรอบสุดท้ายของการฝึกที่ 60 อยู่ที่ 99.27% แบบจำลองที่ 6 มีค่า accuracy ในรอบสุดท้ายของการฝึกที่ 60 อยู่ที่ 99.39% แบบจำลองที่ 3 และ 8 มีค่าเฉลี่ย accuracy อยู่ที่ 99.49%

ตารางที่ 4.2 ประสิทธิภาพจากการฝึกอบรมแบบจำลองการตรวจจับร่องรอยความเสียหาย

No	Model	Backbone	object detection						semantic segmentation						AVG
			mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	
1	Mask R-CNN	ResNet50	21.60	50.00	13.90	0.00	7.60	34.70	16.80	41.90	12.30	0.00	3.70	31.20	21.56
2		ResNet101	26.10	58.50	16.60	0.00	18.20	37.70	23.00	58.90	13.60	0.00	14.80	36.00	26.99
3	HTC	ResNet50	27.80	56.10	26.20	1.30	16.10	40.90	20.20	54.30	18.30	1.00	8.60	32.80	27.79
4	GCNet	ResNet50	22.90	57.30	13.50	0.00	13.40	33.70	19.40	48.60	9.60	0.00	9.40	30.30	22.90
5		ResNet101	30.20	54.90	32.20	0.05	17.10	43.70	23.30	54.40	17.00	0.00	9.30	37.00	29.74
6	HRNet	HRNet	28.70	63.40	29.90	0.00	16.70	41.40	22.50	55.20	15.20	0.00	14.00	32.10	28.96
7	Cascade	ResNet50	25.80	54.30	28.80	0.00	15.40	37.00	18.10	47.20	14.00	0.00	9.30	28.30	25.52
8	Mask R-CNN	ResNet101	27.40	55.80	28.40	0.00	14.00	41.10	20.40	49.20	14.80	0.00	8.50	33.90	27.23

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการฝึกอบรมแบบจำลองสำหรับการตรวจจับร่องรอยความเสียหายในตาราง 4.2 เป็นข้อมูลประสิทธิภาพของแต่ละแบบจ าสง แสดงให้เห็นว่า GCNet ResNet101 (แบบจ าสงที่ 5) เป็นแบบจำลองที่มีประสิทธิภาพดีที่สุดโดยมีอันดับที่ 1 ในการประเมินผลลัพธ์โดยมีค่า mAP ที่ 30.20% ในการตรวจจับวัตถุ และค่า mAP ที่ 23.30% ในการแบ่งส่วนภาพ ตามมาด้วย HRNet (แบบจำลองที่ 6) ที่มีประสิทธิภาพดีเป็นอันดับที่ 2 และ HTC (แบบจำลองที่ 3) เป็นอันดับที่ 3 อย่างไรก็ตาม Mask R-CNN ResNet50 (แบบจำลองที่ 1) เป็นแบบจำลองที่มีประสิทธิภาพต่ำที่สุดในการฝึกอบรมโดยมีค่า mAP ที่ 21.60% ในการตรวจจับวัตถุ และค่า mAP ที่ 16.80% ในการแบ่งส่วนภาพ ซึ่งอยู่ในอันดับสุดท้ายในการประเมินผลลัพธ์



รูปที่ 4.4 ตัวอย่างรูปภาพผลการท านายของแบบจ าสงการตรวจจับร่องรอยความเสียหาย

จากรูปที่ 4.4 ในการตรวจจับร่องรอยความเสียหาย (damage) จะเห็นได้ว่าแต่ละแบบจ าสง มีผลการท านายในการ mask ภาพ ในจุดที่ใกล้เคียงกับ Annotated image ซึ่งเป็นผลลัพธ์ที่คาดหวัง ยกเว้น HTC ResNet50 ที่ไม่มีตรวจพบร่องรอยความเสียหาย

4.1.3 ประสิทธิภาพของแบบจ าสงในการแบ่งประเภทร่องรอยความเสียหาย

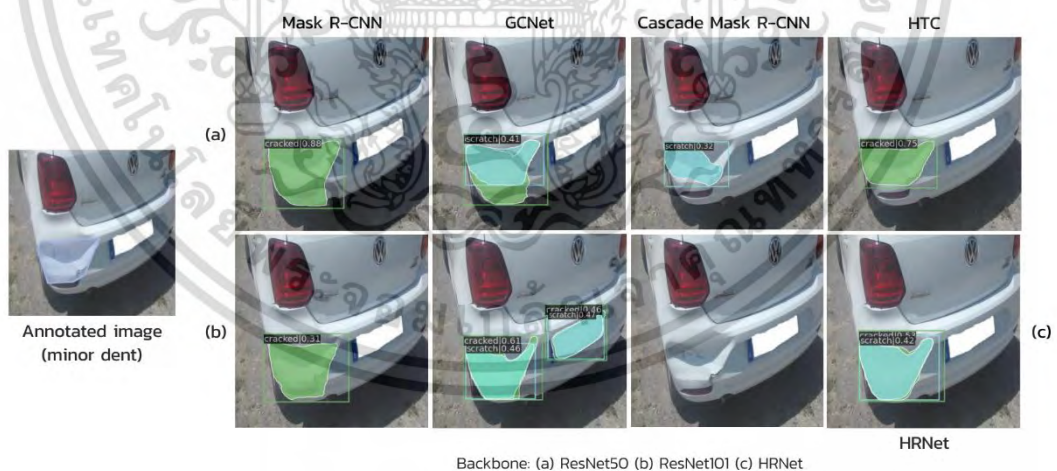
การทดสอบประสิทธิภาพการทำงานของแบบจำลองสำหรับการแบ่งประเภทร่องรอยความเสียหาย ซึ่งปรับค่าตัวแปรอัตราการเรียนรู้ที่ 0.0025 และรอบการเรียนรู้ที่ 60 โดยแบบจ าสงที่ 3 มีค่าเฉลี่ย accuracy ในรอบสุดท้ายของการฝึกอบรมที่ 60 อยู่ที่ 99.14% แบบจำลองที่ 4 และ 5 มีค่า accuracy อยู่ที่ 98.42% และ 98.69% ตามล าดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ประสิทธิภาพจากการฝึกอบรมแบบจำลองการแบ่งประเภทร่องรอยความเสียหาย

No	Model	Backbone	object detection						semantic segmentation						AVG
			mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	
1	Mask R-CNN	ResNet50	17.90	40.70	14.40	0.00	12.70	26.40	14.10	32.20	11.70	0.00	11.90	21.30	18.32
2	CNN	ResNet101	19.70	39.70	15.40	0.00	31.10	27.10	16.40	37.10	15.40	0.00	20.80	21.60	21.37
3	HTC	ResNet50	23.20	40.60	26.00	2.20	31.00	38.10	18.30	46.00	16.10	1.30	20.40	27.00	26.03
4	GCNet	ResNet50	22.50	40.90	26.70	0.00	30.20	23.30	19.30	38.40	16.10	0.00	18.00	19.10	22.54
5		ResNet101	20.10	49.00	16.70	0.00	28.10	26.50	17.80	42.20	10.90	0.00	18.90	24.00	21.95
6	HRNet	HRNet	14.80	32.00	13.70	0.00	24.70	19.70	12.40	27.20	8.00	0.00	15.00	18.10	16.17
7	Cascade	ResNet50	19.70	37.60	19.90	16.70	34.00	20.40	16.30	35.20	15.70	20.00	16.00	17.30	21.36
8	Mask R-CNN	ResNet101	20.90	38.80	15.30	0.00	18.20	34.20	15.50	38.50	7.00	0.00	15.20	23.30	20.31

จากการฝึกอบรมแบบจำลองสำหรับการแบ่งประเภทร่องรอยความเสียหายในตาราง 4.3 เป็นข้อมูลประสิทธิภาพของแต่ละแบบจำลอง แสดงให้เห็นว่า HTC ResNet50 (แบบจำลองที่ 3) เป็นแบบจำลองที่มีประสิทธิภาพดีที่สุดในอันดับที่ 1 ในการประเมินผลลัพธ์โดยมีค่า mAP ที่ 23.20% ในการตรวจจับวัตถุ และค่า mAP ที่ 18.30% ในการแบ่งส่วนภาพ ตามมาด้วย GCNet ResNet50 (แบบจำลองที่ 4) ที่มีประสิทธิภาพดีเป็นอันดับที่ 2 และ GCNet ResNet101 (แบบจำลองที่ 5) เป็นอันดับที่ 3 อย่างไรก็ตาม HRNet (แบบจำลองที่ 6) เป็นแบบจำลองที่มีประสิทธิภาพต่ำที่สุดในการฝึกอบรม โดยมีค่า mAP ที่ 14.80% ในการตรวจจับวัตถุ และค่า mAP ที่ 12.40% ในการแบ่งส่วนภาพ ซึ่งอยู่ในอันดับสุดท้ายในการประเมินผลลัพธ์



