

เทคโนโลยีการเรียนรู้เชิงลึกบนแอปพลิเคชันสำหรับการตรวจวัดแอลกอฮอล์
ด้วยกล้องอินฟราเรด

A DEEP LEARNING BASED APPLICATION FOR DRUNKS DETECTION
WITH INFRARED CAMERA



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมชีวการแพทย์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2564

KMITL-2021-EN-M-317-049

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A DEEP LEARNING BASED APPLICATION FOR DRUNKS DETECTION
WITH INFRARED CAMERA



PISIT EIAMUDOMCHAI

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN BIOMEDICAL ENGINEERING
SCHOOL OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2021

KMITL-2021-EN-M-317-049

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2021

SCHOOL OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	เทคโนโลยีการเรียนรู้เชิงลึกบนแอปพลิเคชันสำหรับการตรวจวัดแอลกอฮอล์ด้วยกล้องอินฟราเรด
นักศึกษา	นายพิสิษฐ์ เอี่ยมอุดมชัย
รหัสประจำตัว	62601087
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมชีวการแพทย์
พ.ศ.	2564
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ดร.วิบูลย์ ปิยวัฒน์เมธา

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้จัดขึ้นโดยมีจุดประสงค์เพื่อช่วยในการแก้ไขปัญหาความล่าช้าในการตรวจวัดแอลกอฮอล์และลดปัญหาของการปนเปื้อนจากการใช้เครื่องเป่าวัดแอลกอฮอล์ (Alcohol breath analyzer) งานวิจัยนี้มุ่งเน้นการออกแบบระบบในการตรวจวัดแอลกอฮอล์ด้วยเทคโนโลยีการเรียนรู้เชิงลึก (Deep Learning) โดยระบบนี้จะประกอบไปด้วยสองส่วนคือ ส่วนแรกเป็นกล้องอินฟราเรดสำหรับบันทึกภาพอินฟราเรดและภาพปกติและส่วนที่สองคือ ระบบการประมวลผลภาพในการตรวจจับแอลกอฮอล์ โดยระบบดังกล่าวจะใช้เทคโนโลยีการเรียนรู้เชิงลึก (Deep learning technology) ซึ่งทำงานอยู่บนระบบปฏิบัติการ iOS และผลลัพธ์ของงานวิจัยนี้สามารถตรวจวัดได้จากค่าความแม่นยำและ confusion matrix โดยเป็นความแม่นยำของ 4 levels of classification (0 แก้ว, 1 แก้ว, 2 แก้ว หรือ 3 แก้ว) 57.14% (ประชากร 28) และความแม่นยำ 78.57% สำหรับ binary identification ไม่เมา (0 แก้ว, 1 แก้ว) หรือเมา (2 แก้ว, 3 แก้ว) โดยแต่ละแก้วมีเปียร์ 200 มิลลิลิตร (แอลกอฮอล์ 6.4% โดยปริมาตร) คิดเป็นแอลกอฮอล์ 10 g และมีปริมาณแอลกอฮอล์ในเลือด BAC 0.026% โดยประมาณ และมีการวิจัยการเพิ่มฐานข้อมูลรูปภาพด้วยการพลิกกลับของรูปภาพแต่ไม่พบการเพิ่มขึ้นอย่างมีนัยยะสำคัญของประสิทธิภาพของโมเดล

Thesis	A Deep Learning based Application for Drunks Detection with Infrared Camera
Student	Mr. Pisit Eiamudomchai
Student ID.	62601087
Degree	Master of Engineering
Program	Biomedical Engineering
Year	2021
Thesis Advisor	Dr.Wibool Piyawattanametha

ABSTRACT

The aim of this thesis is to solve the problems of alcohol measurement delays and reduce the issue of contamination that arise from alcohol breath analyzer. Therefore, we designed a novel infrared (IR) camera-based alcohol detection system with deep learning technology which can operate on a mobile phone. This system consists of 2 parts. The first part is an infrared camera (FLIR) used for collecting both IR and normal images and the second part is an image processing part for alcohol detection with based on deep learning technology operating on an iPhone operating system (iOS) mobile phone. Our handheld IR based detection system achieves accuracy of 57.14% (28 population) accuracy with 4 levels of classification (sober, 1 glass, 2 glasses, or 3 glasses) and 78,57% with binary identification (Normal that contain sober and 1 glass or Drunk that contain 2 glasses and 3 glasses). Each glass contains 200 ml of beer (alcohol 6.4% vol). That is 10 grams of pure alcohol then equal to BAC 0.026% approximately. After that we increase the image database by horizontally flip them but it has not significant improve the efficiency of model.

กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ดร.วิบูลย์ ปิยวัฒน์เมธา ที่ให้ความช่วยเหลือ ให้คำชี้แนะช่วยแก้ปัญหาตลอดจนให้ความรู้และประสบการณ์ที่ดีแก่ข้าพเจ้า

ขอขอบคุณพี่ๆ น้องๆ ในห้องปฏิบัติการในภาควิชาวิศวกรรมชีวการแพทย์สำหรับทุกความช่วยเหลือและความเอาใจใส่ในทุกรายละเอียดที่ทำให้วิทยานิพนธ์เล่มนี้สำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายต้องขอขอบคุณเพื่อนๆ ของข้าพเจ้าที่เรียนสาขาวิชาวิศวกรรมชีวการแพทย์ ที่เป็นเสมือนคู่คิดและเป็นกำลังใจที่ดีตลอดมา

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้กับบิดามารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้และถ่ายทอดประสบการณ์ที่ดีให้แก่ข้าพเจ้า

พิสิษฐ์ เอี่ยมอุดมชัย



สารบัญ

	หน้า
บทคัดย่อ.....	I
ABSTRACT.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญรูปภาพ.....	VII
สารบัญตาราง.....	X
บทที่ 1.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์.....	1
1.3 ขอบเขตของการวิจัย.....	2
1.4 ประโยชน์ของการวิจัย.....	2
1.5 แผนการดำเนินงานวิทยานิพนธ์.....	2
บทที่ 2.....	3
2.1 อุณหภูมิของใบหน้า.....	3
2.2 Blood alcohol concentration (BAC).....	5
2.3 ทฤษฎีอินฟราเรด.....	8
2.3.1 อินฟราเรด คืออะไร.....	9
2.3.2 Infrared thermography / กล้องอินฟราเรด.....	10
2.4 โครงข่ายประสาทเทียม.....	14
2.4.1 โครงข่ายประสาทเทียมคอนโวลูชัน.....	15
2.4.2 โครงข่ายประสาทเทียมแบบวนกลับ.....	16
2.5 การเรียนรู้เชิงลึก (Deep Learning).....	16
2.5.1 ประโยชน์ของการเรียนรู้เชิงลึก.....	21
2.6 Confusion matrix.....	25

