โปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้

Program for classify a plant from leaf image

พัทธนันท์ สุขเกษม

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Program for classify a plant from leaf image

PATTANANT SUKKASEM

A SPECIAL PROBLEM SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR

THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)

DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2018

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	โปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้ Program for classify a plant from leaf image
ชื่อนักศึกษา	นางสาวพัทธนันท์ สุขเกษม รหัสนักศึกษา 58050343
ปริญญา	วิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา	วิทยาการคอมพิวเตอร์
ปีการศึกษา	2561
อาจารย์ที่ปรึกษา	ผศ.ดร.นวลสวาท หิรัญสกลวงศ์

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์) ประจำปีการศึกษา 2561

คณะกรรมการสอบ	ลายมือชื่อ
รศ.ดร.จีรพร วีรพันธุ์ ประธานกรรมการ	
ดร.กัญณัฏฐ์ หอมทรัพย์ กรรมการ	
ผศ.ดร.นวลสวาท หิรัญสกลวงศ์ กรรมการและอาจารย์ที่ปรึกษา	

ลิขสิทธิ์ของคณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ โปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้
ชื่อนักศึกษา นางสาวพัทธนันท์ สุขเกษม รหัสนักศึกษา 58050343
ปริญญา วิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา วิทยาการคอมพิวเตอร์
คณะ วิทยาศาสตร์
มหาวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)
ปีการศึกษา 2561
อาจารย์ที่ปรึกษา ผศ.ดร.นวลสวาท หิรัญสกลวงศ์

บทคัดย่อ

โปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้มีจุดประสงค์ในการนำวิธีการทาง Deep learning มาประยุกต์ใช้ในการทำนาย class ของรูปภาพใบไม้ที่มี class ชนิดเดียวกับโมเดลที่ได้รับการเรียนรู้ ในการวิจัยครั้งนี้มีรูปภาพใบไม้ 548 รูป นำรูปภาพไปแบ่งตาม class ในแต่ละ class มีจำนวนรูปภาพไม่เท่ากันจะนำรูปภาพทั้งหมด 5 class ไปสร้างโมเดลแยกชนิดพืชจากรูปภาพใบไม้ใน 5 class ด้วยการแย่งอัตราส่วนของรูปภาพในแต่ละ class เป็น Train 70 % ของรูปภาพใน class และ Validation 30% ของรูปภาพใน class ด้วยวิธีการ Deep neural network โดยสามารถวัดผลประสิทธิภาพด้วยการนำรูปภาพใบไม้ที่มี class เดียวกับรูปภาพในโมเดลแต่ไม่เคยถูก Train ในโมเดล นำมาใช้ในการทดสอบประสิทธิภาพในการแยก class ของรูปที่เป็น Input ใหม่ด้วยการทำนายค่าความมั่นใจเพื่อแยกว่าเป็นรูปภาพตรงกับ class ไหนกี่ % จากการทดสอบประสิทธิภาพจากรูปภาพตัวอย่าง 10 รูป พบว่าโมเดลที่ได้รับการเรียนรู้สามารถทำนายผลถูกต้อง 7 รูป และทำนายผลไม่ถูกต้อง 3 รูป ซึ่งประสิทธิภาพของโมเดล สามารถนำไปใช้แยกชนิดพืชได้แต่ยังไม่มีประสิทธิภาพพอ ซึ่งยังต้องปรับปรุงในเรื่องของรูปภาพในการฝึกฝน จำนวน class ของรูปภาพให้มีความหลากหลายมากยิ่งขึ้น เพื่อให้สามารถนำไปเป็นแนวทางการปรับปรุงประสิทธิภาพในโปรแกรมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสำคัญ : โมเดล , Deep learning , Deep neural network

Title	Program for classify a plant from a leaf image
Students	Miss Pattanant Sukkase Student ID 58050343
Degree	Bachelor of Science (Computer Science)
Department	Computer Science
Faculty	Science
University	King Mongkut's Institute of Technology Ladkrabang
Academic Year	2561
Advisor	Asst.Prof.Dr. Nualsawat Hiransakolwong

Abstract

An application for separating plant species from leaf images aims to apply the Deep learning method to predict the class of leaf images with the same class type as the model that has been learned. In this research, there are 548 pictures of leaves. Pictures can be divided into 5 classes. There are not equal numbers of pictures in each class. Images in each class are taken randomly with amount of 70 % using for training to create a separate model of the plant and 30% of the images in the class are using for validation with using the Deep neural network method. It can measure performance by bringing the image of the leaves with the same class as the image in the model but never training. From the testing performance, 10 sample images were used to test with the model that has been learned predicting correctly with 7 results and predicting incorrectly with 3 results. The efficiency of the model can be used to separate plant species with effective 70 %. This is able to be used as a guideline for improving more efficiency in the future work.

Keywords : Model , Deep learning , Deep neural network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การทำปัญหาพิเศษเล่มนี้สำเร็จลุล่วงไปได้ด้วยดีจากการช่วยเหลือและสนับสนุนของบุคคลหลายท่าน ผู้จัดทำขอขอบพระคุณ ผศ.ดร.นวลสวาท หิรัญสกลวงศ์ อาจารย์ที่ปรึกษาปัญหาพิเศษที่คอยให้คำแนะนำ ให้คำปรึกษาแนวทางการทำงานอย่างใกล้ชิด รวมถึงช่วยแนะนำการแก้ไขปัญหาที่เกิดขึ้นในการทำปัญหาพิเศษฉบับนี้

ขอขอบพระคุณประธานกรรมการ รศ.ดร.จีรพร วีรพันธุ์ และกรรมการปัญหาพิเศษ ดร.กัญณัฏฐ์ หอมทรัพย์ อาจารย์ภาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ช่วยให้คำแนะนำและชี้จุดบุกพร่องที่ควรแก้ไข และตรวจสอบเพิ่มความสมบูรณ์ให้กับปัญหาพิเศษฉบับนี้

ขอขอบพระคุณคณาจารย์ในภาพวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ ซึ่งได้ให้ความรู้ทางวิชาการและความรู้อันเป็นพื้นฐานแก่ผู้จัดทำมาตลอดระยะเวลา4ปี ท้ายสุดนี้ผู้จัดทำขอขอบคุณ บิดา มารดา และบุคคลในครอบครัว รวมทั้งขอบคุณเพื่อนๆ พี่ๆ น้องๆ ที่ให้ความช่วยเหลือ คอยสนับสนุนและให้กำลังใจตลอดมาจนปัญหาพิเศษนี้สำเร็จลุล่วงไปได้ด้วยดี

พัทธนันท์ สุขเกษม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ก
บทคัดย่อภาษาอังกฤษ ..... ข
กิตติกรรมประกาศ ..... ค
สารบัญ ..... ค
สารบัญ ..... ง
สารบัญรูป ..... ฉ
บทที่ 1 บทนำ 1
1.1 ที่มาและความสำคัญ 1
1.2 วัตถุประสงค์ในการทำวิจัย 2
1.3 ขอบเขตการวิจัย 2
1.4 สมมุติฐานการวิจัย 2
1.5 ประโยชน์ที่คาดว่าจะได้รับ 2
1.6 อุปกรณ์ที่ใช้ในการทำงานวิจัย 3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 4
2.1 Python 4
2.2 Anaconda 4
2.3 Jupyter notebook 5
2.4 Keras 6
2.5 Tensorflow 6
2.6 Deep learing 7
2.7 Neural network 8
2.8 Convolutional Neural Network 9
2.9 Pre-processing 10
2.10 Image classification 10
บทที่ 3 วิธีการดำเนินงานวิจัย 11
3.1 ภาพรวมในการเตรียมการทำงาน 11
3.2 การแบ่งข้อมูลในการทำงาน 12
3.3 Algorithm ที่ใช้ 18
3.4 การสร้างโมเดล 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.5 การเช็ค performance 20

บทที่ 4 ผลการวิจัยและการอภิปรายผล 23

4.1 การทดลองประสิทธิภาพในใช้งาน 23

4.2 สรุปการทดสอบประสิทธิภาพ 27

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ 29

5.1 สรุปผลการดำเนินงาน 29

5.2 ข้อจำกัดของโปรแกรม 29

5.3 ข้อเสนอแนะ 29

เอกสารอ้างอิง 30

ภาคผนวก 32

ภาคผนวก ก ขั้นตอนการติดตั้งโปรแกรม 33

ภาคผนวก ข โค้ดตัวอย่างของโปรแกรม 40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 python	4
2.2 หน้าตาของAnaconda	5
2.3 หน้าตาของ Jupyter notebook	6
2.4 Keras	6
2.5 Tensorflow	7
2.6 Deep learning	8
2.7 Neural Network	9
2.8 Convolutional Neural Network	9
3.1 ภาพรวมในการเตรียมการทำงาน	11
3.2 การแบ่งข้อมูลในการทำงาน	13
3.3 ตัวอย่างการแบ่งข้อมูลใน Train data	13
3.4 Folder train class kaffir lime	14
3.5 Folder train class mango	14
3.6 Folder train class mint	15
3.7 Folder train class wildbetal Leafbush	15
3.8 Folder train class jackfruit	15
3.9 ตัวอย่างการแบ่งข้อมูลใน Validation data	16
3.10 Folder validation class wildbetal Leafbush	16
3.11 Folder validation class jackfruit	16
3.12 Folder validation class mint	17
3.13 Folder validation class mango	17
3.14 Folder validation class kaffir lime	17
3.15 Algorithm Deep Learning	18
3.15 ตัวอย่างการทำงานในส่วนของระบบ layer	19
3.16 Train model	20
3.17 Accuracy	21
3.18 loss	21
4.1 ผลลัพท์หลังใช้โมเดลทำนาย	23
4.2 ผลลัพท์หลังใช้โมเดลทำนาย	24
4.3 ผลลัพท์หลังใช้โมเดลทำนาย	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ **หน้า**

4.4 ผลลัพท์หลังใช้โมเดลทำนาย 25
4.5 ผลลัพท์หลังใช้โมเดลทำนาย..... 25
4.6 ผลลัพท์หลังใช้โมเดลทำนาย.....26
4.7 ผลลัพท์หลังใช้โมเดลทำนาย.....26
4.8ผลลัพท์หลังใช้โมเดลทำนาย.....27
4.9 ผลลัพท์หลังใช้โมเดลทำนาย..... 27
4.10 ผลลัพท์หลังใช้โมเดลทำนาย.....28
ก.1 เข้าweb anaconda..... 33
ก.2 download 3.6..... 33
ก.3 double click anaconda..... 34
ก.4 I agree..... 34
ก.5 just me..... 35
ก.6 Destination Folder..... 35
ก.7 Add anaconda to my path..... 36
ก.8 installing..... 36
ก.9 Next..... 37
ก.10 Finish..... 37
ก.11 Anaconda prompt..... 38
ก.12 command prompt..... 38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1ที่มาและความสำคัญของปัญหา

ประเทศไทยเป็นประเทศเกษตรกรรม และมีสภาพแวดล้อมที่เหมาะสมต่อการเกษตร ซึ่งมีพื้นที่ทางการเกษตร ชาวเกษตรกรส่วนใหญ่ยึดอาชีพการเกษตรเป็นหลักเพื่อใช้ในการดำรงชีวิต ประเทศไทยมีการผลิตพืชที่หลากหลายชนิด ทั้งนี้เนื่องจากมีสภาพภูมิประเทศที่มีความแตกต่างกันค่อนข้างมาก จึงทำให้ลักษณะภูมิอากาศแตกต่างกันไปด้วย ผลจากสภาพภูมิประเทศและภูมิอากาศที่แตกต่างกันออกไปในแต่ละพื้นที่ทำให้มีความหลากหลายของระบบการปลูกพืช ในปัจจุบันมีพืชหลากหลายชนิด พืช(plant)เป็นสิ่งมีชีวิตที่มีความหลากหลายประเภทหนึ่งของโลก จากทั่วโลกมีพืชอยู่ประมาณ 300,000 ชนิด เพื่อให้การศึกษาเรื่องอาณาจักรพืชเป็นไปได้โดยง่ายนักวิทยาศาตร์จึงจัดระบบพืชประเภทต่างๆที่มีลักษณะคล้ายคลึงกันมาไว้ในประเภทเดียวกัน พบได้ทั้งบนบกและในน้ำ เป็นสิ่งมีชีวิตที่เนื้อเยื่อส่วนใหญ่ประกอบด้วยหลายเซลล์ นิวเคลียสมีผนังเซลล์ ห่อหุ้ม เคลื่อนที่ไม่ได้ ไม่มีอวัยวะเกี่ยวกับความรู้สึก มีคลอโรฟิลล์ซึ่งเป็นสารสีเขียว ช่วยในการสังเคราะห์แสง นอกจากนี้ยังมีลักษณะพิเศษที่ต่างไปจากสิ่งมีชีวิตชนิดอื่นก็คือการสังเคราะห์ด้วยแสงที่เกิดขึ้นบริเวณส่วนใบที่เป็นสีเขียวของพืช

ใบไม้ (Leaf) เป็นอวัยวะหรือส่วนของพืชที่เจริญเติบโตยื่นออกมาจากด้านข้างของลำต้นบริเวณข้อ ใบทำหน้าที่สังเคราะห์อาหาร ปกติใบมีสีเขียวของคลอโรฟิลล์ซึ่งทำหน้าที่สังเคราะห์ด้วยแสง นอกจากนั้นอาจมีรงควัตถุอื่นปนอยู่ด้วยแต่มีปริมาณน้อยกว่าคลอโรฟิลล์ จึงไม่เห็นสีของสารประกอบเหล่านั้น ใบมีขนาดและรูปร่างแตกต่างกันไปตามชนิดของพืช แต่ส่วนใหญ่แล้วใบมีลักษณะแผ่ แบน บางชนิดใบอาจลดขนาดมีรูปร่างคล้ายเข็ม บางชนิดอาจเป็นแผ่นหรือเกล็ดเล็ก ๆ ไม่มีสีเขียว หรือบางชนิดอาจเปลี่ยนเป็นรูปทรงกระบอกกลวง

การแยกชนิดพืชจากใบไม้ด้วยตาเปล่าสามารถทำได้แต่การที่มีใบไม้ลักษณะคล้ายกันมันยังสามารถทำได้ยากเพราะใบไม้มีหลากหลายชนิดมากบนโลก และแต่ละชนิดอาจมีลักษณะคล้ายกันมากเพราะใบไม้มีสี รูปร่าง พื้นผิว แตกต่างกัน เส้นใบที่เรียงตัวไม่เหมือนกันทำให้เกิดปัญหาในการแยกชนิดของใบไม้ดังนั้นผู้วิจัยจึงจัดทำโปรแกรมเพื่อใช้ในการแยกชนิดพืชจากภาพถ่ายใบไม้ให้ถูกต้องและมีประสิทธิภาพ โดยโปรแกรมนี้จะพัฒนาขึ้นด้วยภาษา python ซึ่งใช้ keras เป็น library ใน python โดยวิจัยนี้ได้ทำโมเดลการเรียนรู้จากภาพถ่ายใบไม้เพื่อนำมาเรียนรู้เพื่อสอนคอมพิวเตอร์ให้สามารถแยกชนิดของใบไม้และสร้างโปรแกรมขึ้นมาจากการทำโมเดลการเรียนรู้ โดยตัวโปรแกรมนั้นใช้หลักการจำลองโครงข่ายประสาทเทียม (Deep learning) เพื่ออำนวยความสะดวกในการแยกชนิดของใบไม้และสามารถที่จะประยุกต์ปรับใช้และนำไปพัฒนาต่อในอนาคตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2วัตถุประสงค์ในการทำวิจัย

1.) เพื่อสร้างโปรแกรมแยกชนิดพืชของใบไม้จากภาพถ่ายใบไม้ได้

2.) เพื่อประยุกต์ความรู้จากการสร้างโมเดลการเรียนรู้ของเครื่องจักรโดยใช้ภาพถ่ายมาสร้างโปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้

3.) เพื่อศึกษาและพัฒนาขั้นตอนการแยกชนิดของใบไม้ จากภาพถ่ายโดยใช้หลักการทาง deep learning

1.3 ขอบเขตการวิจัย

1.) การสร้างโมเดลการเรียนรู้ของการจำลองโครงข่ายประสาทเทียมโดยใช้ข้อมูลจากรูปภาพใบไม้ชนิดต่างๆ ได้แก่ ใบมะม่วง ใบชะพลู ใบขนุน ใบสาระแหน่ และใบมะกรูด

2.) การสร้างโปรแกรมแยกชนิดพืชจากรูปภาพใบไม้พัฒนามาจากโมเดลการเรียนรู้ของการจำลองโครงข่ายประสาทเทียมโดยใช้ข้อมูลจากภาพถ่ายใบไม้ชนิดต่างๆ