รูปที่ 4.5 ตัวอย่างรูปภาพผลการทำนายของแบบจำลองการแบ่งประเภทร่องรอยความเสียหาย

จากรูปที่ 4.5 ในการตรวจจับและแบ่งประเภทร่องรอยความเสียหายจะเห็นว่าแต่ละแบบจำลองมีผลการทำนายในการ mask ภาพ ในจุดที่ใกล้เคียงกับ Annotated image ที่ระบุว่าเป็นรอยบวมขนาดเล็ก โดยในแบบจำลอง Mask R-CNN และ HTC ทำนายว่าเป็นรอยแตก ส่วนในไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบจำลอง GCNet และ HRNet ทานายว่าเป็นรอยแตกและรอยขีดข่วน และ Cascade Mask R-CNN ResNet50 ทานายว่าเป็นรอยขีดข่วน ยกเว้น Cascade Mask R-CNN ResNet101 ที่ไม่มีตรวจพบร่องรอยความเสียหายใด ๆ

4.1.4 ประสิทธิภาพของแบบจำลองในการแบ่งประเภทการเป็นผู้ชนจากร่องรอยความเสียหาย

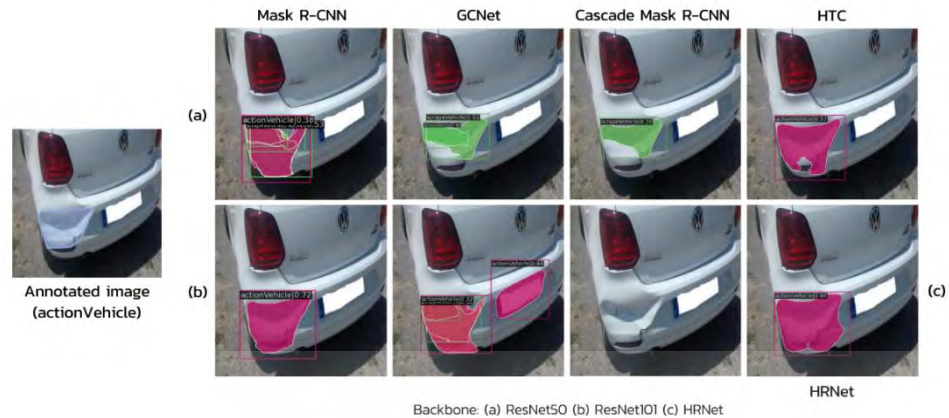
การทดสอบประสิทธิภาพการท างานของแบบจำลองสำหรับการแบ่งประเภทผู้ชนจากร่องรอยความเสียหาย ซึ่งปรับค่าตัวแปรอัตราการเรียนรู้ที่ 0.0025 และรอบการเรียนรู้ที่ 60, 120 แต่ผลลัพธ์ที่ได้ค่อนข้างเอนไปเป็นรถที่กระแทกเกือบทุกภาพ แบบจำลองที่ 6 มีค่าเฉลี่ย accuracy ในรอบสุดท้ายของการฝึกอบที่ 60 อยู่ที่ 98.28% ส่วนรอบสุดท้ายของการฝึกอบที่ 120 อยู่ที่ 98.86% และในแบบจำลองที่ 3 มีค่าเฉลี่ย accuracy ในรอบสุดท้ายของการฝึกอบที่ 60 อยู่ที่ 98.78% ส่วนรอบสุดท้ายของการฝึกอบที่ 120 อยู่ที่ 99.38%

ตารางที่ 4.4 ประสิทธิภาพจากการฝึกอบแบบจำลองการแบ่งประเภทการเป็นผู้ชน

No	Model	Backbone	object detection						semantic segmentation						AVG
			mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	
1	Mask R-	ResNet50	15.90	29.70	13.60	0.00	13.20	19.80	13.50	32.30	12.90	0.00	9.80	18.20	16.34
2	CNN	ResNet101	15.20	28.90	14.60	0.00	13.40	23.20	13.20	28.30	10.60	0.00	11.00	21.30	16.48
3	HTC	ResNet50	18.30	36.20	20.50	0.00	17.30	26.70	14.70	31.90	7.20	0.00	11.70	23.90	18.95
4	GCNet	ResNet50	16.10	33.00	13.00	0.00	13.10	25.10	14.10	31.40	14.10	0.00	8.30	23.80	17.69
5		ResNet101	16.90	29.10	12.30	0.00	22.30	17.80	14.50	34.10	13.60	0.00	13.60	18.20	16.88
6	HRNet	HRNet	18.70	34.20	13.50	0.00	18.50	29.90	15.50	34.60	12.50	0.00	13.30	25.40	19.66
7	Cascade	ResNet50	16.00	28.30	17.20	0.00	15.00	17.50	13.30	27.80	14.50	0.00	11.90	17.30	16.25
8	Mask R-CNN	ResNet101	19.00	33.80	19.40	0.00	15.20	28.00	15.20	30.90	13.00	0.00	11.10	22.70	19.28

จากการฝึกอบแบบจำลองสำหรับการแบ่งประเภทการเป็นผู้ชนจากร่องรอยความเสียหาย ในตาราง 4.4 เป็นข้อมูลประสิทธิภาพของแต่ละแบบจำลอง แสดงให้เห็นว่า HRNet (แบบจำลองที่ 6) เป็นแบบจำลองที่มีประสิทธิภาพดีที่สุดโดยมีอันดับที่ 1 ในการประเมินผลลัพธ์โดยมีค่า mAP ที่ 18.70% ในการตรวจจับวัตถุ และค่า mAP ที่ 15.50% ในการแบ่งส่วนภาพ ตามมาด้วย Cascade Mask R-CNN ResNet101 (แบบจำลองที่ 8) ที่มีประสิทธิภาพดีเป็นอันดับที่ 2 และ HTC ResNet50 (แบบจำลองที่ 3) เป็นอันดับที่ 3 อย่างไรก็ตาม Cascade Mask R-CNN ResNet50 (แบบจำลองที่ 7) เป็นแบบจำลองที่มีประสิทธิภาพต่ำที่สุดในการฝึกอบ โดยมีความ mAP ที่ 15.20% ในการตรวจจับวัตถุ และค่า mAP ที่ 13.20% ในการแบ่งส่วนภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 ตัวอย่างรูปภาพผลการทนายของแบบจำลองการเป็นผู้ชนจากร่องรอยความเสียหาย

จากรูปที่ 4.6 ในการตรวจจับและแบ่งประเภทร่องรอยความเสียหายจะเห็นได้ว่าแต่ละแบบจำลองมีผลการทนายในการ mask ภาพ ในจุดที่ใกล้เคียงกับ Annotated image ที่ระบุว่าเป็นรถที่กระแทกหรือรถคันที่ไปชน โดยในแบบจำลอง Mask R-CNN ResNet101, HTC, HRNet ทนายว่าเป็นรถที่กระแทก ส่วนในแบบจำลอง GCNet ResNet50 และ Cascade Mask R-CNN ResNet50 ทนายว่าเป็นรถเฉี่ยว และ Mask R-CNN ResNet50, GCNet ResNet101 ทนายว่าเป็นรถที่กระแทกและรถเฉี่ยว ยกเว้น Cascade Mask R-CNN ResNet101 ที่ไม่มีตรวจพบร่องรอยความเสียหายใด ๆ

ตารางที่ 4.5 ประสิทธิภาพจากการฝึกอบรมแบบจำลองในแต่ละชุดข้อมูล

No	Model	Backbone	Rank of Dataset			
			Part	Damage	Damage Type	Crash Type
1	Mask R-	ResNet50	7	8	7	7
2	CNN	ResNet101	6	5	4	6
3	HTC	ResNet50	1	3	1	3
4	GCNet	ResNet50	8	7	2	4
5		ResNet101	4	1	3	5
6	HRNet	HRNet	5	2	8	1
7	Cascade	ResNet50	2	6	5	8
8	Mask R-CNN	ResNet101	3	4	6	2