2.7 ระบบปฏิบัติการ iOS	29
2.8 Xcode.....	31
2.8.1 การใช้งาน Xcode แบบเบื้องต้น	32
2.8.2 Swift.....	34
2.8.3 Objective-C	34
2.9 Core ML	35
2.10 โรคที่สามารถติดต่อได้ทางน้ำลาย	36
2.10.1 ไข้หวัด	36
2.10.2 คออักเสบ (Strep throat).....	36
2.10.3 โรคมือเท้าปาก (Hand-foot-mouth disease).....	37
2.10.4 คางทูม (Mumps).....	37
2.10.5 โรคหูดข้าวสุก (Molluscum contagiosum).....	38
2.10.6 โรคเริมที่ปาก (Herpes Simplex).....	38
2.10.7 ไวรัสตับอักเสบ (Hepatitis).....	38
2.10.8 คอตีบ (Diphtheriae).....	38
2.10.9 เยื่อหุ้มสมองอักเสบ (Meningitis).....	39
2.10.10 ไข้กาฬหลังแอ่น (Meningococcal meningitis).....	39
บทที่ 3	40
3.1 การเก็บข้อมูลภาพ	41
3.2 การออกแบบอัลกอริทึมของโปรแกรมประมวลผล.....	43
3.2.1 การเตรียมข้อมูล	43
3.2.2 การเตรียมและสอนโมเดลที่จะใช้	44
3.2.3 โครงสร้างของโมเดลที่ใช้	45
3.2.4 การตรวจสอบความถูกต้องและการใช้งานของโมเดล	47
3.2.5 การ FLIP ภาพและปรับพารามิเตอร์ต่าง ๆ	47
3.2.6 แปลงโมเดลเพื่อนำไปใช้ในแอปพลิเคชัน	50

3.2.7 สร้าง Machine Learning บน iOS ด้วย Core ML.....	50
3.3 ฮาร์ดแวร์.....	53
บทที่ 4	54
4.1 การเก็บข้อมูลภาพ	54
4.2 ผลจากการสอนโมเดลและการปรับพารามิเตอร์ต่าง ๆ	55
4.3 ผลการทดสอบการประมวลผลของ CoreML บน iOS.....	65
4.4 การประกอบฮาร์ดแวร์	67
บทที่ 5	69
5.1 สรุปผลการวิจัย	69
5.2 ข้อเสนอแนะ	71
บรรณานุกรม.....	72
ภาคผนวก ก.....	76
ภาคผนวก ข.....	80
ภาคผนวก ค.....	86
ภาคผนวก ง.....	93
ประวัติผู้เขียน.....	96

สารบัญญรูปภาพ

หน้า

รูปที่ 2.1 แสดงรูปภาพอินฟราเรดของใบหน้าและมือที่สามารถใช้สำหรับยืนยันตัวตนแบบไบโอเมตริกซ์ได้ [4].....	3
รูปที่ 2.2 แสดงความแตกต่างของอุณหภูมิบนใบหน้าเมื่อมีการดื่มแอลกอฮอล์ [6]	4
รูปที่ 2.3 แสดงภาพอินฟราเรดเมื่อก่อนดื่มแอลกอฮอล์และหลังจากดื่มแอลกอฮอล์ [5].....	4
รูปที่ 2.4 แสดงปริมาณแอลกอฮอล์สำหรับ 1 ดื่มมาตรฐาน [8].....	6
รูปที่ 2.5 แสดงปริมาณเครื่องดื่มชนิดต่าง ๆ ต่อ 1 ดื่มมาตรฐาน [8]	6
รูปที่ 2.6 แสดงปริมาณดื่มมาตรฐานในเครื่องดื่มแอลกอฮอล์ที่มีบรรจุภัณฑ์แตกต่างกัน [8]	7
รูปที่ 2.7 ภาพถ่ายจากกล้องอินฟราเรด [12]	8
รูปที่ 2.8 รังสีอินฟราเรด [12]	9
รูปที่ 2.9 ภาพถ่ายมนุษย์ในย่าน mid-infrared เป็นภาพที่เกิดจากรังสีความร้อนที่แผ่ออกมาจากคน [13]	10
รูปที่ 2.10 ระบบการวัดอุณหภูมิโดยอาศัยการแผ่รังสีอินฟราเรดของวัตถุ [13]	11
รูปที่ 2.11 การตรวจวัดผู้ที่สงสัยที่จะเป็นโรคไข้หวัดนกและไข้หวัดใหญ่สายพันธุ์ใหม่ 2009 [12].....	12
รูปที่ 2.12 การประยุกต์ใช้งานกล้องถ่ายภาพความร้อนในอุตสาหกรรมอาหาร [13]	12
รูปที่ 2.13 การประยุกต์ใช้งานกล้องถ่ายภาพความร้อนในอุตสาหกรรมอื่นๆ [13].....	13
รูปที่ 2.14 การประยุกต์ใช้งานกล้องอินฟราเรดกับการตรวจสอบระบบไฟฟ้า [12]	13
รูปที่ 2.15 การประยุกต์ใช้งานกล้องอินฟราเรดกับอุตสาหกรรมปิโตรเคมี [12].....	13
รูปที่ 2.16 การประยุกต์ใช้งานกล้องอินฟราเรดในงานเครื่องกล [12].....	14
รูปที่ 2.17 แสดงโครงข่ายประสาทเสมือนที่มีการเชื่อมต่อกันผ่านกลุ่มโหนด [14]	15
รูปที่ 2.18 โครงสร้างของโครงข่ายประสาทเทียมแบบวนกลับ [15].....	16
รูปที่ 2.19 ความแตกต่างระหว่าง Non-deep และ Deep neural network [19].....	17
รูปที่ 2.20 การเชื่อมต่อกันของเซลล์ประสาท [17].....	18
รูปที่ 2.21 การเชื่อมต่อของเซลล์ประสาท [17].....	18
รูปที่ 2.22 โครงข่ายใยประสาทเสมือน [17].....	19
รูปที่ 2.23 Traditional Programming & Machine learning [17].....	19
รูปที่ 2.24 ตัวอย่างของการเรียนรู้เชิงลึก [17].....	20
รูปที่ 2.25 การเรียนรู้เชิงลึกเมื่อมีหลาย Hidden Layer [17].....	20
รูปที่ 2.26 การทำงานของการเรียนรู้เชิงลึก [19].....	21
รูปที่ 2.27 ประเภทของ Machine Learning [20]	22
รูปที่ 2.28 การทำงานของ Supervised learning [20].....	22