3.) โปรแกรมสามารถใช้ได้กับรูปภาพใบไม้ 5 ชนิด ได้แก่ ใบมะม่วง ใบชะพลู ใบขนุน ใบสาระแหน่ และใบมะกรูด

4.) โปรแกรมถูกพัฒนาขึ้นด้วยภาษา python

1.4 สมมุติฐานการวิจัย

1.) เพื่อสร้างโปรแกรมในการแยกชนิดพืชจากภาพถ่ายใบไม้ได้ โดยใช้หลักการทาง Deep learning

2.) เพื่อค้นหาเทคนิคในการสร้างโมเดลการเรียนรู้ที่มีประสิทธิภาพนำมาพัฒนาเป็นโปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้

3.) เพื่อสร้างโปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้ที่มีประสิทธิภาพมาพัฒนาให้สามารถใช้งานได้จริง

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ผลต่อผู้วิจัย

1.) เพื่อพัฒนาความรู้ความสามารถจากการประยุกต์ใช้ความรู้โดยการปฏิบัติจริง

2.) เพื่อพัฒนาโปรแกรมแยกชนิดพืชจากถ่ายใบไม้ให้สามารถใช้งานได้จริง

1.5.2 ผลต่อสังคม

1.) เพื่อสามารถนำโปรแกรมนี้ไปประยุกต์ใช้และพัฒนาต่อได้

2.) เพื่อช่วยให้การแยกชนิดพืชจากภาพถ่ายใบไม้ทำได้ง่ายขึ้น

1.5.3 ผลต่อประเทศ

1.) เพื่อทำให้สะดวกต่อการศึกษาในการแยกชนิดพันธุ์ใบไม้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.) เพื่ออำนวยความสะดวกในการแยกชนิดของใบไม้ทำได้ง่ายขึ้นโดยใช้งานโปรแกรมจากโมเดลการเรียนรู้ที่มีประสิทธิภาพ

1.6 อุปกรณ์ที่ใช้ในการทำงานวิจัย

1.6.1 ฮาร์ดแวร์

1.)เครื่องคอมพิวเตอร์

2.)คอมพิวเตอร์แบบพกพา

1.6.2 ซอฟแวร์

1.)ระบบปฏิบัติการ window 10

2.)Python 3.6

3.)Anaconda

4.)Library Keras 2.2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่2
เอกสาร ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

งานวิจัยนี้จะกล่าวถึงการประยุกต์ใช้โมเดล สำหรับใช้ในโปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้ โดยใช้หลักการและทฤษฎีต่างๆที่จะกล่าวต่อไปนี้มาประยุกต์ใช้กับการจำลองโครงข่ายประสาทเทียม (Neural Network) เพื่อให้สามารถแยกชนิดของภาพถ่ายใบไม้ได้ เพื่อนำมาใช้กับโปรแกรมที่สามารถนำไปพัฒนาต่อในอนาคตได้

2.1 ไพทอน (Python)

Python คือชื่อภาษาที่ใช้ในการเขียนโปรแกรมภาษาหนึ่ง ซึ่งถูกพัฒนาขึ้นมาโดยไม่ยึดติดกับแพลตฟอร์ม กล่าวคือสามารถรันภาษา Python ได้ทั้งบนระบบ Unix, Linux , Windows NT, Windows 2000, Windows XP หรือแม้แต่ระบบ FreeBSD อีกอย่างหนึ่งภาษาตัว นี้เป็น OpenSource เหมือนอย่าง PHP ทำให้ทุกคนสามารถที่จะนำ Python มาพัฒนาโปรแกรมของเราได้ฟรีๆโดยไม่ต้องเสียค่าใช้จ่าย และความเป็น Open Source ทำให้มีคนเข้ามาช่วยกันพัฒนาให้ Python มีความสามารถสูงขึ้น และใช้งานได้ครบคุมกับทุกลักษณะงาน

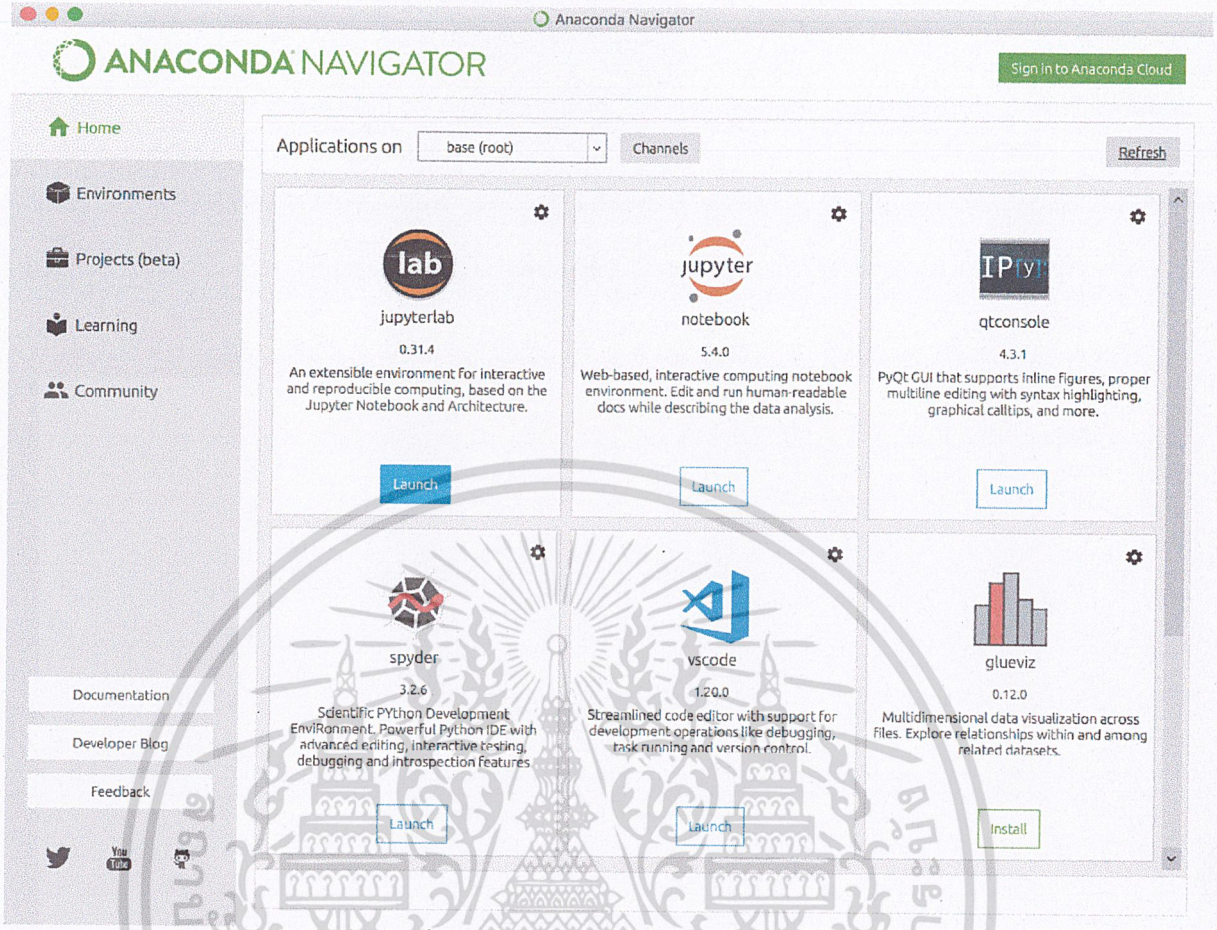
รูปที่2.1 python

ที่มาจาก (https://www.aosoft.co.th/article/322/Python-คืออะไร-ภาษา-python-ใช้ทำอะไร.html)

2.2 Anaconda

คือแพลตฟอร์ม data science **ที่อิงกับภาษาเขียนโค้ด** Python เป็น Python distribution ที่รวมแพกเกจมากมายสำหรับทำ data preparation, data analysis, data visualization, machine learning และ interactive data science โดยเราสามารถดาวน์โหลดและติดตั้งลงในเครื่องคอมพิวเตอร์ของเราได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



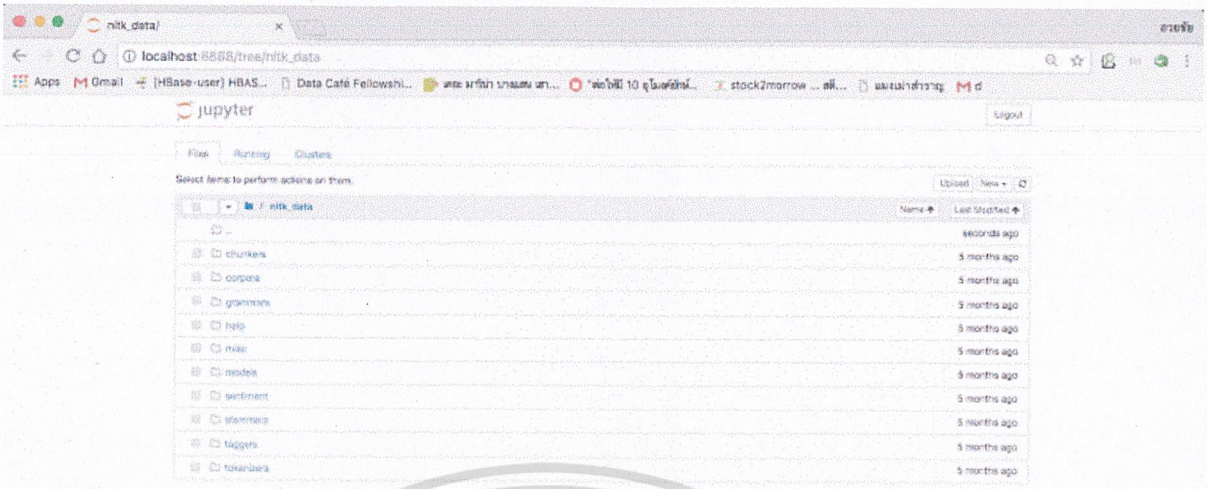
รูปที่2.2 หน้าตาของAnaconda

ที่มาจาก (https://medium.com/@saichonjaiyen/)

2.3 Jupyter notebook

เป็น Interactive computing environment ที่ให้ผู้ใช้สามารถป้อนคำสั่ง แล้วให้ประมวลผลคำสั่งนั้น ๆ พร้อมแสดงผลลัพธ์ต่อท้ายชุดคำสั่งตัว notebook สามารถเก็บ code และตัว markdown ที่ช่วยให้แสดงข้อมูลในแบบต่าง ๆ ได้ ถ้าต้องการใช้งานต้องติดตั้ง Anacondaก่อนถึงจะใช้งานได้ โดย Jupyter notebook สามารถเรียกใช้งานได้จากทุก directory location หรือถ้าทำไม่เป็นสามารถเปิดได้จากAnaconda >> Jupyter notebook ตัว Jupyter notebook จะทำงานในโหมด Wep application โดยสามารถเขียนโค้ด บนWeb UI ได้เลย และ run program แบบ realtime

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 หน้าตาของ Jupyter notebook

ที่มาจาก(https://www.auoychai.com/programmer)

2.4 Keras

คือ deep learning framework สำหรับ python ที่ทำให้วิธีของการฝึกสอน โมเดลด้วย deep learning ทำได้ง่ายขึ้น โดยในตอนแรก Keras ถูกพัฒนาเพื่อนักวิจัยโดยมีจุดประสงค์เพื่อทำให้การทดลองทำได้เร็วมากขึ้น เราจะเห็นว่า ถ้าใช้ Keras จะเขียน code ไม่กี่บรรทัดก็เสร็จ และ ถ้าไม่ใช้ Keras หน้าตาของ code ที่ใช้ในการเทรนโมเดลอยู่ภายใต้ MIT license ที่หมายความว่าเราสามารถใช้งานมันได้อย่างอิสระเพื่อธุรกิจที่หวังกำไรได้ โดยไม่ต้องจ่ายเงินให้ใคร และยังใช้ได้กับทุกเวอร์ชั่นของ python จาก 2.7 ถึง 3.6 และ อนุญาติให้ใช้ code เดียวกันในการรันทั้ง cpu และ gpu ได้เหมือนกัน มี API ที่ง่ายต่อการใช้งานและทำให้ ตัวต้นแบบในการสร้าง deep learning model เสร็จเร็ว

รูปที่ 2.4 Keras

ที่มาจาก (http://drtanet.blogspot.com/2018/09/keras.html)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 Tensorflow

ในการพัฒนา Deep Learning นั้น อาจจะเป็นเรื่องที่ยากในการเขียน เพราะว่ามีกระบวนการที่ซับซ้อน ทำให้มีการพัฒนาไลบรารีที่ช่วยในการเขียนหรือพัฒนา Deep Learning โดยทาง Google ได้ทำการพัฒนาไลบรารีซอฟต์แวร์โอเพนซอร์ส สำหรับการเขียนโปรแกรม dataflow ในงานหลายประเภท เรียกว่า TensorFlowเป็นไลบรารีที่ใช้ในการพัฒนา Machine Learning ได้รับการพัฒนาโดยบริษัท Google ได้ทำการเปิดตัวเมื่อวันที่ 11 กุมภาพันธ์ 2017 ซึ่ง TensorFlow นั้นจะเป็น Open source ที่จะใช้ python ในการเขียน รองรับเวอร์ชั่นทั้ง python 2 และ Python 3 โดย TensorFlow สามารถทำงานบน CPU และ GPUs รองรับระบบปฏิบัติการ Linux, macOS, Windows และ Android

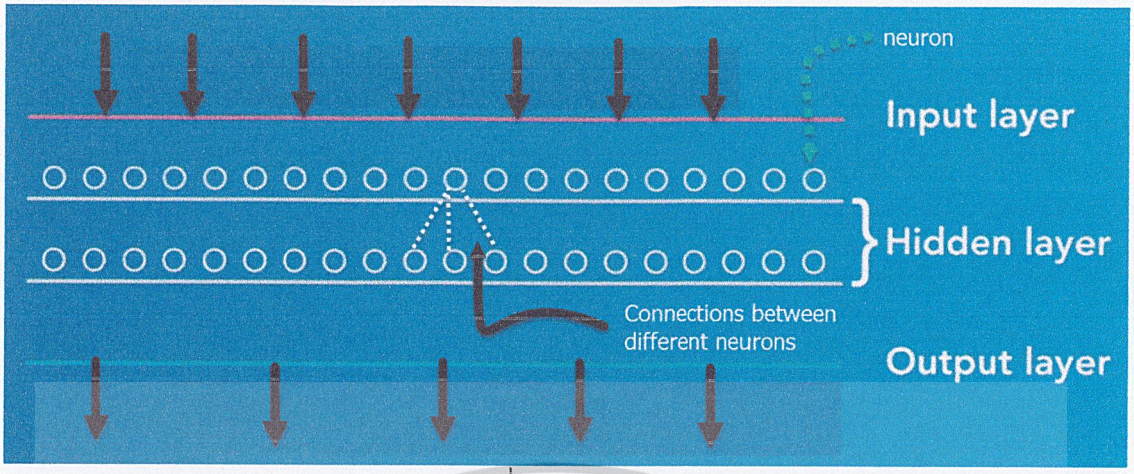
รูปที่ 2.5 Tensorflow

ที่มาจาก (https://mindphp.com/คู่มือ/73-คืออะไร/5964-tensorflow.html)

2.6 Deep learning

คือ ซอฟแวร์คอมพิวเตอร์ที่เลียนแบบการทำงานของระบบโครงข่ายประสาทในสมองมนุษย์ เป็นซับเซ็ตของ machine learning Algorithm ของ deep learning ถูกสร้างขึ้นจากการนำเอา neural network หลายๆ layer มาต่อกัน โดย layer แรกสุดจะทำหน้าที่ในการรับข้อมูล layer สุดท้ายจะทำหน้าที่ส่งผลลัพธ์ การประมวลผลออกมา ส่วน hidden layer เป็น layer ที่อยู่ระหว่าง layer แรกกับ layer สุดท้าย เป็นการต่อกันของ neural network หลายๆอันมาต่อกัน เนื่องจาก layer เหล่านี้เป็นโครงสร้างที่ถูกจัดเก็บแบบเป็นกองซ้อน จึงได้เปรียบเพราะมี layer ที่จำนวนเยอะๆ ก็จะทำให้มีโครงสร้างที่ลึกยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