จากตาราง 4.5 แสดงให้เห็นว่า จากผลการทดลองทั้งสี่ชุดข้อมูลพบว่า แบบจำลองที่มีประสิทธิภาพแม่นยำที่สุดในการแบ่งส่วนพื้นที่ของรถยนต์คือ HTC ResNet50 การตรวจจับร่องรอยความเสียหายคือ GCNet ResNet101 การแบ่งประเภทร่องรอยความเสียหายคือ HTC ResNet50 การแบ่งประเภทการเป็นผู้ชนจากร่องรอยความเสียหายคือ HRNet อย่างไรก็ตามรูปภาพที่นำมาแสดงเอกสารนี้เป็นเพียงตัวอย่างของผลการทนายบางส่วนเท่านั้น แบบจำลองแต่ละแบบจำลองมีประสิทธิภาพที่ไม่ต่างกันในแต่ละรูปภาพที่ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.5 ประสิทธิภาพของแบบจาลอง หลงส สำหรับการแบ่งส่วนพื้นที่รถและความเสียหาย

เนื่องจากแบบจาลอง HTC ResNet50 เป็นแบบจาลองอันดับที่ 1 ในการแบ่งส่วนพื้นที่ของรถยนต์ (จากตารางที่ 4.1) และ GCNet ResNet101 เป็นแบบจาลองอันดับที่ 1 ในการตรวจจับร่องรอยความเสียหาย (จากตารางที่ 4.2) จึงได้นำ มาทดสอบประสิทธิภาพการทางานสำหรับการแบ่งส่วนพื้นที่รถและความเสียหาย ซึ่งปรับค่าตัวแปรอัตราการเรียนรู้ที่ 0.0025 และรอบการเรียนรู้ที่ 60 รอบ

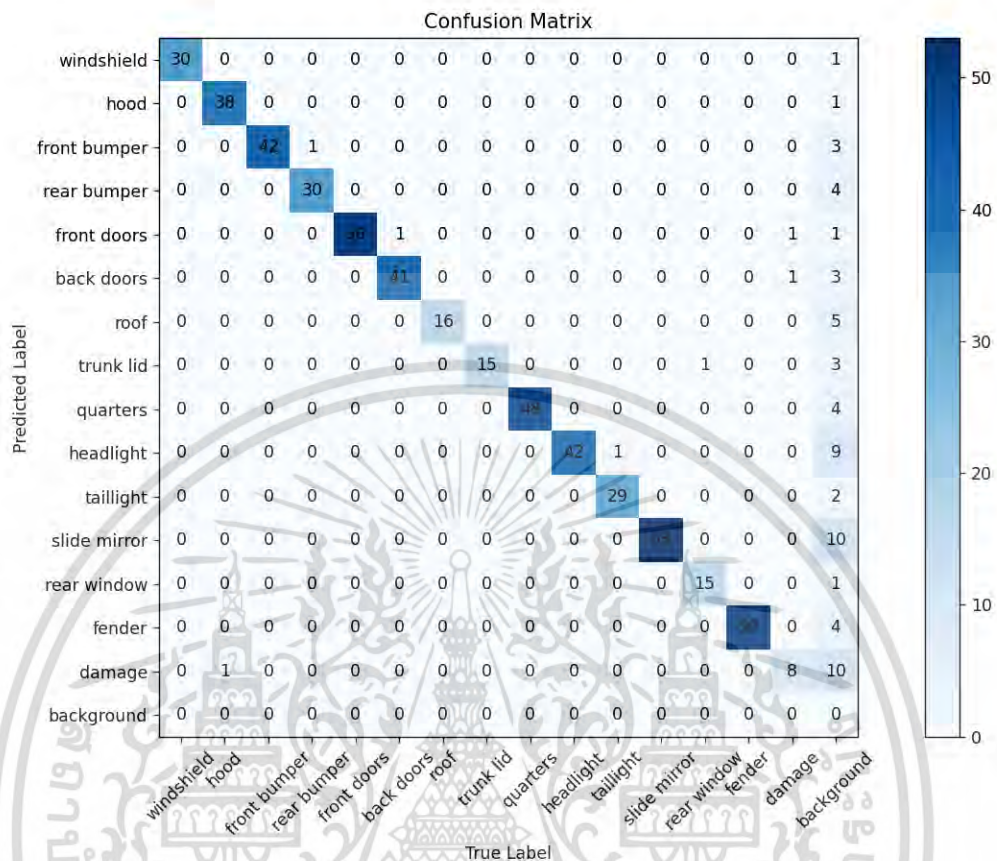
ตารางที่ 4.6 ประสิทธิภาพจากการฝึกอบรมแบบจาลอง หลงส สำหรับการแบ่งส่วนพื้นที่รถ ความเสียหาย

Model	Backbone	object detection						semantic segmentation						AVG
		mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	mAP	AP ^{IoU=50}	AP ^{IoU=75}	AP ^{Small}	AP ^{Medium}	AP ^{Large}	
HTC	ResNet50	56.10	82.50	63.00	40.50	57.50	65.00	53.00	83.00	57.20	29.50	52.10	67.50	61.29
GCNet	ResNet101	56.40	83.10	64.40	35.50	58.60	64.80	53.20	83.00	58.20	30.20	52.70	67.20	61.59

จากการฝึกอบรมแบบจาลองสำหรับการแบ่งส่วนพื้นที่รถและร่องรอยความเสียหายในตาราง 4.6 เป็นข้อมูลประสิทธิภาพของแบบจาลอง HTC ResNet50 และ GCNet ResNet101 เมื่อนำทั้งสองแบบจาลองมาฝึกอบรม แสดงให้เห็นว่าทั้งสองแบบจาลองมีประสิทธิภาพใกล้เคียงกันมาก โดย HTC ในการประเมินผลลัพธ์มีค่า mAP ที่ 56.10% ในการตรวจจับวัตถุ และค่า mAP ที่ 53.00% ในการแบ่งส่วนภาพ ส่วน GCNet ResNet101 ในการประเมินผลลัพธ์มีค่า mAP ที่ 56.40% ในการตรวจจับวัตถุ และค่า mAP ที่ 53.20% ในการแบ่งส่วนภาพ และมีค่าความแม่นยำเฉลี่ยของทั้ง 6 เมตริกที่ใช้ในการวัดต่างกับเพียง 0.30 % เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.6 Confusion Matrix ของแบบจ ลองส หรับการแบ่งส่วนพื้นที่รถและความเสียหาย



รูปที่ 4.7 Confusion Matrix HTC ResNet50 ส าหรับการแบ่งส่วนพื้นที่รถและความเสียหาย

จากตารางรูปที่ 4.7 Confusion Matrix HTC ResNet50 ส าหรับการแบ่งส่วนพื้นที่รถและความเสียหาย วัดที่ค่า IoU = 0.50 ได้ค่า True Positive (TP), False Positive (FP), False Negative (FN), และ True Negative (TN) ของทุกหมวดหมู่ทั้ง 15 หมวดหมู่ ดังนี้

$$TP = 30 + 38 + 42 + 30 + 50 + 41 + 16 + 15 + 48 + 42 + 29 + 53 + 15 + 50 + 8 + 0 = 507$$

$$FP = 1 + 1 + 4 + 4 + 3 + 4 + 5 + 4 + 4 + 10 + 2 + 10 + 1 + 4 + 11 + 0 = 68$$

$$FN = 0 + 1 + 0 + 1 + 0 + 1 + 0 + 0 + 0 + 0 + 1 + 0 + 1 + 0 + 2 + 61 = 68$$

$$TN = 544 + 535 + 529 + 540 + 522 + 529 + 554 + 556 + 523 + 523 + 543 + 512 + 558 + 521 + 554 + 514 = 8557$$

$$Accuracy = \frac{507 + 8557}{507 + 8557 + 68 + 68} = 0.9852$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 Precision = $\frac{507}{507 + 68} = 0.8817$
 ไม่ว่าจะผลิตขึ้นกี่ชิ้นก็ให้แบ่งเงินออกมาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{Sensitivity} = \frac{507}{507 + 68} = 0.8817$$

$$\text{f1score} = \frac{2(0.8817 \times 0.8817)}{(0.8817 + 0.8817)} = 0.8817$$

จากการคำนวณสามารถสรุปค่าประสิทธิภาพที่สำคัญของระบบการตรวจจับวัตถุในงาน Object Detection สำหรับการแบ่งส่วนพื้นที่รถและความเสียหาย จาก Confusion Matrix ที่ได้ระบุไว้ได้ดังนี้ ค่า Accuracy คือสัดส่วนของวัตถุที่ถูกตรวจจับและทำนายถูกต้องทั้งหมดเทียบกับจำนวนวัตถุทั้งหมดบนภาพ ในที่นี้ค่า Accuracy เท่ากับ 0.9852 หมายความว่าระบบตรวจจับและทำนายวัตถุบนภาพได้ถูกต้องและแม่นยำมากถึง 98.52%