รูปที่ 2.29	แสดงการทำงานของ Unsupervised learning [20]	23
รูปที่ 2.30	แสดงการทำงานของ Reinforcement learning [20]	24
รูปที่ 2.31	Machine Learning Techniques [21]	25
รูปที่ 2.32	ตัวอย่าง confusion matrix ขนาด 2x2 [23].....	26
รูปที่ 2.33	ตัวอย่าง confusion matrix ทำนายอีเมลทั้งหมด N=100 [24]	28
รูปที่ 2.34	สัญลักษณ์ของระบบปฏิบัติการ iOS [25].....	29
รูปที่ 2.35	สัญลักษณ์ของ Android [26]	30
รูปที่ 2.36	โปรแกรม Xcode [27]	31
รูปที่ 2.37	คลิกที่ Create New Xcode Project [27]	32
รูปที่ 2.38	รูปแบบของการตั้งค่าแอปพลิเคชัน [27]	32
รูปที่ 2.39	ตั้งชื่อแอปพลิเคชัน [27].....	33
รูปที่ 2.40	เลือกโพลเดอร์ที่จะสร้างโปรเจค [27].....	33
รูปที่ 2.41	Swift Logo [28]	34
รูปที่ 2.42	Objective – C Logo [29].....	35
รูปที่ 2.43	Core ML Logo [31].....	35
รูปที่ 2.44	ใช้หัวัดใหญ่ [32]	36
รูปที่ 2.45	โรคมือเท้าปาก [32].....	37
รูปที่ 3.1	แผนภาพการทำงานของระบบ	40
รูปที่ 3.2	ขั้นตอนการเก็บข้อมูลภาพ	42
รูปที่ 3.3	วิธีการเก็บข้อมูลภาพ	42
รูปที่ 3.4	การแบ่งประเภทของรูปภาพ	43
รูปที่ 3.5	ขั้นตอนการสอนโมเดล	44
รูปที่ 3.6	แสดงโครงสร้างของ Stem และ Final Layers ของ EfficientNet [37].....	46
รูปที่ 3.7	แสดงโครงสร้างของ Module ทั้ง 5 ของ EfficientNet [37].....	46
รูปที่ 3.8	แสดงโครงสร้างของโมเดล EfficientNetB0 โดยที่ x2 หมายถึง Module ภายในวงเล็บนี้นำมาวนซ้ำ 2 รอบ [37].....	47
รูปที่ 3.9	หน้าต่างโปรแกรม ImageJ	48
รูปที่ 3.10	แสดงการ Import Image Sequence ใน ImageJ	48
รูปที่ 3.11	แสดงวิธีการ Flip Horizontally ในโปรแกรม ImageJ.....	49
รูปที่ 3.12	แสดงตัวอย่างรูปภาพที่ทำการพลิกกลับ โดยภาพซ้ายเป็นภาพตั้งต้น ภาพขวาเป็นภาพที่พลิกกลับแล้ว.....	49
รูปที่ 3.13	เลือก Single View App.....	51
รูปที่ 3.14	ตั้งชื่อและเลือกภาษาที่ใช้เป็น Swift.....	51

รูปที่ 3.15 วาด View ใน Storyboard	52
รูปที่ 3.16 รายละเอียดของ ResnetV2.mlmodel ที่ Xcode อ่าน	52
รูปที่ 3.17 แต่ละหน้าของแอปพลิเคชันเมื่อติดตั้งลงบน iOS.....	53
รูปที่ 3.18 CAD ตัวยัดระหว่างโทรศัพท์และกล้องอินฟราเรด.....	53
รูปที่ 4.1 a) filter contrast, b) filter grey, c) filter antarctic	54
รูปที่ 4.2 แสดง confusion matrix ของโมเดล EfficientNetB0 โดยที่ 0 คือไม่มา และ 1 คือมา.....	56
รูปที่ 4.3 แสดงค่าตัวแปรวัดผลต่าง ๆ ที่ได้จากโมเดล EfficientNetB0.....	56
รูปที่ 4.4 แสดง confusion matrix ของโมเดล mobilenet_v1_025_244 โดยเลขในทั้งแกนตั้งและ นอนแทนจำนวนแก้วของแอลกอฮอล์ เช่น 0 แก้ว, 1 แก้ว เป็นต้น	58
รูปที่ 4.5 แสดงค่าตัวแปรวัดผลต่าง ๆ ที่ได้จากโมเดล mobilenet_v1_025_244	59
รูปที่ 4.6 แสดง confusion matrix และค่าตัวแปรวัดผลต่าง ๆ ของโมเดล EfficientNetB0 ที่สอนโดย รูปภาพที่ FLIP แล้ว โดยที่ 0 คือไม่มา และ 1 คือมา	61
รูปที่ 4.7 แสดง confusion matrix ของโมเดล mobilenet_v1_075_244 ที่สอนโดยรูปภาพที่ FLIP แล้วโดยเลขในทั้งแกนตั้งและนอนแทนจำนวนแก้วของแอลกอฮอล์ เช่น 0 แก้ว, 1 แก้ว เป็นต้น	62
รูปที่ 4.8 แสดงค่าตัวแปรวัดผลต่าง ๆ ที่ได้จากโมเดล mobilenet_v1_075_244 ที่สอนโดยรูปภาพที่ FLIP แล้ว	63
รูปที่ 4.9แสดงตัวอย่างของรูปภาพ (a) รูปภาพก่อน Flip (b) รูปภาพหลัง Flip ซึ่งไม่มีความแตกต่างที่ ส่งผลต่อประสิทธิภาพของโมเดล	64
รูปที่ 4.10 การทำงานของแอปพลิเคชัน Binary Classification (a) แสดงผลลัพธ์ไม่มา (b)แสดงผลลัพธ์ มา	66
รูปที่ 4.11 การทำงานของแอปพลิเคชัน 4 Level of Classification (a)แสดงผลลัพธ์ 3 แก้ว (b) แสดงผลลัพธ์ 2 แก้ว	66
รูปที่ 4.12 มุมด้านหลังเมื่อประกอบตัวยัด	67
รูปที่ 4.13 มุมด้านหน้าเมื่อประกอบตัวยัด.....	67
รูปที่ 4.14 มุมด้านข้างเมื่อประกอบตัวยัด.....	68

สารบัญตาราง

หน้า

ตารางที่ 1.1	แผนการดำเนินวิทยานิพนธ์	2
ตารางที่ 2.1	แสดงผลกระทบทางร่างกายและจิตใจของปริมาณแอลกอฮอล์ในเลือด	5
ตารางที่ 3.1	แสดงรายละเอียดต่าง ๆ ของกล้องอินฟราเรด FLIR ONE Pro โหมด video คร่าว ๆ	41
ตารางที่ 4.1	แสดงค่าความแม่นยำ 5 อันดับสูงสุดของโมเดลสำหรับ Binary classification	55
ตารางที่ 4.2	แสดงค่าความแม่นยำ 5 อันดับสูงสุดของโมเดลสำหรับ 4level of Classification.....	57
ตารางที่ 4.3	แสดงค่าความแม่นยำสูงสุด 3 อันดับของ Binary classification โมเดลที่ทำการสอนกับ ข้อมูลที่ FLIP แล้ว	60
ตารางที่ 4.4	แสดงค่าความแม่นยำสูงสุด 3 อันดับของ 4Level of classification โมเดลที่ทำการสอนกับ ข้อมูลที่ FLIP แล้ว	61
ตารางที่ 4.5	แสดงผลการปรับ Learning rate โดยค่าอื่น ๆ คงที่.....	64
ตารางที่ 4.6	แสดงผลการปรับ Training step โดยค่าอื่น ๆ คงที่	65
ตารางที่ 4.7	แสดงผลการปรับ Train batch size โดยค่าอื่น ๆ คงที่.....	65

