

การจำแนกเกรดเรียงนกแอ่นกันรังด้วยขั้นตอนวิธีการเรียนรู้เชิงลึก

เอมี สูงศิริ

TNI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

บัณฑิตวิทยาลัย สาขาวิชาเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

ปีการศึกษา 2565

THE CLASSIFICATION OF EDIBLE-NEST SWIFTLETS USING DEEP LEARNING

Amy Sungsiri



A Thesis Submitted in Partial Fulfillment of the Requirements

For the Degree of Master of Science Program in Information Technology

Graduate School

Thai-Nichi Institute of Technology

Academic Year 2022

หัวข้อวิทยานิพนธ์

การจำแนกเกรตริงนกแอ่นกินรังด้วยขั้นตอนวิธี

การเรียนรู้เชิงลึก

โดย

เอมี สูงศิริ

สาขาวิชา

เทคโนโลยีสารสนเทศ

อาจารย์ที่ปรึกษาวิทยานิพนธ์

ดร.ศรายุทธ นนท์ศิริ

บัณฑิตวิทยาลัย สถาบันเทคโนโลยีไทย-ญี่ปุ่น อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็น
ส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาโทบริหารธุรกิจ

.....คณบดีบัณฑิตวิทยาลัย

(รองศาสตราจารย์ ดร.วรากร ศรีเชวงทรัพย์)

วันที่.....เดือน.....พ.ศ.....

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(รองศาสตราจารย์ ดร.อรรณพ หมั่นสกุล)

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.จิตติพร เลิศรัตน์เดชากุล)

.....กรรมการ

(ดร.ประมุข บุญเสียง)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์

(ดร.ศรายุทธ นนท์ศิริ)

เอมี สุงสิริ: การจำแนกเกรดรังนกแอ่นกินรังด้วยขั้นตอนวิธีการเรียนรู้เชิงลึก.
อาจารย์ที่ปรึกษา: ดร.ศรายุทธ นนท์ศิริ, 50 หน้า.

รังนกแอ่นกินรังเป็นที่ต้องการสูงเนื่องจากมีการบริโภคกันอย่างแพร่หลายในประเทศจีนและหลายประเทศในเอเชียตะวันออกเฉียงใต้ โดยอ้างว่าเป็นอาหารเพื่อสุขภาพ และมีประโยชน์ทางการแพทย์มากมาย การทำฟาร์มรังนกแอ่นกินรังเป็นอุตสาหกรรมที่ร่ำรวยเพราะถือว่าเป็นหนึ่งในผลิตภัณฑ์จากสัตว์ที่แพงที่สุด อย่างไรก็ตาม สำหรับการคัดเกรดคุณภาพ กระบวนการนี้ใช้เวลานานเนื่องจากผู้เชี่ยวชาญดำเนินการด้วยตนเอง โดยพิจารณาจากสี รูปร่าง พื้นผิว และขนาด

ในงานวิจัยนี้นำเสนอวิธีการของคอมพิวเตอร์วิทัศน์ (Computer Vision) เพื่อใช้สำหรับการจำแนกเกรดรังนกแอ่นกินรังด้วยการประยุกต์ใช้วิธีโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks: CNN) ถูกนำไปใช้เพื่อจำแนกภาพของรังนกแอ่นกินรัง ได้แก่ เกรดเอ (Grade A) เกรดบี (Grade B) และรังมุม (Corner) ในการทดลอง มีการเปรียบเทียบสถาปัตยกรรม CNN ทั้งหมด 7 สถาปัตยกรรม ได้แก่ MobileNetV2, Xception, DenseNet121, DenseNet169, DenseNet201, VGG16 และ VGG19

ผลวิจัยพบว่า DenseNet201 และ VGG16 มีค่าความถูกต้องเท่ากันคือ 99.34% ดังนั้นงานวิจัยนี้ DenseNet201 และ VGG16 มีความเหมาะสมมากที่สุดสำหรับการใช้วิเคราะห์จำแนกเกรดรังนกแอ่นกินรัง

บัณฑิตวิทยาลัย

สาขาวิชา เทคโนโลยีสารสนเทศ

ปีการศึกษา 2565

ลายมือชื่อนักศึกษา.....

ลายมือชื่ออาจารย์ที่ปรึกษา.....

AMY SUNGSIRI: THE CLASSIFICATION OF EDIBLE-NEST SWIFTLETS USING DEEP LEARNING. ADVISOR: DR. SARAYUT NONSIRI, 50 PP.

The edible bird's nest (EBN) is in high demand due to being widely consumed in China and many Southeast Asia countries. It claims that it is healthy food and there are many medical benefits. EBN farming is a lucrative industry because it is considered one of the most expensive animal products. However, for quality grading, this process is time-consuming because it is done manually by the expert based on color, shape, texture, and size.

This paper presents a method of computer vision to grade EBN automatically. Deep convolutional neural network (CNN) algorithms are applied to classify the images of edible-nest swiftlets in the following categories Grade A, Grade B, and Corner. In the experiment, seven CNN architectures are compared in terms of the performance: MobileNetV2, Xception, DenseNet121, DenseNet169, DenseNet201, VGG16, and VGG19. As a result, DenseNet201 and VGG16 have the same highest accuracy of 99.34%. Therefore, this work is useful for automatically quality grade EBN

Graduate School

Field of Study Information Technology

Academic Year 2022

Student's Signature.....

Advisor's Signature.....

กิตติกรรมประกาศ

การทำวิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดีนั้น ผู้วิจัยขอขอบคุณ ดร.ศรายุทธ นนท์ศิริ อาจารย์ที่ปรึกษาที่ได้สละเวลาในการให้การแนะนำ การสนับสนุน ตลอดจนแก้ไขข้อบกพร่องต่าง ๆ ของการทำวิทยานิพนธ์ตลอดจนสำเร็จสมบูรณ์ รวมทั้งขอขอบคุณคณาจารย์คณะและเจ้าหน้าที่บัณฑิตวิทยาลัยของสถาบันเทคโนโลยีไทย-ญี่ปุ่น ที่อำนวยความสะดวกต่าง ๆ ในการจัดทำและตรวจสอบเล่มวิทยานิพนธ์ฉบับนี้ให้สมบูรณ์และประสบความสำเร็จไปได้ด้วยดี

ทั้งนี้ผู้วิจัยตระหนักถึงความตั้งใจจริงและความทุ่มเท ขอขอบพระคุณคณะกรรมการสอบทุกท่าน ได้แก่ รองศาสตราจารย์ ดร.อรรณพ หมั่นสกุล ผู้ช่วยศาสตราจารย์ ดร.ฐิติพร เลิศรัตนเดชากุล และ ดร.ประมุข บุญเสียง ที่ได้กรุณาให้ข้อเสนอแนะและช่วยตรวจสอบข้อบกพร่องของวิทยานิพนธ์ฉบับนี้ให้เสร็จสมบูรณ์

เอมี สูงศิริ

TNI

THAI - JAPAN
INSTITUTE OF TECHNOLOGY

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ	จ
กิตติกรรมประกาศ	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ญ
สารบัญรูป.....	ฎ
 บทที่	
1 บทนำ.....	1
1.1 ที่มาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์.....	2
1.3 ขอบเขตงานวิจัย.....	2
1.4 ผลที่คาดว่าจะได้รับ	2
 2 เอกสารและงานวิจัยที่เกี่ยวข้อง	3
2.1 ลักษณะของรังนก	3
2.1.1 รังนกสีขาว.....	4
2.1.2 รังนกสีแดง	4
2.1.3 รังนกสีดำ.....	4
2.1.4 รังนกบ้าน.....	4
2.2 ความเป็นมาของปัญญาประดิษฐ์.....	4
2.3 ปัญญาประดิษฐ์.....	5
2.3.1 ปัญญาประดิษฐ์ (Artificial Intelligence).....	5
2.3.2 การเรียนรู้ของเครื่อง (Machine Learning).....	5

สารบัญ (ต่อ)

บทที่		หน้า
2	2.3.3 การเรียนรู้เชิงลึก (Deep Learning)	6
2.4	โครงข่ายประสาทเทียม.....	6
	2.4.1 ชั้นข้อมูลเข้า (Input Layer)	8
	2.4.2 ชั้นซ่อน (Hidden Layer)	8
	2.4.3 ชั้นข้อมูลออก (Output Layer)	8
2.5	โครงข่ายประสาทเทียมแบบคอนโวลูชัน	9
	2.5.1 ชั้นนำเข้า (Input Layer)	9
	2.5.2 ชั้นคอนโวลูชัน (Convolutional Layers)	9
	2.5.3 ชั้นพูลลิง (Pooling Layers).....	10
	2.5.4 ฟังก์ชันกระตุ้น (Activation Function)	10
	2.5.5 ชั้นการเชื่อมต่อ (Fully Connected Layers).....	12
	2.5.6 ชั้นนำออก (Output Layer)	12
2.6	การถ่ายโอนการเรียนรู้.....	12
	2.6.1 MobileNetV2	13
	2.6.2 Xception	14
	2.6.3 VGG	14
	2.6.4 DenseNet.....	15
2.7	การวัดประสิทธิภาพของโมเดล.....	16
	2.7.1 Confusion Matrix.....	17
	2.7.2 Accuracy	17
	2.7.3 Precision.....	18
	2.7.4 Sensitivity.....	18
	2.7.5 Specificity.....	18
	2.7.6 F1-Score.....	18
2.8	งานวิจัยที่เกี่ยวข้อง	18

สารบัญ (ต่อ)

บทที่		หน้า
3	วิธีการดำเนินงาน.....	14
	3.1 การศึกษาข้อมูลและเก็บรวบรวมข้อมูลรูปภาพ	25
	3.2 สร้างแบบจำลองสำหรับจำแนกเกรตริงนกแอ่นกินรัง	29
	3.3 การทดสอบประสิทธิภาพ	33
4	สรุปผลการดำเนินงาน การวิเคราะห์ และสรุปผลต่าง ๆ.....	34
	4.1 ผลการดำเนินงาน	34
	4.1.1 MobileNetV2	34
	4.1.2 Xception	36
	4.1.3 VGG16.....	37
	4.1.4 VGG19.....	39
	4.1.5 DenseNet121	40
	4.1.6 DenseNet169	41
	4.1.7 DenseNet201	45
5	สรุปและอภิปราย.....	45
	5.1 สรุปผลวิจัย.....	45
	5.2 ข้อเสนอแนะ.....	45
	5.3 ปัญหาและอุปสรรค	46
	บรรณานุกรม.....	47
	ประวัติย่อผู้วิจัย.....	50

สารบัญตาราง

ตาราง	หน้า
2.1 แสดงการเปรียบเทียบประสิทธิภาพของโมเดล.....	13
2.2 แสดงงานวิจัยที่เกี่ยวข้อง.....	21
3.1 จำนวนภาพเกรตริงนกแอ่นกินรังในแต่ละชุดข้อมูล.....	29
3.2 แสดงพารามิเตอร์ฝึกสอนโมเดล.....	29
4.1 แสดงผลจากการดำเนินงาน.....	34
4.2 ตารางแสดงประสิทธิภาพ ของ MobileNetV2.....	36
4.3 ตารางแสดงประสิทธิภาพ ของ Xception.....	37
4.4 ตารางแสดงประสิทธิภาพ ของ VGG16.....	38
4.5 ตารางแสดงประสิทธิภาพ ของ VGG19.....	40
4.6 ตารางแสดงประสิทธิภาพ ของ DenseNet121.....	41
4.7 ตารางแสดงประสิทธิภาพ ของ DenseNet169.....	42
4.8 ตารางแสดงประสิทธิภาพ ของ DenseNet201.....	44

TNI

THAI - NICHI INSTITUTE OF TECHNOLOGY

สารบัญรูป

รูปที่	หน้า
2.1 ส่วนย่อยของปัญญาประดิษฐ์.....	5
2.2 ความแตกต่างระหว่างโครงข่ายประสาทเทียมและการเรียนรู้เชิงลึก.....	6
2.3 โครงข่ายประสาทชีวภาพ (Biological neural network)	7
2.4 โครงข่ายประสาทเทียม (Artificial Neural Network).....	7
2.5 ส่วนประกอบของโครงข่ายประสาทเทียม.....	8
2.6 โครงสร้างโครงข่ายประสาทแบบคอนโวลูชัน.....	9
2.7 การคำนวณด้วยวิธีการคอนโวลูชัน.....	10
2.8 การคำนวณด้วยวิธีการคอนโวลูชัน.....	10
2.9 ฟังก์ชันซิกมอยด์ (Sigmoid Function)	11
2.10 ฟังก์ชันไฮเพอร์โบลิค (Hyperbolic Tangent Function: tanh).....	11
2.11 ฟังก์ชันเรกติไฟด์ไลน์เนี่ยยูนิต (Rectified Linear Units: ReLu).....	12
2.12 โครงสร้างของ MobileNetV2.....	13
2.13 โครงสร้างของ Xception.....	14
2.14 ฟิลเตอร์ 3×3 2 ชั้นครอบคลุมพื้นที่ 5×5 แล้ว.....	15
2.15 สถาปัตยกรรมของ VGG16.....	15
2.16 ตัวอย่างการเชื่อมกันของแต่ละชั้นของ DenseNet.....	16
2.17 ชุดข้อมูลทั่วไปแบ่งออกเป็นชุดฝึกสอน ชุดตรวจสอบและชุดทดสอบ.....	17
2.18 Confusion Matrix.....	17
3.1 ระบบจำแนกเกรตริงนกแอ่นกินรัง.....	24
3.2 ตัวอย่างชุดข้อมูล Grade A.....	25
3.3 ตัวอย่างชุดข้อมูล Grade B.....	25
3.4 ตัวอย่างชุดข้อมูล Corner.....	25
3.5 ตัวอย่างชุดข้อมูล Grade A พลิกแนวนอน.....	26
3.6 ตัวอย่างชุดข้อมูล Grade B พลิกแนวนอน.....	26

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.7 ตัวอย่างชุดข้อมูล Corner พลิกแนวนอน.....	26
3.8 ตัวอย่างชุดข้อมูล Grade A พลิกแนวตั้ง.....	27
3.9 ตัวอย่างชุดข้อมูล Grade B พลิกแนวตั้ง.....	27
3.10 ตัวอย่างชุดข้อมูล Corner พลิกแนวตั้ง.....	27
3.11 ตัวอย่างการหมุนภาพชุดข้อมูล Corner.....	28
2.12 การแบ่งชุดข้อมูลภาพถ่าย.....	28
3.13 แสดงพารามิเตอร์ฝึกสอนโมเดล MobileNetV2.....	30
3.14 แสดงพารามิเตอร์ฝึกสอนโมเดล Xception.....	30
3.15 แสดงพารามิเตอร์ฝึกสอนโมเดล DenseNet121.....	30
3.16 แสดงพารามิเตอร์ฝึกสอนโมเดล DenseNet169.....	31
3.17 แสดงพารามิเตอร์ฝึกสอนโมเดล DenseNet201.....	31
3.18 แสดงพารามิเตอร์ฝึกสอนโมเดล VGG16.....	32
3.19 แสดงพารามิเตอร์ฝึกสอนโมเดล VGG19.....	32
3.20 การทดสอบประสิทธิภาพของโมเดล.....	33
3.21 การหา Confusion Metrix ของโมเดล.....	33
3.22 การหาค่าความถูกต้องของโมเดล.....	33
4.1 แสดงค่า Confusion Matrix ของ MobileNetV2.....	35
4.2 กราฟแสดงค่า Accuracy และ Loss ของ MobileNetV2.....	35
4.3 แสดงค่า Confusion Matrix ของ Xception.....	36
4.4 กราฟแสดงค่า Accuracy และ Loss ของ Xception.....	37
4.5 แสดงค่า Confusion Matrix ของ VGG16.....	38
4.6 กราฟแสดงค่า Accuracy และ Loss ของ VGG16.....	38
4.7 แสดงค่า Confusion Matrix ของ VGG19.....	39
4.8 กราฟแสดงค่า Accuracy และ Loss ของ VGG19.....	39

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.9 แสดงค่า Confusion Matrix ของ DenseNet121.....	40
4.10 กราฟแสดงค่า Accuracy และ Loss ของ DenseNet121.....	41
4.11 แสดงค่า Confusion Matrix ของ DenseNet169.....	42
4.12 กราฟแสดงค่า Accuracy และ Loss ของ DenseNet169.....	42
4.13 แสดงค่า Confusion Matrix ของ DenseNet201.....	43
4.14 กราฟแสดงค่า Accuracy และ Loss ของ DenseNet201.....	43