ค่า Precision คือสัดส่วนของวัตถุที่ถูกตรวจจับและทำนายว่าเป็นวัตถุนภาพจริง เทียบกับวัตถุที่ระบบตรวจจับว่าเป็นวัตถุนภาพ ในที่นี้ค่า Precision เท่ากับ 0.8817 หมายความว่าระบบมีความแม่นยำในการตรวจจับวัตถุนภาพอยู่ที่ประมาณ 88.17%

ค่า Sensitivity หรือเรียกว่า Recall หรือความจำเป็น คือสัดส่วนของวัตถุที่ถูกตรวจจับและทำนายถูกต้อง เทียบกับจำนวนวัตถุทั้งหมดที่เป็นวัตถุจริง ในที่นี้ค่า Sensitivity เท่ากับ 0.8817 หมายความว่าระบบมีความครอบคลุมในการตรวจจับวัตถุนภาพอยู่ที่ประมาณ 88.17%

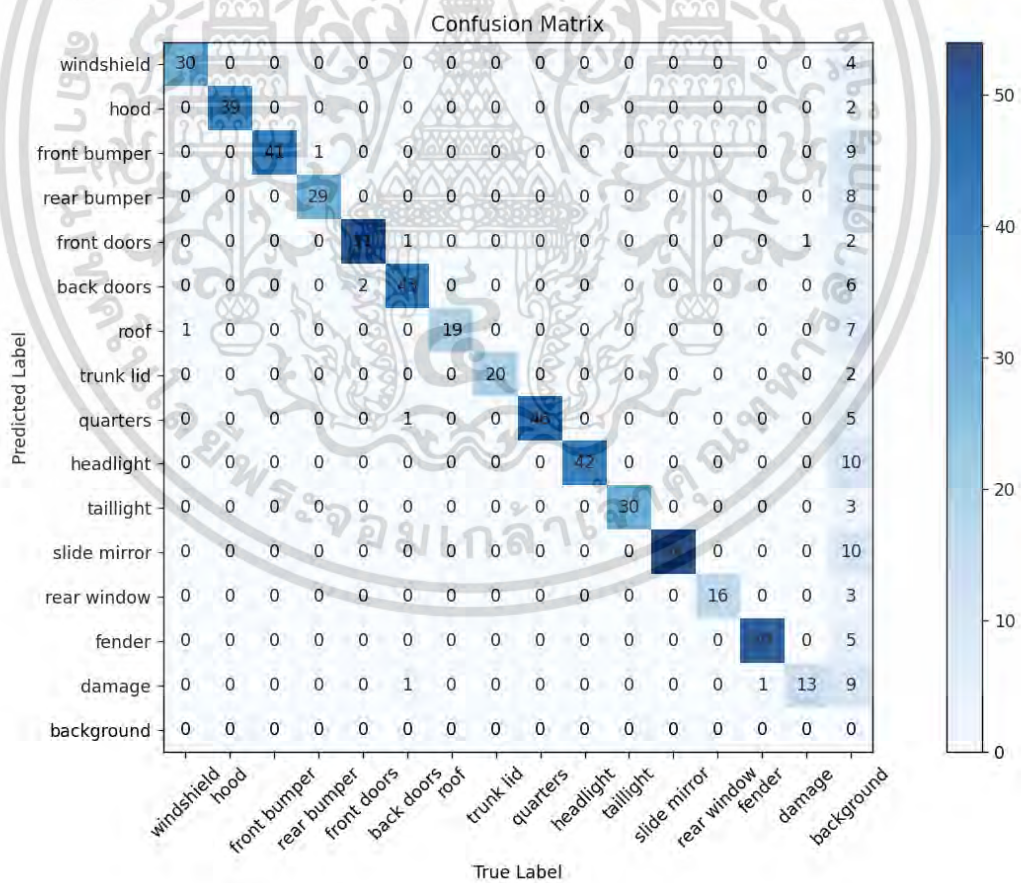
การเฉลี่ยความแม่นยำและความครอบคลุมเพื่อให้ได้ค่าฐานนิยมระหว่างความแม่นยำและความครอบคลุม ในที่นี้ค่า F1-Score เท่ากับ 0.8817 หมายความว่าระบบมีประสิทธิภาพทั้งในการตรวจจับวัตถุและทำนายว่าเป็นวัตถุนภาพอยู่ที่ประมาณ 88.17%

ตารางที่ 4.7 ประสิทธิภาพแบบจ ของ HTC ResNet50 จาก Confusion Matrix

Classes	Actual	Predicted	TP	FP	FN	TN	Accuracy	Precision	Recall	F1
windshield	31	31	30	1	0	544	0.9983	1.0000	0.9677	0.9836
hood	38	39	38	1	1	535	0.9965	0.9744	0.9744	0.9744
front bumper	43	46	42	4	0	529	0.9930	1.0000	0.9130	0.9545
rear bumper	38	34	30	4	1	540	0.9913	0.9677	0.8824	0.9231
front doors	54	53	50	3	0	522	0.9948	1.0000	0.9434	0.9709
back doors	43	45	41	4	1	529	0.9913	0.9762	0.9111	0.9425
roof	25	21	16	5	0	554	0.9913	1.0000	0.7619	0.8649
trunk lid	23	19	15	4	0	556	0.9930	1.0000	0.7895	0.8824
quarters	51	52	48	4	0	523	0.9930	1.0000	0.9231	0.9600
headlight	48	52	42	10	0	523	0.9826	1.0000	0.8077	0.8936
taillight	37	31	29	2	1	543	0.9948	0.9667	0.9355	0.9508
slide mirror	61	63	53	10	0	512	0.9826	1.0000	0.8413	0.9138
rear window	17	16	15	1	1	558	0.9965	0.9375	0.9375	0.9375
fender	54	54	50	4	0	521	0.9930	1.0000	0.9259	0.9615
damage	23	19	8	11	2	554	0.9774	0.8000	0.4211	0.5517
background	0	0	0	0	61	514	0.8939	0.0000	0.0000	0.0000

จากตารางที่ 4.7 แสดงให้เห็นว่าแบบจำลอง HTC ResNet50 มีความแม่นยำสูงในการตรวจจับคลาสส่วนพื้นที่ยี่รถ เช่น คลาส windshield, front bumper, front doors และ fender ที่มีค่า Precision, Recall, และ F1 Score ที่สูงและ Accuracy ทั้งหมดมีค่าสูงอยู่ที่ระดับสูงมากกว่า 99%

ความแม่นยำในการตรวจจับคลาส damage ของแบบจำลอง HTC ResNet50 ได้ความแม่นยำอยู่ที่ประมาณ 97.74% โดยค่า Precision 80.00% หมายความว่าจากตัวอย่างที่แบบจำลองทำนายว่าเป็นคลาส damage อยู่ทั้งหมด 19 ตัวอย่าง แต่มีเพียง 8 ตัวอย่างเท่านั้นที่เป็นคลาส damage จริง ส่วนค่า Recall แสดงให้เห็นว่าแบบจำลองสามารถตรวจจับตัวอย่างที่เป็นคลาส damage ได้เพียง 42.11% เท่านั้น ซึ่งหมายความว่ามตัวอย่างที่เป็นคลาส damage บางส่วนที่โมเดลไม่สามารถตรวจจับได้ นอกจากนี้ F1 Score เท่ากับ 0.5517 แสดงให้เห็นว่าแบบจำลองมีความสามารถในการทำนายและตรวจจับคลาส damage อยู่ที่ 55.17% ถือว่าอยู่ในระดับที่ค่อนข้างต่ำ ความสามารถของแบบจำลองในการทำนายและตรวจจับคลาส damage ที่ระดับนี้อาจยังไม่เพียงพอสำหรับการประยุกต์ใช้จริงที่ต้องการความแม่นยำและประสิทธิภาพสูง สำหรับงานที่ต้องการการตรวจจับคลาส damage อย่างแม่นยำ เช่น ในการตรวจสอบสภาพรถหรือการตรวจแก้ไขความเสียหาย