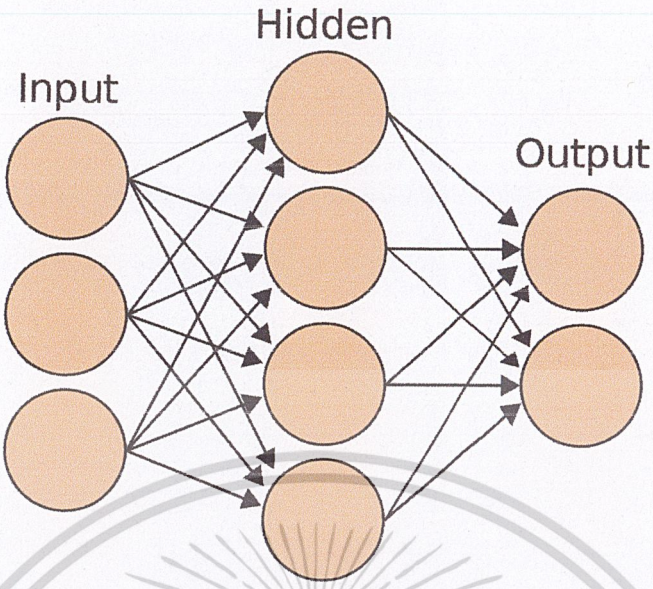
รูปที่ 2.6 Deep learning

ที่มาจาก (https://www.thaiprogrammer.org/2018/12/deep-learning-คืออะไร/)

2.7 Neural Network

โมเดลทางคอมพิวเตอร์ที่ใช้คณิตศาสตร์เข้ามาช่วยสำหรับประมวลผลสารสนเทศด้วยการคำนวณแบบคอนเนคชันนิสต์ (connectionist) แนวคิดเริ่มต้นของเทคนิคนี้ได้มาจากการศึกษาโครงข่ายไฟฟ้าชีวภาพ (bioelectric network) ในสมอง ซึ่งประกอบด้วย เซลล์ประสาท (neurons) และ จุดประสานประสาท (synapses) ตามหลักการของโครงข่ายประสาทเกิดจากการที่เซลล์ประสาทเชื่อมต่อระหว่างเซลล์ประสาท จนเกิดเป็นเครือข่ายที่ทำงานร่วมกัน โดย Input layer จะเป็นตัวรับข้อมูลนำเข้า Hidden layers จะทำการคำนวณข้อมูลที่ได้รับมาจาก layer ก่อนหน้า โดยวิธีการทางคณิตศาสตร์ และส่งไปยัง layer ชั้นถัดไป การสร้าง neural networks คือการตัดสินใจว่าเราควรมี hidden layers กี่ชั้น และมี neuron ในแต่ละ layer จำนวนเท่าใด Output layer จะเป็นตัวให้ผลลัพธ์ของการประมวลผลออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

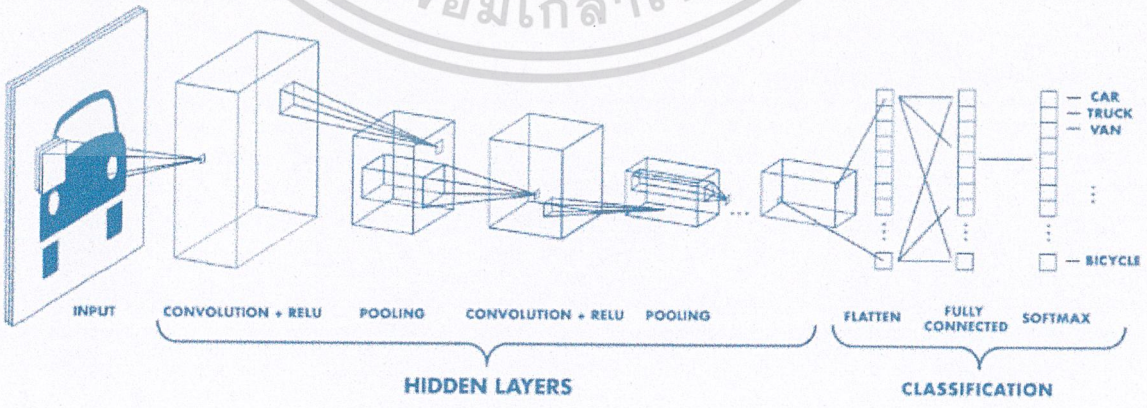


รูปที่ 2.7 Neural Network

ที่มาจาก (https://th.wikipedia.org/wiki/โครงข่ายประสาทเทียม)

2.8 Convolutional Neural Network

โครงข่ายประสาทแบบคอนโวลูชัน เป็นโครงข่ายประสาทเทียมหนึ่งในกลุ่ม bio-inspired โดยที่ CNN จะจำลองการมองเห็นของมนุษย์ที่มองพื้นที่เป็นที่ย่อยๆ และนำกลุ่มของพื้นที่ย่อยๆมาผสานกัน เพื่อดูว่าสิ่งที่เห็นอยู่เป็นอะไรกันแน่ การมองพื้นที่ย่อยของมนุษย์จะมีการแยกคุณลักษณะ (feature) ของพื้นที่ย่อยนั้น เช่น ลายเส้น และการตัดกันของสี ซึ่งการที่มนุษย์รู้ว่าพื้นที่ตรงนี้เป็นเส้นตรงหรือสีตัดกัน เพราะมนุษย์ดูทั้งจุดที่สนใจและบริเวณรอบ ๆ ประกอบกัน



รูปที่ 2.8 Convolutional Neural Network

ที่มาจาก (https://blog.datawow.io/มาลองดูวิธีการคิดของ-cnn-กัน-e3f5d73eebaa)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

2.9 Pre-processing

ขั้นตอน pre-processing เป็นขั้นตอนที่จะต้องเรียนรู้เพื่อเข้าใจข้อมูลอะไรสักอย่างที่เรามี แต่ข้อมูลที่มีเยอะยิ่งทำให้เราเข้าใจได้ง่ายขึ้น ในการใช้ในคอมพิวเตอร์เราต้องสอนโมเดลที่ใช้ในการจำแนกชนิดของใบไม้ เรามีข้อมูลรูปภาพที่เป็นใบไม้ มากพอที่จะให้คอมพิวเตอร์รู้จำได้ เราจะส่งข้อมูลเหล่านั้นเข้าไปฝึกสอน แต่ถ้าข้อมูลเหล่านั้นมีองค์ประกอบส่วนอื่นที่ไม่เกี่ยวข้องอยู่มากมันก็อาจจะยิ่งทำให้ โมเดลของเรานั้นมีความแปรปรวนสูง ดังนั้นกระบวนการ pre-processing จึงเป็นเรื่องสำคัญอย่างมากเพราะขั้นตอนนี้จะเป็นการ คัดแยกคุณลักษณะเด่นออกมาจากข้อมูล หรือเรียกว่า Feature Extraction

2.10 Image classification

เป็นการวิเคราะห์เชิงสถิติเพื่อคิดแยกกลุ่มของข้อมูล (จุดภาพ) ที่มีคุณสมบัติทางแสงที่เหมือนกันหรือใกล้เคียงกันให้เป็นกลุ่มข้อมูลประเภท (Class) เดียวกัน และคิดแยกกลุ่มข้อมูลที่มีคุณสมบัติทางแสงต่างกันให้ออกจากกัน

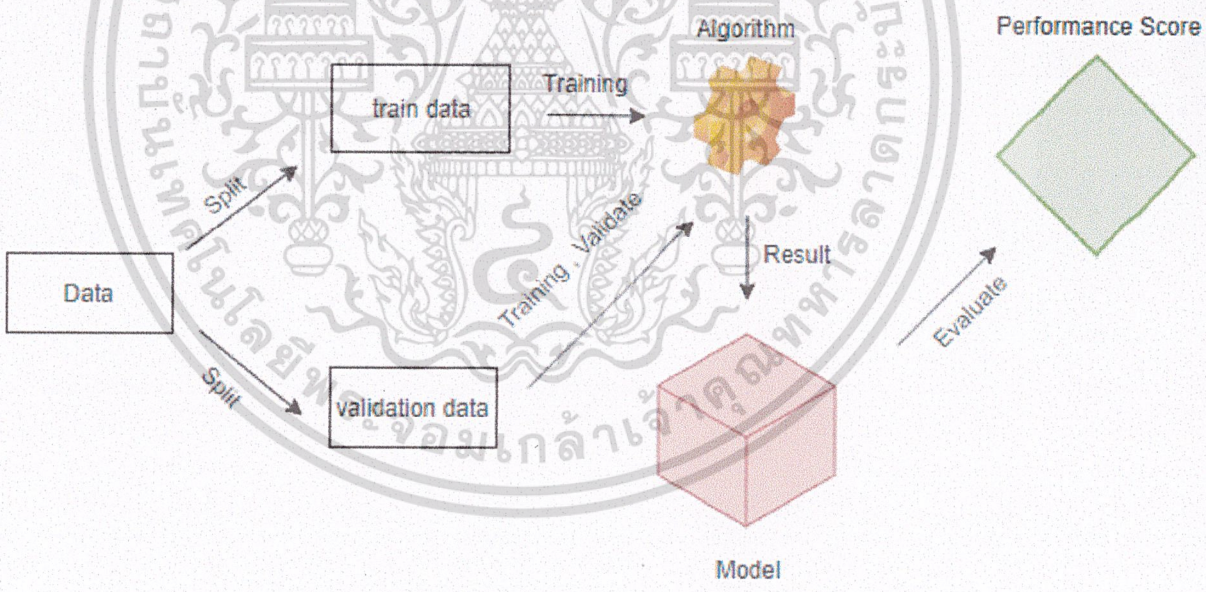
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีดำเนินการวิจัย

โปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้ ผู้วิจัยได้ทำการออกแบบโปรแกรมจากการเลียนแบบการทำงานของระบบโครงข่ายประสาทเทียมโดยใช้ algorithm ของ deep learning ที่ถูกสร้างขึ้นเพื่อทำการแยกชนิดของใบไม้จากรูปภาพให้ถูกต้องและมีประสิทธิภาพมากที่สุด โดยโปรแกรมแยกชนิดพืชจากรูปภาพใบไม้สามารถใช้งานได้กับใบไม้ 5 ชนิดได้แก่ ใบมะม่วง ใบชะพลู ใบขนุน ใบสาระแหน่ และใบมะกรูด ตัวโปรแกรมถูกพัฒนาขึ้นด้วยภาษา python โดยการวิจัยนี้ได้ทำการพัฒนาเฉพาะโปรแกรมเท่านั้น และพัฒนาโปรแกรมโดยใช้ซอฟแวร์ โปรแกรม Anaconda การทำงานของวิจัยนี้ได้พัฒนาโปรแกรมบนเครื่องคอมพิวเตอร์ Notebook โดยโปรแกรมจะรับรูปภาพ จากผู้ใช้งานเพื่อนำรูปภาพไปประมวลผลตามขั้นตอน

3.1 ภาพรวมในการเตรียมการทำงาน

การออกแบบการทำงานสามารถแบ่งการทำงานได้ดังรูปที่ 3.1



รูปที่ 3.1 ภาพรวมในการเตรียมการทำงาน

จากรูปที่ 3.1 สามารถอธิบายได้ดังนี้

1.)Data :

คือ ข้อมูลในการวิจัยครั้งนี้คือรูปภาพ 548 รูป โดยจะนำรูปภาพทั้งหมดมาแบ่งตามจำนวน class ของรูป ก่อนจะถูกแบ่งสัดส่วน 70% ของรูปจาก class นั้นให้ข้อมูลฝึกสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Train data) และ 30% ของรูปจาก class นั้นให้ข้อมูลทดสอบประสิทธิภาพ (Validation data)

2.)Train data :

คือรูปภาพที่ใช้ฝึกสอนในการสร้างโมเดลในการทำงาน โดยรูปภาพในส่วนนี้จะมี 70% ของแต่ละ class รูปภาพจะถูกแบ่งตาม class แล้ว รูปภาพในส่วนนี้จะถูกสอนให้โมเดลเรียนรู้ในสิ่งที่ต้องการให้เรียนรู้

3.)Validation data :

คือรูปภาพที่ใช้ทดสอบประสิทธิภาพจากการฝึกสอนของโมเดลโดยรูปภาพในส่วนนี้จะมี30% ของแต่ละ class รูปภาพจะถูกแบ่งตาม class แล้ว รูปภาพในส่วนนี้จะทดสอบโมเดลหลังจากเรียนรู้ในสิ่งที่ได้ฝึกสอนไป

4.)Algorithm :

คือกระบวนการที่ใช้ในการที่จะสร้างโมเดลในการทำงาน โดยจะใช้กระบวนการ Deep learning เป็นการฝึกสอนและทดสอบประสิทธิภาพในการสร้างโมเดล

5.)Model:

คือแบบจำลองที่ผ่านการฝึกสอนข้อมูลและทดสอบประสิทธิภาพมาเพื่อผ่านกระบวนการทำงานมาเพื่อใช้ทำนายผลลัพธ์ออกมาเป็น class ของรูปภาพ

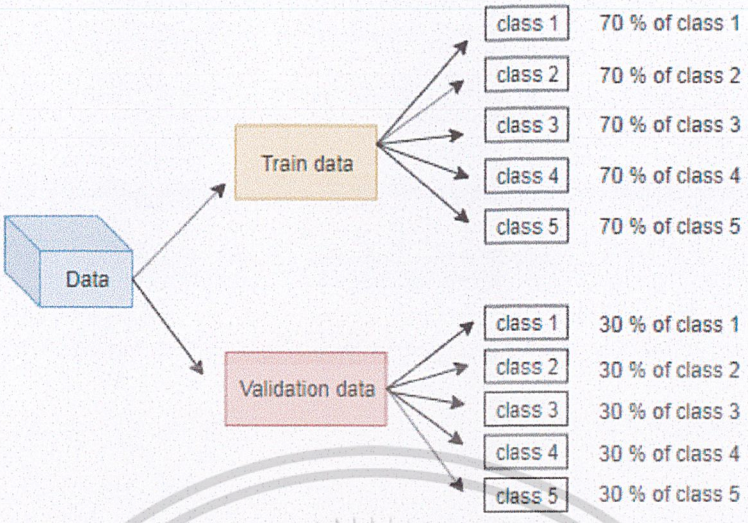
6.)Performance Score :

คือประสิทธิภาพการทำงานของโมเดลที่สร้างขึ้นมาว่าโมเดลที่ถูกสร้างขึ้นมีค่าความแม่นยำเท่าไร ส่วนนี้จะทำให้สามารถสร้างโมเดลจนกว่าจะพอใจในประสิทธิภาพของโมเดล ถ้าค่าความแม่นยำยังไม่น่าพอใจ ก็ยังสามารถไปปรับปรุงโมเดลขึ้นมาใหม่ได้

3.2 การแบ่งข้อมูลในการทำงาน

การแบ่งข้อมูลในการทำงานสามารถแบ่งได้โดยการนำรูปภาพทั้งหมดมาแบ่งตาม class ของรูปภาพ สามารถแบ่งสัดส่วนออกเป็น Train data ใช้รูปภาพ 70% จากจำนวนข้อมูลทั้งหมดของแต่ละ class และ Validation data ใช้รูปภาพ 30% จากจำนวนข้อมูลทั้งหมดของแต่ละ class โดยทุกรูปภาพจะมีการแบ่งข้อมูลตาม class อยู่แล้ว สามารถอธิบายการทำงานได้ดังรูปที่ 3.2 การแบ่งข้อมูลในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 การแบ่งข้อมูลในการทำงาน

จากรูปภาพที่3.2 สามารถอธิบายได้ดังนี้

1.)Data :

คือข้อมูลของเราทั้งหมด ข้อมูลในที่นี้หมายถึงข้อมูลที่เป็นรูปภาพ นำรูปภาพมาแบ่งเป็นแต่ละ class ของรูป โดยมีรูปภาพทั้งหมด 548 รูปภาพ จาก 5 class

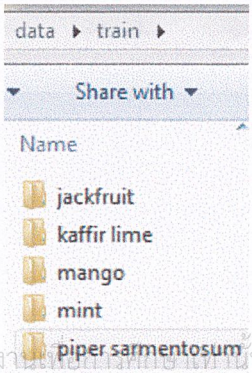
2.)Train Data :

คือข้อมูลรูปภาพที่เราจะนำไปฝึกสอนโมเดลสำหรับการทำนาย class ของรูปภาพ โดยจะนำรูปภาพมาฝึกฝน 70% ของข้อมูลรูปภาพทั้งหมดในแต่ละ class

3.)Validation Data:

คือข้อมูลรูปภาพที่ใช้ทดสอบประสิทธิภาพหลังจากการฝึกสอนของโมเดล โดยจะนำรูปภาพมาทดสอบประสิทธิภาพ 30% ของรูปภาพทั้งหมดในแต่ละ class