TNI

TAI - NICHU INSTITUTE OF TECHNOLOGY

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

รังนกแอ่นเป็นผลิตภัณฑ์อาหารเพื่อสุขภาพ และจัดว่าเป็นอาหารบำรุงที่ดีที่สุดในชนิดหนึ่ง ซึ่งในปัจจุบันมีความต้องการสูง เนื่องจากการบริโภคกันอย่างแพร่หลายในประเทศจีน และประเทศในเอเชียตะวันออกเฉียงใต้หลายแห่ง เนื่องจากมีรสชาติเข้มข้น มีโปรตีนสูง และหาได้ยาก นกนางแอ่นที่พบมากที่สุดแบ่งออกเป็น 3 ชนิด คือ นกนางแอ่นรังขาว (*Aerodramus fuciphagus*) นกนางแอ่นรังดำ (*Aerodramus Maximus*) และนกนางแอ่นรังกินได้ [1] รังนกแอ่นอุดมไปด้วยสารอาหาร และมีประโยชน์ทางการแพทย์มากมาย [2] ซึ่งองค์ประกอบหลัก ๆ จะประกอบไปด้วยโปรตีน คาร์โบไฮเดรต และโปรตีน โดยพบมากจะประกอบด้วย 50-60% และ 30-40% ตามลำดับ [3] รังนกแอ่น ส่วนใหญ่ตั้งอยู่ในภาคใต้ของประเทศจีน และประเทศในเอเชียตะวันออกเฉียงใต้ เช่น ไทย มาเลเซีย อินโดนีเซีย เวียดนาม เมียนมาร์ กัมพูชา และฟิลิปปินส์ อย่างไรก็ตาม คุณค่าทางโภชนาการ องค์ประกอบ และคุณภาพของรังนกแอ่น นั้นแตกต่างกันไปตามภูมิศาสตร์ และชนิดของนกนางแอ่น [4] ความต้องการรังนกที่เพิ่มมากขึ้นทำให้เกิดมูลค่าทางเศรษฐกิจจาก 1000.00 ดอลลาร์สหรัฐฯ เป็น 10,000.00 ดอลลาร์สหรัฐฯ ต่อกิโลกรัม จึงทำให้รังนกแอ่นแต่ละชนิดมีราคาที่ไม่เท่ากัน ขึ้นอยู่กับเกรด รูปร่าง สายพันธุ์ แหล่งกำเนิด และสีของรังนก ซึ่งอาจมีการจำหน่ายไม่ตรงประเภทในท้องตลาด

โดยงานวิจัยนี้จะนำความรู้ทางด้านการเรียนรู้เชิงลึก (Deep Learning) มาเพื่อช่วยแก้ปัญหาในการจำแนกเกรดรังนกแอ่นกินรัง เนื่องจากการเรียนรู้เชิงลึกได้รับความสนใจและมีความก้าวหน้าเป็นอย่างมาก โดยเฉพาะการใช้โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Network) โดยการใช้โครงข่ายประสาทแบบคอนโวลูชันจะเป็นการจำลองการมองเห็นของมนุษย์ ที่มองพื้นที่เป็นที่ย่อย ๆ จากนั้นนำกลุ่มของพื้นที่ย่อย ๆ นั้นมาผสมกันซึ่งมีจุดเด่นที่สามารถสกัดคุณลักษณะเด่นที่สำคัญ (Feature Extraction) แล้วจึงนำคุณลักษณะนั้นมาสร้างโมเดล โดยประเภทของรังนกแอ่นกินรังที่ใช้ในงานวิจัยมี 3 ประเภท คือ เกรดเอ (Grade A) เกรดบี (Grade B) และรังมุม (Corner)

1.2 วัตถุประสงค์

เพื่อศึกษาและเปรียบเทียบประสิทธิภาพโครงข่ายประสาทเทียม เพื่อนำไปใช้ในการจำแนกเกรตริงนกแอ่นกินรัง

1.3 ขอบเขตงานวิจัย

1. รูปภาพรังกนกแอ่นกินรังที่ใช้ในงานวิจัย ได้ทำการรวบรวมเกรตริงนกแอ่นกินรัง 3 ประเภท คือ เกรตเอ (Grade A) เกรตบี (Grade B) และรังมุม (Corner)
2. ใช้เทคนิคการเรียนรู้เชิงลึก ด้วยวิธีคอนโวลูชันนอลนิวรอลเน็ตเวิร์ก (CNN) สำหรับจำแนกเกรตริงนกแอ่นกินรัง
3. ทำการเปรียบเทียบประสิทธิภาพกับ 7 สถาปัตยกรรม ได้แก่ MobileNetV2, Xception, DenseNet121, DenseNet169, DenseNet201, VGG16, และ VGG19
4. ตรวจสอบประสิทธิภาพของโมเดล จาก Confusion Metrix, Precision, Recall, F1-Score และค่าความแม่นยำของโมเดล

1.4 ผลที่คาดว่าจะได้รับ

1. สามารถจำแนกเกรตริงนกแอ่นกินรังได้ถูกต้อง
2. สามารถหาแบบจำลองโมเดลสำหรับจำแนกเกรตริงนกแอ่นกินรังที่เหมาะสม

บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

ในการศึกษาวิจัยครั้งนี้ ผู้วิจัยได้ทำการศึกษาบนพื้นฐานของแนวคิดและทฤษฎีที่เกี่ยวข้อง เพื่อใช้เป็นแนวทางและข้อมูลสำหรับการวิจัย โดยมีสาระสำคัญดังนี้

- 2.1 ลักษณะของรังนก
- 2.2 ความเป็นมาของปัญหาประดิษฐ์
- 2.3 ปัญหาประดิษฐ์
- 2.4 โครงข่ายประสาทเทียม
- 2.5 โครงข่ายประสาทเทียมแบบคอนโวลูชัน
- 2.6 การถ่ายโอนการเรียนรู้
- 2.7 การวัดประสิทธิภาพของโมเดล
- 2.8 งานวิจัยที่เกี่ยวข้อง

2.1 ลักษณะของรังนก

รังนก (Bird's Nest) เกิดจากสารคัดหลั่งจากต่อมน้ำลาย (Salivary Glands) ของนกแอ่นกินรัง ซึ่งมีชื่อวิทยาศาสตร์ว่า *Aerodramus germani* อยู่ในสกุล *Aerodramus* หรือสกุล *Collocalia* ปัจจุบันรังนกมีมูลค่าสูงมาก และกำลังได้รับความนิยม เพราะช่วยบำรุงกำลังให้ร่างกายสดชื่น แข็งแรง แก้ไอ แก้อาการอ่อนเพลีย ไอเป็นเลือด ขับเสมหะ [1] ในประเทศไทยมีนกแอ่นกินรังจำนวน 12 สายพันธุ์ ซึ่งรังนกที่มีมูลค่า และมีความสำคัญทางเศรษฐกิจคือ นกแอ่นกินรัง ซึ่งในประเทศไทยมีอยู่ด้วยกัน 3 ชนิด คือ นกแอ่นกินรัง (*Collocalia Fuciphaga*) นกแอ่นกินรังตะโพกขาว (*Collocalia Germani*) และนกแอ่นรังดำ (*Collocalia Maxima*) ซึ่งรังของนกนางแอ่นทั้งสามชนิดนั้นสามารถนำมารับประทานได้ โดยรังนกทำมาจากน้ำลายของนกนางแอ่นกินรัง ผลิตมาจากต่อมน้ำลายของพ่อแม่ นกก่อนการผสมพันธุ์ และใช้เป็นที่พักไข่ อีกทั้งยังใช้เป็นที่อยู่ของลูกนกก่อนที่จะเริ่มหัดบิน ส่วนประกอบของรังนก ประมาณ 85–90% เป็นน้ำลาย และ 3–15% เป็นขนอ่อน ซึ่งรังนกมี 4 ประเภท ได้แก่ รังนกสีขาว, รังนกสีแดง, รังนกสีดำ และรังนกบ้าน

2.1.1 รังนกสีขาว

รังนกสีขาวเป็นรังนกได้รับความนิยมมากในประเทศไทย เพราะมีคุณภาพดีและมีราคาสูง เป็นรังนกที่เก็บจากถ้ำตามธรรมชาติมีหลากหลายสี ขึ้นอยู่กับแหล่งอาหาร และพันธุ์ของนก โดยเนื้อรังนกจะสะอาดและมีเนื้อหนา

2.1.2 รังนกสีแดง

รังนกสีแดงเป็นรังนกที่มีสีแดง เนื่องจากปฏิกิริยาเคมีของสารประกอบประเภททองแดงที่มีปะปนอยู่ภายในผนังถ้ำ จึงทำให้รังนกมีสีแดงปน

2.1.3 รังนกสีดำ

รังนกสีดำเป็นรังนกที่มีคุณภาพและราคาต่ำกว่าชนิดอื่น ๆ เนื่องจากมีเศษขนของนกเจือปนอยู่มาก

2.1.4 รังนกบ้าน

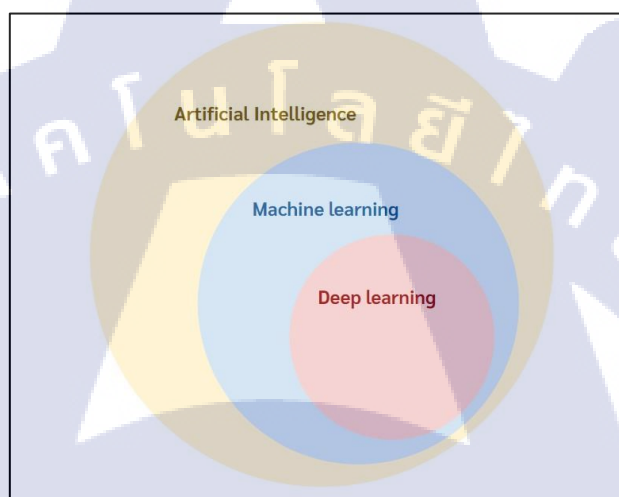
รังนกบ้านเป็นรังนกที่นกนางแอ่น สร้างไว้ตามบ้านเรือนของผู้คน มีสีขาว บางและขนาดเล็ก

2.2 ความเป็นมาของปัญญาประดิษฐ์

ในปี ค.ศ.1950 ได้มีการเริ่มศึกษาปัญญาประดิษฐ์ โดยอาจารย์จากประเทศอเมริกาและอังกฤษ จนในปี ค.ศ. 1956 ได้มีการกำหนดนิยามของปัญญาประดิษฐ์ โดย John McCarthy ได้มีการศึกษาและพัฒนางานด้านปัญญาประดิษฐ์ อีกทั้งมีการตั้งเกณฑ์ทดสอบเพื่อที่จะระบุว่าเครื่องจักรกลหรือระบบคอมพิวเตอร์สามารถเลียนแบบการคิดได้เหมือนสิ่งมีชีวิต ออกมาโดย Alan Turing เป็นนักคณิตศาสตร์ชาวอังกฤษ แต่จนบัดนี้เครื่องจักรกลหรือระบบคอมพิวเตอร์ก็ยังไม่สามารถผ่านเกณฑ์ของ Alan Turing ได้เลย ณ ปัจจุบันระบบปัญญาประดิษฐ์ยังไม่สามารถสร้างคำตอบที่แปลกใหม่หรือคำตอบที่มาจากการคิดค้นขึ้นมาใหม่ของระบบเองได้ ทำได้เพียงแค่เป็นการลอกเลียนความสามารถของสิ่งมีชีวิตได้เท่านั้น

2.3 ปัญญาประดิษฐ์

ปัญญาประดิษฐ์ทำงานโดยรวบรวมข้อมูลปริมาณมหาศาลด้วยความรวดเร็ว ประมวลผลซ้ำ ๆ ผ่านขั้นตอนการประมวลผลที่ชาญฉลาด อันช่วยให้ซอฟต์แวร์สามารถเรียนรู้จากรูปแบบและลักษณะของข้อมูลได้อย่างอัตโนมัติ [2] ซึ่งปัญญาประดิษฐ์ถูกแบ่งออกเป็น 3 สาขาย่อย คือ ปัญญาประดิษฐ์ (Artificial Intelligence) การเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning) ดังรูปที่ 2.1



รูปที่ 2.1 ส่วนย่อยของปัญญาประดิษฐ์

2.3.1 ปัญญาประดิษฐ์ (Artificial Intelligence)

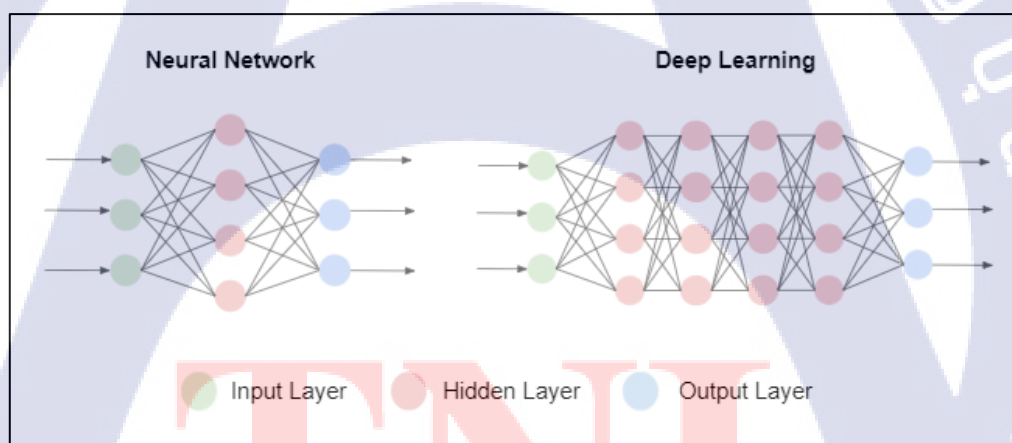
ปัญญาประดิษฐ์ (Artificial Intelligence) เป็นศาสตร์แขนงหนึ่งทางด้านวิทยาศาสตร์และเทคโนโลยีที่มีพื้นฐานมาจากวิชาวิทยาการคอมพิวเตอร์ ชีววิทยา จิตวิทยา ภาษาศาสตร์ คณิตศาสตร์ และวิศวกรรมศาสตร์ เป้าหมายคือ การพัฒนาระบบคอมพิวเตอร์ให้มีพฤติกรรมเลียนแบบมนุษย์ รวมทั้งเลียนแบบความเป็นอัจฉริยะของมนุษย์

2.3.2 การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง (Machine Learning) ในการสร้างแบบการวิเคราะห์แบบอัตโนมัติ โดยใช้วิธีการจากโครงข่ายประสาทเทียม สถิติ การวิจัยดำเนินการ (operations research) และการค้นหาแบบแผน (pattern) ข้อมูลเชิงลึกที่ซ่อนอยู่ในข้อมูลโดยไม่จำเป็นต้องเขียนโปรแกรมในการกำหนดเงื่อนไขแบบตายตัว

2.3.3 การเรียนรู้เชิงลึก (Deep Learning)

การเรียนรู้เชิงลึก (Deep Learning) เป็นวิธีการเรียนรู้ข้อมูลแบบอัตโนมัติ ที่มีแรงบันดาลใจหรือเลียนแบบการทำงานของโครงข่ายประสาทของสิ่งมีชีวิต (Neurons) ซึ่งเป็นการนำระบบโครงข่ายประสาทเทียม (Artificial Neural Network) มาเชื่อมต่อซ้อนทับกันจำนวนหลายชั้น (Layer) และทำการเรียนรู้จากข้อมูลตัวอย่าง ซึ่งข้อมูลดังกล่าวจะถูกนำไปใช้ในการตรวจจับรูปแบบ (Pattern) หรือจำแนกข้อมูล (Classification) โดยโครงข่ายประสาทเทียมแบบดั้งเดิมจะมีจำนวน Hidden layer เพียงไม่กี่ชั้น เพื่อที่จะทำให้ Neural Network ของเรานั้นสามารถคิดและประมวลผลซับซ้อนได้เหมือนสมองมนุษย์ ชั้นที่เป็น Hidden Layer จึงต้องมีหลาย ๆ ชั้น ให้มันส่งข้อมูลประมวลผลต่อ ๆ กันไป ทำให้มันสามารถคำนวณอะไรที่ ซับซ้อนได้เยอะขึ้น เนื่องจากยังไม่มีข้อมูลสอน (Training Data) หรือความสามารถด้านคอมพิวเตอร์ยังไม่สูงพอ อย่างไรก็ตาม ในปัจจุบันเทคโนโลยีได้มีการพัฒนามากขึ้นคอมพิวเตอร์มีประสิทธิภาพมากยิ่งขึ้น จึงทำให้มีข้อมูลชั้นของโครงข่ายได้ง่ายขึ้นและมากขึ้น ยังมีข้อกันหลายชั้นโครงข่ายก็ยังคงมีความซับซ้อนและลึกขึ้นจึงเป็นที่มาของ การเรียนรู้เชิงลึก ดังรูปที่ 2.2

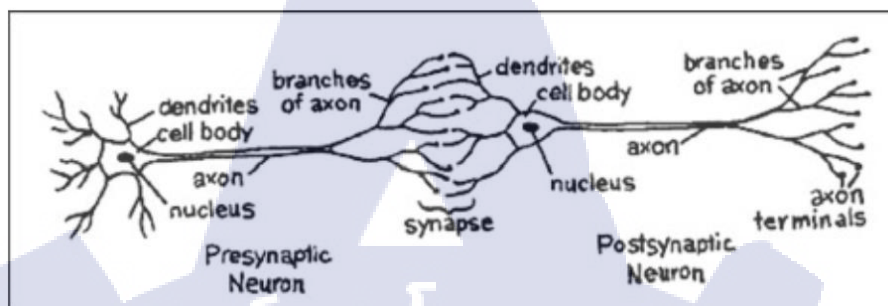


รูปที่ 2.2 ความแตกต่างระหว่างโครงข่ายประสาทเทียมและการเรียนรู้เชิงลึก

2.4 โครงข่ายประสาทเทียม

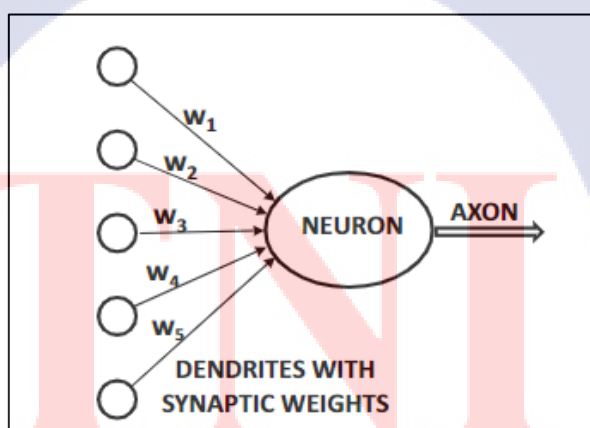
โครงข่ายประสาทเทียม (Artificial neural network : ANN) [3] เป็นเทคนิคการเรียนรู้ของเครื่องที่ได้รับความนิยม ซึ่งได้รับแรงบันดาลใจและเลียนแบบกลไกการเรียนรู้ในสิ่งมีชีวิตทางชีววิทยา ระบบประสาทของมนุษย์ประกอบด้วยเซลล์ซึ่ง เรียกว่า เซลล์ประสาท (Neuron) เซลล์