รูปที่ 4.8 Confusion Matrix GCNet ResNet101 สำหรับการแบ่งส่วนพื้นที่ยี่รถและความเสียหาย เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางรูปที่ 4.8 Confusion Matrix GCNet ResNet101 สำหรับการแบ่งส่วนพื้นที่รถและความเสียหาย วัดที่ค่า IoU = 0.50 ได้ค่า True Positive (TP), False Positive (FP), False Negative (FN), และ True Negative (TN) ของทุกหมวดหมู่ทั้ง 15 หมวดหมู่ ดังนี้

$$\begin{aligned} TP &= 30 + 39 + 41 + 29 + 51 + 43 + 19 + 20 + 46 + 42 + 30 + 54 + 16 + 50 + 13 \\ &= 523 \end{aligned}$$

$$FP = 4 + 2 + 10 + 8 + 4 + 8 + 8 + 2 + 6 + 10 + 3 + 10 + 3 + 5 + 11 + 0 = 94$$

$$FN = 1 + 0 + 0 + 1 + 2 + 3 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 1 + 1 + 85 = 94$$

$$\begin{aligned} TN &= 582 + 576 + 566 + 579 + 560 + 563 + 590 + 595 + 565 + 565 + 584 + 553 \\ &+ 598 + 561 + 592 + 532 = 9161 \end{aligned}$$

$$\text{Accuracy} = \frac{523 + 9161}{523 + 9161 + 94 + 94} = 0.9810$$

$$\text{Precision} = \frac{523}{523 + 94} = 0.8476$$

$$\text{Sensitivity} = \frac{523}{523 + 94} = 0.8476$$

$$f1\text{score} = \frac{2(0.8476 \times 0.8476)}{(0.8476 + 0.8476)} = 0.8476$$

จากการคำนวณสามารถสรุปค่าประสิทธิภาพที่สำคัญของระบบการตรวจจับวัตถุในงาน Object Detection สำหรับการแบ่งส่วนพื้นที่รถและความเสียหาย จาก Confusion Matrix ที่ได้ระบุไว้ได้ดังนี้ ค่า Accuracy เท่ากับ 0.9810 หมายความว่าระบบตรวจจับและทำนายวัตถุบนภาพได้ถูกต้องและแม่นยำมากถึง 98.10% ค่า Precision เท่ากับ 0.8476 หมายความว่าระบบมีความแม่นยำในการตรวจจับวัตถุบนภาพอยู่ที่ประมาณ 84.76% ค่า Sensitivity หรือเรียกว่า Recall หรือความจำเป็น เท่ากับ 0.8476 หมายความว่าระบบมีความครอบคลุมในการตรวจจับวัตถุบนภาพอยู่ที่ประมาณ 84.76% ในส่วนของค่า F1-Score เท่ากับ 0.8476 หมายความว่าระบบมีประสิทธิภาพทั้งในการตรวจจับวัตถุและทำนายว่าเป็นวัตถุบนภาพอยู่ที่ประมาณ 84.76%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.8 ประสิทธิภาพแบบจำลอง GCNet ResNet101 จาก Confusion Matrix

Classes	Actual	Predicted	TP	FP	FN	TN	Accuracy	Precision	Recall	F1
windshield	31	34	30	4	1	582	0.9919	0.9677	0.8824	0.9231
hood	38	41	39	2	0	576	0.9968	1.0000	0.9512	0.9750
front bumper	43	51	41	10	0	566	0.9838	1.0000	0.8039	0.8913
rear bumper	38	37	29	8	1	579	0.9854	0.9667	0.7838	0.8657
front doors	54	55	51	4	2	560	0.9903	0.9623	0.9273	0.9444
back doors	43	51	43	8	3	563	0.9822	0.9348	0.8431	0.8866
roof	25	27	19	8	0	590	0.9870	1.0000	0.7037	0.8261
trunk lid	23	22	20	2	0	595	0.9968	1.0000	0.9091	0.9524
quarters	51	52	46	6	0	565	0.9903	1.0000	0.8846	0.9388
headlight	48	52	42	10	0	565	0.9838	1.0000	0.8077	0.8936
taillight	37	33	30	3	0	584	0.9951	1.0000	0.9091	0.9524
slide mirror	61	64	54	10	0	553	0.9838	1.0000	0.8438	0.9153
rear window	17	19	16	3	0	598	0.9951	1.0000	0.8421	0.9143
fender	54	55	50	5	1	561	0.9903	0.9804	0.9091	0.9434
damage	23	24	13	11	1	592	0.9806	0.9286	0.5417	0.6842
background	0	0	0	0	85	532	0.8622	0.0000	0.0000	0.0000

จากตารางที่ 4.8 แสดงให้เห็นว่าแบบจำลอง GCNet ResNet101 มี Accuracy อยู่ในระดับสูง เช่น rear bumper, front doors, back doors, roof, trunk lid, quarters, headlight, taillight, slide mirror, rear window, fender แบบจำลองมีความแม่นยำ และประสิทธิภาพในการทำนายที่สูง โดยมีค่า Accuracy, Precision, Recall, และ F1-Score ใกล้เคียงกันและสูงมาก

ความแม่นยำในการตรวจจับคลาส damage ของแบบจำลอง GCNet ResNet101 ได้ Accuracy อยู่ที่ประมาณ 98.06% โดยค่า Precision 92.86% หมายความว่าจากตัวอย่างที่แบบจำลองทำนายว่าเป็นคลาส damage อยู่ทั้งหมด 24 ตัวอย่าง แต่มีเพียง 13 ตัวอย่างเท่านั้นที่เป็นคลาส damage จริง ส่วนค่า Recall แสดงให้เห็นว่าแบบจำลองสามารถตรวจจับตัวอย่างที่เป็นคลาส damage ได้เพียง 54.17% เท่านั้น ซึ่งหมายความว่ามีความเป็นไปได้ว่าตัวอย่างที่เป็นคลาส damage บางส่วนที่โมเดลไม่สามารถตรวจจับได้ นอกจากนี้ F1 Score เท่ากับ 0.6842 แสดงให้เห็นว่าแบบจำลองมีความสามารถในการทำนายและตรวจจับคลาส damage อยู่ที่ 68.42%

กรณีตัวอย่าง

ในงานวิจัยนี้เราได้ศึกษาและฝึกอบรมแบบจำลองจำนวน 8 แบบจำลอง (ตารางที่ 3.2) โดยผลการทดสอบทั้งสี่ชุดข้อมูลพบว่าแบบจำลองที่มีประสิทธิภาพแม่นยำที่สุดในการแบ่งส่วนพื้นที่ของรถยนต์คือ HTC ResNet50 การตรวจจับร่องรอยความเสียหายคือ GCNet ResNet101 จากนั้นเราจึงได้นำทั้งสองแบบจำลองมาฝึกอบรมในชุดข้อมูลการแบ่งส่วนพื้นที่และความเสียหาย พบว่าทั้งสองโมเดลที่เลือกไว้ทั้งนี้ อีกทั้งยังมีให้คำปรึกษาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบจำลองมีค่า mAP ที่ใกล้เคียงกันมาก โดย HTC ResNet50 มีค่า mAP 56.10% ในการตรวจจับวัตถุ และค่า mAP 53.00% ในการแบ่งส่วนภาพ ส่วน GCNet ResNet101 ในการประเมินผลลัพธ์มีค่า mAP 56.40% ในการตรวจจับวัตถุ และค่า mAP 53.20% ในการแบ่งส่วนภาพ

เนื่องจากทั้งสองแบบจำลองมีความแม่นยำเฉลี่ย (mAP) ที่ใกล้เคียงกันมาก แต่เมื่อตรวจสอบประสิทธิภาพการตรวจจับแต่ละคลาสจากตารางที่ 4.7 และ 4.8 พบว่า ในส่วนการตรวจจับร่องรอยความเสียหาย GCNet ResNet101 ได้ Accuracy อยู่ที่ 98.06% ค่า Precision 92.86% Recall ได้ 54.17% และ F1 Score เท่ากับ 68.42% ซึ่งมีค่าดีกว่า HTC ResNet50 ที่ได้ Accuracy อยู่ที่ 97.74% ค่า Precision 80.00% Recall ได้ 42.11% และ F1 Score เท่ากับ 55.17% ดังนั้นเราจึงเลือกแบบจำลอง GCNet ResNet101 มาใช้ในการพัฒนาเว็บแอปพลิเคชันต่อไป

ในกรณีตัวอย่างนี้เราได้ทดลองโดยใช้ภาพรถยนต์ที่มีความเสียหายด้านข้างทั้งหมด 18 ภาพรวม 23 รอยความเสียหาย ซึ่งประกอบด้วย 14 ภาพที่มี 1 รอยความเสียหายในภาพ, 3 ภาพที่มี 2 รอยความเสียหายในภาพ และ 1 ภาพที่มี 3 รอยความเสียหาย โดยพื้นที่ส่วนของรถด้านข้างที่นำมาทดลองประกอบด้วย 4 ส่วน ได้แก่ ประตูหน้า (front doors), ประตูหลัง (back doors), บังโคลนหลังหรือแผงที่ปิดล้อหลัง (quarters), บังโคลนหน้าหรือแผงที่ปิดล้อหน้า (fender) โดยในการทดลองนี้กำหนดให้ True Positives คือมีคะแนน IoU สูงกว่า 0.5 (สีแดงหมายถึงส่วนรถที่ไม่ถูกตรวจพบความเสียหาย)