ตัวอย่างการแบ่งข้อมูลใน Train data



รูปที่ 3.3 ตัวอย่างการแบ่งข้อมูลใน Train data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปภาพที่3.3 สามารถอธิบายได้ดังนี้

สร้าง Folder ในส่วนของ Train data เพื่อแบ่งรูปภาพตาม class โดยใช้รูปภาพ 70 % ของรูปภาพทั้งหมดใน class นั้น แล้วแบ่งตาม Folder ของ class ตัวเอง

ตัวอย่างการเก็บรูปภาพใน Train data ของ Folder class ใบมะกรูด (kaffir lime)

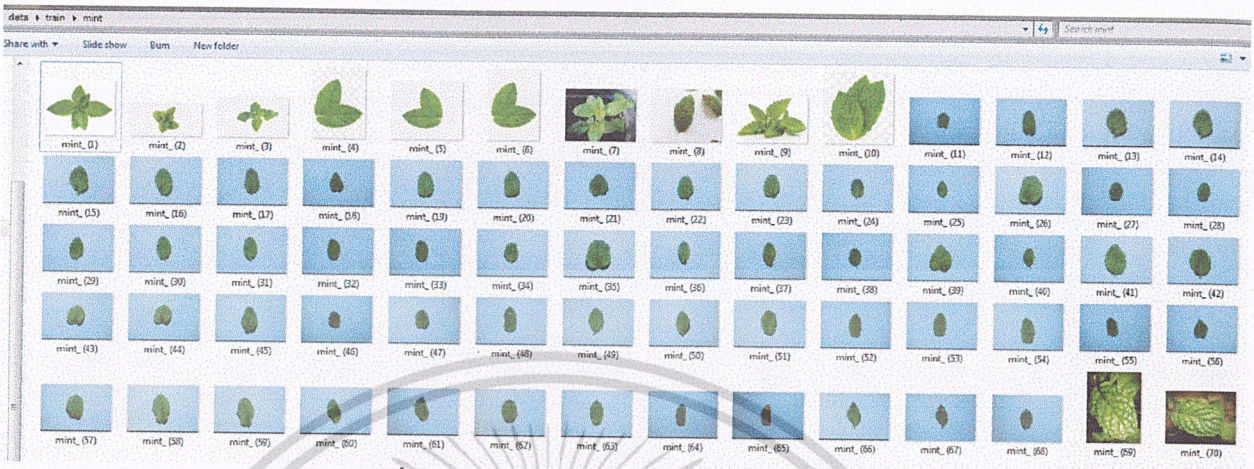
รูปที่ 3.4 Folder train class kaffir lime

ตัวอย่างการเก็บรูปภาพใน Train data ของ Folder class ใบมะม่วง (mango)

รูปที่ 3.5 Folder train class mango

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการเก็บรูปภาพใน Train data ของ Folder class ใบสาระแหน่ (mint)



รูปที่ 3.6 Folder train class mint

ตัวอย่างการเก็บรูปภาพใน Train data ของ Folder class ใบชะพลู (wildbetal Leafbush)

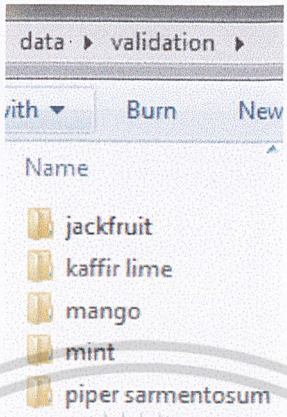
รูปที่ 3.7 Folder train class wildbetal Leafbush

ตัวอย่างการเก็บรูปภาพใน Train data ของ Folder class ใบขนุน (jackfruit)

รูปที่ 3.8 Folder train class jackfruit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการแบ่งข้อมูลใน Validation data

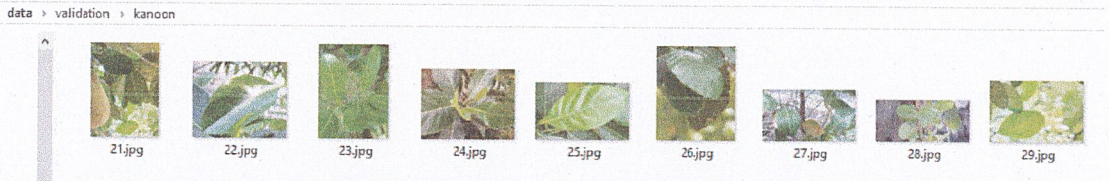


รูปที่ 3.8 ตัวอย่างการแบ่งข้อมูลใน Validation data

ตัวอย่างการเก็บรูปภาพใน Validation data ของ Folder class ใบชะพลู (wildbetal Leafbush)

รูปที่ 3.9 Folder validation class wildbetal Leafbush

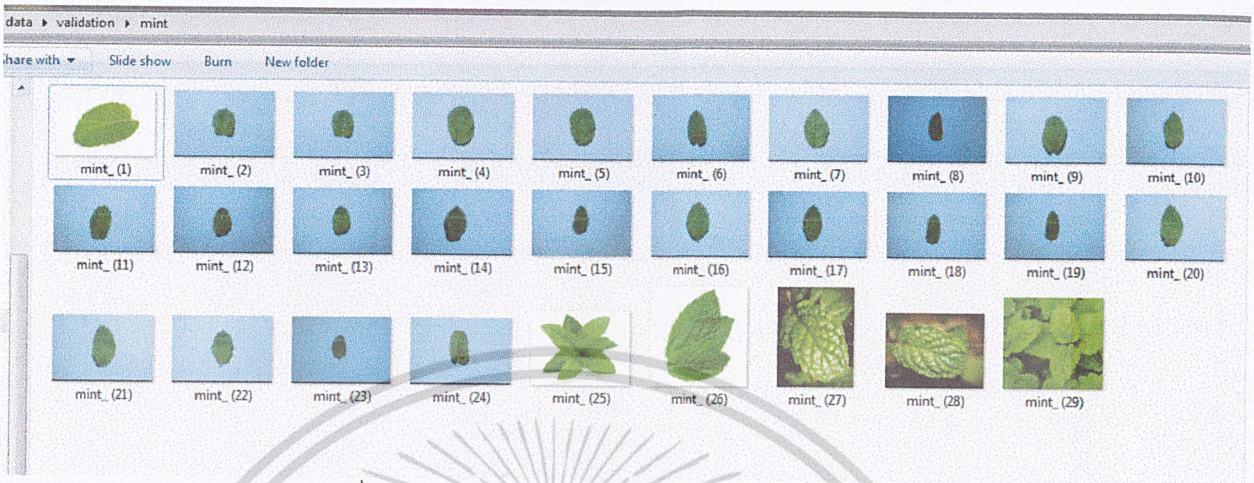
ตัวอย่างการเก็บรูปภาพใน Validation data ของ Folder class ใบขนุน (jackfruit)



รูปที่ 3.10 Folder validation class jackfruit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการเก็บรูปภาพใน Validation data ของ Folder class ใบสาระแหน่ (mint)

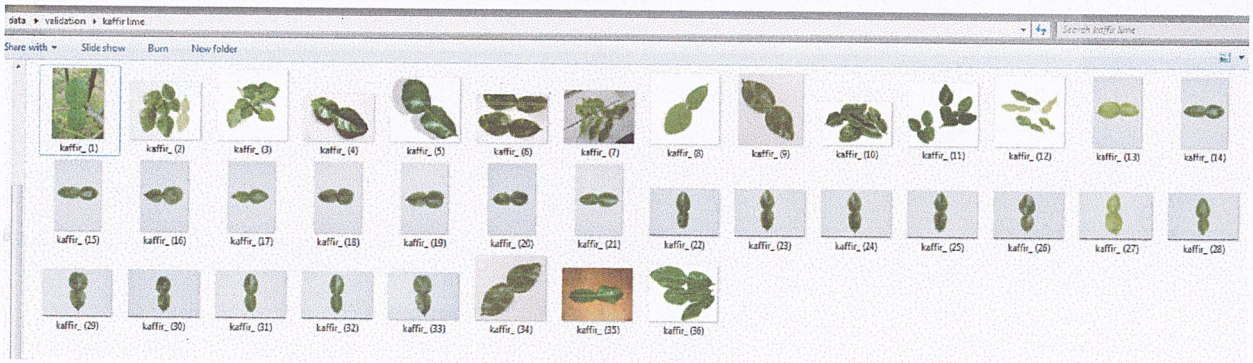


รูปที่ 3.11 Folder validation class mint

ตัวอย่างการเก็บรูปภาพใน Validation data ของ Folder class ใบมะม่วง (mango)

รูปที่ 3.12 Folder validation class mango

ตัวอย่างการเก็บรูปภาพใน Validation data ของ Folder class ใบมะกรูด (kaffir lime)



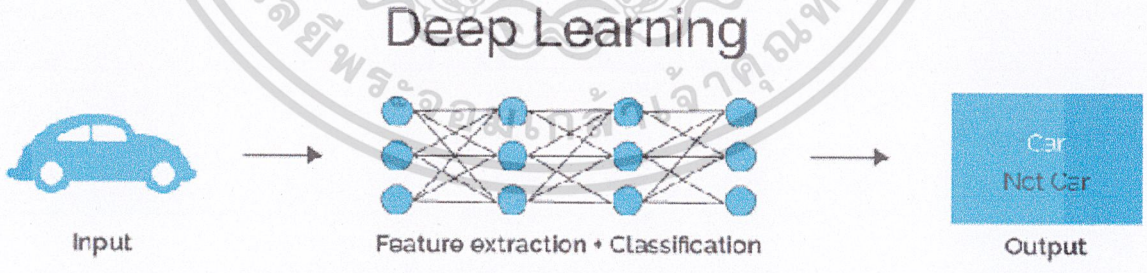
รูปที่ 3.13 Folder validation class kaffir lime

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 Algorithm ที่ใช้งาน

หลังจากแบ่งสัดส่วนของข้อมูลรูปภาพแล้ว ต้องนำข้อมูลรูปภาพไปปรับขนาดของรูปภาพให้มีขนาดความกว้าง ความยาวให้เท่ากันทุกรูปก่อนที่จะนำไปปรับขนาดการเรียนรู้ให้เล็กลงเนื่องจากรูปภาพเป็นภาพสี RGB ค่า pixel เลยมีค่าอยู่ระหว่าง 0 ถึง 255 ทำให้มีเมทริกซ์ของ pixel 3 ตระกูล ได้แก่ สีแดง สีเขียว สีน้ำเงิน เลยต้องทำให้มีขนาดเมทริกซ์ที่น้อยลงด้วยการปรับให้ pixel มีค่าระหว่าง 0 ถึง 1 เป็นภาพขาวดำแทนเพื่อการฝึกสอนโมเดลที่เร็วขึ้น type ของ array 1มิติ ที่สร้างใหม่จะเป็น float โดยค่าที่เป็นไปได้ของความกว้าง และ ความยาว จะอยู่ในช่วงเปิด [-0.2 , +0.2] จะมีการเติม parameters ค่า pixel ใหม่หลังจากที่ใช้ตัวดำเนินการโดยการหมุนรูปภาพ 20 องศา จะเติมค่า pixel ใหม่ด้วยค่า pixel โดยรอบที่ใกล้ที่สุด ในการทดสอบประสิทธิภาพจะมีการปรับขนาดให้เล็กลงให้อยู่ในรูป array 1 มิติ เช่นกัน

ในการสร้าง array feature , label ของการtrain และ validation นั้นจะสร้าง array ที่มีแต่เลข 0 ตามขนาดที่กำหนด รูปร่างของ array (ขนาดแถว=0,หลัก=2048) ต่อมาจะมีการดึงรูปภาพและ class ของรูปภาพโดยอัตโนมัติสำหรับชุดข้อมูล train data และ validation data โดยขนาดของรูปภาพที่เป็นintจะถูกปรับขนาด ต้องกำหนดให้ class mode เป็น categorical เพราะมี 5 class เรียกวิธีนี้ว่าการ normalized data ในส่วนของ algorithm นั้นต้องนำข้อมูลมาผ่านกระบวนการ Deep Learning โดยกระบวนการนี้จะไม่ต้องจัดการในเรื่องของ Feature Extraction เพราะ Deep learning สามารถเรียนรู้จากตัวอย่างที่เกิดขึ้น โดยใช้ข้อมูลที่มีเป็นตัวอย่าง ทำให้สามารถตัดสินใจได้เหมือนกับสมองของมนุษย์ มีความรวดเร็วในการเรียนรู้ Deep learning จะมีการทำงานในส่วนของระบบ layer หรือชั้นของระบบเครือข่ายประสาทเทียม



รูปที่ 3.14 Algorithm Deep Learning
ที่มาจาก (https://bizion.co/content/63986/)

ในการสร้างระบบ layers จะต้องมีการใช้ activation function เพื่อคำนวณค่าของข้อมูลที่เข้ามา โดย activation ที่เลือกใช้ในสองชั้นแรกคือ Relu activation

$$relu(x) = max(o, x)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็น function ที่เมื่อเอาค่าใดๆมาเข้า function นี้ถ้าค่านั้นน้อยกว่า 0 ผลลัพธ์จะออกมาเป็น 0 แต่ถ้ามีค่ามากกว่า 0 ผลลัพธ์ก็จะออกมาเป็นค่า x นั้นๆ เลย

ตัวอย่างการทำงานในส่วนของระบบ layers

ชั้นแรก Dense layer หมายถึงโครงสร้างแบบ fully connected ระหว่างชั้นที่ติดกัน โดยมี activation function ชนิด Relu , input node =2048 , hidden node = 1024

ชั้นสอง Dense layer หมายถึงโครงสร้างแบบ fully connected ระหว่างชั้นที่ติดกัน โดยมี activation function ชนิด Relu โดยมี hidden node = 120

ชั้นสาม Dropout layer หมายถึงการสุ่มปิดบาง node ในเครือข่ายประสาทเพื่อลดปัญหาการกระจัดกระจายที่ลดความเป็นไปได้ของ overfitting

ชั้นสุดท้าย Dense layer หมายถึงโครงสร้างแบบ fully connected ระหว่างชั้นที่ติดกัน โดยมี activation function ชนิด Softmax เพื่อ classify โดยมี output node = 5 ตามจำนวนชนิดของข้อมูล

เนื่องจากเป็นงาน classification ดังนั้นเราจะใช้ loss function คือ Cross Entropy การใช้ loss function ต้องใช้คู่กับ softmax activation เพราะในการคำนวณ entropy นั้นต้องการตีความข้อมูลส่งออกจาก neural network นี้ในเชิงความน่าจะเป็น ซึ่งหนึ่งในข้อแม้ก็คือผลรวมของค่าส่งออกทุกค่าต้องเท่ากับ 1 ซึ่งเงื่อนไขนี้จะเป็นจริงเมื่อใช้ softmax activation เมื่อทำงานในส่วนของระบบ layers แล้วจะมีการ compile เพื่อให้ได้ parameters ต่างๆออกมา

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 1024)	2098176
dense_2 (Dense)	(None, 120)	123000
dropout_1 (Dropout)	(None, 120)	0
dense_3 (Dense)	(None, 5)	605

Total params: 2,221,781
Trainable params: 2,221,781
Non-trainable params: 0

รูปที่ 3.15 ตัวอย่างการทำงานในส่วนของระบบ layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การสร้างโมเดล

จากการทำงานในส่วนของระบบlayer จะมีการ train model ทั้งหมด 100 รอบ โดยให้ขนาดของ mini-batch เป็น 10 ตัวอย่าง (คือเราจะสะสมค่าเกรเดียนต์บนตัวอย่าง 10 ตัวก่อนจะทำการปรับ 1 ครั้ง)