ประสาทเชื่อมต่อกันโดยใช้เดนไดรต์ (Dendrite) และ แอซซอน (Axon) ซึ่งบริเวณที่เชื่อมต่อระหว่าง แอซซอนและเดนไดรต์ เรียกว่า ซินแนปติก (Synapses) ดังรูปที่ 2.3

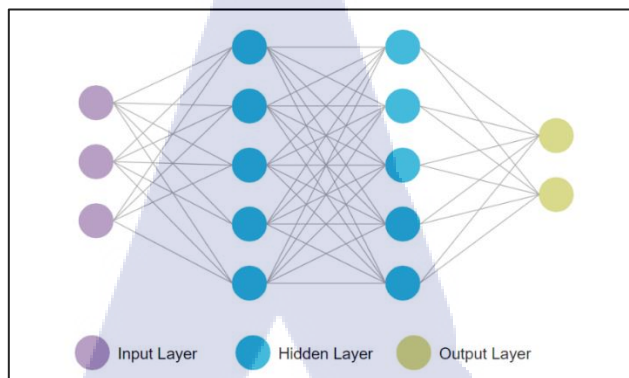


รูปที่ 2.3 โครงข่ายประสาทชีวภาพ (Biological neural network)

จุดแข็งของการเชื่อมต่อแบบซินแนปติกมักจะเปลี่ยนไปตามสิ่งเร้าภายนอก การเปลี่ยนแปลงนี้เป็นวิธีที่การเรียนรู้เกิดขึ้นในสิ่งมีชีวิต ซึ่งกลไกทางชีววิทยานี้ถูกเลียนแบบขึ้นในโครงข่ายประสาทเทียม ซึ่งมีหน่วยคำนวณที่เรียกว่าเซลล์ประสาท เราจะเรียกแทนว่า "โครงข่ายประสาทเทียม" เพื่ออ้างถึงโครงข่ายประสาทเทียม แทนที่จะเป็นโครงข่ายทางชีววิทยา หน่วยคำนวณเชื่อมต่อกันผ่านค่าน้ำหนัก ซึ่งทำหน้าที่เหมือนกัน ดังรูปที่ 2.4



รูปที่ 2.4 โครงข่ายประสาทเทียม (Artificial Neural Network)



รูปที่ 2.5 ส่วนประกอบของโครงข่ายประสาทเทียม

ส่วนประกอบของโครงข่ายประสาทเทียม จากรูปที่ 2.5 จะเห็นว่าส่วนประกอบของโครงข่ายประสาทเทียมประกอบด้วย 3 ส่วนหลัก ดังนี้

2.4.1 ชั้นข้อมูลเข้า (Input Layer)

ชั้นข้อมูลเข้า (Input Layer) ที่ชั้นนี้จะเป็นข้อมูลเข้า จำนวนของโหนดในชั้นนี้ขึ้นอยู่กับจำนวนของ ข้อมูลเข้าว่ามีข้อมูลอะไรบ้างที่จะนำเข้ามาคิดในแบบจำลอง เช่น ถ้าข้อมูลของลูกค้าเป็นข้อมูลเข้าที่ประกอบด้วย อายุ เพศ จังหวัดที่อาศัย รวมทั้งสิ้น 4 อย่าง ดังนั้นชั้นข้อมูลเข้าก็จะมี 4 โหนด ซึ่งอาจจะเรียก ปัจจัยที่นำมาวิเคราะห์เหล่านี้ว่าคุณลักษณะ (Feature)

2.4.2 ชั้นซ่อน (Hidden Layer)

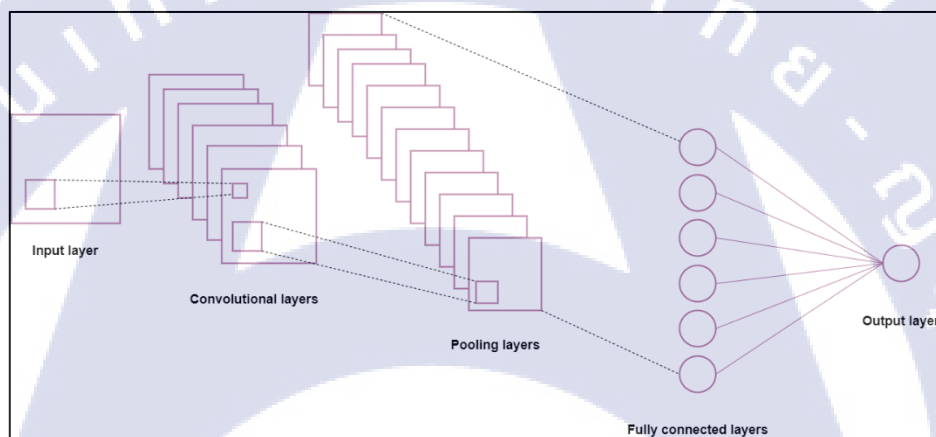
ชั้นซ่อน (Hidden Layer) เป็นชั้นที่อยู่ระหว่างกลาง ซึ่งจะมีผลอย่างมากต่อประสิทธิภาพในการเรียนรู้ของแบบจำลอง ซึ่งในชั้นซ่อนนั้นจะมีกี่ชั้นก็ได้ และแต่ละชั้นจะมีจำนวนของนิวรอนจำนวนเท่าไรก็ได้ ซึ่งการเพิ่มชั้นและจำนวนนิวรอนจะส่งผลต่อการทำงานของโมเดล ในส่วนของชั้นซ่อนมีการทำงาน เปรียบเสมือนส่วนที่เรียนรู้ข้อมูลเชิงลึก

2.4.3 ชั้นข้อมูลออก (Output Layer)

ชั้นข้อมูลออก (Output Layer) ชั้นที่จะนำเอาข้อมูลจากการคำนวณไปใช้ และจำนวนของโหนดในชั้นนี้ขึ้นอยู่กับรูปแบบของข้อมูลออกที่จะเอาไปใช้ ตัวอย่างเช่น ถ้างานที่ทำเป็น สมการถดถอย (Regression) จะกำหนดให้ชั้นข้อมูลออกเป็นแบบ 1 โหนด เพราะต้องการคำตอบเพียงค่าเดียว หากเป็นหลายค่าก็เพิ่มไปตามที่ต้องการ เช่น ในบางงานอาจต้องการทำนายหาตำแหน่งของภาพในแกน x และ y พร้อม ๆ กัน ในกรณีนี้ก็ต้องกำหนดชั้นข้อมูลออกเป็น 2 โหนด เป็นต้น

2.5 โครงข่ายประสาทเทียมแบบคอนโวลูชัน

โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network: CNN) [4] เป็นโครงข่ายประสาทเทียมแบบ feed-forward โครงสร้างของโครงข่ายประสาทเทียมเลียนแบบเซลล์ประสาทในสมองของมนุษย์ ข้อดีของโครงข่ายประสาทเทียมแบบคอนโวลูชันคือโมเดลจะสามารถเรียนรู้และเลือกลักษณะเด่นของภาพได้ดีกว่ามนุษย์ จึงทำให้ผลจากการประมวลผลของโมเดลมีความแม่นยำมากกว่า ซึ่งรูปภาพที่เข้า CNN จะถูกแปลงเป็นเมทริกซ์ (Matrix) โดยโครงสร้างของ CNN จะประกอบด้วย ชั้นนำเข้า (Input Layer) ชั้นคอนโวลูชัน (Convolutional Layers) ชั้นพูลลิง (Pooling Layers) ฟังก์ชันการเปิดใช้งาน (Activation Function) ชั้นการเชื่อมต่อ (Fully Connected Layers) และชั้นนำออก (Output Layer) ดังรูปที่ 2.6



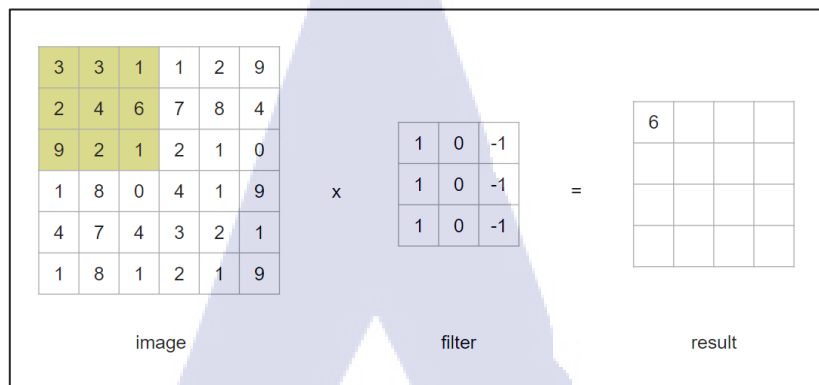
รูปที่ 2.6 โครงสร้างโครงข่ายประสาทแบบคอนโวลูชัน

2.5.1 ชั้นนำเข้า (Input Layer)

ชั้นนำเข้าของ input data เช่น เป็นรูปภาพ matrix ขนาด $64 \times 64 \times 3$ pixel เป็นรูปภาพแบบสี RGB

2.5.2 ชั้นคอนโวลูชัน (Convolutional Layers)

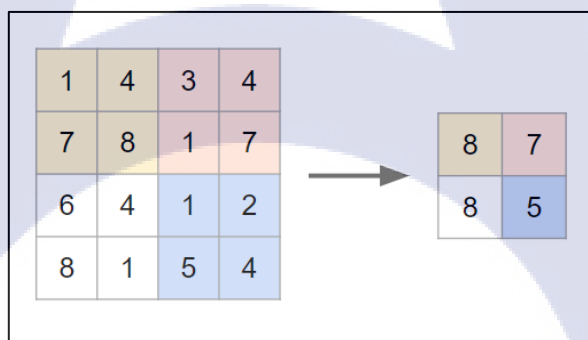
ทำหน้าที่สกัดเอาส่วนต่าง ๆ ของภาพออกมาโดยหาจำนวนผลลัพธ์ของนิวรอน (Neuron) ที่เชื่อมต่อจากพื้นที่ส่วนย่อย ๆ ของภาพด้วยวิธีการคอนโวลูชัน เพื่อให้โมเดลสามารถเรียนรู้ลักษณะของภาพได้อย่างมีประสิทธิภาพและมีแม่นยำมากยิ่งขึ้น ดังรูปที่ 2.7



รูปที่ 2.7 การคำนวณด้วยวิธีการคอนโวลูชัน

2.5.3 ชั้นพูลิ่ง (Pooling Layers)

เป็นการสกัดเอาส่วนที่สำคัญที่สุดของข้อมูล โดยจะนำค่าสูงสุดของช่องเก็บไว้ในส่วนนำออก (Output) เรียกว่า ค่าสูงสุดพูลิ่ง (Max Pooling) ดัง รูปที่ 2.8

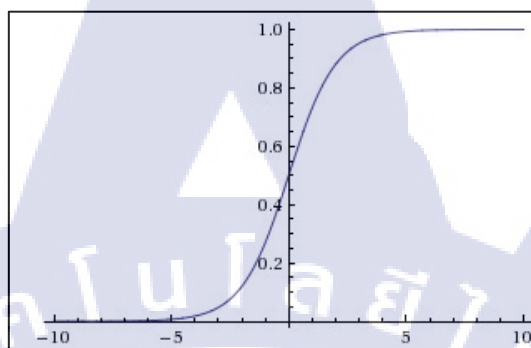


รูปที่ 2.8 การคำนวณด้วยวิธีการคอนโวลูชัน

2.5.4 ฟังก์ชันกระตุ้น (Activation Function)

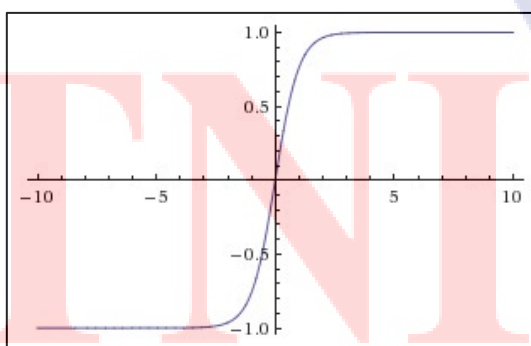
ฟังก์ชันกระตุ้น [5] จะใช้สำหรับการแบ่งประเภท การตัดสินใจหรือการแบ่งรูปแบบ โดยจะใช้ในการกำหนดข้อมูลออก (Output) ของโครงข่ายประสาทเทียม (Neural Network) ฟังก์ชันกระตุ้นที่นิยมใช้มีหลากหลายรูปแบบ แต่ที่นิยมนำมาใช้ในโครงข่ายประสาทเทียมแบบคอนโวลูชัน ได้แก่ ฟังก์ชันซิกมอยด์ (Sigmoid Function) ฟังก์ชันไฮเพอร์โบลิค (Hyperbolic Tangent Function) ฟังก์ชันเรกติไฟด์ไลน์เนี่ยยูนิต (Rectified Linear Units Function)

1) ฟังก์ชันซิกมอยด์ (Sigmoid Function) เป็นฟังก์ชันที่มีลักษณะเป็นเส้นโค้งที่มีรูปเป็นตัวเอส (S-curve) จะมีค่าระหว่าง 0 และ 1 ดังรูปที่ 2.9 ฟังก์ชันนี้จึงเหมาะที่จะนำไปประยุกต์ใช้ในการทำนายความน่าจะเป็น (Probability) ของข้อมูลออก (Output)



รูปที่ 2.9 ฟังก์ชันซิกมอยด์ (Sigmoid Function)

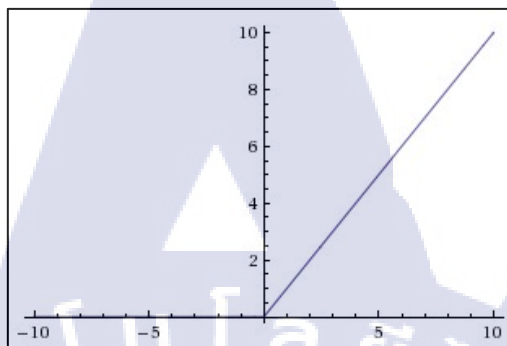
2) ฟังก์ชันไฮเพอร์โบลิค (Hyperbolic Tangent Function: $\tanh$) เป็นฟังก์ชันที่มีลักษณะการทำงานคล้ายฟังก์ชันซิกมอยด์ (Sigmoid Function) แต่มีประสิทธิภาพดีกว่า จะมีค่าระหว่าง $[-1, 1]$ ดังรูปที่ 2.10 ซึ่งข้อดีของฟังก์ชันไฮเพอร์โบลิคแทงก์เจนต์คือ สามารถแปลงค่าข้อมูลเข้าที่มีค่าเป็นลบมาก ๆ ให้เป็นข้อมูลออกที่ติดลบได้ ส่วนข้อมูลที่มีค่าเป็นศูนย์จะถูกแปลงเป็นข้อมูลออกที่มีค่าใกล้ศูนย์ (near-zero output)



รูปที่ 2.10 ฟังก์ชันไฮเพอร์โบลิค (Hyperbolic Tangent Function: $\tanh$)

3) ฟังก์ชันเรคตีไฟด์ไลน์เนียยูนิต (Rectified Linear Units: ReLu) เป็นฟังก์ชันที่นิยมใช้งานมากที่สุดในโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks:

CNN) ซึ่งฟังก์ชันนี้จะมีค่าอยู่ระหว่าง $[-\infty, \infty)$ หมายถึงถ้าข้อมูลเข้ามีค่ามากกว่าศูนย์ ข้อมูลออกจากเป็นค่าบวก และถ้าข้อมูลเข้ามีค่าศูนย์หรือติดลบ ข้อมูลออกจะมีค่าเป็นศูนย์



รูปที่ 2.11 ฟังก์ชันเรกติไฟด์ไลน์เนี่ยยูนิต (Rectified Linear Units: ReLu)

2.5.5 ชั้นการเชื่อมต่อ (Fully Connected Layers)

เป็นชั้นที่ประกอบด้วย Hidden Layer และ Output Layer ของโครงข่ายประสาทเทียม

2.5.6 ชั้นนำออก (Output Layer)

เป็นชั้นที่นำข้อมูลออก ซึ่งเป็นผลลัพธ์การเรียนรู้ (Training) หรือจำแนกประเภทของวัตถุของโมเดล

2.6 การถ่ายโอนการเรียนรู้

การถ่ายโอนข้อมูล (Transfer Learning) [6] เป็นเทคนิคที่ช่วยลดเวลาในการเรียนรู้ของโมเดล Deep Learning โดยจะนำบางส่วนของโมเดลที่เรียนรู้ชุดข้อมูลแล้ว มาใช้เป็นส่วนหนึ่งของโมเดลใหม่ที่จะสร้างเพื่อนำมาเป็นโมเดลตั้งต้นเพื่อต่อกับชุดข้อมูลขนาดเล็ก หรือสกัดคุณลักษณะ (Feature) สำหรับงานที่ต้องการออกมา ดังตารางที่ 2.1