ตารางที่ 4.9 กรณีตัวอย่างการทดลองใช้ GCNet ResNet101

ภาพที่	Classes	Actual	Perdicted	TP	FP
1	back doors, quarters	1	2	2	-
2	front doors, fender	1	2	1	1
3	front doors, back doors	1	2	2	-
4	front doors, back doors	1	1	1	-
5	front doors	1	0	-	-
6	front doors	1	0	-	-
7	back doors	1	1	-	1
8	front doors	1	1	1	-
9	back doors, quarters	2	1	1	-
10	back doors	1	1	-	1
11	front doors	1	1	-	1
12	front doors, back doors	1	2	2	-
13	front doors, back doors	2	2	-	2
14	back doors	1	1	1	-
15	front doors, fender	1	2	1	1
16	fender	3	1	1	-
17	quarters	2	2	-	2
18	front doors, back doors	1	2	2	2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งการตีพิมพ์โดยหน่วยงานและทางอ้างอิงถึงเจ้าของเอกสารทุกฉบับที่มีการนำไปใช้

จากตารางที่ 4.9 กรณีตัวอย่างการทดลองใช้ GCNet ResNet101 แสดงให้เห็นโดยแบ่งตามพื้นที่ส่วนของรถที่เกิดความเสียหายได้ดังนี้

ประตูหน้า (front doors) จากตารางแสดงให้เห็นว่ามี TP เท่ากับ 11 ซึ่งบ่งชี้ว่าแบบจำลองระบุความเสียหายที่ประตูหน้าได้ถูกต้อง 11 ครั้ง และมี FP เท่ากับ 3 หมายถึงมีการระบุความเสียหายที่ประตูหน้าไม่ถูกต้อง 3 ครั้ง

ประตูหลัง (back doors) จากตารางแสดงให้เห็นว่ามี TP เท่ากับ 6 ซึ่งบ่งชี้ว่าแบบจำลองระบุความเสียหายที่ประตูหน้าได้ถูกต้อง 6 ครั้ง และมี FP เท่ากับ 4 หมายถึงมีการระบุความเสียหายที่ประตูหน้าไม่ถูกต้อง 4 ครั้ง

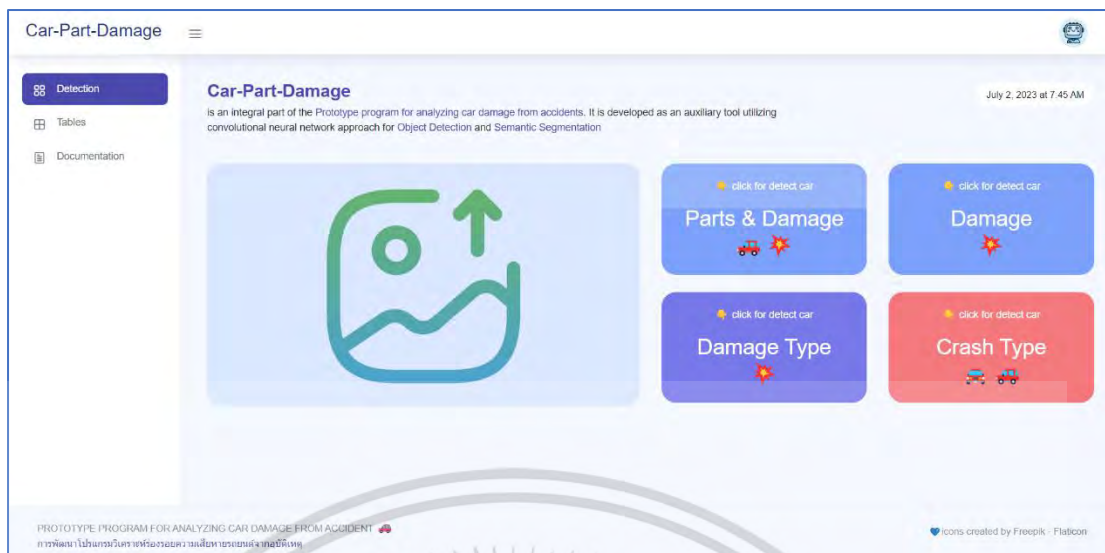
บังโคลนหลัง (quarters) จากตารางแสดงให้เห็นว่ามี TP เท่ากับ 3 ซึ่งบ่งชี้ว่าแบบจำลองระบุความเสียหายที่ประตูหน้าได้ถูกต้อง 3 ครั้ง และมี FP เท่ากับ 2 หมายถึงมีการระบุความเสียหายที่ประตูหน้าไม่ถูกต้อง 2 ครั้ง

บังโคลนหน้า (fender) จากตารางแสดงให้เห็นว่ามี TP เท่ากับ 1 ซึ่งบ่งชี้ว่าแบบจำลองระบุความเสียหายที่ประตูหน้าได้ถูกต้อง 1 ครั้ง และมี FP เท่ากับ 2 หมายถึงมีการระบุความเสียหายที่ประตูหน้าไม่ถูกต้อง 2 ครั้ง

4.1.7 ส่วนติดต่อกับผู้ใช้ของเว็บแอปพลิเคชัน

1) หน้าจอหลักของแอปพลิเคชัน

ในหน้าจอหลักของเว็บแอปพลิเคชันนั้นมีปุ่มสำหรับกดอัปโหลดรูปภาพที่ต้องการให้แบบจำลองตรวจจับ และมี 4 ปุ่มให้เลือกแบบจำลองที่ต้องการใช้งานโดยประกอบด้วย ปุ่ม Part & Damage สำหรับใช้แบบจำลองเพื่อตรวจจับส่วนพื้นที่ของรถและร่องรอยความเสียหาย พร้อมระบุส่วนของรถที่มีความเสียหาย, ปุ่ม Damage สำหรับใช้แบบจำลองเพื่อตรวจจับร่องรอยความเสียหายเท่านั้น, ปุ่ม Damage Type สำหรับใช้แบบจำลองเพื่อแบ่งประเภทร่องรอยความเสียหาย, ปุ่ม Crash Type สำหรับใช้แบบจำลองเพื่อแบ่งประเภทการเป็นผู้ชนจากร่องรอยความเสียหาย ดังรูปที่ 4.9