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

จากข้อมูลจากกรมคุมประพฤติ ที่กล่าวว่ามึคดีเกี่ยวกับการเมาแล้วขับในช่วงปีใหม่ 2562 (27-29 ธันวาคม 2561) เพิ่มขึ้นอย่างมากถึงร้อยละ 50 จากปี 2561 (27-29 ธันวาคม 2560) [1] ซึ่งการเมาแล้วขับจัดเป็นสาเหตุสำคัญของการเกิดอุบัติเหตุบนท้องถนน ก่อให้เกิดการบาดเจ็บและความสูญเสียเป็นจำนวนมาก ทั้งทรัพย์สินและชีวิต และสร้างความเสียหายมากมายให้กับครอบครัวหรือแม้กระทั่งคนรอบตัวที่ไม่เกี่ยวข้องเลย ดังนั้นเพื่อลดจำนวนอุบัติเหตุที่มีสาเหตุจากการเมาแล้วขับ ประเทศไทยจึงมีการออกกฎหมายว่าผู้ขับรถทุกชนิดมีปริมาณแอลกอฮอล์ในเลือดต้องไม่เกิน 50 มิลลิกรัมเปอร์เซ็นต์ จึงต้องมีการตรวจวัดแอลกอฮอล์เพื่อการควบคุมที่มีประสิทธิภาพ โดยวิธี [2] การตรวจวัดในเลือด วิธีนี้เป็นวิธีที่น่าเชื่อถือที่สุดแต่ต้องเจาะเลือด รวดนานและราคาแพง [3] การใช้เครื่องตรวจวัดระดับแอลกอฮอล์ในกระแสเลือด โดยวิธีการเป่าลมหายใจ (Breath Analyzer Test) เป็นวิธีที่นิยมที่สุดในปัจจุบัน ซึ่งมีข้อดีคือรวดเร็ว พกพาง่าย และราคาถูก แต่มีข้อเสียคือต้องเป่าผ่านหลอดเป่า หากใช้ซ้ำจะติดเชื้อโรคผ่านทางน้ำลาย เสียค่าใช้จ่ายเพิ่มในการซื้อหลอดเป่า การปฏิเสธที่จะให้ความร่วมมือในการตรวจวัดแอลกอฮอล์ของคนที่เมาแล้วขับ ส่งผลให้เกิดความล่าช้าในการตรวจวัด

ผู้จัดทำเล็งเห็นถึงความสำคัญของปัญหาที่กล่าวมาในข้างต้น จึงออกแบบระบบการตรวจวัดระดับแอลกอฮอล์ด้วยกล้องอินฟราเรดมาเพื่อแก้ไขปัญหานี้ โดยการทำงานของระบบดังกล่าวประกอบไปด้วยกล้องอินฟราเรดที่ถ่ายรูปแบบบุคคลที่ต้องการตรวจวัดแอลกอฮอล์ จากนั้นทำการประมวลผลภาพก่อนแล้วจึงประมวลผลด้วยการเรียนรู้เชิงลึก (Deep learning) ออกมาเป็นปริมาณแอลกอฮอล์บนแอปพลิเคชัน (iOS)

1.2 วัตถุประสงค์

1. เพื่อพัฒนาระบบการตรวจวัดปริมาณแอลกอฮอล์ด้วยเทคโนโลยีการเรียนรู้เชิงลึก
2. เพื่อสร้างระบบการประมวลผลภาพจากกล้องอินฟราเรดด้วยการเรียนรู้เชิงลึกบนแอปพลิเคชัน (iOS) ไว้สำหรับคัดกรอง
3. เพื่อลดระยะเวลาที่ใช้ในการตรวจวัดปริมาณแอลกอฮอล์
4. เพื่อป้องกันการติดโรคติดต่อจากทางน้ำลาย
5. เพื่อลดค่าใช้จ่ายในการซื้ออุปกรณ์เสริม เช่น หลอดเป่า เป็นต้น

1.3 ขอบเขตของการวิจัย

วิทยานิพนธ์นี้ได้มุ่งเน้นการออกแบบและพัฒนาระบบในการตรวจวัดปริมาณแอลกอฮอล์ด้วยกล้องอินฟราเรด ซึ่งระบบนี้จะประกอบไปด้วย 1) กล้องอินฟราเรดสำหรับบันทึกภาพบุคคลที่ต้องการวัดปริมาณแอลกอฮอล์ 2) ระบบการประมวลผลภาพเพื่อให้ได้ปริมาณแอลกอฮอล์ โดยระบบดังกล่าวจะใช้กระบวนการประมวลผลภาพก่อนและใช้เทคโนโลยีการเรียนรู้เชิงลึก (Deep learning) ซึ่งทำงานอยู่บนระบบปฏิบัติการ iOS

1.4 ประโยชน์ของการวิจัย

1. วิธีการตรวจวัดปริมาณแอลกอฮอล์โดยใช้กล้องอินฟราเรด จะช่วยป้องกันการสะสมของเชื้อโรคจากการเป่าเครื่องวัดแอลกอฮอล์ สร้างความสบายใจให้แก่ผู้ที่ต้องถูกตรวจวัด อีกทั้งยังสะดวกต่อการพกพา
2. การใช้เทคโนโลยีการเรียนรู้เชิงลึกในการเพิ่มประสิทธิภาพของการประมวลผลภาพ ทำให้ลดระยะเวลาในการตรวจวัดให้มีการตอบสนองที่รวดเร็วและแม่นยำขึ้น
3. โมเดลการเรียนรู้เชิงลึกที่ได้เป็นโมเดลพื้นฐานที่สามารถนำไปประยุกต์ใช้กับ
4. ลดค่าใช้จ่ายในการซื้อหลอดเป่า

1.5 แผนการดำเนินงานวิทยานิพนธ์

แผนการดำเนินงานวิทยานิพนธ์แสดงในตารางที่ 1.1

ที่	แผนการ
1	รวบรวมเนื้อหาและงานวิจัยอื่น ๆ ที่เกี่ยวข้อง
2	ศึกษาวิธีการเขียนโปรแกรมการเรียนรู้เชิงลึก (Deep Learning) โดยใช้ภาษา Python
3	ศึกษาวิธีการเพิ่มประสิทธิภาพของโมเดลการเรียนรู้เชิงลึก
4	ทำการสอนโมเดลการเรียนรู้เชิงลึกใหม่อีกครั้ง
5	ทดสอบความแม่นยำของโมเดลการเรียนรู้เชิงลึก
6	ทดสอบและปรับปรุงโมเดลการเรียนรู้เชิงลึก
7	ทำรูปเล่มวิจัย

ตารางที่ 1.1 แผนการดำเนินงานวิทยานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 อุณหภูมิของใบหน้า

ภาพถ่ายใบหน้าของมนุษย์นั้นมีคุณลักษณะที่มีเอกลักษณ์ซึ่งสามารถมองเห็นการเปลี่ยนแปลงของการไหลเวียนเลือดได้อย่างง่ายดาย [4] อีกทั้งบริเวณใบหน้านั้นมักจะไม่มีการปกคลุมจึงทำให้กล้องอินฟราเรดนั้นสามารถจับอุณหภูมิจากใบหน้าได้โดยตรง [5] การจดจำใบหน้าโดยใช้กล้องอินฟราเรดจะกำจัดผลกระทบของการเปลี่ยนแปลงของสภาพแวดล้อมทางแสงและลดขั้นตอนการแบ่งสัดส่วนออก แต่ว่าการใช้งานกล้องอินฟราเรดก็ยังเป็นสิ่งที่ท้าทายเนื่องจากความแตกต่างกันของรูปแบบของความร้อนของแต่ละบุคคลทำให้เกิดระดับการทำงานที่แตกต่างกันหรือความรู้สึกต่าง ๆ เช่น ความวิตกกังวล