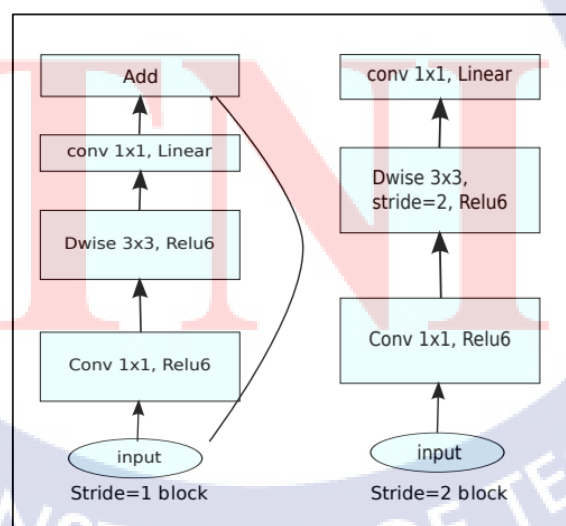
ซึ่งในการวิจัยนี้จะนำเสนอการจำแนกเกรตริงนกแอนกิ้งริง โดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันเป็นหลัก ในการประมวลผลข้อมูลของรูปภาพทั้งหมด และทำการศึกษาเปรียบเทียบโดยใช้ลักษณะของสถาปัตยกรรม (Architecture) ที่ต่างกัน 7 สถาปัตยกรรม ได้แก่ MobileNetV2, Xception, VGG16, VGG19, DenseNet121, DenseNet169 และ DenseNet201

ตารางที่ 2.1 แสดงการเปรียบเทียบประสิทธิภาพของโมเดล

No.	Model	Year	Depth	Size (MB)	Parameters	Top-1 Accuracy	Top-5 Accuracy
1	Xception	2017	81	88	22.9M	79.00%	94.50%
2	VGG16	2015	16	528	138.4M	71.30%	90.10%
3	VGG19	2015	19	549	143.7M	71.30%	90.00%
4	MobileNetV2	2018	105	14	3.5M	71.30%	90.10%
5	DenseNet121	2017	242	33	8.1M	75.00%	92.30%
6	DenseNet169	2017	338	57	14.3M	76.20%	93.20%
7	DenseNet201	2017	402	80	20.2M	77.30%	93.60%

2.6.1 MobileNetV2

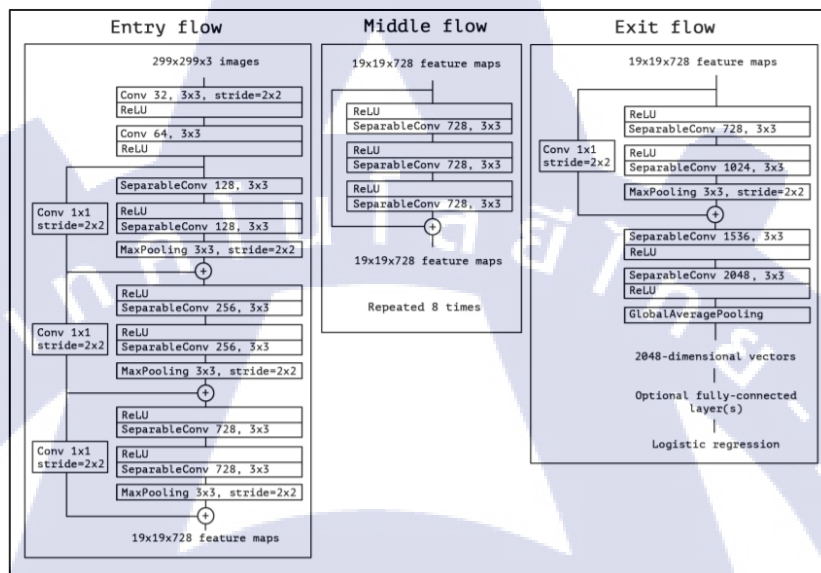
เป็นสถาปัตยกรรมโครงข่ายประสาทเทียม [7] ที่มีการปรับปรุงขนาด และ ประสิทธิภาพของโมเดลให้ดีขึ้น มีชั้นคอนโวลูชัน 53 ชั้น ซึ่งมีองค์ประกอบหลัก คือ Inverted Residual Block และ Bottleneck Residual Block โดยมีชั้นคอนโวลูชัน 2 ชนิดคือ 1x1 Convolution และ 3x3 Depthwise Convolution แต่ละบล็อกมี 3 ชั้นที่แตกต่างกัน คือ 1x1 Convolution with Relu6, Depthwise Convolution และ 1x1 Convolution without any linearity ดังรูปที่ 2.12



รูปที่ 2.12 โครงสร้างของ MobileNetV2

2.6.2. Xception

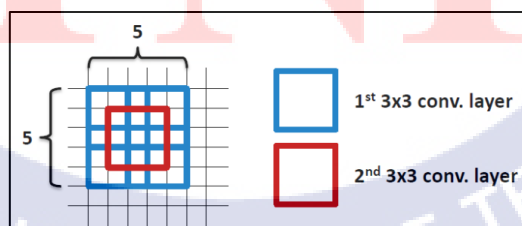
เป็น [8] โครงสร้างเครือข่ายที่มีการปรับปรุงประสิทธิภาพที่ดีกว่า Inception ซึ่ง Xception มีจำนวนพารามิเตอร์เท่ากับ Inception V3 โดย Xception มี 3 ส่วนหลัก คือ Entry flow , Middle flow และ Exit flow ดังรูปที่ 2.13



รูปที่ 2.13 โครงสร้างของ Xception

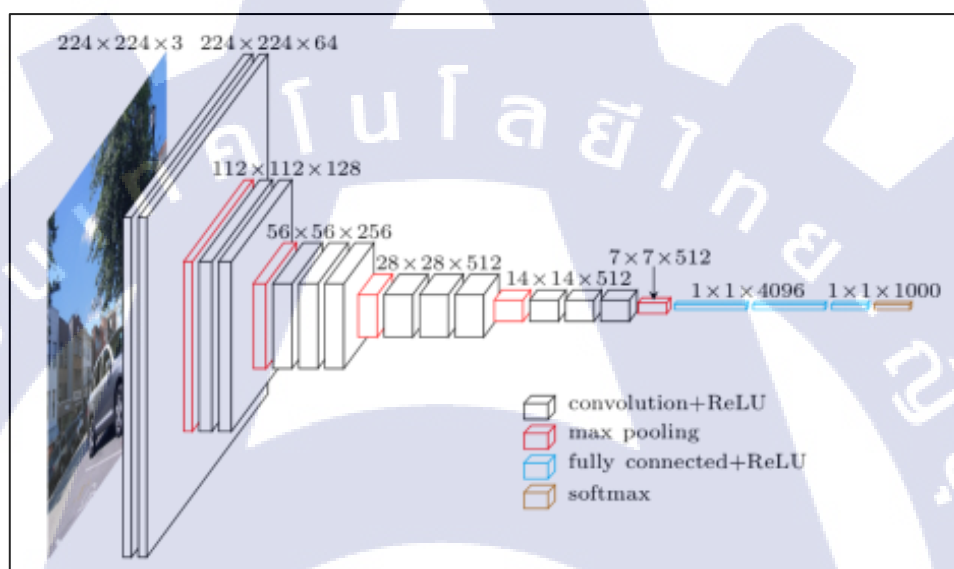
2.6.3 VGG

เป็นโครงสร้างเครือข่ายโดย [9] ใช้ฟิลเตอร์ 3×3 แทนการใช้ฟิลเตอร์ขนาดใหญ่ ซึ่งการใช้ฟิลเตอร์ 3×3 2 ชั้น จะสามารถครอบคลุมพื้นที่ 5×5 แสดงดังรูปที่ 2.14 โดยหากใช้ฟิลเตอร์ 3×3 3 ชั้น จะสามารถครอบคลุมพื้นที่ 7×7 ดังนั้นจึงทำให้ไม่จำเป็นต้องใช้ฟิลเตอร์ขนาดใหญ่ และจะสามารถช่วยลดจำนวนพารามิเตอร์ให้น้อยลงได้



รูปที่ 2.14 ฟิลเตอร์ 3×3 2 ชั้นครอบคลุมพื้นที่ 5×5

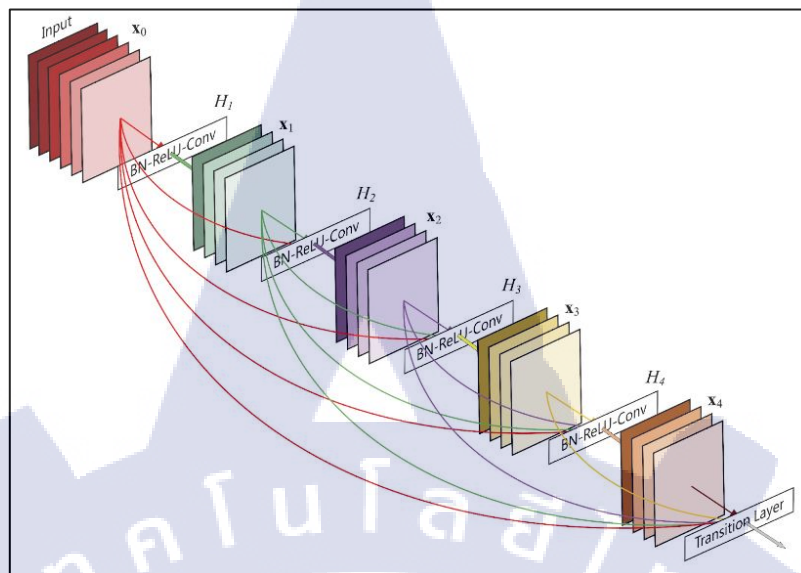
VGG16 เป็นสถาปัตยกรรมโครงข่ายประสาทเทียม ซึ่งเคยชนะเลิศการแข่งขัน ILSVR (Imagenet) ในปี 2014 ซึ่งเป็นสถาปัตยกรรมแบบจำลองการมองเห็น ซึ่ง VGG16 คือเน้นการมีเลเยอร์ฟิลเตอร์ 3×3 ในช่องแรก และใช้ padding และ maxpool ของฟิลเตอร์ 2×2 ของช่องที่สอง ไปตามการจัดเรียง Convolution และจนถึงขั้นสูงสุดอย่างสม่ำเสมอตลอดสถาปัตยกรรมทั้งหมด ในช่วงท้ายจะมี Fully connected เป็นเลเยอร์ที่เชื่อมต่อ ตามด้วย softmax แสดงดังรูปที่ 2.15 สำหรับเอาต์พุต 16 ใน VGG16 หมายถึงมี 16 ชั้น หรือหากมีเอาต์พุต 19 ใน VGG19 หมายถึงมี 19 ชั้น



รูปที่ 2.15 สถาปัตยกรรมของ VGG16

2.6.4 DenseNet

เป็นโครงสร้างโครงข่าย [10] โดยแต่ละชั้นจะเชื่อมต่อโดยตรงกับชั้นอื่น ๆ ในลักษณะฟีดฟอร์เวิร์ด แสดงดังรูปที่ 2.16 โดย CNN แบบทั่วไปถ้ามี L ชั้น ก็จะมีการไหลผ่านของข้อมูลจำนวน $L-1$ แต่ถ้าเป็น DenseNet จะมีจำนวนเป็น $1+2+3+\dots+L = L*(L-1)/2$ โดย DenseNet มีความแม่นยำเทียบเท่ากับ ResNet แต่ใช้พารามิเตอร์น้อยกว่าครึ่งหนึ่ง

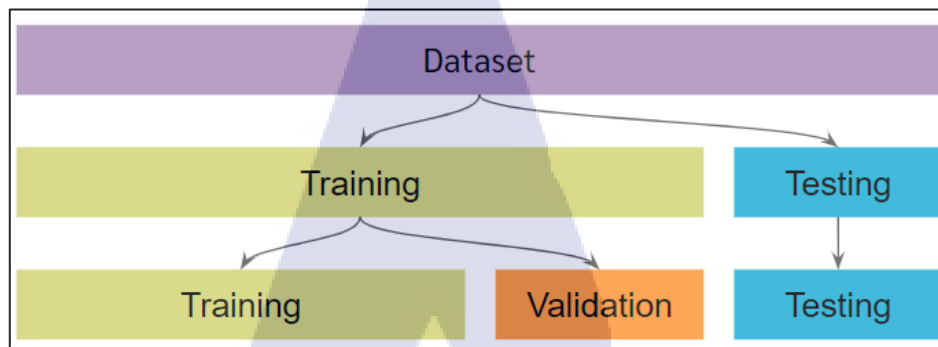


รูปที่ 2.16 ตัวอย่างการเชื่อมกันของแต่ละชั้นของ DenseNet

ซึ่ง DenseNet มีหลากหลายโครงสร้าง สำหรับการวิจัยครั้งนี้จะนำใช้ DenseNet121, DenseNet169 และ DenseNet201 ที่ถูกออกแบบให้มีจำนวนชั้น 121, 169 และ 201 ตามลำดับ

2.7 การวัดประสิทธิภาพของโมเดล

Cross-Validation วิธีนี้เป็นวิธีที่นิยมในการทำงานวิจัย ซึ่งผลที่ได้จะมีความน่าเชื่อถือ [11] การวัดโดยแบ่งกลุ่มข้อมูลออกเป็น k กลุ่ม เช่น กำหนด $k = 5$ -Fold Cross-Validation คือ การแบ่งข้อมูลออกเป็น 5 ส่วน โดยที่แต่ละส่วนมีจำนวนข้อมูลเท่ากัน หรือ กำหนด $k = 10$ -Fold Cross-Validation คือ การแบ่งข้อมูลออกเป็น 10 ส่วน โดยที่แต่ละส่วนมีจำนวนข้อมูลเท่ากัน หลังจากนั้นข้อมูลหนึ่งส่วนจะใช้เป็นตัวแทนทดสอบประสิทธิภาพของโมเดล ทำวนไปเช่นนี้จนครบจำนวนที่แบ่งไว้ โดยจะแบ่งชุดข้อมูลออกเป็น 2 ชุด ได้แก่ ชุดสำหรับการฝึกสอน (Training Set) และชุดสำหรับการทดสอบ (Test Set) หลังจากนั้นจะแบ่งชุดฝึกสอนอีกเป็นชุดสำหรับการฝึกสอน (Training Set) กับชุดสำหรับการตรวจสอบ (Validation Set) เพื่อใช้ส่วนนี้ในการฝึกสอนโมเดล โดยจะจำลองตัวอย่างการทดสอบดังรูปที่ 2.17



รูปที่ 2.17 ชุดข้อมูลทั่วไปแบ่งออกเป็นชุดฝึกสอน ชุดตรวจสอบ และชุดทดสอบ

การวัดประสิทธิภาพของโมเดล สามารถตรวจสอบอัตราความถูกต้อง ได้ดังนี้

2.7.1 Confusion Matrix

เป็นเครื่องมือสำคัญในการประเมินผลลัพธ์ของการทำนาย จะอยู่ในรูปของ Matrix ที่เป็นแนวนอน (Row) และแนวตั้ง (Column) นั่นคือมีลักษณะข้อมูลเป็น 2 มิติ ซึ่งเราจะนำข้อมูลที่ Confusion Matrix มาใช้คำนวณค่าต่าง ๆ ดังรูปที่ 2.18

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

รูปที่ 2.18 Confusion Matrix

2.7.2 Accuracy

ค่าความถูกต้องหรือค่าที่โมเดลทายถูกทั้งหมด คำนวณได้จากสูตรสมการที่ 2.1

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (2.1)$$

2.7.3 Precision

ค่าความแม่นยำ คำนวณได้จากสูตรสมการที่ 2.2

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \quad (2.2)$$

2.7.4 Sensitivity

Sensitivity หรือ Recall ค่าความแม่นยำที่มองว่าโมเดลทำนายถูกกี่ % เมื่อเทียบกับความเป็นจริง คำนวณได้จากสูตรสมการที่ 2.3

$$\text{Sensitivity/Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \quad (2.3)$$

2.7.5 Specificity

ความจำเพาะ คำนวณได้จากสูตรสมการที่ 2.4

$$\text{Specificity} = \frac{\text{TN}}{(\text{TN} + \text{FP})} \quad (2.4)$$

2.7.6 F1-Score

ค่าที่แสดงประสิทธิภาพของโมเดล เป็นค่าที่ได้จากการเอาค่า precision และ recall มาคำนวณรวมกัน หากมีค่ามากถือว่ามีประสิทธิภาพที่ดี คำนวณได้จากสูตรสมการที่ 2.5

$$\text{F1} = 2 \times \frac{(\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})} \quad (2.5)$$

2.8 งานวิจัยที่เกี่ยวข้อง

D. M. Asriny et al. [12] ได้นำเสนอการจำแนกรูปภาพผลไม้ส้มโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network) แบ่งประเภทส้ม ออกเป็น 5 ประเภท คือ ส้มดีเกรด1 ส้มดีเกรด2 ส้มยังไม่สุก ส้มเน่าเสีย และส้มได้รับความเสียหาย ใช้รูปประเภทละ 200 รูป รวมใช้รูปภาพทั้งหมด 1,000 รูป โดยข้อมูลถูกแบ่งออกเป็น 60% สำหรับข้อมูลชุดเรียนรู้ และ 20% สำหรับข้อมูลการตรวจสอบ และ 20% สำหรับข้อมูลการทดสอบ ในการทดลองได้ปรับรูปภาพให้มี