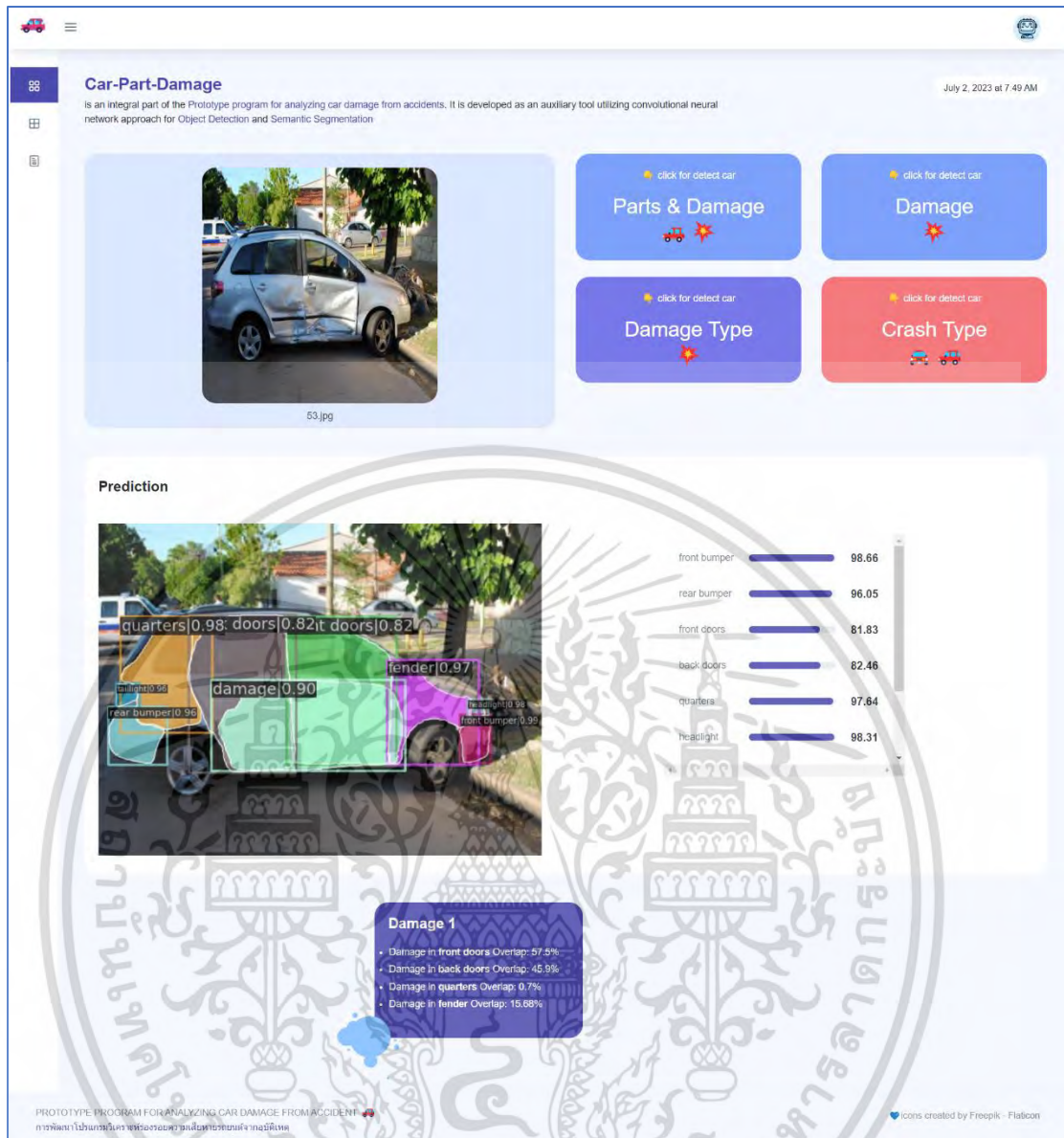


รูปที่ 4.9 หน้าจอหลักของแอปพลิเคชัน

2) หน้าจอแสดงผลการตรวจจับของแบบจำลอง

หน้าจอแสดงผลการตรวจจับของแบบจำลองจะอยู่ในหน้าเดียวกันกับหน้าจอหลักที่ใช้อัปโหลดภาพ โดยในรูปที่ 4.10 จะเป็นตัวอย่างเมื่ออัปโหลดภาพและกดปุ่ม Part & Damage สำหรับใช้แบบจำลองเพื่อตรวจจับส่วนพื้นที่ของรถและร่องรอยความเสียหาย โดยผลลัพธ์ที่ได้ประกอบด้วยภาพที่ถูกมาส์กผลการตรวจจับ รายชื่อคลาสที่ตรวจจับได้ และระบุรายการของคลาสร่องรอยความเสียหาย (damage) ที่ตรวจพบและระบุเปอร์เซ็นต์การทับซ้อนของ damage กับ ส่วนพื้นที่รถ (part) โดยในรูปที่ 4.10 ระบุว่าตรวจพบ 1 damage ที่อยู่บน front doors, back doors, quarters, fender ซึ่งมีเปอร์เซ็นต์การทับซ้อนดังที่ระบุในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 หน้าจอแสดงผลการตรวจจับของแบบจำลอง

3) หน้าจอแสดงข้อมูลตารางในฐานข้อมูล

หน้าจอแสดงข้อมูลตารางในฐานข้อมูลจะแสดงข้อมูลในตาราง ซึ่งประกอบด้วย ตารางที่ 1 ตารางรูปภาพ (images) ที่อัปโหลด ซึ่งประกอบด้วย รหัสภาพ ชื่อภาพ วัน/เวลาที่อัปโหลด ภาพ และ ปุ่มลบ, ตารางที่ 2 ตารางผลการทำนาย (prediction) ซึ่งประกอบด้วย รหัสการทำนาย รหัสภาพ รหัสแบบจำลองที่ใช้ ภาพที่ได้จากการตรวจจับ ปุ่มลบ และ JSON ที่เก็บผลการทำนาย, ตารางที่ 3 ซึ่งประกอบด้วย รหัสแบบจำลอง และ ชื่อแบบจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Images Table

Images table is a data table of the original uploaded images.

Delete All

ID	Image Name	Upload Date / Time	Preview	Delete
210	53.jpg	2023-07-02 07:48:20.820456		Delete
211	53.jpg	2023-07-02 07:49:38.920789		Delete
205	copart001.jpg	2023-07-02 07:46:44.864678		Delete
206	copart002.jpg	2023-07-02 07:47:00.683702		Delete

Prediction Table

prediction table is the prediction result data table of each image in each model.

Delete All

ID	Image ID	Model ID	Image	Delete	JSON
122	205	1		Delete	{ "annotations": [{"bbox": [160, 228, 134, 552, 734, 66, 558, 704, 67, 38, 26, 359, 88, 156, 127, 92, 969, 139, 876, 235, 96, 19, 1406], "score": 0.970957219600}
123	206	1		Delete	{ "annotations": [{"bbox": [163, 824, 602, 197, 265, 63, 112, 854, 0, 38, 62, 5, 317, 236, 297, 607, 42, 19, 127, 74, 91, 989, 135, 74, 22], "score": 0.866089344024}
124	207	1		Delete	{ "annotations": [{"bbox": [162, 0, 82, 26, 367, 198, 76, 7, 97, 6, 164, 55, 0, 76, 1, 342, 0, 107, 42, 167, 5, 131, 6, 77, 0, 477, 2, 94, 922], "score": 0.926451934337616,
125	208	1		Delete	{ "annotations": [{"bbox": [142, 469, 329, 338, 438, 78, 42, 54, 37, 27, 24, 61, 2, 90, 94, 76, 2, 87, 1, 0, 93, 7, 141, 6, 114, 1, 96, 7, 73, 4, 38], "score": 0.743145048618}

Model Table

prediction table is the prediction result data table of each image in each model.

ID	Model Name
1	part
2	damage
3	damagetype
4	crashtype

PROTOTYPE PROGRAM FOR ANALYZING CAR DAMAGE FROM ACCIDENT
การพัฒนาระบบประมวลผลภาพรถชนเพื่อวิเคราะห์ความเสียหาย

Icons created by Freepik - Flaticon

รูปที่ 4.11 หน้าจอแสดงข้อมูลตารางในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

งานวิจัยนี้มีจุดประสงค์ในการพัฒนาโปรแกรมต้นแบบสำหรับการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุเพื่อเป็นโปรแกรมเสริมการทำงานของการทำงานของพนักงานปกติจากบุคคลหรือบุคลากรที่รับผิดชอบในบริษัทประกันภัยรถยนต์ เพื่อลดเวลาและข้อผิดพลาดในการวิเคราะห์ร่องรอยความเสียหาย

5.1 สรุปผลการวิจัย

โปรแกรมต้นแบบสำหรับการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุมีการใช้งานผ่านเว็บแอปพลิเคชัน ซึ่งสามารถระบุชื่อประเภทของส่วนพื้นที่รถยนต์จากรูปภาพ ส่วนพื้นที่ของรถยนต์ที่มีร่องรอยความเสียหาย อีกทั้งระบุลักษณะหรือประเภทของร่องรอยความเสียหายที่เกิดขึ้น และแนะนำการเป็นผู้ชนจากร่องรอยความเสียหายของรถที่กระทำและรถที่ถูกกระทำ ซึ่งผลการทดสอบทั้งสี่ชุดข้อมูลพบว่าแบบจำลองที่มีประสิทธิภาพแม่นยำที่สุดในการแบ่งส่วนพื้นที่ของรถยนต์คือ HTC ResNet50 การตรวจจบบร่องรอยความเสียหายคือ GCNet ResNet101 การแบ่งประเภทร่องรอยความเสียหายคือ HTC ResNet50 การแบ่งประเภทการเป็นผู้ชนจากร่องรอยความเสียหายคือ HRNet เนื่องจากโปรแกรมต้องการระบุส่วนพื้นที่ของรถที่มีความความเสียหาย จึงได้มีการทดลองนำ HTC ResNet50 และ GCNet ResNet101 มาฝึกในชุดข้อมูลการแบ่งส่วนพื้นที่และความเสียหาย และพบว่า HTC ResNet50 และ GCNet ResNet101 มีค่าความแม่นยำใกล้เคียงกัน แต่เมื่อดู Confusion Matrix แล้วพบว่าทั้งสองแบบจำลองมีความแม่นยำในการตรวจจบบส่วนพื้นที่ของรถต่างกันเพียงเล็กน้อย และในส่วนของ การตรวจจบบร่องรอยความเสียหาย พบว่า ทั้งสองแบบจำลองมีประสิทธิภาพที่ค่อนข้างต่ำ แต่ GCNet ResNet101 มีผลของประสิทธิภาพดีกว่า ดังนั้นจึงได้เลือก GCNet ResNet101 มาใช้ในการพัฒนาเว็บแอปพลิเคชัน