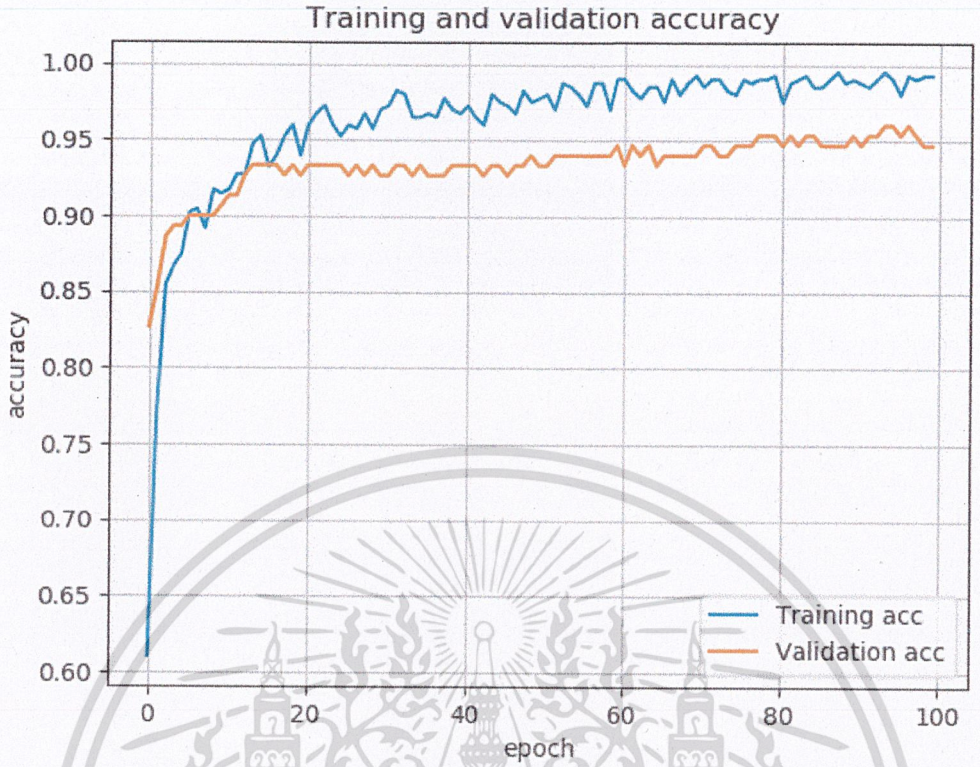
```
Epoch 92/100
398/398 [==============================] - 1s 3ms/step - loss: 0.1005 - acc: 0.9849 - val_loss: 0.1841 - val_acc: 0.9533
Epoch 93/100
398/398 [==============================] - 1s 3ms/step - loss: 0.0798 - acc: 0.9899 - val_loss: 0.1839 - val_acc: 0.9533
Epoch 94/100
398/398 [==============================] - 1s 3ms/step - loss: 0.0868 - acc: 0.9950 - val_loss: 0.1832 - val_acc: 0.9600
Epoch 95/100
398/398 [==============================] - 1s 3ms/step - loss: 0.0975 - acc: 0.9899 - val_loss: 0.1839 - val_acc: 0.9600
Epoch 96/100
398/398 [==============================] - 1s 3ms/step - loss: 0.1015 - acc: 0.9799 - val_loss: 0.1817 - val_acc: 0.9533
Epoch 97/100
398/398 [==============================] - 1s 3ms/step - loss: 0.0760 - acc: 0.9925 - val_loss: 0.1814 - val_acc: 0.9600
Epoch 98/100
398/398 [==============================] - 1s 3ms/step - loss: 0.1028 - acc: 0.9899 - val_loss: 0.1804 - val_acc: 0.9533
Epoch 99/100
398/398 [==============================] - 1s 3ms/step - loss: 0.0852 - acc: 0.9925 - val_loss: 0.1796 - val_acc: 0.9467
Epoch 100/100
398/398 [==============================] - 1s 3ms/step - loss: 0.0771 - acc: 0.9925 - val_loss: 0.1787 - val_acc: 0.9467
```

รูปที่ 3.16 Train model

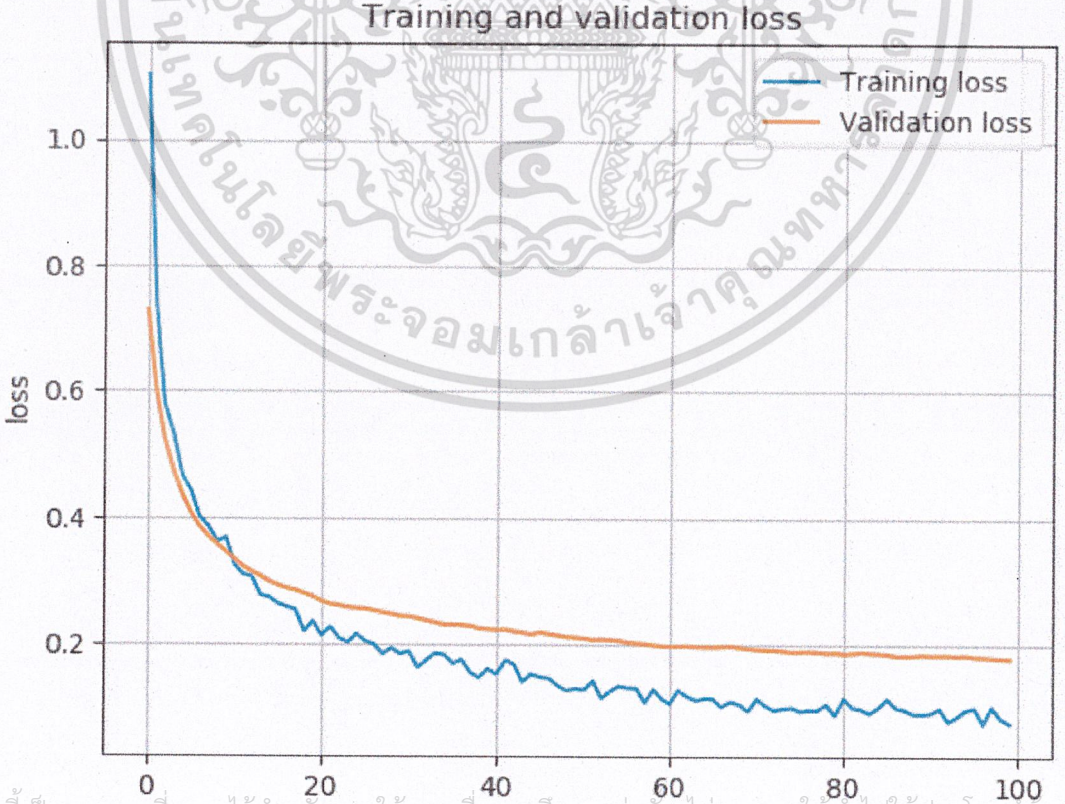
3.5 การเช็ค performance

จากการ train model 100 รอบ จะมีการปรับค่าweightของโมเดลทุก 10 ข้อมูล โดยค่า accuracy ในรอบที่100 จะมีค่า accuracy ของการtrain อยู่ที่ 0.9925 , ค่า accuracy ของการ validate อยู่ที่ 0.9467 และค่า loss ของการ train = 0.0771 , ค่า loss ของการ validate = 0.1787

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.17 Accuracy



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.18 loss

จะเห็นได้ว่ายิ่งจำนวนรอบที่ model ถูก training และ validation เพิ่มขึ้น ค่า accuracy ก็จะเพิ่มขึ้นตามด้วย ส่วนค่า loss ค่าจะลดลงจากเดิม แต่ไม่ได้ลดเยอะ ยังอยู่ในระดับค่าที่ใกล้เคียงกันอยู่ไม่ห่างเกินไป ในส่วนของการ validate รูปภาพ 150 รูป จาก 5 class หลังจากการ train model พบว่าเจอข้อผิดพลาด 8 รูป จาก 150 รูป คิดเป็น 5.33 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

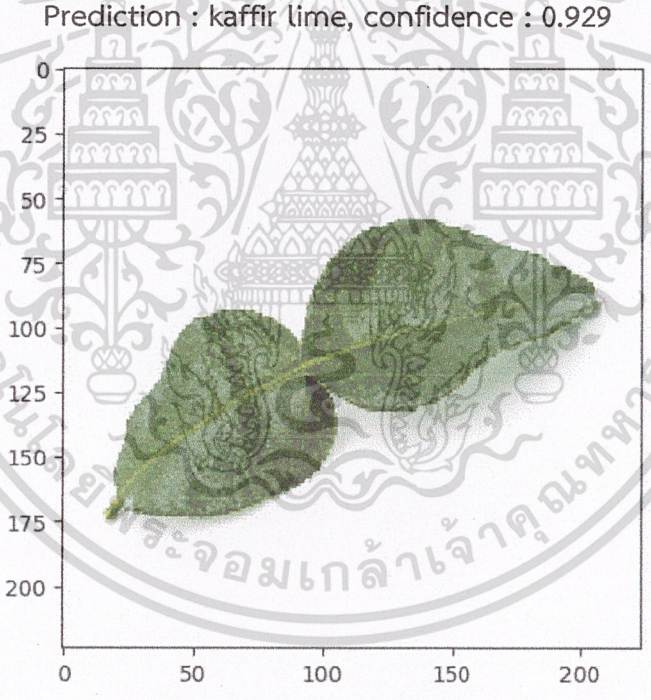
บทที่4
ผลการดำเนินการและอธิปรายผล

ผลการดำเนินการจะกล่าวถึงผลลัพธ์ที่ได้จากการดำเนินงานวิจัยในครั้งนี้ในแต่ละส่วนและแสดงตัวอย่างโค้ดในการพัฒนา

4.1 การทดลองประสิทธิภาพในใช้งาน

ในการทดลองประสิทธิภาพ ได้ทำการนำรูปภาพที่ไม่เคยผ่าน train data และ validation data มาเพื่อทดสอบประสิทธิภาพ โดยการนำรูปภาพเข้าสู่โมเดลเพื่อให้โมเดลทำนายผลลัพธ์ออกมาว่าเป็นใบไม้ชนิดอะไร มีค่าความมั่นใจที่เท่าไร โดยได้ทำการทดสอบทั้งหมด 10ครั้ง

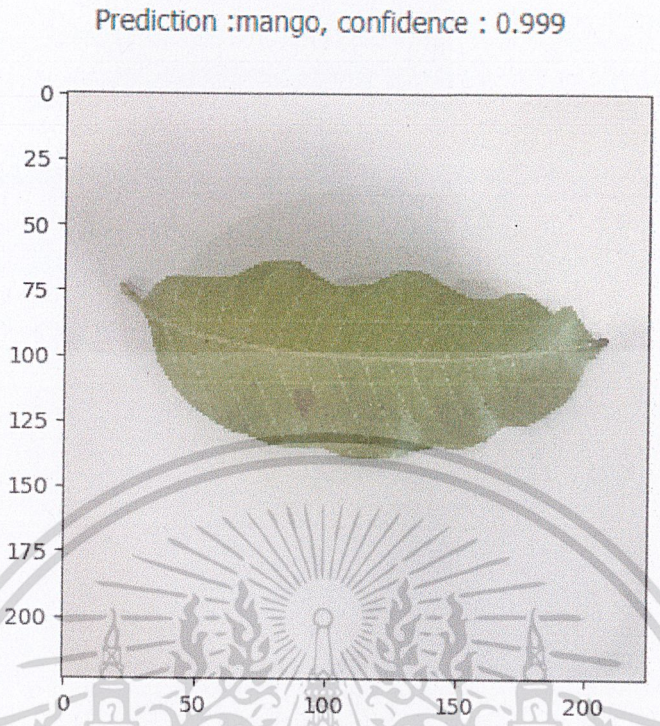
การทดสอบครั้งที่1 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบมะกรูด ว่าเป็นใบมะกรูด ด้วยค่าความมั่นใจที่ 0.929 ผลลัพธ์ตรงกับคำตอบ



รูปที่ 4.1 ผลลัพธ์หลังใช้โมเดลทำนาย

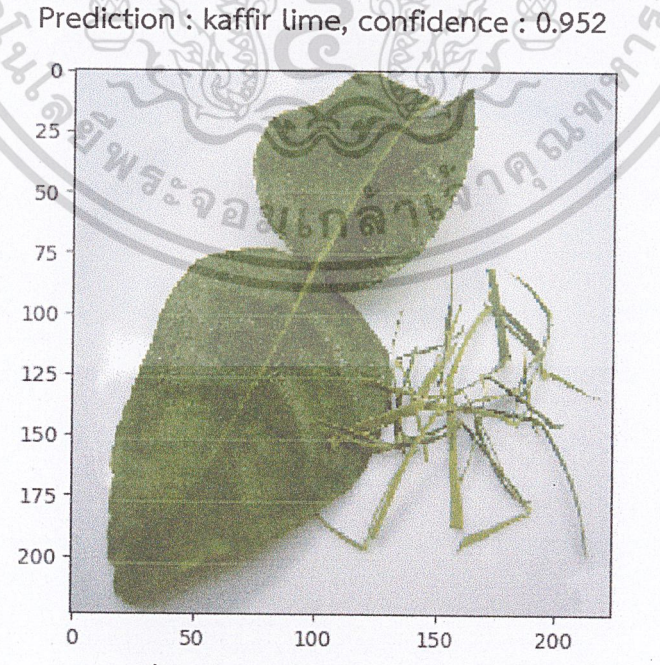
การทดสอบครั้งที่2 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบมะม่วง ว่าเป็นใบมะม่วง ด้วยค่าความมั่นใจที่ 0.999 ผลลัพธ์ตรงกับคำตอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 ผลลัพท์หลังใช้โมเดลทำนาย

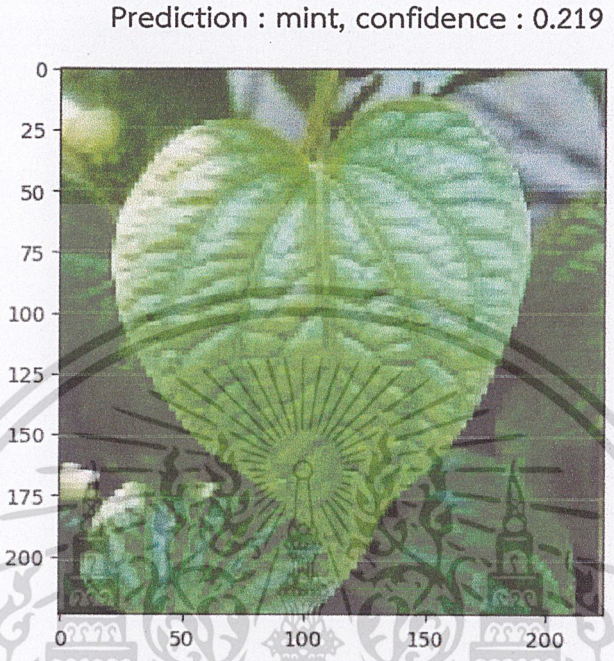
การทดสอบครั้งที่3 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบมะกรูด ว่าเป็นใบมะกรูด ด้วยค่าความมั่นใจที่ 0.952 ผลลัพธ์ตรงกับคำตอบ



รูปที่ 4.3 ผลลัพท์หลังใช้โมเดลทำนาย

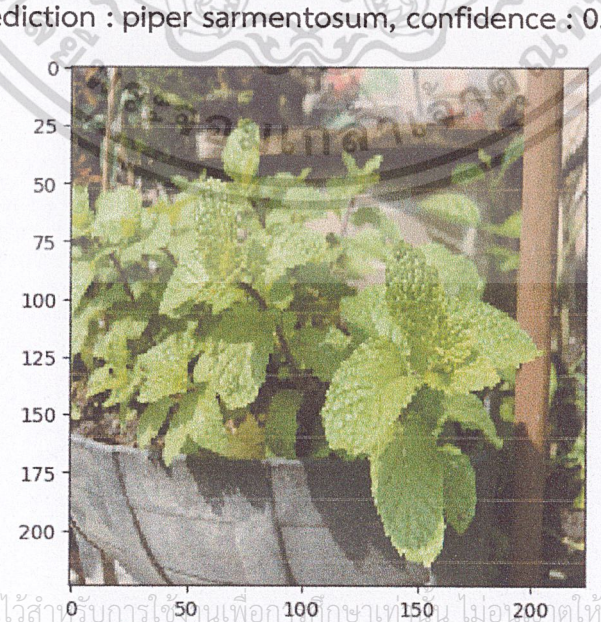
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบครั้งที่4 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบชะพลู ว่าเป็นใบสาระแหน่ ด้วยค่าความมั่นใจที่ 0.219 ผลลัพธ์ไม่ตรงกับคำตอบ



รูปที่ 4.4 ผลลัพธ์หลังใช้โมเดลทำนาย

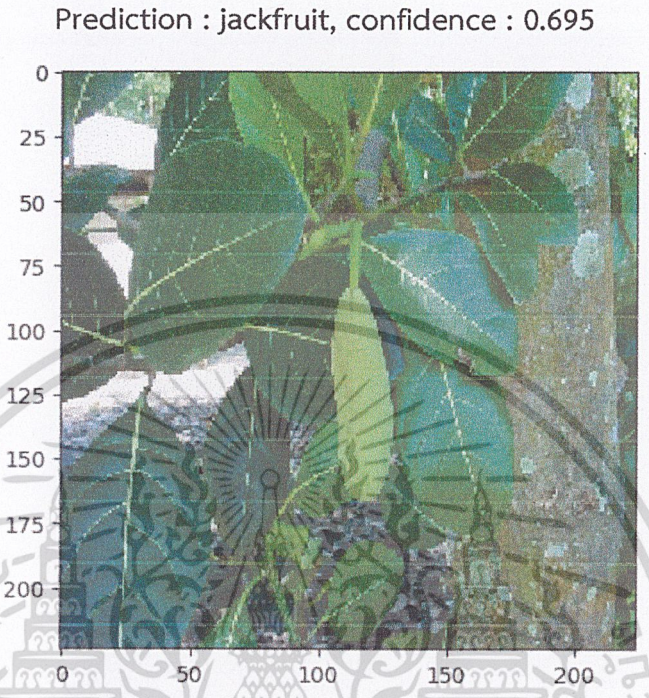
การทดสอบครั้งที่5 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบสาระแหน่ ว่าเป็นใบชะพลู ด้วยค่าความมั่นใจที่ 0.530 ผลลัพธ์ไม่ตรงกับคำตอบ