ขนาด 60x60 พิกเซล สร้างโมเดลขนาดชั้นคอนโวลูชัน (Convolution Layer) 3x3 และขนาดชั้นพูลลิ่ง (Pooling Layer) 2x2 จำนวนตัวกรองใช้กับเลเยอร์ 1 และ 2 คือ 32 ตัวกรอง ในขณะที่จำนวนตัวกรองบนเลเยอร์ 3 และ 4 คือ 64 ฟิลเตอร์ ใช้วิธี K-Fold Cross Validation เพื่อใช้สำหรับการทดสอบประสิทธิภาพของโมเดล และทำการเปรียบเทียบการใช้งาน ฟังก์ชัน ReLU และฟังก์ชัน Tanh ผลลัพธ์ค่าความถูกต้องของฟังก์ชันที่เปิดการใช้งาน ReLU คือ 96% ค่าความถูกต้องของฟังก์ชันที่เปิดการใช้งาน Tanh คือ 93,8% ซึ่งฟังก์ชันการเปิดใช้งาน ReLU ให้ความแม่นยำกว่าฟังก์ชันการเปิดใช้งาน Tanh

N. Hongboonmee และ N. Jantawong [13] ได้นำเสนอการพัฒนาระบบอัตโนมัติสำหรับวิเคราะห์ความหวานและพันธุ์แตงโมด้วยภาพถ่ายโดย ประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึก สำหรับใช้งานบนสมาร์ตโฟน ใช้โครงข่ายประสาทเทียมการเรียนรู้เชิงลึก โดยใช้ไลบรารี TensorFlow โดยนำอัลกอริทึม 2 อัลกอริทึม คือ MobileNet และ InceptionV3 มาทำการทดลองเปรียบเทียบการจำแนกภาพ ซึ่งฝึกสอนให้สามารถจำแนกภาพจำนวน 4 ประเภท ประเภทละ 100 ภาพ ฝึกสอนจำนวน 500 รอบ จากการเปรียบเทียบพบว่าโมเดลจากอัลกอริทึม MobileNet และ InceptionV3 มีค่าความถูกต้องเท่ากันคือ 97.20% โดยได้นำ MobileNet ไปใช้ในการพัฒนาต่อเนื่องจาก InceptionV3 มีขนาด 85.40 MB MobileNet มีขนาด 16.73 MB จึงเลือก MobileNet หลังจากนำ MobileNet ไปพัฒนาต่อได้ให้ค่าความถูกต้องในการจำแนกพันธุ์ และความหวานแตงโมเท่ากับ 87.50%

M. A. I. Aquil และ W. H. W. Ishak [14] ได้ดำเนินการศึกษาเปรียบเทียบโครงข่ายประสาทเทียมแบบคอนโวลูชันสำหรับการจำแนกโรคพืชมะเขือเทศ ซึ่งเปรียบเทียบโครงข่ายประสาทเทียมแบบคอนโวลูชันจำนวน 7 โครงสร้าง ประกอบด้วย VGG-16, VGG-19, Squeeze-ne, ResNet-18, ResNet-34, Resnet-50, ResNet-101 และ DenseNet-120 โดยชุดข้อมูลที่ใช้ได้นำมาจากชุดข้อมูล PlantVillage ซึ่งประกอบด้วยรูปพืชผล 14 ชนิด และรูปโรคพืช 30 ชนิด โดยในงานวิจัยนี้ได้นำเฉพาะรูปใบมะเขือเทศ ซึ่งประกอบด้วยรูปโรคมะเขือเทศ 9 ประเภท นอกจากนี้ได้เพิ่มรูปที่ไม่เป็นโรคพืช รวมรูปทั้งหมด 18,160 รูป ในการทดลองได้ปรับรูปภาพให้มีขนาด 256x256 พิกเซล โดยข้อมูลถูกแบ่งออกเป็น 20-80 (20% ของชุดข้อมูลสำหรับการฝึกอบรม และ 80% ที่ใช้สำหรับการทดสอบ) , 40-60 (40% ของชุดข้อมูลสำหรับการฝึกอบรม และ 60% ที่ใช้สำหรับการทดสอบ) , 50-50 (50% ของชุดข้อมูลสำหรับการฝึกอบรม และ 50% ที่ใช้สำหรับการทดสอบ) , 60-40 (60% ของชุดข้อมูลสำหรับการฝึกอบรม และ 40% ที่ใช้สำหรับการทดสอบ) และ 80-20 (80% ของชุดข้อมูลสำหรับการฝึกอบรม และ 20% ที่ใช้สำหรับการทดสอบ) หลังจากนั้นทำการฝึกฝนกับโมเดลที่ฝึกฝนมาตั้งแต่ต้น และโมเดลที่ฝึกฝนล่วงหน้าแล้ว สำหรับการทดสอบใช้การวิเคราะห์ทางสถิติแบบหลาย

คลาตาม F-Score, ความจำเพาะ, ความไว, ความแม่นยำ และความถูกต้อง จากผลการทดลองวิธี CNN โดยใช้โครงสร้างของ Densenet-120 ที่ผ่านการฝึกฝนล่วงหน้าทำงานได้ดีที่สุด โดยให้ความถูกต้อง 99.68%, คะแนน F-1 99.84% และความแม่นยำ 99.81% ซึ่งสูงกว่าเมื่อเทียบกับแบบจำลองที่ไม่ได้ฝึกฝน

R. Dandavate และ V. Patodkar [15] ได้นำเสนอการจำแนกประเภทผลไม้ ซึ่งสามารถระบุได้ว่าเป็นประเภทที่สามารถรับประทานได้หรือไม่สามารถรับประทานได้ โดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน โดยจำแนกภาพผลไม้ 4 ประเภท ได้แก่ กล้วย มะละกอ มะม่วง และฝรั่ง แบ่งเป็น 3 ระยะ ดิบ สุก และสุกเกินไป รวมรูปภาพทั้งหมด 12,559 รูป รวบรวมจาก Google images ในการทดลองได้ปรับรูปภาพให้มีขนาด 60x60 พิกเซล โดยมีการนำเทคนิคการเสริมข้อมูล (Data Augmentation) เอียง หมุน เลื่อนในแนวตั้ง และแนวนอน โดยผลจากการทดสอบให้ค่าความถูกต้อง 97.74%

S. Lu et al. [16] ได้นำเสนอการจำแนกภาพผลไม้ โดยโครงข่ายประสาทเทียมแบบคอนโวลูชันจำนวน 6 ชั้น ประกอบด้วย ชั้นคอนโวลูชัน (Convolution Layers) ชั้นพูลลิง (Pooling Layers) และ ชั้นเชื่อมต่อแบบสมบูรณ์ (fully Connected Layers) เพื่อจำแนกภาพผลไม้ 9 ชนิด คือ ลูกแพร์แองเจิ้ล (Anjou Pear) แบล็กเบอร์รี่ (Blackberry) องุ่นดำ (Black grape) บลูเบอร์รี่ (Blueberry) แคนตาลูป (Cantaloupe) สับปะรดสีทอง (Golden Pineapple) แพร์บอส (Bosc Pear) แอปเปิ้ลแกรนสมิท (Granny Smith Apple) และ องุ่นเขียว (Green Grape) รวมรูปภาพทั้งหมด 1,800 รูป ในการทดลองได้ปรับรูปภาพให้มีขนาด 256x256 พิกเซล โดยข้อมูลถูกแบ่งออกเป็นสำหรับข้อมูลชุดเรียนรู้ 900 รูป และข้อมูลชุดทดสอบ 900 รูป โดยผลจากการทดสอบให้ค่าความถูกต้อง 91.44%

TNI

THAI - NICHI INSTITUTE OF TECHNOLOGY

ตารางที่ 2.2 แสดงงานวิจัยที่เกี่ยวข้อง

ลำดับ	ชื่อหัวข้อวิจัย	ปี	วัตถุประสงค์	เทคนิค	ชุดข้อมูล	ผลลัพธ์ของงานวิจัย
1	Orange Fruit Images Classification using Convolutional Neural Networks	2019	การจำแนกรูปภาพผลไม้ส้ม 5 ประเภท คือ ส้มดีเกรด1, ส้มดีเกรด 2, ส้มยังไม่สุก, ส้มเน่าเสีย, และส้ม ได้รับความเสียหายโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน และ เปรียบเทียบประสิทธิภาพการทำงานของ ReLU และ Tanh	-ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน สร้างโมเดล ขนาดชั้นคอนโวลูชัน 3x3 และขนาดชั้นพูลลิง 2x2 จำนวนตัวกรองใช้กับเลเยอร์ 1 และ 2 คือ 32 ตัวกรอง ในขณะที่จำนวนตัวกรองบนเลเยอร์ 3 และ 4 คือ 64 ฟิลเตอร์ - ปรับรูปภาพให้มีขนาด 32x32 พิกเซล	- ใช้รูปภาพทั้งหมด 1,000 รูป ประเภทละ 200 รูป จัดเก็บเอง โดยใช้กล้องถ่ายรูปดิจิตอล - ข้อมูลถูกแบ่งเป็น 60% ข้อมูลชุดเรียนรู้ 20% ข้อมูลการตรวจสอบ และ 20% ข้อมูลการทดสอบ	ค่าความถูกต้องของฟังก์ชันที่เปิดการใช้งาน ReLU คือ 96% ซึ่งมากกว่าฟังก์ชันการเปิดใช้งาน Tanh
2	Apply of Deep Learning Techniques to Measure the Sweetness Level of Watermelon via Smartphone	2020	จำแนกพันธุ์ และระดับความหวานของแตงโม 4 ประเภท พันธุ์กินรีรสชาติหวาน, พันธุ์กินรี รสชาติไม่หวาน, พันธุ์ตอปีโต รสชาติหวาน และ พันธุ์ตอปี โต รสชาติไม่หวาน โดยประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึก	-ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน เปรียบเทียบ InceptionV3 และ MobileNet - การกำหนดระดับความหวานใช้การเทียบความหวานกับเครื่องวัดค่าความหวาน 0-32% Brix - ปรับรูปภาพให้มีขนาด 224x224 พิกเซล	- ใช้รูปภาพทั้งหมด 400 รูป ประเภทละ 100 รูป จัดเก็บเอง โดยใช้กล้องถ่ายรูปดิจิตอล - แบ่งชุดข้อมูลเป็น 2 ส่วน ชุดเรียนรู้ 90 % ชุดทดสอบ 10%	MobileNet และ InceptionV3 ให้ค่าความถูกต้องเท่ากัน คือ 97.20% แต่ MobileNet มีขนาดที่เล็กกว่า จึงเลือก MobileNet

ตารางที่ 2.2 แสดงงานวิจัยที่เกี่ยวข้อง (ต่อ)

ลำดับ	ชื่อหัวข้อวิจัย	ปี	วัตถุประสงค์	เทคนิค	ชุดข้อมูล	ผลลัพธ์ของงานวิจัย
3	Evaluation of scratch and pre-trained convolutional neural networks for the classification of Tomato plant diseases	2021	เปรียบเทียบโครงข่ายประสาทเทียมแบบคอนโวลูชันสำหรับการจำแนกโรคพืชมะเขือเทศ 9 ประเภท	- ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันจำนวน 7 โครงสร้าง ประกอบด้วย VGG-16, VGG-19, Squeeze-ne, ResNet-18, ResNet-34, Resnet-50, ResNet-101 และ DenseNet-120 - ปรับรูปภาพให้มีขนาด 256x256 พิกเซล	- ใช้รูปภาพทั้งหมด 18,160 รูป - แบ่งชุดข้อมูลสำหรับการฝึกอบรม และสำหรับการทดสอบ เป็น 20%-80%, 40%-60%, 50%-50%, 60%-40% และ 80%-20% ตามลำดับ	Densenet-120 ที่ผ่านการฝึกฝนล่วงหน้าในการทำงานได้ดีที่สุด โดยให้ค่าความถูกต้อง 99.68%
4	CNN and Data Augmentation Based Fruit Classification Model	2022	จำแนกประเภทผลไม้ และสามารถระบุได้ว่าเป็นประเภทที่รับประทานได้ และไม่สามารถรับประทานได้ โดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน	- ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน สร้างโมเดล conv2d เลเยอร์แรกประกอบด้วย 32 และเลเยอร์ที่สองมี 64 ตัวกรอง หลังจากนั้นใช้ Max Pooling เพื่อลดมิติเชิงพื้นที่ของปริมาณการส่งออก - มีการนำเทคนิคการเสริมข้อมูล (Data Augmentation) เอียง หมุน เลื่อนในแนวตั้ง และแนวนอน - รูปภาพถูกปรับขนาดเป็น 60x60 พิกเซล	- ใช้รูปภาพทั้งหมด 12,559 รูป - ข้อมูลแบ่งให้ 80% ของภาพใช้สำหรับการฝึกอบรม และการตรวจสอบความถูกต้อง และส่วนที่เหลืออีก 20% สำหรับในการทดสอบ	ค่าความถูกต้อง 97.74%

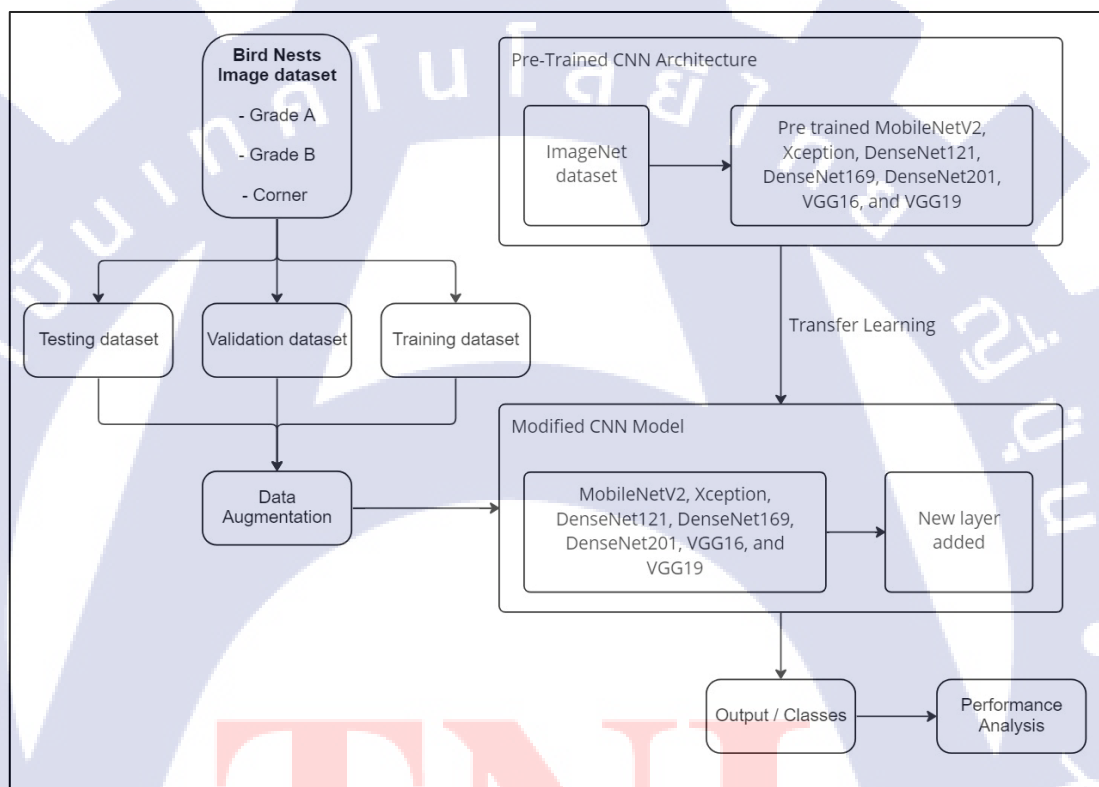
ตารางที่ 2.2 แสดงงานวิจัยที่เกี่ยวข้อง (ต่อ)

ลำดับ	ชื่อหัวข้อวิจัย	ปี	วัตถุประสงค์	เทคนิค	ชุดข้อมูล	ผลลัพธ์ของงานวิจัย
5	Fruit Classification Based on Six Layer Convolutional Neural Network	2018	จำแนกภาพผลไม้ 9 ชนิด คือ ลูกแพร์แองเจิ้ล, แบล็กเบอร์รี่, องุ่นดำ, บลูเบอร์รี่, แพร์บอส, แคนตาลูป, สับปะรดสีทอง, แอปเปิ้ลแกรนสมิท และ องุ่นเขียวโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน	- ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน จำนวน 6 ชั้น ประกอบด้วย ชั้นคอนโวลูชัน 4 ชั้น และชั้นเชื่อมโยงแบบสมบรูณ์ 2 ชั้น - ปรับรูปภาพให้มีขนาด 256x256 พิกเซล	- ใช้รูปภาพทั้งหมด 1,800 รูป ประเภทละ 200 รูป - แบ่งสำหรับข้อมูลชุดเรียนรู้ 900 รูป และข้อมูลชุดทดสอบ 900 รูป	ค่าความถูกต้อง 91.44%

บทที่ 3

วิธีการดำเนินงาน

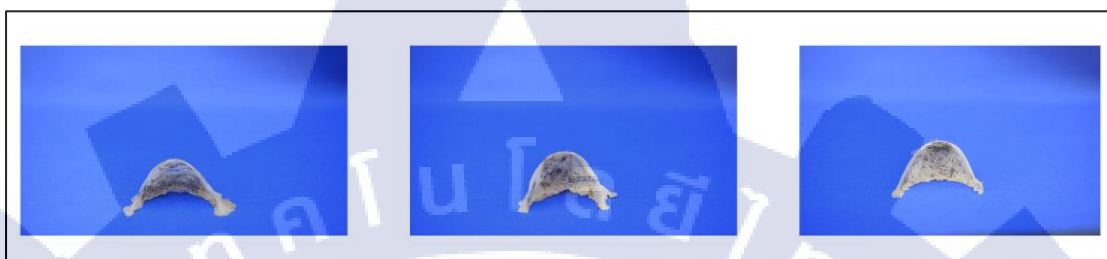
ในบทนี้กล่าวถึงการออกแบบของระบบจำแนกเกรตริงนกแอ่นกินรังโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolution Neural Network: CNN) โดยงานวิจัยได้แบ่งการดำเนินการออกเป็น 3 ส่วน และมีขั้นตอนการดำเนินงานวิจัยดังรูปที่ 3.1 ดังนี้