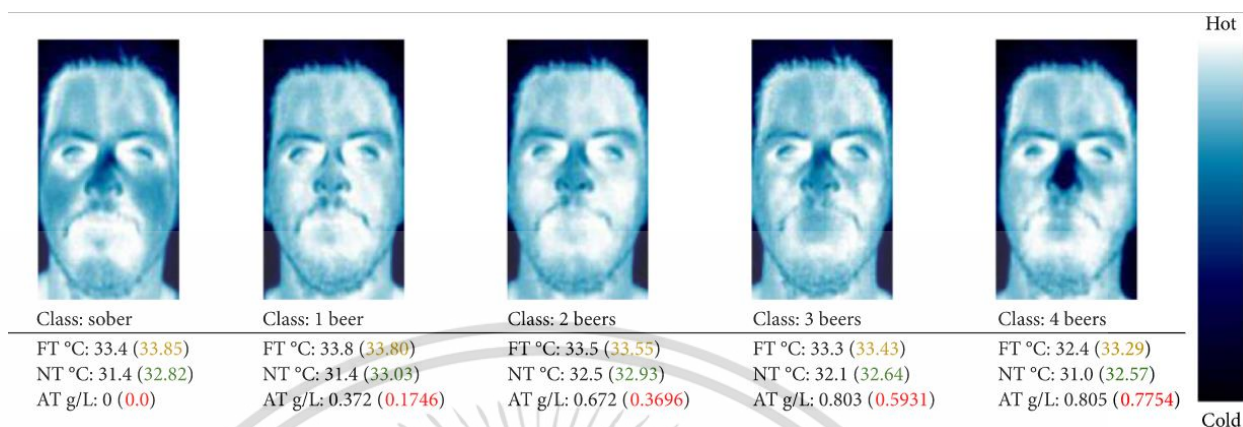


รูปที่ 2.1 แสดงรูปภาพอินฟราเรดของใบหน้าและมือที่สามารถใช้สำหรับยืนยันตัวตนแบบไบโอเมตริกซ์ได้ [4]

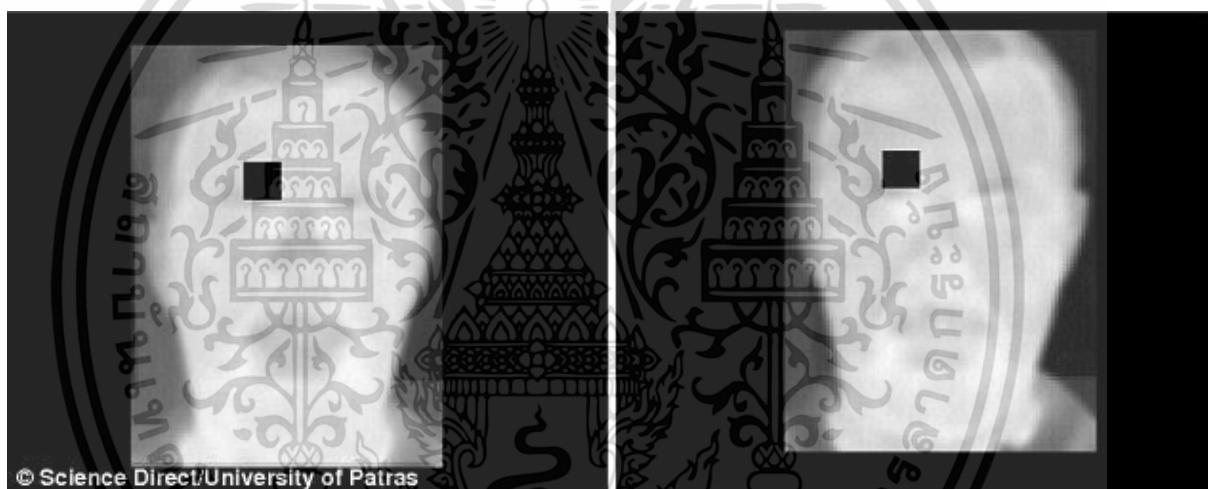
ภาพถ่ายอินฟราเรดของใบหน้า ประกอบไปด้วยข้อมูลที่บ่งบอกถึงอุณหภูมิของใบหน้า ซึ่งอุณหภูมินี้จะขึ้นอยู่กับลักษณะทางกายภาพของแต่ละบุคคล จากกฎของ Wien ได้บอกไว้ว่า ใบหน้าของมนุษย์นั้นจะมีการปล่อยรังสีที่มีอุณหภูมิประมาณ 300 เคลวิน ออกมา และจะมีค่าการสะท้อนรังสี (Emissivity) อยู่ระหว่าง 0.98 และ 0.99 ซึ่งความยาวคลื่นมากที่สุดที่สามารถสะท้อนรังสีได้คือ $9.5 \mu\text{m}$ ซึ่งจากคุณสมบัติของคลื่นแม่เหล็กไฟฟ้าที่มีความยาวคลื่นอยู่ในช่วง 7 ถึง $13 \mu\text{m}$ จึงทำให้ทราบข้อมูลต่างๆบนใบหน้าได้

จากงานวิจัยของ University of Patras พบวิธีการในการตรวจสอบคนที่มีการตี้มแอลกอฮอล์แบบใหม่ ซึ่งเป็นการตรวจสอบการเปลี่ยนแปลงของอุณหภูมิของใบหน้า ในการตรวจสอบวิธีการแรก นั้นเป็นการตรวจสอบโดยใช้ pixels ในภาพ thermal ของใบหน้าเทียบกันระหว่างคนปกติกับคนที่ตี้มแอลกอฮอล์ เป็นสิ่งที่ใช้ในการตรวจสอบ ซึ่งพบว่าคนที่มีการตี้มแอลกอฮอล์จะมีการเปลี่ยนแปลงของอุณหภูมิในบริเวณหน้าผาก และจมูก และวิธีการที่สอง จะตรวจสอบด้วยการสังเกตเส้นเลือดบนใบหน้า ซึ่งจะมีการทำงานที่มากขึ้นเมื่อมีการตี้มแอลกอฮอล์[6]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 แสดงความแตกต่างของอุณหภูมิบนใบหน้าเมื่อมีการดื่มแอลกอฮอล์ [6]



รูปที่ 2.3 แสดงภาพอินฟราเรดเมื่อก่อนดื่มแอลกอฮอล์และหลังจากดื่มแอลกอฮอล์ [5]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 Blood alcohol concentration (BAC)