รูปที่ 4.5 ผลลัพธ์หลังใช้โมเดลทำนาย

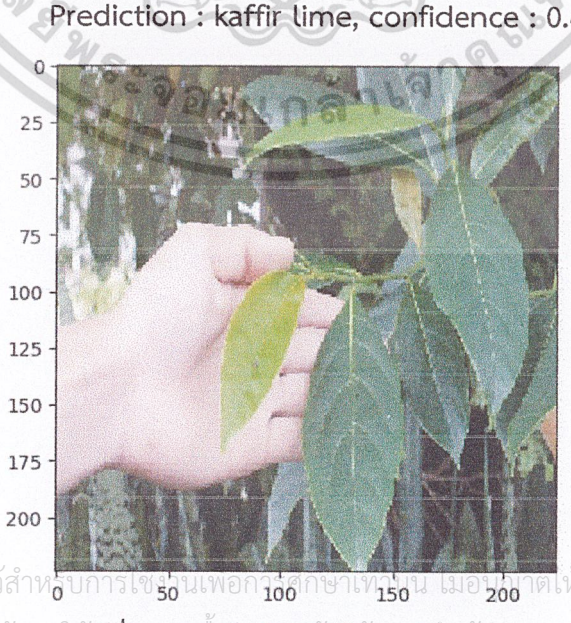
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบครั้งที่6 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบขนุน ว่าเป็นใบขนุน ด้วยค่าความมั่นใจที่ 0.695 ผลลัพธ์ตรงกับคำตอบ



รูปที่ 4.6 ผลลัพธ์หลังใช้โมเดลทำนาย

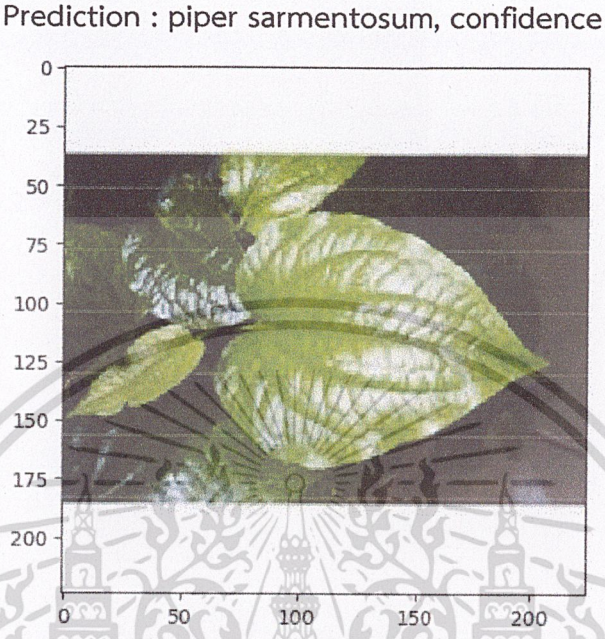
การทดสอบครั้งที่7 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบขนุน ว่าเป็นมะกรูด ด้วยค่าความมั่นใจที่ 0.400 ผลลัพธ์ไม่ตรงกับคำตอบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

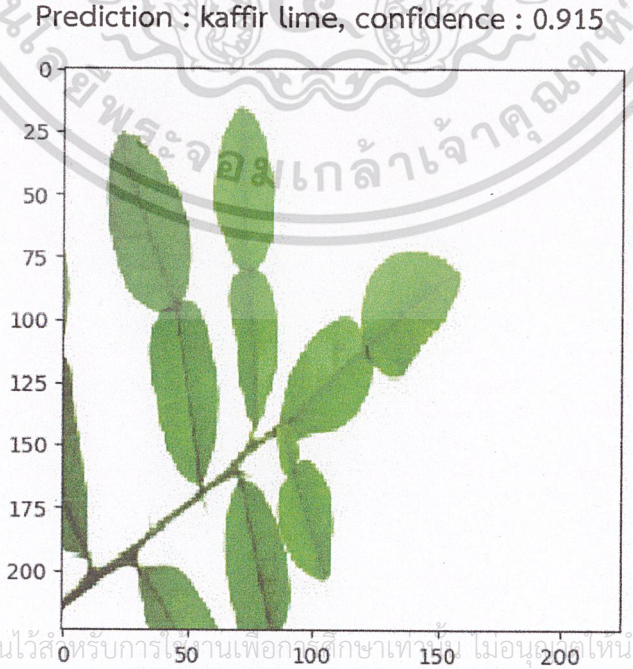
รูปที่ 4.7 ผลลัพธ์หลังใช้โมเดลทำนาย

การทดสอบครั้งที่8 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบชะพลู ว่าเป็นใบชะพลู ด้วยค่าความมั่นใจที่ 0.587 ผลลัพธ์ตรงกับคำตอบ



รูปที่ 4.8 ผลลัพท์หลังใช้โมเดลทำนาย

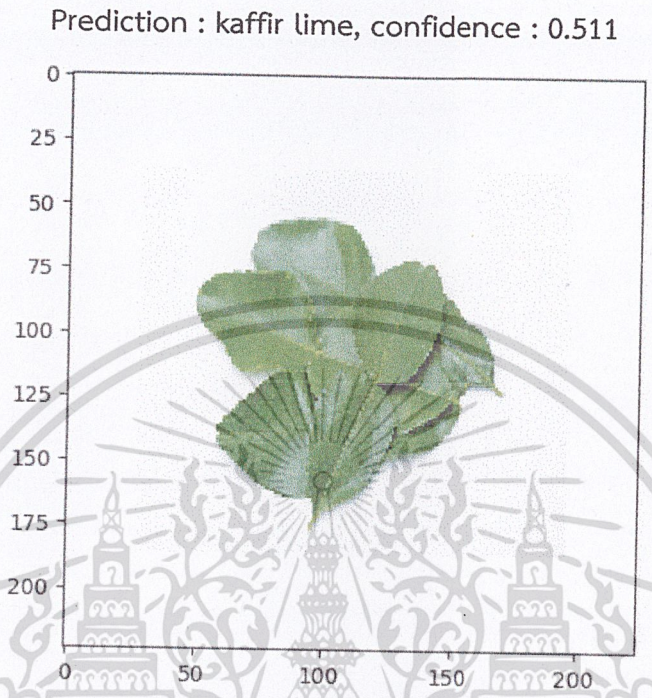
การทดสอบครั้งที่9 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบมะกรูด ว่าเป็นใบมะกรูด ด้วยค่าความมั่นใจที่ 0.915 ผลลัพธ์ตรงกับคำตอบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.9 ผลลัพท์หลังใช้โมเดลทำนาย

การทดสอบครั้งที่10 รูปที่นำเข้าโปรแกรมเพื่อให้โมเดลทำนายว่าเป็นใบไม้ชนิดอะไร ผลที่ได้คือทำนายใบมะกรูด ว่าเป็นใบมะกรูด ด้วยค่าความมั่นใจที่ 0.511 ผลลัพธ์ตรงกับคำตอบ



รูปที่ 4.10 ผลลัพท์หลังใช้โมเดลทำนาย

4.2 สรุปการทดสอบประสิทธิภาพ

จากการทดสอบประสิทธิภาพของโมเดลโดยการนำรูปภาพที่ไม่เคยผ่านการ train , validate มาทดสอบ จากการทดสอบ 10 ครั้ง ทำนายผลลัพธ์ถูกต้องตามclassของชนิดใบไม้ทั้งหมด 7 ครั้ง และ ทำนายผลลัพธ์ไม่ตรงตาม class ของชนิดใบไม้ทั้งหมด 3 ครั้ง จากการทดสอบคิดเป็นเปอร์เซ็นต์การทำนายผลลัพธ์ถูกต้องตาม class ได้ 70% จาก 100% จากการ Train data 398 รูป และ Validation data 150 รูป จากทั้งหมด 548 รูปภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่5
สรุปผลและข้อเสนอแนะ

5.1สรุปผลการดำเนินงาน

โปรแกรมแยกชนิดพืชจากภาพถ่ายใบไม้นั้นเป็นโปรแกรมเชิงวิจัย เป็นโปรแกรมที่ใช้ภาษา python ในการพัฒนาที่อยู่บน anaconda โดยใช้เครื่องมือ Jupyter notebook เป็นเครื่องมือเพื่อนำไปใช้กับalgorithmด้าน Deep learning ในการสร้างโมเดลเพื่อทำนายผลลัพธ์ของรูปภาพใบไม้ 5 class ได้แก่ ใบมะม่วง ใบชะพลู ใบขนุน ใบสาระแหน และ ใบมะกรูด มีข้อมูลรูปภาพทั้งหมด 548 รูป แบ่งเป็น train data 398 รูป และ validation data 150 รูปนำไปผ่าน layer การเรียนรู้ของ deep learning ทั้งหมด 4 layers เพื่อนำไปสร้างโมเดลในการทำนายclassของรูปภาพใบไม้ โดยจากการทดสอบได้นำรูปภาพที่ไม่เคยผ่านการ train , validate มาทดสอบประสิทธิภาพการใช้งาน 10 ครั้ง พบว่าทำนายผลลัพธ์ถูกต้องตรงตาม class ของข้อมูลรูปภาพ 7 ครั้ง และ ทำนายผลลัพธ์ไม่ถูกต้องตาม class ของข้อมูลรูปภาพ 3 ครั้ง จากผลการทดสอบพบว่าประสิทธิภาพการทำงานของโมเดลยังไม่แม่นยำพอเนื่องจาก รูปภาพของการtrain model , validate ยังไม่หลากหลายพอ จึงทำให้เมื่อเจอinput ใหม่ ที่ไม่เคยผ่านการฝึกสอนมา อาจทำให้ผลลัพธ์การทำนาย class ไม่ถูกต้องตาม class คำตอบ ซึ่งยังต้องนำไปปรับปรุงพัฒนาให้มีประสิทธิภาพมากยิ่งขึ้น ให้สามารถใช้กับรูปภาพที่หลากหลายและรองรับชนิดของใบไม้มากกว่านี้

5.2ข้อจำกัดของโปรแกรม

1.)สามารถใช้ได้กับข้อมูลที่เคยผ่านการฝึกสอน ได้แก่ ใบมะม่วง ใบชะพลู ใบสาระแหน ใบมะกรูด ใบขนุน เท่านั้น

2.)สามารถใช้ได้กับซอฟแวร์ที่รองรับภาษา python เท่านั้น

3.)เครื่องคอมพิวเตอร์ต้องติดตั้ง libray ให้ครบตามที่โปรแกรมต้องการก่อนถึงจะใช้งานได้

4.)Jupyter notebook ไม่สามารถใช้งานแบบออฟไลน์ได้

5.3)ข้อเสนอแนะ

1.)พัฒนาเป็นแอพลิเคชันให้สามารถรองรับความต้องการใช้งานมากขึ้น

2.)พัฒนาโปรแกรมให้สามารถใช้งานได้กับรูปภาพหลายชนิดมากขึ้น

3.)ทำส่วนติดต่อผู้ใช้งานให้สะดวก สวยงาม และน่าใช้งานมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] Saichon Jaiyen. 2561. **การติดตั้ง Anaconda บน Windows**. สืบค้นเมื่อวันที่ 10 สิงหาคม,2561 [Online]. เข้าถึงได้จาก https://medium.com/@saichonjaiyen/การติดตั้ง-anaconda-บน-windows-4dbf02068792

[2] A0S0FT. 2561. **Python คืออะไร**. สืบค้นเมื่อวันที่. 10 สิงหาคม,2561 [Online]. เข้าถึงได้จาก https://www.aosoft.co.th/article/322/Python-คืออะไร-ภาษา-python-ใช้ทำอะไร.html

[3] อวยชัย ภิรมย์รื่น. 2561. **Jupyter notebook**. สืบค้นเมื่อวันที่ 10 สิงหาคม,2561 [Online]. เข้าถึงได้จาก https://www.auoychai.com/programmer/python/มาเรียนรูปการใช้งาน-jupyter-notebook/

[4] ธเนศ วงศ์หงษ์. 2561. **Keras คืออะไร**. สืบค้นเมื่อวันที่ 22 กันยายน,2561 [Online]. เข้าถึงได้จาก http://drtanet.blogspot.com/2018/09/keras.html

[5] mindphp. 2561. **Tensorflow คืออะไร**. สืบค้นเมื่อวันที่ 14 เมษายน,2561 [Online]. เข้าถึงได้จาก https://mindphp.com/คู่มือ/73-คืออะไร/5964-tensorflow.html

[6] nessessence. 2561. Deep learning คืออะไร. สืบค้นเมื่อวันที่ 2 มกราคม,2562 [Online]. เข้าถึงได้จาก https://www.thaiprogrammer.org/2018/12/deep-learning-คืออะไร/

[7] วิกิพีเดีย สารานุกรมเสรี. 2560. **โครงข่ายประสาทเทียม**. สืบค้นเมื่อวันที่ 2 มกราคม,2562 [Online]. เข้าถึงได้จาก https://th.wikipedia.org/wiki/โครงข่ายประสาทเทียม

[8] Natthawat Phongchit. 2560. **Convolutional Neural Network คืออะไร**. สืบค้นเมื่อวันที่ 10 มิถุนายน, 2562 [Online]. เข้าถึงได้จาก https://blog.datawow.io/มาลองดูวิธีการคิดของ-cnn-กัน-e3f5d73eebaa

[9] Kissada Pornratthanapong. 2560 **แยกประเภทรูปภาพด้วย Deep Learing**. สืบค้นเมื่อวันที่ 10 ตุลาคม,2561 [Online]. เข้าถึงได้จาก https://life.wongnai.com/internship-image-classification-wongnai-a1dbc2890766

[10] Jon Shlens. 2558. **Train your own image classifier with Inception in Tensorflow**. สืบค้นเมื่อวันที่ 3 มีนาคม,2561 [Online]. เข้าถึงได้จาก https://ai.googleblog.com/2016/03/train-your-own-image-classifier-with.html

[11] Keras Documentation. 2560. **Keras:The Python Deep Learning library**. สืบค้นเมื่อวันที่ 10 มกราคม,2562 [Online]. เข้าถึงได้จาก https://keras.io

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[12] Vijayabhaskar J. 2561. **Tutorial on using Keras flow_from_directory and generators**. สืบค้นเมื่อวันที่ 1 มิถุนายน ,2562 [Online]. เข้าถึงได้จาก https://medium.com/@vijayabhaskar96/tutorial-image-classification-with-keras-flow-from-directory-and-generators-95f75ebe5720

[13] Vikas Gupta. 2561. **Keras Tutorial: Transfer Learning using pre-trained models**. สืบค้นเมื่อวันที่ 1 มิถุนายน,2562 [Online]. เข้าถึงได้จาก https://www.learnopencv.com/keras-tutorial-transfer-learning-using-pre-trained-models/

[14] Sanparith Marukatat. 2560. **สนุกกับ Neural Network**. สืบค้นเมื่อวันที่ 10 มิถุนายน ,2562 [Online]. เข้าถึงได้จาก https://medium.com/@sanparithmarukatat/สนุกกับ-neural-network-657fa293c4d1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
ขั้นตอนการติดตั้งโปรแกรม