รูปที่ 3.1 ระบบจำแนกเกรตริงนกแอ่นกินรัง

3.1 การศึกษาข้อมูลและเก็บรวบรวมข้อมูลรูปภาพ

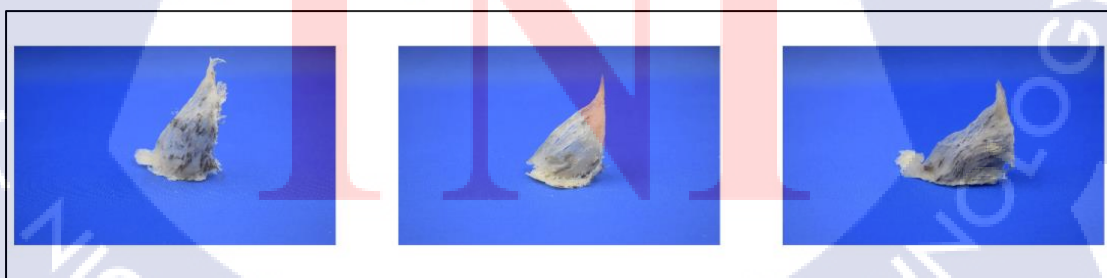
ในงานวิจัยนี้ได้ดำเนินการศึกษาข้อมูลที่เกี่ยวข้องกับเกรตริงนกแอ่นกินรัง และงานวิจัยที่เกี่ยวข้อง โดยได้เก็บรวบรวมรูปภาพของเกรตริงนกแอ่นกินรังชนิดรังนกบ้าน โดยได้รวบรวม 3 ประเภท คือ เกรตเอ (Grade A) เกรตบี (Grade B) และรังมุม (Corner) โดยเก็บภาพถ่ายรังนกแอ่นกินรังรวมทั้งหมดจำนวน 2,256 ภาพ ประเภทละ 752 ภาพ ดังรูปที่ 3.2-3.4



รูปที่ 3.2 ตัวอย่างชุดข้อมูล Grade A



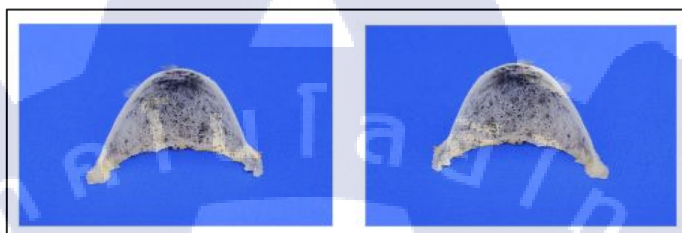
รูปที่ 3.3 ตัวอย่างชุดข้อมูล Grade B



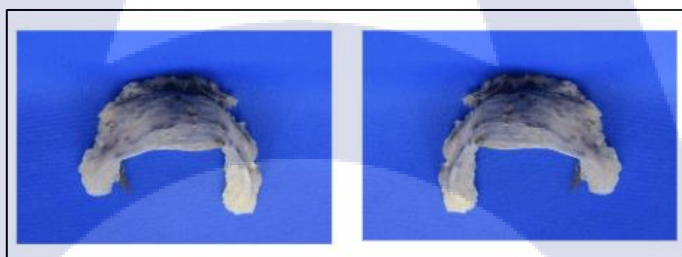
รูปที่ 3.4 ตัวอย่างชุดข้อมูล Corner

ในระหว่างขั้นตอนการเตรียมข้อมูล (Augmented Data) ภาพรังนกจะถูกสร้างขึ้นจากข้อมูลที่มีอยู่โดยใช้การเสริมข้อมูลเพื่อเพิ่มจำนวนชุดข้อมูลการฝึกอบรม โมเดลสามารถเรียนรู้รูปแบบต่าง ๆ ได้มากขึ้น มีการใช้เทคนิคการประมวลผลภาพ (Image Processing) เช่น การหมุนภาพ พลิกแนวนอน และพลิกแนวตั้ง หลังจากนั้น ภาพทั้งหมดจะถูกอ่าน และปรับขนาดเป็น 224x224 พิกเซลเพื่อใช้สำหรับการฝึกในโมเดล

1. การพลิกแนวนอน (Horizontal Flip) ดังรูปที่ 3.5–3.7



รูปที่ 3.5 ตัวอย่างชุดข้อมูล Grade A พลิกแนวนอน



รูปที่ 3.6 ตัวอย่างชุดข้อมูล Grade B พลิกแนวนอน



รูปที่ 3.7 ตัวอย่างชุดข้อมูล Corner พลิกแนวนอน

2. การพลิกแนวตั้ง (Vertical Flip) ดังรูปที่ 3.8–3.10



รูปที่ 3.8 ตัวอย่างชุดข้อมูล Grade A พลิกแนวตั้ง

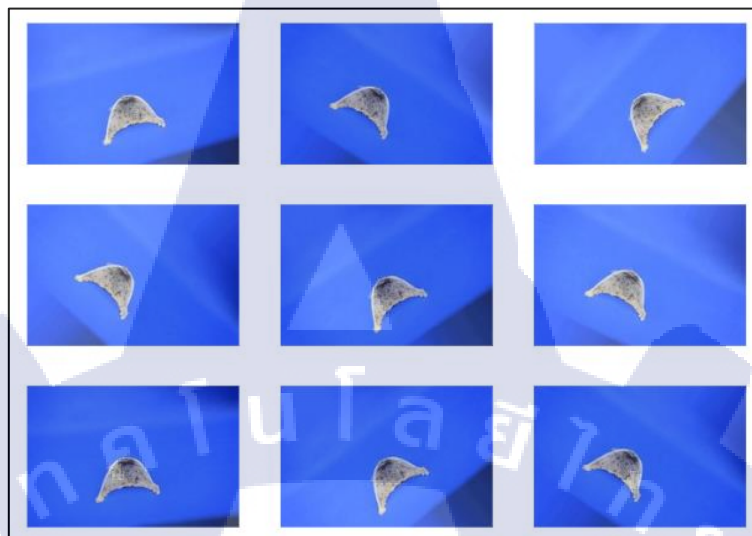


รูปที่ 3.9 ตัวอย่างชุดข้อมูล Grade B พลิกแนวตั้ง



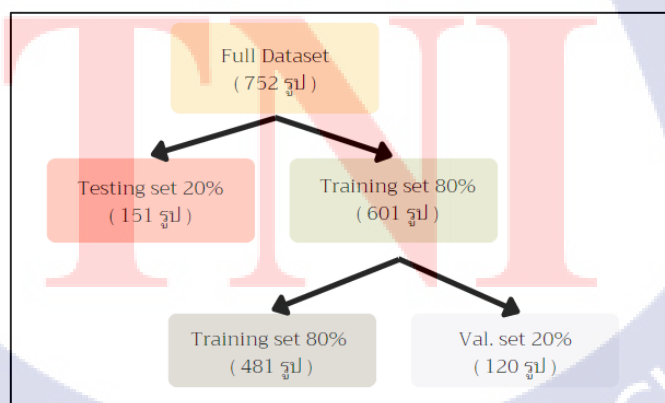
รูปที่ 3.10 ตัวอย่างชุดข้อมูล Corner พลิกแนวตั้ง

3. การหมุนภาพ (Rotation Range) ดังรูปที่ 3.11



รูปที่ 3.11 ตัวอย่างการหมุนภาพชุดข้อมูล Corner

โดยภาพถ่ายที่นำมาใช้แบ่งออกเป็น 3 ส่วนคือ ภาพชุดฝึก (Training data) ภาพชุดตรวจสอบความถูกต้อง (Validation data) และภาพชุดทดสอบ (Testing data) ดังรูปที่ 3.12 ภาพชุดฝึกใช้จำนวนภาพประเภทละ 481 ภาพ รวมภาพทั้งหมด 1,443 ภาพ ชุดตรวจสอบความถูกต้อง ภาพประเภทละ 151 ภาพ รวมภาพทั้งหมด 453 ภาพ ส่วนชุดทดสอบใช้ภาพประเภทละ 120 ภาพ รวมทั้งหมด 360 ภาพ ดังตารางที่ 3.1



รูปที่ 3.12 การแบ่งชุดข้อมูลภาพถ่าย

ตารางที่ 3.1 จำนวนภาพเกรตริงนกแอ่นกินรังในแต่ละชุดข้อมูล

Class	Training data	Validation data	Testing data	Total
Grade A	481	120	151	752
Grade B	481	120	151	752
Corner	481	120	151	752
Total	1,443	360	453	2,256

3.2 สร้างแบบจำลองสำหรับจำแนกเกรตริงนกแอ่นกินรัง

การทดลองจำแนกเกรตริงนกแอ่นกินรังโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน พัฒนาคอมพิวเตอร์ ยี่ห้อ Acer รุ่น Nitro 5 AN515-54 คุณสมบัติของคอมพิวเตอร์เครื่องนี้คือ CPU Intel Core i5-9300H @ 2.40 GHz การ์ดจอรุ่น NVIDIA GeForce GTX 1050 แรมขนาด 16 GB จอแสดงผล 15.6 นิ้ว (1920x1080) ความละเอียดจอคอมพิวเตอร์ Full-HD IPS 120Hz จะสร้างระบบโดยใช้ ภาษา Python ในการพัฒนาระบบ โดยจะใช้ไลบรารี Keras และ TensorFlow ซึ่งจะมีแอปพลิเคชันของโครงสร้างให้ใช้งาน โดยงานวิจัยนี้จะใช้ 7 สถาปัตยกรรม ได้แก่ MobileNetV2, Xception, DenseNet121, DenseNet169, DenseNet201, VGG16, และ VGG19 โดยจะนำชุดข้อมูลรูปภาพมาทำการทดสอบประสิทธิภาพกับ 7 สถาปัตยกรรม โดยพารามิเตอร์ที่ใช้ฝึกสอนโมเดลแสดงดังตารางที่ 3.2 และรูปที่ 3.13–3.19 อีกทั้งมีการนำฟังก์ชัน Callback มาใช้เพื่อที่ว่าปัจจุบันโมเดลสามารถเรียนรู้ไปได้มากน้อยเพียงใด จากนั้นนำไปหาประสิทธิภาพของโมเดล

ตารางที่ 3.2 แสดงพารามิเตอร์ฝึกสอนโมเดล

Parameters	Value
Optimizer	Adam
No of Classes	3
Color	RGB
Input Image Size	224x224
Batch sizes	32
Learning rate	0.0001

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 224, 224, 3) 0		
Conv1 (Conv2D)	(None, 112, 112, 32) 864		input_1[0][0]
bn_Conv1 (BatchNormalization)	(None, 112, 112, 32) 128		Conv1[0][0]
Conv1_relu (ReLU)	(None, 112, 112, 32) 0		bn_Conv1[0][0]
expanded_conv_depthwise (Depthwise Conv2D)	(None, 112, 112, 32) 288		Conv1_relu[0][0]
expanded_conv_depthwise_BN (BatchNormalization)	(None, 112, 112, 32) 128		expanded_conv_depthwise[0][0]
...			
Trainable params: 180,867			
Non-trainable params: 2,257,984			

รูปที่ 3.13 แสดงพารามิเตอร์ฝึกสอนโมเดล MobileNetV2

Layer (type)	Output Shape	Param #	Connected to
input_7 (InputLayer)	[(None, 224, 224, 3) 0		
block1_conv1 (Conv2D)	(None, 111, 111, 32) 864		input_7[0][0]
block1_conv1_bn (BatchNormalization)	(None, 111, 111, 32) 128		block1_conv1[0][0]
block1_conv1_act (Activation)	(None, 111, 111, 32) 0		block1_conv1_bn[0][0]
block1_conv2 (Conv2D)	(None, 109, 109, 64) 18432		block1_conv1_act[0][0]
block1_conv2_bn (BatchNormalization)	(None, 109, 109, 64) 256		block1_conv2[0][0]
...			
Trainable params: 279,171			
Non-trainable params: 20,861,480			

รูปที่ 3.14 แสดงพารามิเตอร์ฝึกสอนโมเดล Xception

Layer (type)	Output Shape	Param #	Connected to
input_4 (InputLayer)	[(None, 224, 224, 3) 0		
zero_padding2d (ZeroPadding2D)	(None, 230, 230, 3) 0		input_4[0][0]
conv1/conv (Conv2D)	(None, 112, 112, 64) 9408		zero_padding2d[0][0]
conv1/bn (BatchNormalization)	(None, 112, 112, 64) 256		conv1/conv[0][0]
conv1/relu (Activation)	(None, 112, 112, 64) 0		conv1/bn[0][0]
zero_padding2d_1 (ZeroPadding2D)	(None, 114, 114, 64) 0		conv1/relu[0][0]
...			
Trainable params: 148,099			
Non-trainable params: 7,037,504			

รูปที่ 3.15 แสดงพารามิเตอร์ฝึกสอนโมเดล DenseNet121

Layer (type)	Output Shape	Param #	Connected to
input_5 (InputLayer)	[(None, 224, 224, 3) 0		
zero_padding2d_2 (ZeroPadding2D)	(None, 230, 230, 3) 0		input_5[0][0]
conv1/conv (Conv2D)	(None, 112, 112, 64) 9408		zero_padding2d_2[0][0]
conv1/bn (BatchNormalization)	(None, 112, 112, 64) 256		conv1/conv[0][0]
conv1/relu (Activation)	(None, 112, 112, 64) 0		conv1/bn[0][0]
zero_padding2d_3 (ZeroPadding2D)	(None, 114, 114, 64) 0		conv1/relu[0][0]
...			
Trainable params: 230,019			
Non-trainable params: 12,642,880			

รูปที่ 3.16 แสดงพารามิเตอร์ฝึกสอนโมเดล DenseNet169

Layer (type)	Output Shape	Param #	Connected to
input_6 (InputLayer)	[(None, 224, 224, 3) 0		
zero_padding2d_4 (ZeroPadding2D)	(None, 230, 230, 3) 0		input_6[0][0]
conv1/conv (Conv2D)	(None, 112, 112, 64) 9408		zero_padding2d_4[0][0]
conv1/bn (BatchNormalization)	(None, 112, 112, 64) 256		conv1/conv[0][0]
conv1/relu (Activation)	(None, 112, 112, 64) 0		conv1/bn[0][0]
zero_padding2d_5 (ZeroPadding2D)	(None, 114, 114, 64) 0		conv1/relu[0][0]
...			
Trainable params: 262,787			
Non-trainable params: 18,321,984			

รูปที่ 3.17 แสดงพารามิเตอร์ฝึกสอนโมเดล DenseNet201

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
...		
Trainable params: 82,563		
Non-trainable params: 14,714,688		

รูปที่ 3.18 แสดงพารามิเตอร์ฝึกสอนโมเดล VGG16

Layer (type)	Output Shape	Param #
input_3 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
...		
Trainable params: 82,563		
Non-trainable params: 20,024,384		

รูปที่ 3.19 แสดงพารามิเตอร์ฝึกสอนโมเดล VGG19

3.3 การทดสอบประสิทธิภาพ

การหาประสิทธิภาพของระบบโดยใช้ไลบรารี Sklearn เพื่อคำนวณหาค่า Confusion Metrix, Precision, Recall, F1-Score และค่าความถูกต้องของโมเดล ดังรูปที่ 3.20–3.22 จากนั้นนำโมเดลที่มีความถูกต้องมากที่สุดไปใช้ในการจำแนกเกรตริงนกแอ่นกินรัง

```
predictions = np.argmax(model.predict(test_images),axis=1)

cm = confusion_matrix(test_images.labels,predictions)
clr = classification_report(test_images.labels,predictions,target_names=test_images.class_indices)
```

รูปที่ 3.20 การทดสอบประสิทธิภาพของโมเดล

```
plt.figure(figsize=(3,3))
sns.heatmap(cm,annot=True,fmt='g',vmin=0,cmap='Blues',cbar=False)
plt.xticks(ticks=np.arange(3)+0.5,labels=test_images.class_indices,rotation=90)
plt.yticks(ticks=np.arange(3)+0.5,labels=test_images.class_indices,rotation=90)
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.title("Confusion Matrix")
plt.show()
```

รูปที่ 3.21 การหา Confusion Metrix ของโมเดล

```
results = model.evaluate(test_images,verbose=0)
print("Test Accuracy : {:.2f}%" .format(results[1]*100))
```

รูปที่ 3.22 การหาค่าความถูกต้องของโมเดล

บทที่ 4

สรุปผลการดำเนินงาน การวิเคราะห์ และสรุปผลต่าง ๆ

การจำแนกเกรดเรียงนกแอ่นกินรังด้วยขั้นตอนวิธีการเรียนรู้เชิงลึก ผู้วิจัยได้ดำเนินการนำชุดข้อมูลภาพ (Dataset) มาดำเนินการประมวลผลตามสถาปัตยกรรม ซึ่งมีผลดำเนินงานดังต่อไปนี้

4.1 ผลการดำเนินงาน

จากการวิจัยผลจากการดำเนินงาน เมื่อทำการถ่ายโอนการเรียนรู้ครบทั้ง 7 สถาปัตยกรรม คือ MobileNetV2, Xception, VGG16, VGG19, DenseNet121, DenseNet169 และ DenseNet201 ทำให้ได้ผลจากการดำเนินงาน ดังนี้