Blood alcohol concentration (BAC) [7][8] นั้นคือ ปริมาณแอลกอฮอล์ (เอทิลแอลกอฮอล์ หรือ เอทานอล) ในกระแสเลือดของคน ซึ่ง BAC 0.10% นั้นหมายความว่าในกระแสเลือดของคนมี ปริมาณของแอลกอฮอล์หนึ่งในส่วนใน 1000 ส่วนของเลือด ซึ่งในประเทศไทยกฎหมายได้ระบุไว้ว่าผู้ขับรถทุกชนิดที่มีปริมาณแอลกอฮอล์ในเลือดเกิน 0.05% หรือ 50 มิลลิกรัมเปอร์เซ็นต์นั้นผิดกฎหมาย ซึ่ง ปัจจัยต่าง ๆ ที่ส่งผลต่อปริมาณแอลกอฮอล์ในเลือดนั้นมีดังต่อไปนี้

- จำนวนของดื่มมาตรฐาน
- เวลาที่ผ่านไปหลังจากดื่มแอลกอฮอล์
- น้ำหนักตัวของผู้ที่ดื่มแอลกอฮอล์
- องค์ประกอบของน้ำในร่างกายของผู้ดื่มแอลกอฮอล์
- การสร้างปริมาณเอนไซม์ต่าง ๆ และระดับเอนไซม์ของผู้ดื่มแอลกอฮอล์
- เพศและระดับฮอร์โมนของผู้ดื่มแอลกอฮอล์
- การให้ยาของผู้ดื่มแอลกอฮอล์
- อาหาร

ซึ่งปัจจัยต่าง ๆ ที่กล่าวมาจะส่งผลต่อกระบวนการเผาผลาญแอลกอฮอล์ในกระแสเลือด ทำให้ปริมาณแอลกอฮอล์ในเลือดนั้นแตกต่างกัน โดยปริมาณแอลกอฮอล์ในเลือดนั้นจะผลกระทบต่อร่างกายดังตารางที่ 2.1 ดังนี้

ปริมาณ BAC (%)	ผลกระทบทางร่างกายและจิตใจ
0.01 – 0.03	ไม่มีผลกระทบที่ชัดเจน อารมณ์ดีขึ้นเล็กน้อย
0.04 – 0.06	รู้สึกผ่อนคลาย, รู้สึกว่าร่างกายอุ่นขึ้น, ลดการใช้เหตุผลและความจำเล็กน้อย ซึ่งในประเทศไทยถือว่าผิดกฎหมายหากมีปริมาณ BAC มากกว่า 0.05%
0.07 – 0.09	การทรงตัว การสนทนา และการมองเห็นบกพร่องเล็กน้อย
0.10 – 0.12	การควบคุมและการตัดสินใจบกพร่องอย่างเห็นได้ชัดและมีอาการเบลอ
0.13 – 0.15	สูญเสียการควบคุมและทรงตัว สายตาพร่ามัว เริ่มมีอาการหายใจลำบาก (วิตกกังวล กระสับกระส่าย)
0.16 – 0.30	มีอาการคลื่นไส้ อาเจียน มีการมีเมมาอย่างรุนแรง ไม่สามารถเดินได้ด้วยตนเอง มีอาการสับสน
0.35 – 0.40	สูญเสียสติ และเริ่มมีอาการโคม่า
0.40 และมากกว่า	อาการโคม่าและมีโอกาสเสียชีวิตเนื่องจากระบบหายใจล้มเหลว

ตารางที่ 2.1 แสดงผลกระทบทางร่างกายและจิตใจของปริมาณแอลกอฮอล์ในเลือด

Standard drinks หรือ ดื่มมาตรฐาน คือหน่วยวัดปริมาณแอลกอฮอล์อย่างง่าย โดยการวัดปริมาณดื่มมาตรฐานเป็นวิธีที่ง่ายที่สุดที่จะวัดว่าได้ดื่มแอลกอฮอล์ไปมากเท่าไรแล้วซึ่งบนบรรจุภัณฑ์ของแอลกอฮอล์ชนิดต่าง ๆ ไม่ว่าจะเป็นไวน์ เบียร์ เหล้า วิสกี้ ก็มีปริมาณดื่มมาตรฐานระบุไว้ที่ฉลากของบรรจุภัณฑ์ ทำให้ง่ายต่อการคิดปริมาณดื่มมาตรฐานที่ดื่ม

โดยดื่มมาตรฐานนั้นจะเป็นปริมาณแอลกอฮอล์บริสุทธิ์ที่เราดื่มเข้าไปโดย 1 ดื่มมาตรฐานจะมีปริมาณแอลกอฮอล์ 10 กรัม



รูปที่ 2.5 แสดงปริมาณเครื่องดื่มชนิดต่าง ๆ ต่อ 1 ดื่มมาตรฐาน [8]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 แสดงปริมาณดื่มมาตรฐานในเครื่องดื่มแอลกอฮอล์ที่มีบรรจุภัณฑ์แตกต่างกัน [8]

สูตรการหาปริมาณแอลกอฮอล์ในเลือดของ Widmark [9][10][11]

ในการหา BAC หรือ ปริมาณแอลกอฮอล์ในเลือดนั้นสามารถประมาณค่าได้โดยสมการ widmark formula ซึ่งคิดค้นโดย Erik Widmark ซึ่งเป็นนักวิจัยชาวสวีเดนและเป็นผู้บุกเบิกในการวิจัยเกี่ยวกับเอทานอลหรือแอลกอฮอล์ภายในร่างกายมนุษย์ ซึ่งสมการ widmark formula จะใช้ข้อมูลเบื้องต้นเช่น น้ำหนัก, เพศ และ ปริมาณแอลกอฮอล์ที่ดื่มเข้าไปในช่วงเวลานั้น ซึ่งจะนับด้วยดื่มมาตรฐานหรือปริมาณแอลกอฮอล์ที่ได้รับเป็นกรัม widmark formula คือสมการ 2.1 ดังนี้

$$BAC \% = \frac{\text{Alcohol consumed in grams}}{\text{Body weight in grams} \times r} \times 100 \quad (2.1)$$

โดยที่ r คือค่าคงที่ของเพศโดยเพศชายมีค่า 0.68 ส่วนเพศหญิงมีค่า 0.5

และถ้าเวลาผ่านไปช่วงเวลาหนึ่งก็จะสามารถคำนวณปริมาณแอลกอฮอล์ที่หลงเหลือภายในร่างกายได้โดยสมการ 2.2 ต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$BAC\% - (\text{elapsed time in hours} \times MR) \quad (2.2)$$

โดยที่ MR คือค่าคงที่ในการเผาผลาญแอลกอฮอล์ซึ่งจะแตกต่างกันโดยเพศชายจะมีค่าประมาณ 0.015 และเพศหญิงจะมีค่าประมาณ 0.017