ก.1 การติดตั้งโปรแกรมสำหรับการพัฒนา

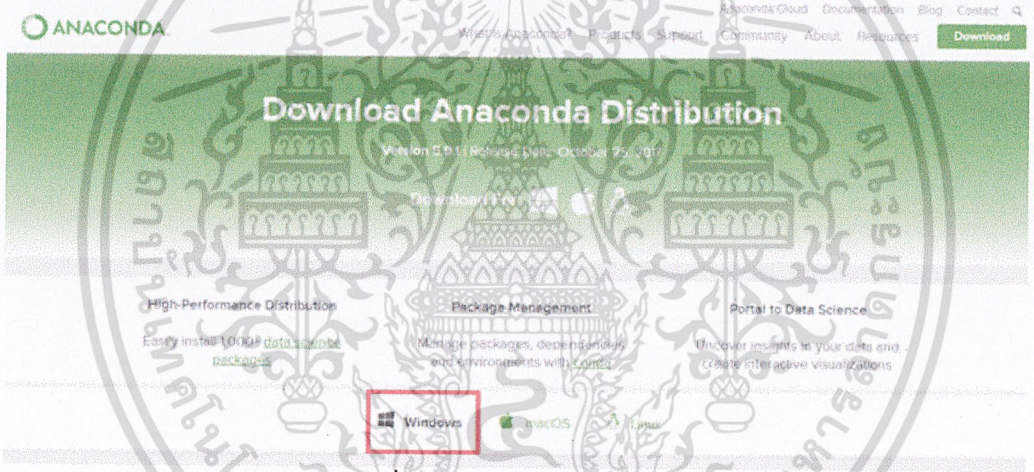
ก.1.1 การติดตั้งโปรแกรม Anaconda

Anaconda

เป็น Python distribution ที่รวมแพกเกจมากมายสำหรับทำ data preparation, data analysis, data visualization, machine learning และ interactive data science ซึ่งสามารถดาวน์โหลดได้จาก https://www.anaconda.com/download/ ซึ่งมีวิธีการติดตั้งดังนี้

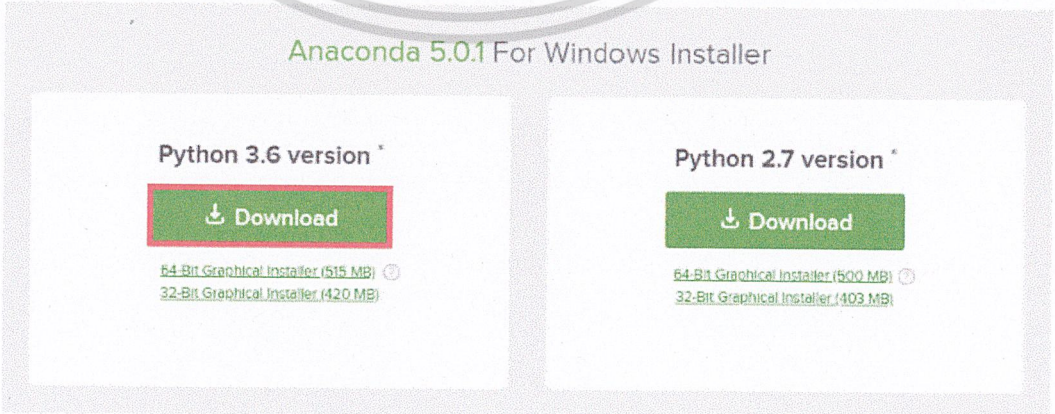
1.)เข้าไปที่เว็บไซต์ https://www.anaconda.com/download/

2.)เลือกดาวน์โหลดเวอร์ชันสำหรับระบบปฏิบัติการ Windows แสดงดังรูปที่ ก.1



รูปที่ ก.1 เข้าweb anaconda

3.)คลิกที่ปุ่ม Download (Python 3.6 version)

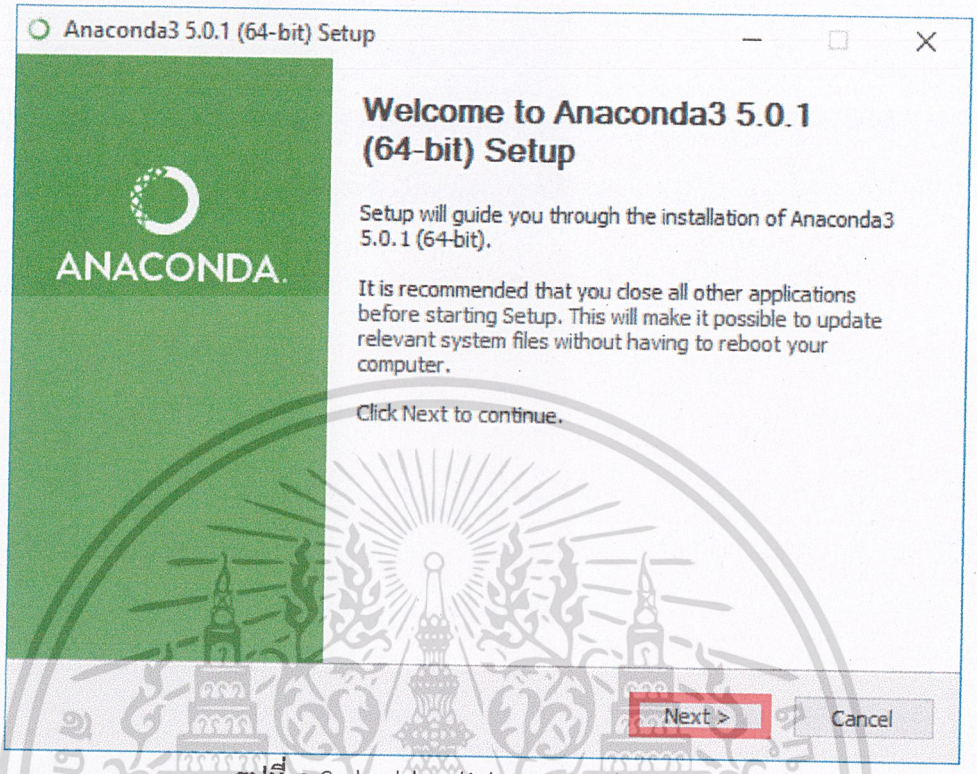


รูปที่ ก.2 download 3.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

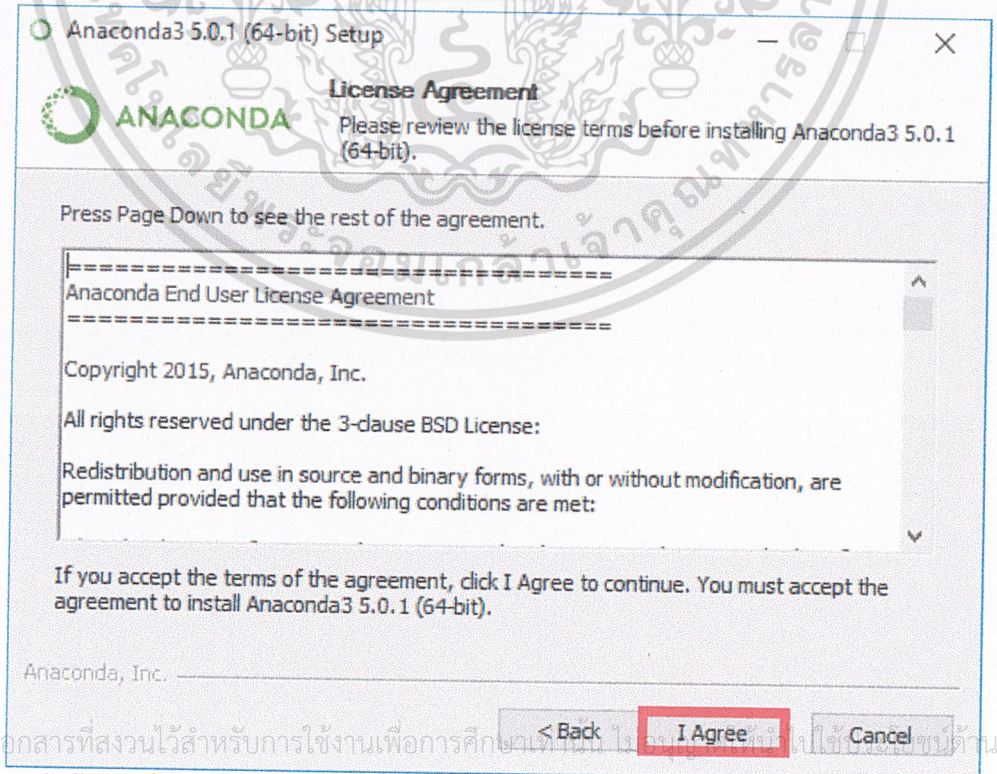
4.)ดับเบิลคลิกไฟล์ที่ดาวน์โหลดมาเพื่อติดตั้ง Anaconda