ตารางที่ 4.1 แสดงผลจากการดำเนินงาน

Model	Accuracy (%)	Precision	Recall	F1 Score	Epochs (รอบ)	Size (MB)	เวลาที่ใช้
MobileNetV2	92.92	0.94	0.93	0.93	8	14	1ชม. 19 นาที
Xception	97.79	0.98	0.98	0.98	7	88	1ชม. 22 นาที
VGG16	99.34	0.99	0.99	0.99	21	528	3ชม. 34 นาที
VGG19	95.58	0.96	0.96	0.96	8	549	1ชม. 18 นาที
DenseNet121	96.90	0.97	0.97	0.97	6	33	1ชม. 6 นาที
DenseNet169	98.23	0.98	0.98	0.98	7	57	1ชม. 12 นาที
DenseNet201	99.34	0.99	0.99	0.99	10	80	1ชม. 45 นาที

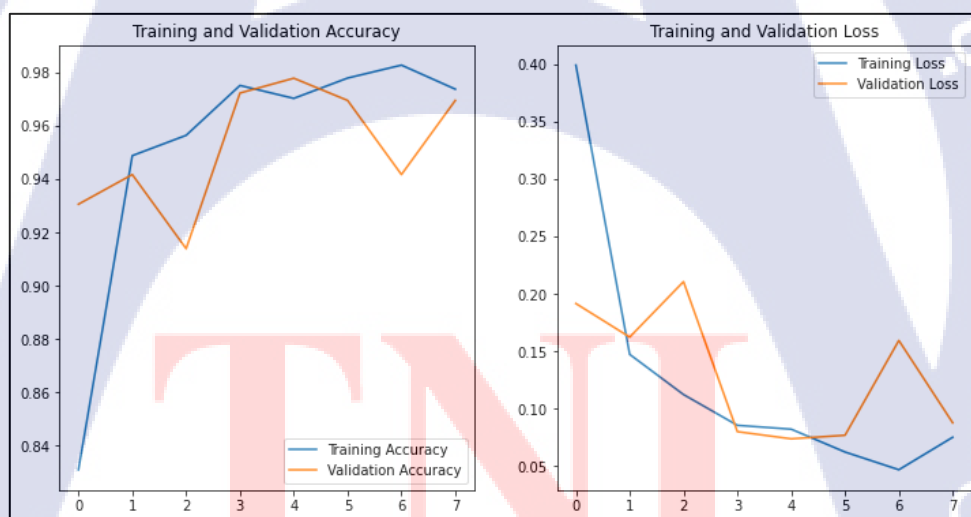
4.1.1 MobileNetV2

จำนวนรอบในการเรียนรู้ (epochs) จำนวน 8 รอบ ใช้เวลาในการเรียนรู้ 1 ชั่วโมง 19 นาที ผลลัพธ์จากการประมวลผลของการประเมินสมรรถนะแบบจำลองด้วย Confusion Matrix จากผลลัพธ์มีค่าความแม่นยำในการทำนายในแต่ละประเภท ซึ่งค่าความแม่นยำของค่า True Positive คือ เกรดเอ (Grade A) 146 รูป เกรดบี (Grade B) 133 รูป และรังมูม (Corner) 141 รูป ดังรูปที่ 4.1-4.2 อีกทั้งให้ค่าความถูกต้อง (Accuracy) เท่ากับ 92.92% ค่าความแม่นยำ Precision

เท่ากับ 0.94 ค่าความแม่นยำ Recall เท่ากับ 0.93 และค่าความแม่นยำ F1-score เท่ากับ 0.93 ดังตารางที่ 4.2

Confusion Matrix			
	Corner	GradeA	GradeB
Actual Corner	141	1	0
Actual GradeA	3	146	0
Actual GradeB	0	28	133
	Corner	GradeA	GradeB
		Predicted	

รูปที่ 4.1 แสดงค่า Confusion Matrix ของ MobileNetV2



รูปที่ 4.2 กราฟแสดงค่า Accuracy และ Loss ของ MobileNetV2

ตารางที่ 4.2 ตารางแสดงประสิทธิภาพ ของ MobileNetV2

	Precision	Recall	F1 Score	Accuracy (%)
Grade A	0.83	0.98	0.90	92.92
Grade B	1.00	0.83	0.90	
Corner	0.98	0.99	0.99	
Average	0.94	0.93	0.93	

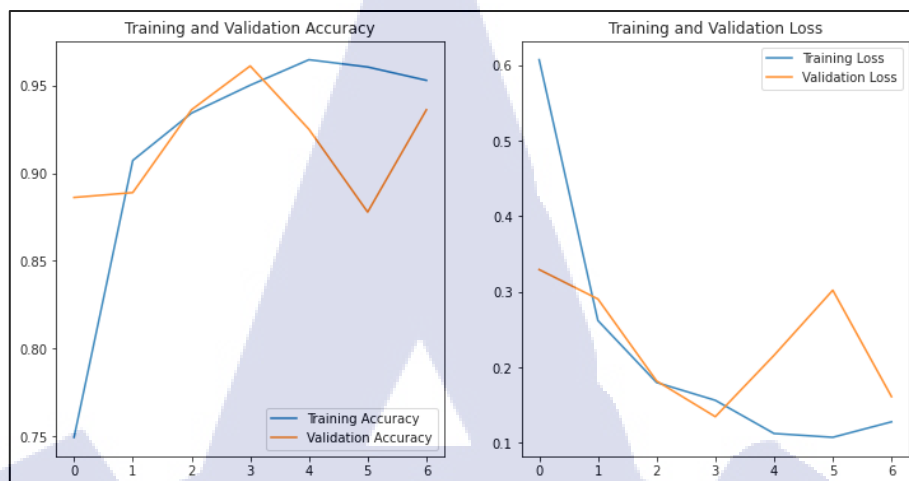
หมายเหตุ Precision, Recall และ F1 Score มีค่าอยู่ในช่วงระหว่าง 0 และ 1

4.1.2 Xception

จำนวนรอบในการเรียนรู้ (epochs) จำนวน 7 รอบ ใช้เวลาในการเรียนรู้ 1 ชั่วโมง 22 นาที ผลลัพธ์จากการผลลัพธ์จากการประมวลผลของการประเมินสมรรถนะแบบจำลองด้วย Confusion Matrix จากผลลัพธ์มีค่าความแม่นยำในการทำนายในแต่ละประเภท ซึ่งค่าความแม่นยำของค่า True Positive คือ เกรดเอ (Grade A) 141 รูป เกรดบี (Grade B) 161 รูป และรั้งมุม (Corner) 140 รูป ดังรูปที่ 4.3-4.4 อีกทั้งให้ค่าความถูกต้อง (Accuracy) เท่ากับ 97.79% ค่าความแม่นยำ Precision เท่ากับ 0.98 ค่าความแม่นยำ Recall เท่ากับ 0.98 และค่าความแม่นยำ F1-score เท่ากับ 0.98 ดังตารางที่ 4.3

Confusion Matrix			
	Corner	GradeA	GradeB
Actual Corner	140	2	0
Actual GradeA	3	141	5
Actual GradeB	0	0	161
	Corner	GradeA	GradeB
		Predicted	

รูปที่ 4.3 แสดงค่า Confusion Matrix ของ Xception



รูปที่ 4.4 กราฟแสดงค่า Accuracy และ Loss ของ Xception

ตารางที่ 4.3 ตารางแสดงประสิทธิภาพ ของ Xception

Class	Precision	Recall	F1 Score	Accuracy (%)
Grade A	0.99	0.95	0.97	97.79
Grade B	0.97	1.00	0.98	
Corner	0.98	0.99	0.98	
Average	0.98	0.98	0.98	

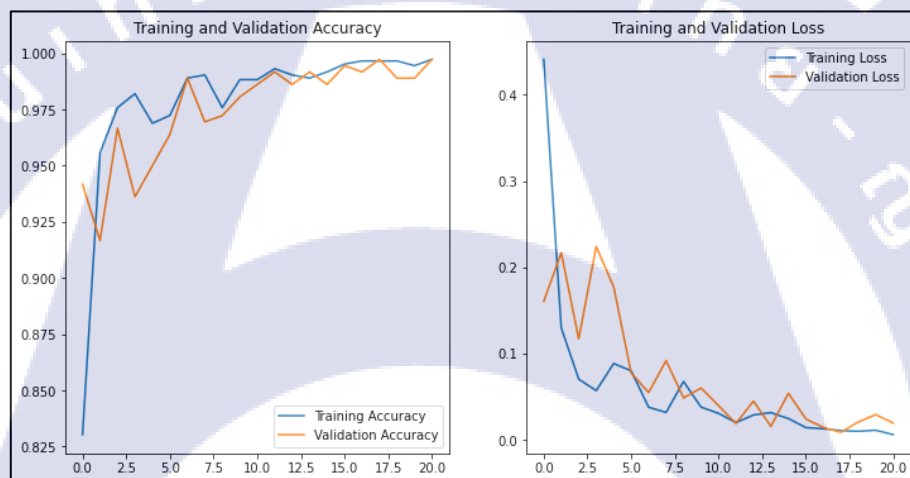
หมายเหตุ Precision, Recall และ F1 Score มีค่าอยู่ในช่วงระหว่าง 0 และ 1

4.1.3 VGG16

จำนวนรอบในการเรียนรู้ (epochs) จำนวน 21 รอบ ใช้เวลาในการเรียนรู้ 3 ชั่วโมง 34 นาที ผลลัพธ์จากการประมวลผลของการประเมินสมรรถนะแบบจำลองด้วย Confusion Matrix จากผลลัพธ์มีค่าความแม่นยำในการทำนายในแต่ละประเภท ซึ่งค่าความแม่นยำของค่า True Positive คือ เกรดเอ (Grade A) 146 รูป เกรดบี (Grade B) 161 รูป และรังมูม (Corner) 142 รูป ดังรูปที่ 4.5-4.6 อีกทั้งให้ค่าความถูกต้อง (Accuracy) เท่ากับ 99.34% ค่าความแม่นยำ Precision เท่ากับ 0.99 ค่าความแม่นยำ Recall เท่ากับ 0.99 และค่าความแม่นยำ F1-score เท่ากับ 0.99 ดังตารางที่ 4.4

Confusion Matrix			
Actual	Corner	GradeA	GradeB
Corner	142	0	0
GradeA	3	146	0
GradeB	0	0	161
Predicted			

รูปที่ 4.5 แสดงค่า Confusion Matrix ของ VGG16



รูปที่ 4.6 กราฟแสดงค่า Accuracy และ Loss ของ VGG16

ตารางที่ 4.4 ตารางแสดงประสิทธิภาพ ของ VGG16

Class	Precision	Recall	F1 Score	Accuracy (%)
Grade A	1.00	0.98	0.99	99.34
Grade B	1.00	1.00	1.00	
Corner	0.98	1.00	0.99	
Average	0.99	0.99	0.99	

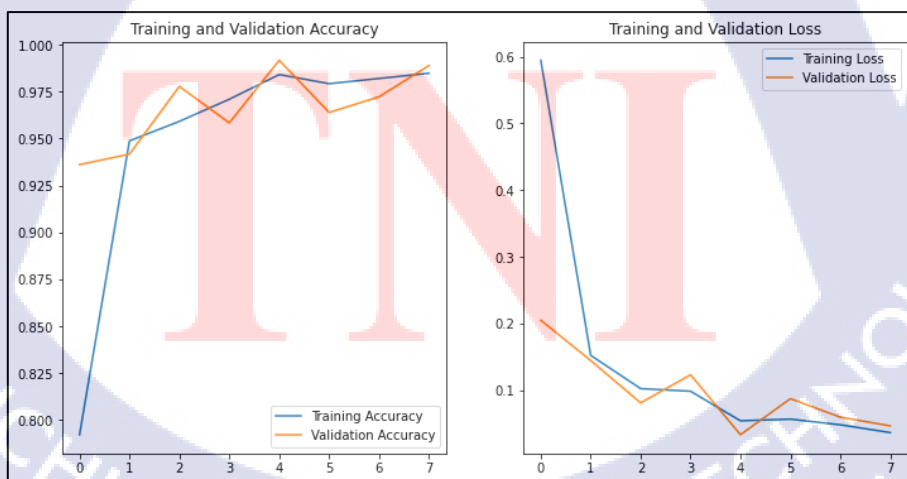
หมายเหตุ Precision, Recall และ F1 Score มีค่าอยู่ในช่วงระหว่าง 0 และ 1

4.1.4 VGG19

จำนวนรอบในการเรียนรู้ (epochs) จำนวน 8 รอบ ใช้เวลาในการเรียนรู้ 1 ชั่วโมง 18 นาที ผลลัพธ์จากการประมวลผลของการประเมินสมรรถนะแบบจำลองด้วย Confusion Matrix จากผลลัพธ์มีค่าความแม่นยำในการทำนายในแต่ละประเภท ซึ่งค่าความแม่นยำของค่า True Positive คือ เกรดเอ (Grade A) 146 รูป เกรดบี (Grade B) 161 รูป และรังมูม (Corner) 142 รูป ดังรูปที่ 4.7-4.8 อีกทั้งให้ค่าความถูกต้อง (Accuracy) เท่ากับ 95.58% ค่าความแม่นยำ Precision เท่ากับ 0.96 ค่าความแม่นยำ Recall เท่ากับ 0.96 และค่าความแม่นยำ F1-score เท่ากับ 0.96 ดังตารางที่ 4.5

Confusion Matrix			
	Actual Corner	Actual GradeA	Actual GradeB
Predicted Corner	142	13	6
Predicted GradeA	0	135	0
Predicted GradeB	0	1	155

รูปที่ 4.7 แสดงค่า Confusion Matrix ของ VGG19



รูปที่ 4.8 กราฟแสดงค่า Accuracy และ Loss ของ VGG19

ตารางที่ 4.5 ตารางแสดงประสิทธิภาพ ของ VGG19

	Precision	Recall	F1 Score	Accuracy (%)
Grade A	1.00	0.91	0.95	95.58
Grade B	0.99	0.96	0.98	
Corner	0.88	1.00	0.94	
Average	0.96	0.96	0.96	

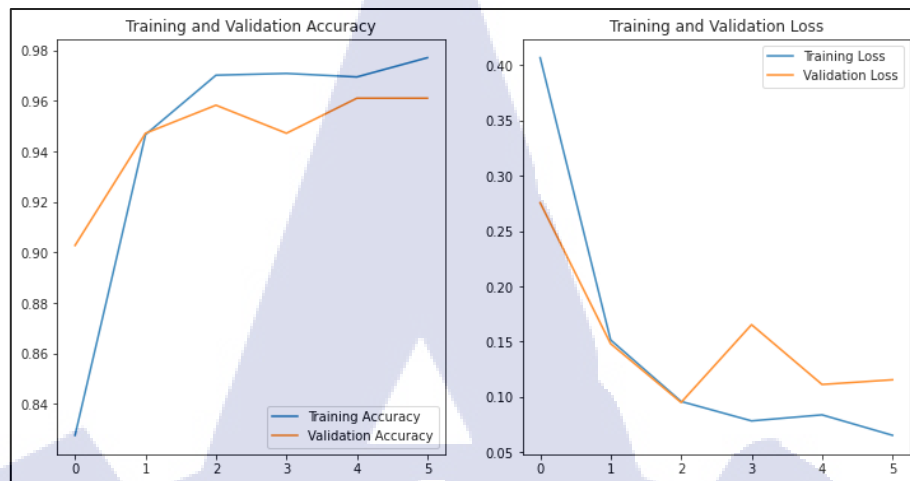
หมายเหตุ Precision, Recall และ F1 Score มีค่าอยู่ในช่วงระหว่าง 0 และ 1

4.4.5 DenseNet121

จำนวนรอบในการเรียนรู้ (epochs) จำนวน 6 รอบ ใช้เวลาในการเรียนรู้ 1 ชั่วโมง 6 นาที ผลลัพธ์จากการประมวลผลของการประเมินสมรรถนะแบบจำลองด้วย Confusion Matrix จากผลลัพธ์มีค่าความแม่นยำในการทำนายในแต่ละประเภท ซึ่งค่าความแม่นยำของค่า True Positive คือ เกรดเอ (Grade A) 136 รูป เกรดบี (Grade B) 161 รูป และวังมุม (Corner) 141 รูป ดังรูปที่ 4.9-4.10 อีกทั้งให้ค่าความถูกต้อง (Accuracy) เท่ากับ 96.90% ค่าความแม่นยำ Precision เท่ากับ 0.97 ค่าความแม่นยำ Recall เท่ากับ 0.97 และค่าความแม่นยำ F1-score เท่ากับ 0.97 ดังตารางที่ 4.6

Confusion Matrix			
	Corner	GradeA	GradeB
Actual Corner	141	1	0
Actual GradeA	10	136	3
Actual GradeB	0	0	161
	Corner	GradeA	GradeB
	Predicted		

รูปที่ 4.9 แสดงค่า Confusion Matrix ของ DenseNet121



รูปที่ 4.10 กราฟแสดงค่า Accuracy และ Loss ของ DenseNet121

ตารางที่ 4.6 ตารางแสดงประสิทธิภาพ ของ DenseNet121

	Precision	Recall	F1 Score	Accuracy (%)
Grade A	0.99	0.91	0.95	96.90
Grade B	0.98	1.00	0.99	
Corner	0.93	0.99	0.96	
Average	0.97	0.97	0.97	