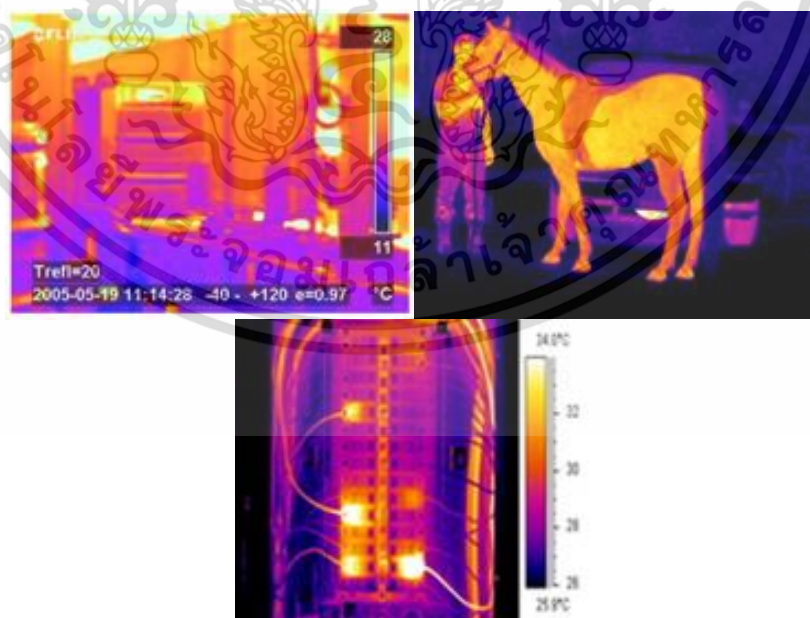
ตัวอย่างในการคำนวณปริมาณแอลกอฮอล์ในเลือดโดยใช้สมการของ Widmark เช่น เพศชาย น้ำหนัก 80 กิโลกรัมดื่มแอลกอฮอล์ 3 ดื่มมาตรฐานไปเป็นเวลา 2 ชั่วโมง เป็นไปตามสมการที่ 2.3 และ 2.4

$$BAC\% = \frac{3 \times 10}{80 \times 1000 \times 0.68} * 100 = 0.055 \quad (2.3)$$

$$BAC\% = 0.055 - (2 * 0.015) = 0.025 \quad (2.4)$$

2.3 ทฤษฎีอินฟราเรด

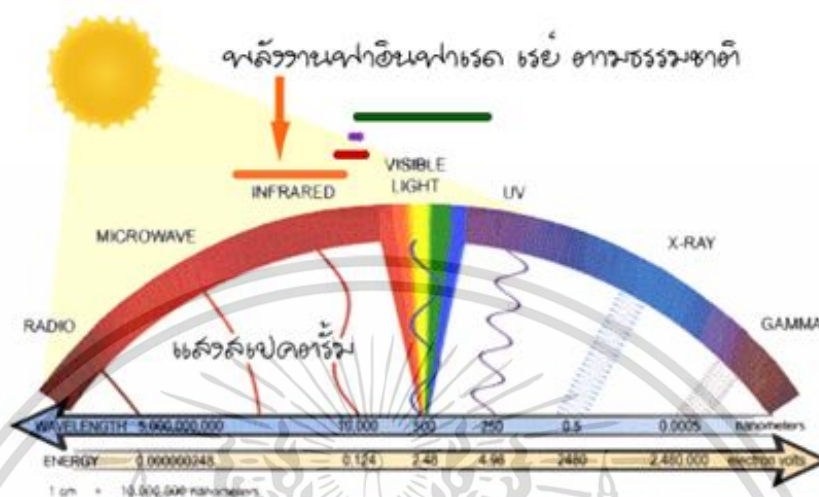
ในการวิจัยหรือค้นคว้าทางด้านฟิสิกส์ของแสงนั้นจะมีด้วยกัน 2 วิธีคือ ทฤษฎีควอนตัมและทฤษฎีแม่เหล็กไฟฟ้าที่เกี่ยวข้องกับทฤษฎีคลื่น แต่ทฤษฎีควอนตัมไม่ได้เป็นที่ยอมรับกัน เนื่องจากไม่สามารถอธิบายเกี่ยวกับปรากฏการณ์การรบกวนและปรากฏการณ์ตีแฟรกชันได้อย่างชัดเจน สำหรับวิศวกรไฟฟ้าหรือช่างที่ได้เรียนวิศวกรรมแสงสว่าง ส่วนใหญ่ได้มีการนำแสงในย่านที่สามารถมองมาประยุกต์ใช้มากกว่า



รูปที่ 2.7 ภาพถ่ายจากกล้องอินฟราเรด [12]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

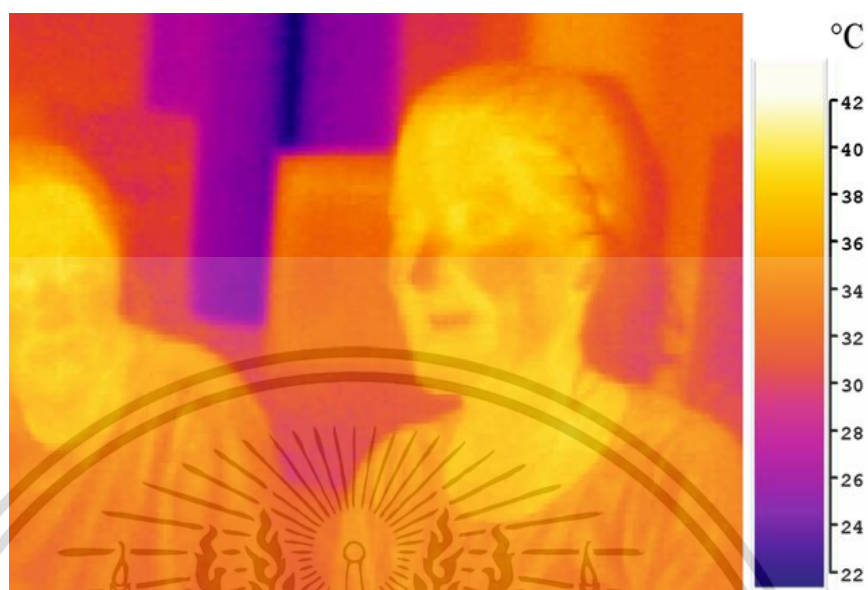
2.3.1 อินฟราเรด คืออะไร [12]



รูปที่ 2.8 รังสีอินฟราเรด [12]