5.2 ข้อจำกัด

1) ชุดข้อมูลที่ใช้ในการทดลองแต่ละแบบจำลองทั้งชุดข้อมูลพื้นที่ส่วนของรถและชุดข้อมูลร่องรอยความเสียหายมีจำนวนไม่มากพอ เฉพาะอย่างยิ่งชุดข้อมูลร่องรอยความเสียหายที่มีเพียง 268 รูปเท่านั้น ซึ่งเป็นจำนวนที่น้อยมากและส่งผลกระทบต่อการเรียนรู้ของแบบจำลองให้มีประสิทธิภาพที่ไม่ดีพอ

2) ชุดข้อมูลร่องรอยความเสียหายที่ใช้เป็นชุดข้อมูลสำหรับการแบ่งประเภทการเป็นผู้ชนจากร่องรอยความเสียหาย ผู้จัดทำเป็นผู้พิจารณาสร้างคำอธิบายประกอบภาพแบ่งประเภทด้วยตนเองไม่

ได้มาจากข้อมูลจริงที่เกิดขึ้นจึงทำให้ไม่อาจมั่นใจได้ว่าข้อมูลที่จัดทำนั้นแบ่งประเภทการชนได้ถูกต้อง และส่งผลต่อความเชื่อมั่นของแบบจำลองด้วย

3) ในการพัฒนาแบบจำลองได้มีการปรับปรุงหลายครั้ง มีการทดลองใช้จากหลายไลบรารี โอเพนซอร์สและมักพบปัญหาในการติดตั้งไลบรารีที่เกี่ยวข้องให้เหมาะสมกับเวอร์ชันที่ใช้ งาน อีกทั้งมีการอัปเดตเวอร์ชันแต่ไม่มีการอัปเดตคู่มือการใช้งาน ทำให้พบปัญหาในการใช้งานในหลาย ๆ ส่วน

4) โปรแกรมต้นแบบสำหรับการวิเคราะห์ร่องรอยความเสียหายรถยนต์จากอุบัติเหตุเป็นเว็บแอปพลิเคชันที่รับข้อมูลจากคลังรูปภาพเท่านั้น ยังไม่สามารถรับข้อมูลจากกล้องถ่ายภาพได้

5) การเตรียมข้อมูลภาพให้อยู่ใน coco format ค่อนข้างใช้เวลานานเนื่องจากต้องวาดเพื่อสร้างคำอธิบายประกอบภาพเอง

5.3 ข้อเสนอแนะ

1) จากที่ได้กล่าวถึงปัญหาชุดข้อมูลที่ใช้ควรมีการเพิ่มชุดข้อมูลภาพให้มากขึ้นทั้งชุดข้อมูลสำหรับการแบ่งประเภทการเป็นผู้ชนจากร่องรอยความเสียหาย

2) ปรับปรุงชุดข้อมูลสำหรับการแบ่งประเภทการเป็นผู้ชนจากร่องรอยความเสียหาย ให้เป็นข้อมูลที่ได้มาจากสถานการณ์จริงที่เกิดขึ้นจากอุบัติเหตุ

3) พัฒนาให้แอปพลิเคชันสามารถรับข้อมูลจากกล้องถ่ายภาพได้

เอกสารอ้างอิง

- [1] ACM. ImageNet Classification with Deep Convolutional Neural Networks. [Online]. Available : <https://cacm.acm.org/magazines/2017/6/217745-imagenet-classification-with-deep-convolutional-neural-networks/fulltext#F2>. 2017.
- [2] AIT Goehner GmbH. DEEP LEARNING. [Online]. Available : <https://www.ait.de/en/deep-learning/>. 2023.
- [3] Arel, D. C. Rose and T. P. Karnowski, "Deep Machine Learning - A New Frontier in Artificial Intelligence Research [Research Frontier]," in IEEE Computational Intelligence Magazine, vol. 5, no. 4, pp. 13-18, Nov. 2010, doi: 10.1109/MCI.2010.938364.
- [4] Athanasios Voulodimos et al. Deep Learning for Computer Vision: A Brief Review. [Online]. Available : <https://doi.org/10.1155/2018/7068349>. 2018.
- [5] Confusion Matrix in Machine Learning. [Online]. Available : <https://www.geeksforgeeks.org/confusion-matrix-machine-learning/>. 2023
- [6] D. Widjojo, E. Setyati and Y. Kristian, "Integrated Deep Learning System for Car Damage Detection and Classification Using Deep Transfer Learning," 2022 IEEE 8th Information Technology International Seminar (ITIS), Surabaya, Indonesia, 2022, pp. 21-26, doi: 10.1109/ITIS57155.2022.10010292.
- [7] D2L. Dive into Deep Learning. [Online]. Available : <https://d2l.ai/index.html>. 2023.
- [8] IBM. Convolutional Neural Networks. [Online]. Available : <https://www.ibm.com/topics/convolutional-neural-networks>. 2023.
- [9] J. Wang et al., "Deep High-Resolution Representation Learning for Visual Recognition," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 10, pp. 3349-3364, 1 Oct. 2021, doi: 10.1109/TPAMI.2020.2983686.
- [10] K. Chen et al. MMDetection: Open MMLab Detection Toolbox and Benchmark. [Online]. Available : <https://arxiv.org/abs/1906.07155>. 2019.
- [11] K. Chen et al., "Hybrid Task Cascade for Instance Segmentation," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 4969-4978, doi: 10.1109/CVPR.2019.00511.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [12] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2980-2988, doi: 10.1109/ICCV.2017.322.
- [13] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.
- [14] K. Patil, M. Kulkarni, A. Sriraman and S. Karande, "Deep Learning Based Car Damage Classification," 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 2017, pp. 50-54, doi: 10.1109/ICMLA.2017.0-179.
- [15] K. Sun, B. Xiao, D. Liu and J. Wang, "Deep High-Resolution Representation Learning for Human Pose Estimation," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 5686-5696, doi: 10.1109/CVPR.2019.00584.
- [16] Medium. Splash of Color: Instance Segmentation with Mask R-CNN and TensorFlow. [Online]. Available : <https://engineering.matterport.com/splash-of-color-instance-segmentation-with-mask-r-cnn-and-tensorflow-7c761e238b46>. 2018.
- [17] Parhizkar, M., Amirfakhrian, M. Car detection and damage segmentation in the real scene using a deep learning approach. Int J Intell Robot Appl 6, 231–245 (2022). <https://doi.org/10.1007/s41315-022-00231-5>.
- [18] Pasupa, K., Kittiworapanya, P., Hongngern, N. et al. Evaluation of deep learning algorithms for semantic segmentation of car parts. Complex Intell. Syst. 8, 3613–3625 (2022). <https://doi.org/10.1007/s40747-021-00397-8>.
- [19] Piotr Dollár et al. Microsoft COCO: Common Objects in Context. [Online]. Available : <https://arxiv.org/abs/1405.0312>. 2014.
- [20] Q. Zhang, X. Chang and S. B. Bian, "Vehicle-Damage-Detection Segmentation Algorithm Based on Improved Mask RCNN," in IEEE Access, vol. 8, pp. 6997-7004, 2020, doi: 10.1109/ACCESS.2020.2964055.
- [21] R. Singh, M. P. Ayyar, T. V. Sri Pavan, S. Gosain and R. R. Shah, "Automating Car Insurance Claims Using Deep Learning Techniques," 2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM), Singapore, 2019, pp. 199-207, doi: 10.1109/BigMM.2019.00-25.

- [22] S. C M, S. Kunjumon and N. R, "Car Damage Identification and Categorization Using Various Transfer Learning Models," 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2021, pp. 1097-1101, doi: 10.1109/ICOEI51242.2021.9452846.
- [23] Shirode, T. Rathod, P. Wanjari and A. Halbe, "Car Damage Detection and Assessment Using CNN," 2022 IEEE Delhi Section Conference (DELCON), New Delhi, India, 2022, pp. 1-5, doi: 10.1109/DELCON54057.2022.9752971.
- [24] Y. Cao, J. Xu, S. Lin, F. Wei and H. Hu, "GCNet: Non-Local Networks Meet Squeeze-Excitation Networks and Beyond," 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Korea (South), 2019, pp. 1971-1980, doi: 10.1109/ICCVW.2019.00246.
- [25] Z. Cai and N. Vasconcelos, "Cascade R-CNN: High Quality Object Detection and Instance Segmentation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 5, pp. 1483-1498, 1 May 2021, doi: 10.1109/TPAMI.2019.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้