5.)คลิกที่ปุ่ม Next



รูปที่ ก.3 double click anaconda

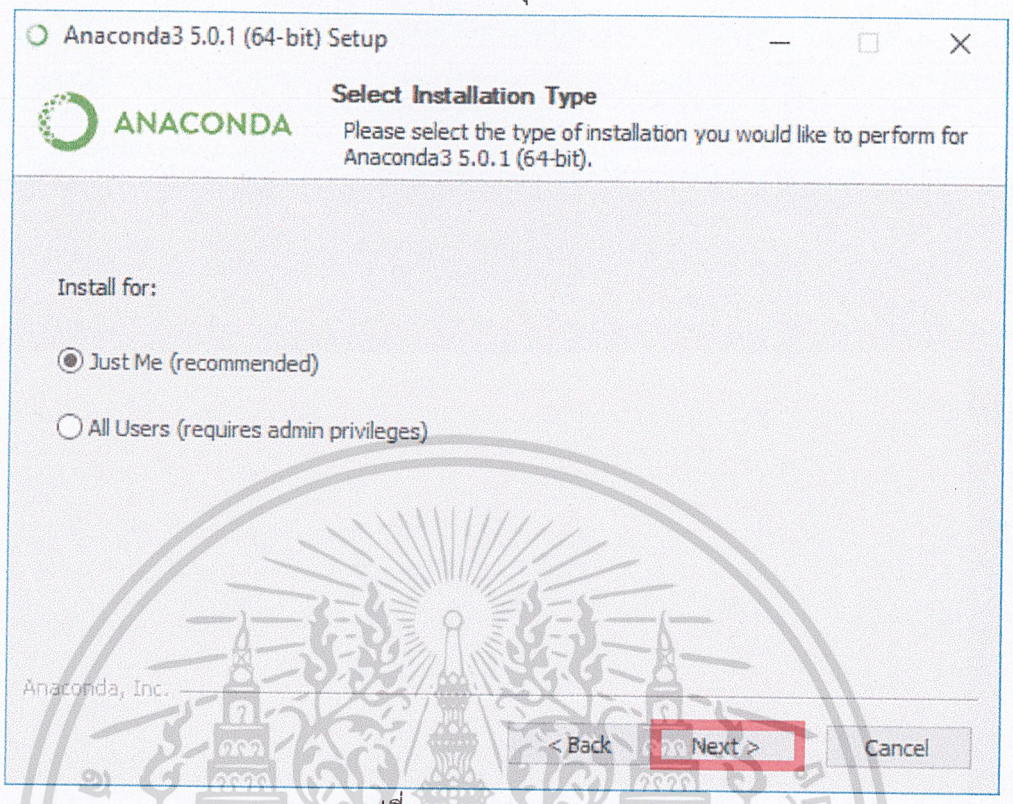
6.)คลิกที่ปุ่ม I Agree



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

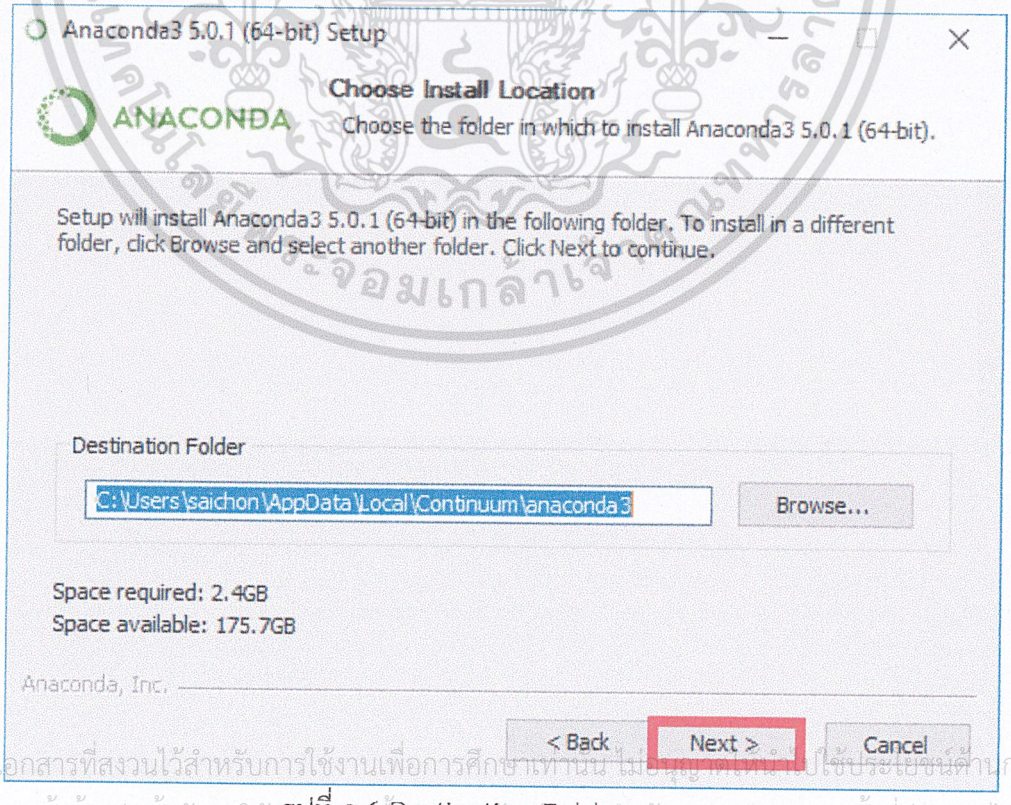
รูปที่ ก.4 I agree

7.)เลือก Just Me (recommended)แล้วคลิกปุ่ม Next



รูปที่ ก.5 just me

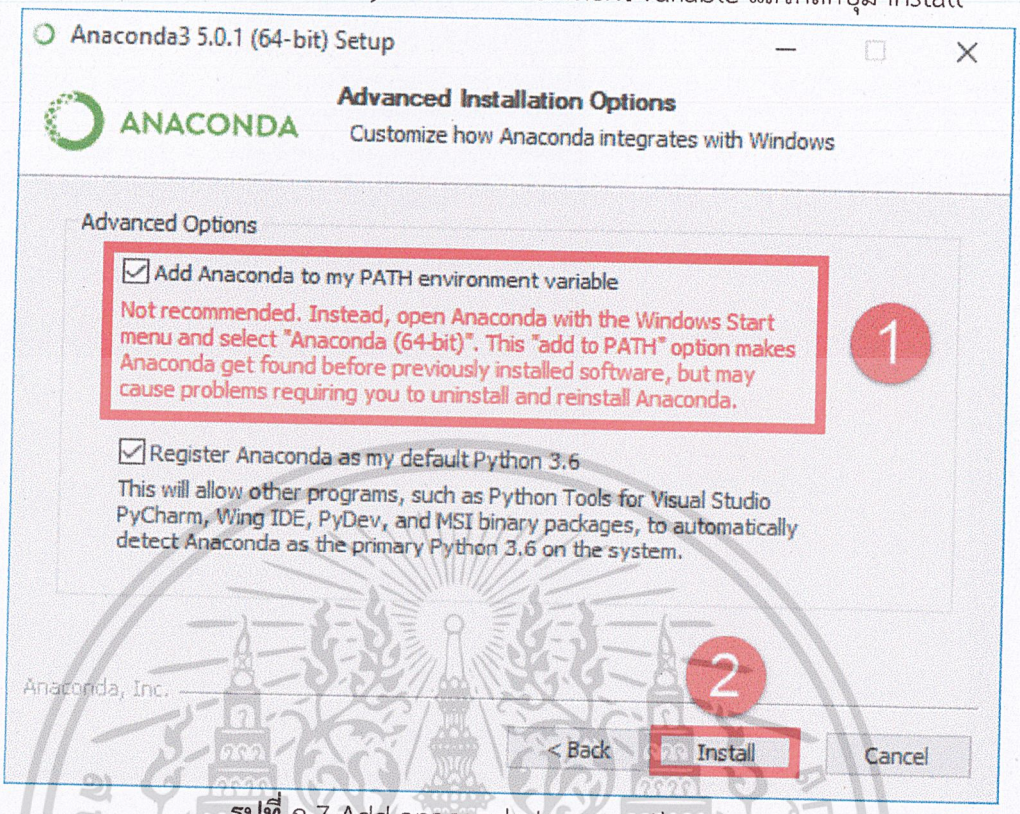
8.)คลิกที่ปุ่ม Next



รูปที่ ก.6 Destination Folder

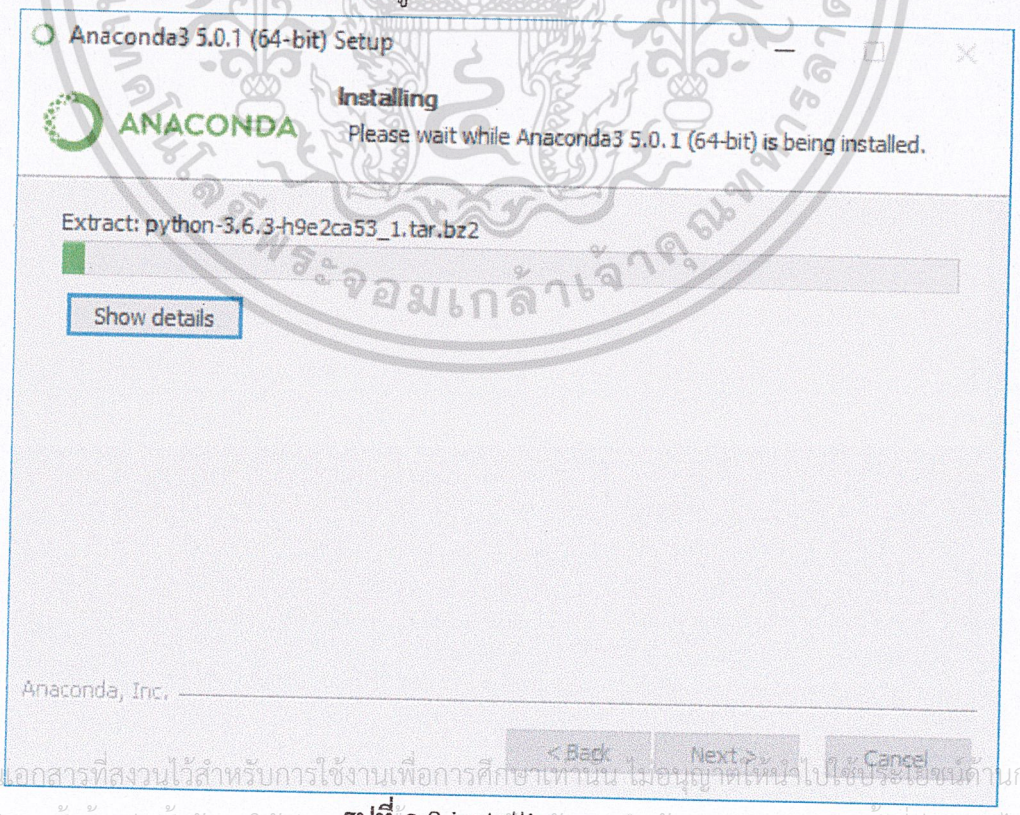
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.)เลือก Add Anaconda to my PATH environment variable แล้วคลิกปุ่ม Install



รูปที่ ก.7 Add anaconda to my path

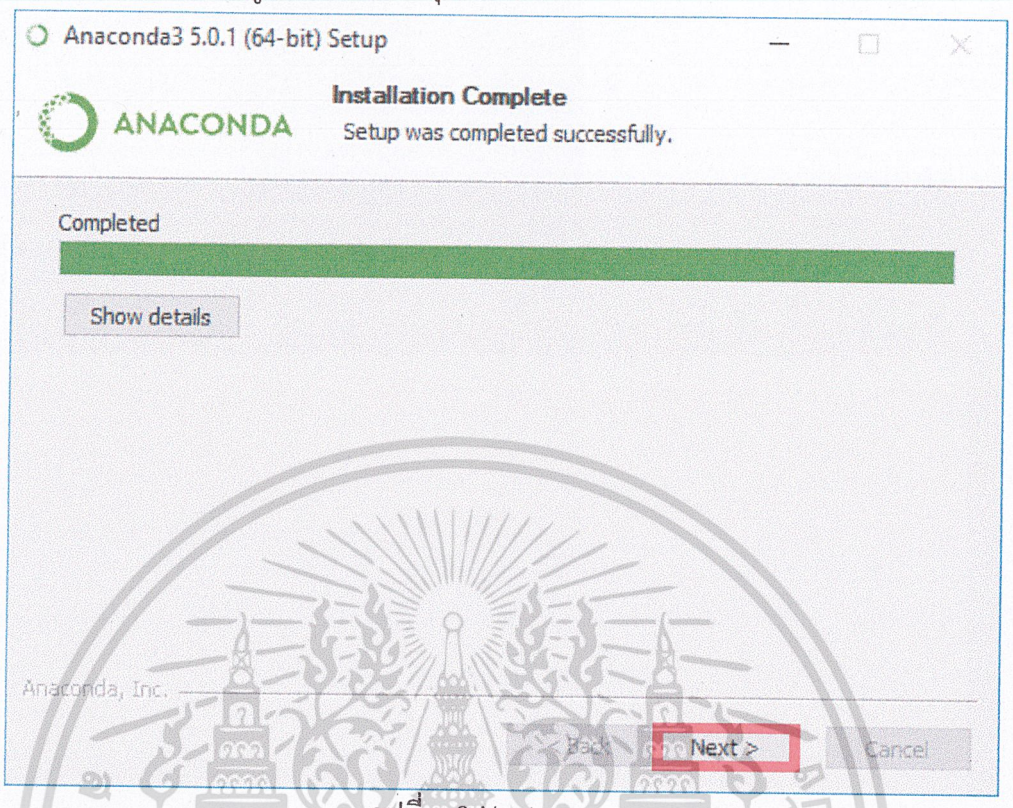
10.)รอจนกว่าการติดตั้งจะเสร็จสมบูรณ์



รูปที่ ก.8 installing

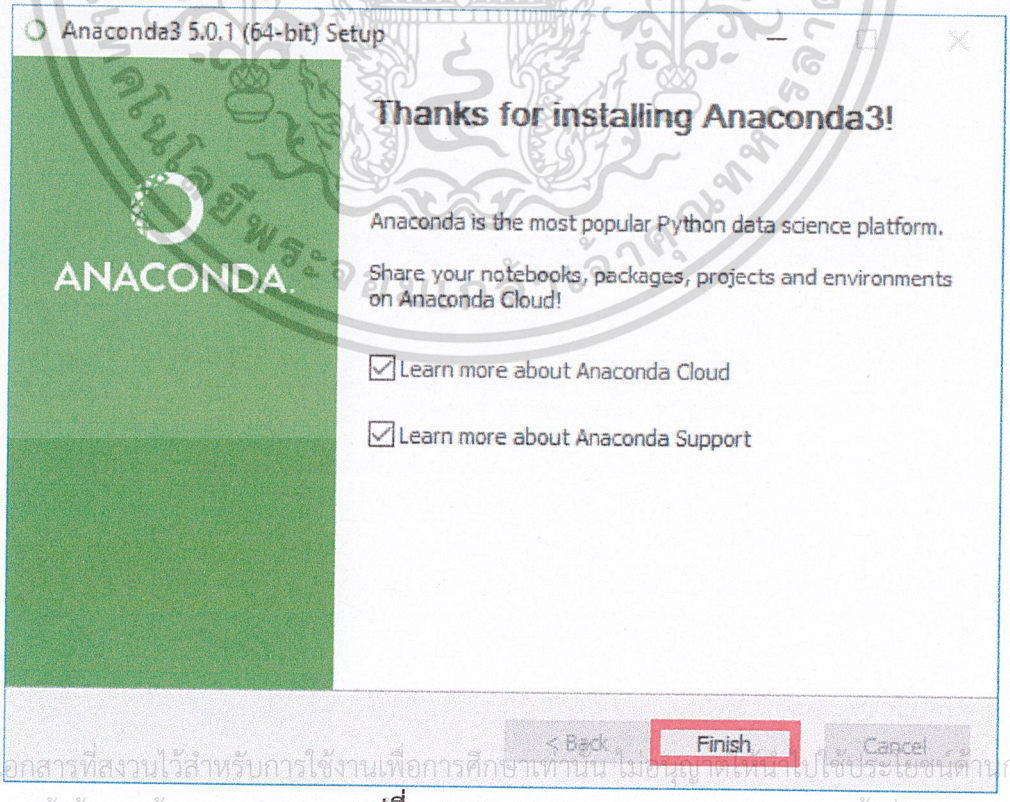
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11.)เมื่อติดตั้งเสร็จสมบูรณ์แล้วให้คลิกปุ่ม Next



รูปที่ ก.9 Next

12.)คลิกที่ปุ่ม Finish

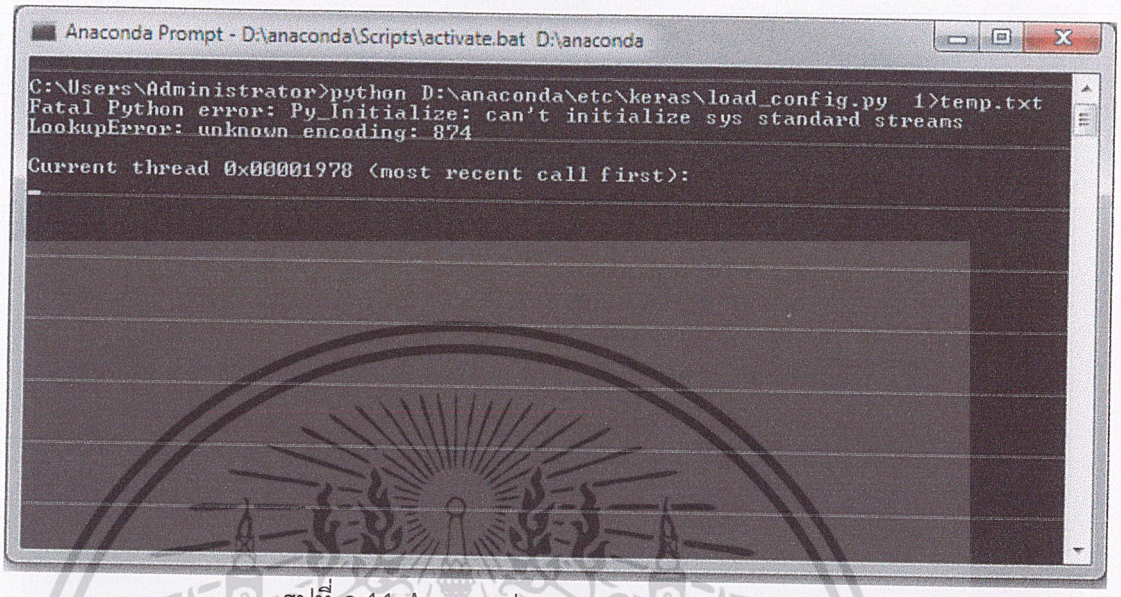


รูปที่ ก.10 Finish

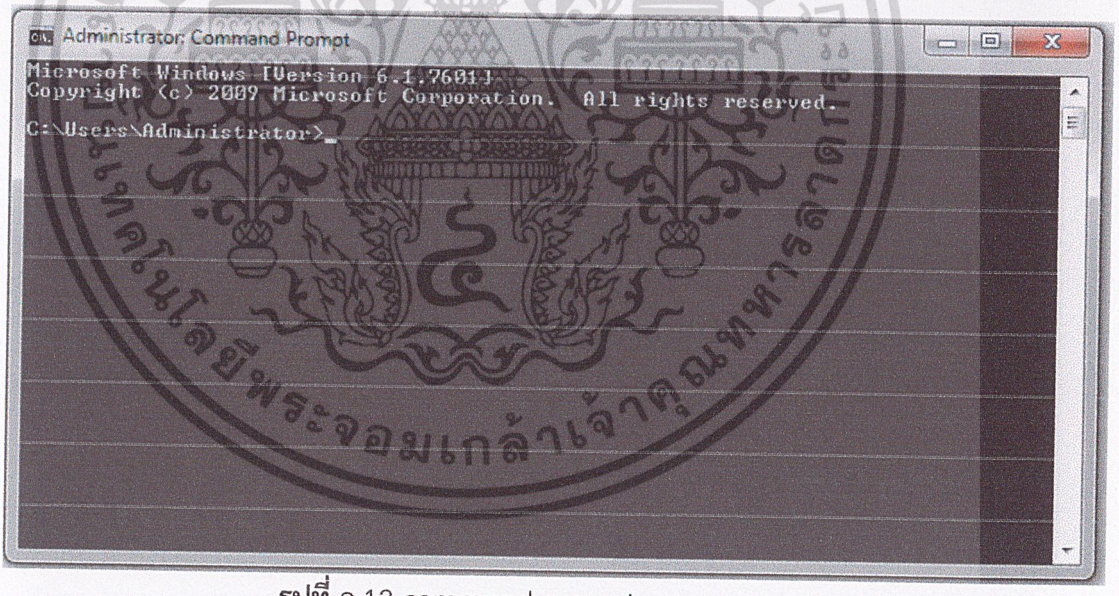
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.1.2 การติดตั้ง library

การติดตั้ง library บน anaconda สามารถทำได้โดยการ เปิดโปรแกรม Anaconda prompt หรือ command prompt โดยสามารถใช้คำสั่งได้ดังต่อไปนี้



รูปที่ ก.11 Anaconda prompt



รูปที่ ก.12 command prompt

1.)Numpy

เป็นlibraryที่ช่วยในการจัดรูปแบบ array หลายมิติเพื่ออำนวยความสะดวกบนภาษาpython สามารถติดตั้งได้โดย การพิมบน Anaconda prompt ว่า “conda install numpy” หรือ การพิมบน command prompt ว่า “pip install numpy”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.)Matplotlib

เป็นlibraryที่ช่วยในเรื่องของการแสดงผลในรูปแบบของ visualiztion ให้ผลลัพธ์ออกมาดูได้ง่ายสามารถติดตั้งได้โดย การพิมบน Anaconda prompt ว่า “conda install matplotlib”หรือ การพิมบน command prompt ว่า “pip install matplotlib”

3.)Keras

เป็นlibraryที่จะช่วยในเรื่องของ deep learning การพิมบน Anaconda prompt ว่า “conda install keras==version 2.2.4”หรือ การพิมบน command prompt ว่า “pip install keras==version 2.2.4”

4.)Tensorflow

เป็นlibraryที่จะช่วยในเรื่องของการพัฒนาmachine learning การพิมบน Anaconda prompt ว่า “conda install tensorflow”หรือ การพิมบน command prompt ว่า “pip install tensorflow”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
โค้ดตัวอย่างของโปรแกรม

ข.1 การรีสเกล ค่า pixel

```
# rescale the pixel values from [0, 255] to [0, 1]
train_datagen = ImageDataGenerator(rescale=1./255,
                                   rotation_range=20,
                                   width_shift_range=0.2,
                                   height_shift_range=0.2,
                                   horizontal_flip=True,
                                   fill_mode='nearest')

validation_datagen = ImageDataGenerator(rescale=1./255)
```

ข.2 การสร้าง layers ของ Deep learning

```
        model=models.Sequential()
        model.add(layers.Dense(1024,activation='relu',input_dim=feature_shape))
        model.add(layers.Dense(120,activation='relu'))
        model.add(layers.Dropout(0.5))
        model.add(layers.Dense(nClass,activation='softmax'))

        model.compile(optimizers.Adagrad(lr=1e-4,epsilon=None,decay=0.0),
                      loss='categorical_crossentropy',
                      metrics=['accuracy'])

        history = models.fit(train_freatures,
                             train_labels,epochs=100,
                             batch_size=10,
                             validation_data=(val_feature,val_labels))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข.3 การแสดงกราฟแบบ visualization

```
acc = history.history['acc']
val_acc = history.history['val_acc']
loss = history.history['loss']
val_loss = history.history['val_loss']
epochs = range(len(acc))
plt.style.use([u'default'])
plt.grid(True)
plt.plot(epochs, acc, 'C0', label='Training acc')
plt.plot(epochs, val_acc, 'C1', label='Validation acc')
plt.title('Training and validation accuracy')
plt.xlabel('epoch')
plt.ylabel('accuracy')
plt.legend()
plt.figure()
plt.grid(True)
plt.plot(epochs, loss, 'C0', label='Training loss')
plt.plot(epochs, val_loss, 'C1', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('epoch')
plt.ylabel('loss')
plt.legend()
plt.show()
```

ข.4 การรับinputเข้ามาทำนายผลลัพธ์

```
# Input :
img_path = ' ใส่path fileรูปภาพที่ต้องการ '
img = image.load_img(img_path, target_size=(224, 224))
x = image.img_to_array(img)
x = np.expand_dims(x, axis=0)
x = preprocess_input(x)
features = pre_model.predict(x)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
features = np.reshape(features, (1, feature_shape)
pred_features = model.predict(features)
pred_class_test = np.argmax(pred_features)
pred_label_test = idx2label[pred_class_test]
print('Prediction :{}, confidence : {:.3f}'.format(
    pred_label_test,
    pred_features[0][pred_class_test]))
plt.imshow(img)
plt.show()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้