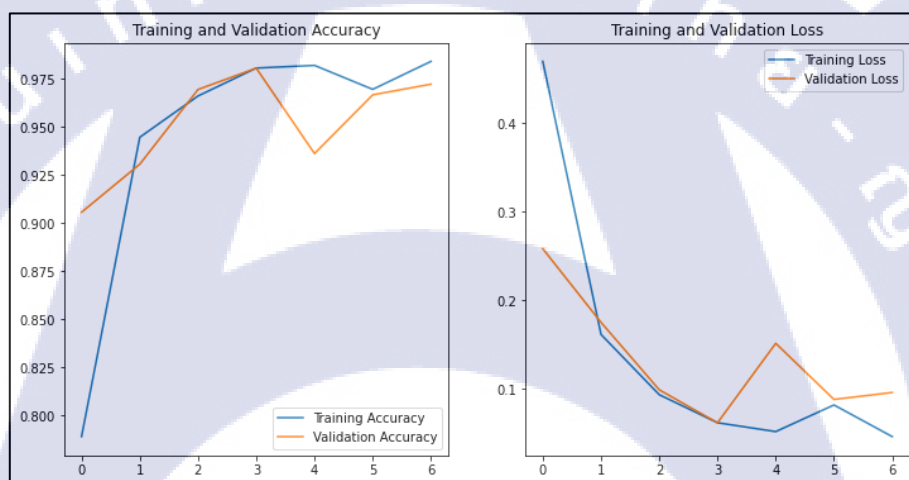
หมายเหตุ Precision, Recall และ F1 Score มีค่าอยู่ในช่วงระหว่าง 0 และ 1

4.1.6 DenseNet169

จำนวนรอบในการเรียนรู้ (epochs) จำนวน 7 รอบ ใช้เวลาในการเรียนรู้ 1 ชั่วโมง 12 นาที ผลลัพธ์จากการประมวลผลของการประเมินสมรรถนะแบบจำลองด้วย Confusion Matrix จากผลลัพธ์มีค่าความแม่นยำในการทำนายในแต่ละประเภท ซึ่งค่าความแม่นยำของค่า True Positive คือ เกรดเอ (Grade A) 142 รูป เกรดบี (Grade B) 161 รูป และรั้งมุม (Corner) 141 รูป ดังรูปที่ 4.11-4.12 อีกทั้งให้ค่าความถูกต้อง (Accuracy) เท่ากับ 98.23% ค่าความแม่นยำ Precision เท่ากับ 0.98 ค่าความแม่นยำ Recall เท่ากับ 0.98 และค่าความแม่นยำ F1-score เท่ากับ 0.98 ดังตารางที่ 4.7

Confusion Matrix			
Actual	Corner	GradeA	GradeB
	141	1	0
	6	142	1
	0	0	161
Predicted			

รูปที่ 4.11 แสดงค่า Confusion Matrix ของ DenseNet169



รูปที่ 4.12 กราฟแสดงค่า Accuracy และ Loss ของ DenseNet169

ตารางที่ 4.7 ตารางแสดงประสิทธิภาพ ของ DenseNet169

	Precision	Recall	F1 Score	Accuracy (%)
Grade A	0.99	0.95	0.97	98.23
Grade B	0.99	1.00	1.00	
Corner	0.96	0.99	0.98	
Average	0.98	0.98	0.98	

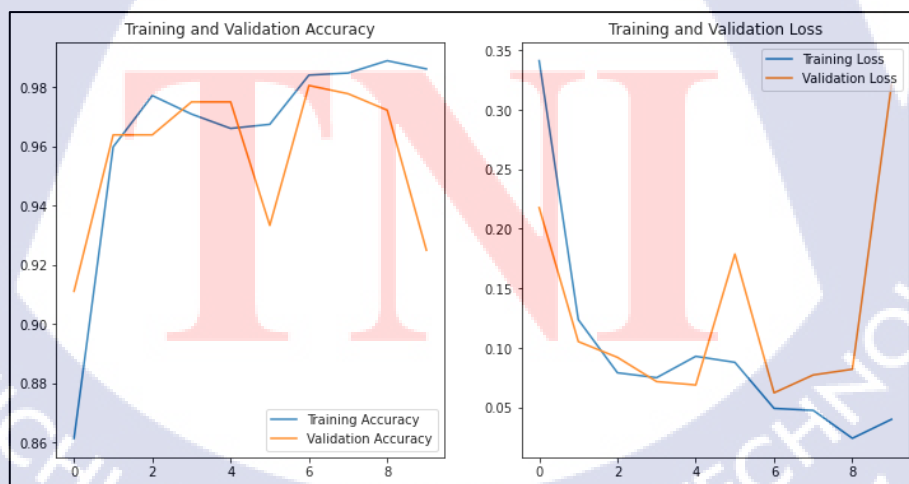
หมายเหตุ Precision, Recall และ F1 Score มีค่าอยู่ในช่วงระหว่าง 0 และ 1

4.1.7 DenseNet201

จำนวนรอบในการเรียนรู้ (epochs) จำนวน 10 รอบ ใช้เวลาในการเรียนรู้ 1 ชั่วโมง 45 นาที ผลลัพธ์จากการประมวลผลของการประเมินสมรรถนะแบบจำลองด้วย Confusion Matrix จากผลลัพธ์มีค่าความแม่นยำในการทำนายในแต่ละประเภท ซึ่งค่าความแม่นยำของค่า True Positive คือ เกรดเอ (Grade A) 146 รูป เกรดบี (Grade B) 161 รูป และรังมูม (Corner) 142 รูป ดังรูปที่ 4.13-4.14 อีกทั้งให้ค่าความถูกต้อง (Accuracy) เท่ากับ 99.34% ค่าความแม่นยำ Precision เท่ากับ 0.99 ค่าความแม่นยำ Recall เท่ากับ 0.99 และค่าความแม่นยำ F1-score เท่ากับ 0.99 ดังตารางที่ 4.8

	Predicted		
Actual	Corner	GradeA	GradeB
	140	2	0
	0	148	1
	0	0	161

รูปที่ 4.13 แสดงค่า Confusion Matrix ของ DenseNet201



รูปที่ 4.14 กราฟแสดงค่า Accuracy และ Loss ของ DenseNet201

ตารางที่ 4.8 ตารางแสดงประสิทธิภาพ ของ DenseNet201

Class	Precision	Recall	F1 Score	Accuracy (%)
Grade A	0.99	0.99	0.99	99.34
Grade B	0.99	1.00	1.00	
Corner	1.00	0.99	0.99	
Average	0.99	0.99	0.99	

หมายเหตุ Precision, Recall และ F1 Score มีค่าอยู่ในช่วงระหว่าง 0 และ 1



บทที่ 5

สรุปและอภิปราย

วิทยานิพนธ์ฉบับนี้ มีวัตถุประสงค์เพื่อศึกษาและออกแบบโครงข่ายประสาทเทียม เพื่อนำไปใช้ในการจำแนกเกรดรังนกแอ่นกินรัง จากการออกแบบและการทดลองประสิทธิภาพของโครงข่ายประสาทเทียม มีผลสรุปดังนี้

5.1 สรุปผลวิจัย

วิทยานิพนธ์ฉบับนี้ได้นำเสนอวิธีการประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึก (Deep Learning) โดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks) ทำการเปรียบเทียบประสิทธิภาพ 7 สถาปัตยกรรม ได้แก่ MobileNetV2, Xception, DenseNet121, DenseNet169, DenseNet201, VGG16, และ VGG19 สำหรับจำแนกเกรดรังนกแอ่นกินรัง จำนวน 3 ประเภท คือ เกรดเอ (Grade A) เกรดบี (Grade B) และรังมุม (Corner) จากผลการทดลองพบว่า DenseNet201 และ VGG16 มีค่าความถูกต้องเท่ากันคือ 99.34% ซึ่งมากที่สุด รองลงมาคือ DenseNet169 มีค่าความถูกต้องคือ 98.23% ซึ่งได้นำเทคนิคการเพิ่มจำนวนรูปภาพ (Data Augmentation) มาใช้เนื่องจากชุดข้อมูลมีจำนวนน้อย ซึ่งทำให้เป็นผลดีต่อโมเดลที่ทำการทดลองเพื่อให้สามารถเรียนรู้ข้อมูลได้หลากหลาย ทำให้โมเดลมีความสามารถในการทำนายได้สูงขึ้น จากการเปรียบเทียบประสิทธิภาพ 7 สถาปัตยกรรม โดยการ pre-trained model สามารถช่วยลดระยะเวลาในการเรียนรู้ได้มาก อีกทั้งสามารถช่วยให้โมเดลมีประสิทธิภาพสูงขึ้น ดังนั้น DenseNet201 และ VGG16 เหมาะสำหรับจำแนกเกรดรังนกแอ่นกินรัง เนื่องจากมีค่าความถูกต้องที่สูง และยังช่วยลดระยะเวลาในการเรียนรู้

5.2 ข้อเสนอแนะ

1. การเพิ่มจำนวนข้อมูลภาพรังนกแอ่นกินรังให้มากขึ้น ซึ่งข้อมูลภาพยังมีน้อยด้วยข้อจำกัดในการเก็บ รวบรวมข้อมูลภาพที่ต้องสร้างขึ้นใหม่ทั้งหมด ในอนาคตหากสามารถเพิ่มจำนวนข้อมูลภาพให้มากขึ้นจะเป็นผลดีมากยิ่งขึ้น
2. เนื่องจากมีการนำเทคนิคการเพิ่มจำนวนข้อมูลภาพ จึงทำให้มีการย่อหรือขยายข้อมูลภาพ ซึ่งอาจมีผลต่อตัวโมเดล จึงควรระมัดระวังมากยิ่งขึ้น

3. การควบคุมพื้นหลัง เนื่องจากข้อจำกัดในการจัดเก็บข้อมูลภาพรังก่อนเกินรังพื้นหลัง จึงเป็นสีฟ้า ซึ่งส่งผลต่อการเรียนรู้ของโมเดล ดังนั้นควรควบคุมพื้นหลังก่อนให้โมเดลเรียนรู้ข้อมูลภาพ อาจเปลี่ยนพื้นหลังเป็นสีขาวหรือไม่มีสี ซึ่งหากไม่ควบคุมพื้นหลังตรงตามข้อมูลภาพที่ใช้ในการสอน จะทำให้ประสิทธิภาพต่ำลง

5.3 ปัญหาและอุปสรรค

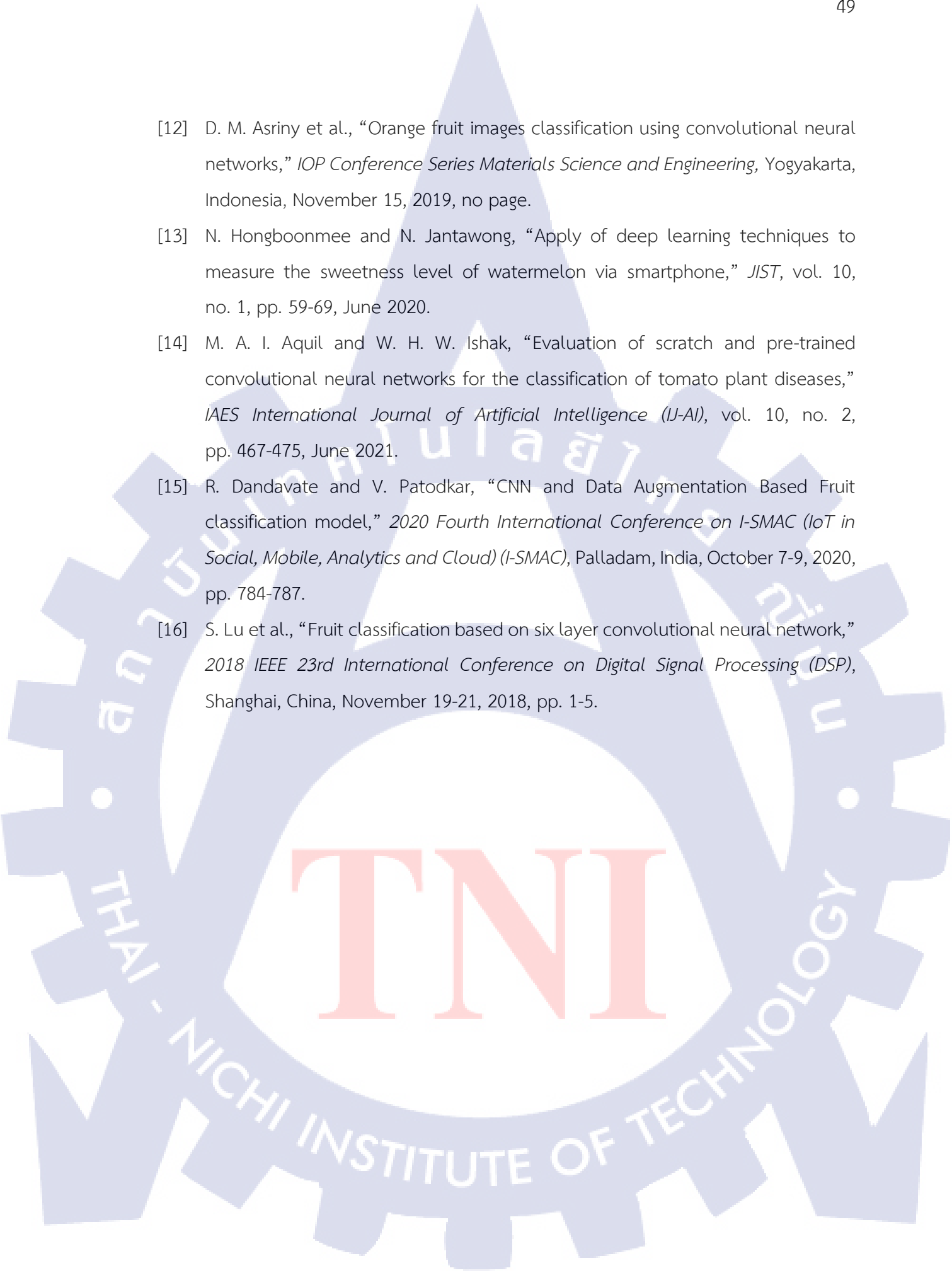
ช่วงของการทดลองมีข้อจำกัดเรื่องของทรัพยากรในการใช้งานหน่วยประมวลผลภาพ กราฟิกส์ (Graphics Processing Unit : GPU) ของเครื่องคอมพิวเตอร์ ซึ่งมีความสำคัญสำหรับ คอมพิวเตอร์ ทำให้มีการประมวลผลที่ช้าและค้างขณะทำการทดลองจึงควรเลือกเครื่องคอมพิวเตอร์ที่มี GPU ที่เหมาะสมสำหรับใช้ทำการทดลอง

บรรณานุกรม

TNI

บรรณานุกรม

- [1] จิตติพันธ์ วสุธาริรมย์, “บล็อกกรวบรวมข้อมูลและขั้นตอนการทำบ้านร้างนกหรือฟาร์มนกแอ่น,” [Online]. Available: <http://swiftletline.blogspot.com>. [Accessed: Oct. 5, 2022].
- [2] S. S. Shwartz and S. B. David, *Understanding Machine Learning - From Theory to Algorithms*, Cambridge: Cambridge University Press, 2014.
- [3] C. C. Aggarwal, *Neural Networks and Deep Learning*, Cham: Springer, 2019.
- [4] M. G. Lim and J. H. Chuah, “Durian types recognition using deep learning techniques,” *2018 9th IEEE Control and System Graduate Research Colloquium (ICSGRC)*, Shah Alam, Malaysia, August 03-04, 2018, pp. 183-187.
- [5] Cs231n, “*Convolutional Neural Networks for Visual Recognition*,” [Online]. Available: <https://cs231n.github.io>. [Accessed: Oct. 5, 2022].
- [6] Keras, “*Keras Applications*,” [Online]. Available: <https://keras.io>. [Accessed: Oct. 5, 2022].
- [7] M. Sandler et al., “MobileNetV2: Inverted residuals and linear bottlenecks,” *IEEE/CVF Conference on Computer Vision and Pattern Recognition, RAMS 2009*, Salt Lake City, UT, USA, June 18-23, 2018, pp. 4510–4520.
- [8] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 21-26, 2017, no page.
- [9] K. Simonyan and A. Zisserman, “*Very Deep Convolutional Networks for Large-Scale Image Recognition*,” [Online]. Available: <http://arxiv.org/abs/1409.1556>. [Accessed: Oct. 5, 2022].
- [10] G. Huang et al., “*Densely Connected Convolutional Networks*,” [Online]. Available: <http://arxiv.org/abs/1608.06993>. [Accessed: Oct. 5, 2022].
- [11] Goadrich and H. Mark, “*Learning Ensembles of First Order Clauses That Optimize Precision Recall Curves*,” [Online]. Available: <https://minds.wisconsin.edu/1793>. [Accessed: Oct. 22, 2022].

- 
- [12] D. M. Asriny et al., "Orange fruit images classification using convolutional neural networks," *IOP Conference Series Materials Science and Engineering*, Yogyakarta, Indonesia, November 15, 2019, no page.
- [13] N. Hongboonmee and N. Jantawong, "Apply of deep learning techniques to measure the sweetness level of watermelon via smartphone," *JIST*, vol. 10, no. 1, pp. 59-69, June 2020.
- [14] M. A. I. Aquil and W. H. W. Ishak, "Evaluation of scratch and pre-trained convolutional neural networks for the classification of tomato plant diseases," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 2, pp. 467-475, June 2021.
- [15] R. Dandavate and V. Patodkar, "CNN and Data Augmentation Based Fruit classification model," *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Palladam, India, October 7-9, 2020, pp. 784-787.
- [16] S. Lu et al., "Fruit classification based on six layer convolutional neural network," *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*, Shanghai, China, November 19-21, 2018, pp. 1-5.