รังสีอินฟราเรด หรือ รังสีความร้อน เป็นคลื่นแม่เหล็กไฟฟ้าชนิดหนึ่งแผ่มาจากดวงอาทิตย์ รังสีแม่เหล็กไฟฟ้าจากดวงอาทิตย์นั้นมีหลายชนิดซึ่งแต่ละชนิดนั้นมีความยาวคลื่นต่างกัน เช่น แสงที่ตามองเห็น (Visible light) มีความยาวคลื่นอยู่ระหว่าง 400 - 700 นาโนเมตร, รังสีอัลตราไวโอเล็ต (Ultraviolet radiation) มีความยาวคลื่น 1 - 400 นาโนเมตร, รังสีแกมมา (Gamma ray) มีความยาวคลื่นน้อยกว่า 0.01 นาโนเมตร, รังสีเอ็กซ์ (X-ray) ความยาวคลื่น 0.01 - 1 นาโนเมตร เป็นต้น สำหรับรังสีอินฟราเรด มีความยาวคลื่น 700 นาโนเมตร- 1 มิลลิเมตร เป็นคลื่นที่มีความถี่ต่ำกว่าความถี่ของสีแดงลงมา มนุษย์จึงไม่สามารถมองเห็นรังสีอินฟราเรด แต่ก็รู้สึกถึงความร้อนได้ ส่วนสัตว์บางชนิด เช่น งูมีประสาทสัมผัสรังสีอินฟราเรดสามารถทราบตำแหน่งของเหยื่อได้ โดยการสัมผัสรังสีอินฟราเรดซึ่งแผ่ออกมาจากร่างกายของเหยื่อ เมื่อโลกได้รับรังสีจากดวงอาทิตย์ โลกจะแผ่รังสีสะท้อนกลับสู่บรรยากาศ เรียกว่า รังสีโลก (Terrestrial radiation) ซึ่งเป็นรังสีอินฟราเรดในคลื่นยาว ซึ่งแตกต่างจากรังสีอินฟราเรดจากดวงอาทิตย์ซึ่งเป็นช่วงคลื่นสั้น ตามปกติแล้วไอน้ำและคาร์บอนไดออกไซด์ในบรรยากาศจะมีปริมาณพอเหมาะและสามารถ ดูดซับพลังงานส่วนนี้ไว้ ทำให้ โลกเก็บความร้อนไว้อยู่ในระดับที่เหมาะสมต่อการดำรงชีวิตของคน สัตว์และพืช ในปัจจุบันมีการนำรังสีอินฟราเรดมาประยุกต์ใช้งานหลายอย่างด้วยกัน เช่น ใช้เป็นตัวกลางในการสื่อสารของอุปกรณ์ไร้สายหลายชนิดทั้งโทรศัพท์มือถือและคอมพิวเตอร์ กล้องอินฟราเรดที่สามารถจับภาพได้แม้ในเวลากลางคืน ล่าสุดมีผลงานการวิจัยจาก University of Arizona แสดงให้เห็นว่ารังสีหรือแสงอินฟราเรดนั้นสามารถนำมาประยุกต์ใช้ประโยชน์อื่นๆ ได้อีก โดยเฉพาะอย่างยิ่งการตรวจวิเคราะห์เซลล์มะเร็ง รวมไปถึงการวิเคราะห์สารปนเปื้อนในแหล่งต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 ภาพถ่ายมนุษย์ในย่าน mid-infrared เป็นภาพที่เกิดจากรังสีความร้อนที่แผ่ออกมาจากคน [13]

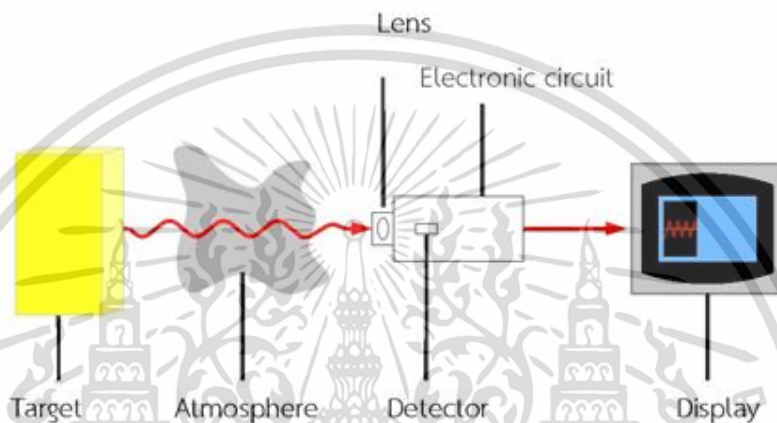
2.3.2 Infrared thermography / กล้องอินฟราเรด [13]

กล้องอินฟราเรด (Infrared thermography) หรือกล้องถ่ายภาพอินฟราเรด หรือกล้องถ่ายภาพความร้อน (thermal image camera หรือ thermal imager หรือ TI - camera) เป็นเครื่องมือวัดอุณหภูมิที่ผิวของวัตถุ ทำงานโดยอาศัยหลักการแผ่รังสีอินฟราเรด (Infrared radiation) ออกจากวัตถุ ซึ่งเป็นการวัดแบบไม่สัมผัสและไม่ทำลายวัตถุ และเป็นการวัดอุณหภูมิแบบพื้นที่ ส่วนประกอบสำคัญของกล้องถ่ายภาพความร้อน ประกอบด้วย เลนส์ (lens) ตัวตรวจจับรังสีอินฟราเรด (Infrared detector) หรือเซนเซอร์ชนิดอินฟราเรด (Infrared sensor) วงจรอิเล็กทรอนิกส์ (Electronic circuit) และส่วนแสดงผล (display)

หลักการทำงานมีดังนี้ ตัวตรวจจับรังสีอินฟราเรด ทำหน้าที่รับรังสีอินฟราเรด (Infrared) ที่แผ่ออกจากวัตถุเป้าหมาย (Target) ผ่านเลนส์ของเครื่องมือวัด (Instrument) แล้วแปลงรังสีอินฟราเรดเหล่านี้ให้อยู่ในรูปของสัญญาณทางไฟฟ้า โดยรังสีอินฟราเรดที่ตัวตรวจจับรับไปนั้นประกอบด้วยรังสีที่วัตถุเป้าหมายแผ่ ออกมารวมกับรังสีที่แผ่จากวัตถุอื่นหรือจากสิ่งแวดล้อมสะท้อนออกจากผิวของ วัตถุเป้าหมาย (ตามทฤษฎีการแผ่รังสีความร้อน: Theory of thermal radiation) จากนั้นวงจรอิเล็กทรอนิกส์ จะทำหน้าที่แปลงข้อมูลที่รับมาจากตัวตรวจจับและนำไปแสดงที่ตัวแสดงผล ซึ่งอาจแสดงผลออกมาในรูปแบบของตัวเลข สี หรือกราฟ หรือทั้ง 3 รูปแบบ

กล้องถ่ายภาพความร้อนประกอบด้วยเซนเซอร์ชนิดอินฟราเรดหลายตัว แต่ละตัวแสดงผลออกมาในรูปแบบของสีที่แตกต่างกันตามอุณหภูมิของจุดนั้น ๆ โดยทั่วไป สีแดงแสดงผลของบริเวณที่มีอุณหภูมิ

สูงและสีน้ำเงินแสดงผลของบริเวณที่มีอุณหภูมิต่ำ หรือโทนสีสว่างแสดงผลของบริเวณที่มีอุณหภูมิสูง และ โทนสีมืดแสดงผลของบริเวณที่มีอุณหภูมิต่ำ ผลของแต่ละจุดที่ได้จากเซนเซอร์แต่ละตัว เมื่อนำมารวมกัน จะประกอบขึ้นเป็นภาพ เรียกว่า "ภาพถ่ายความร้อน (Thermal image)" ซึ่งการแปลงรังสีอินฟราเรดที่ เซนเซอร์ตรวจจับได้ให้อยู่ในหน่วยของอุณหภูมิอาศัยกฎของ Planck (Planck's law) และ กฎของ Stefan-Boltzmann (Stefan-Boltzman's law) ลักษณะการทำงานแสดงดังรูปที่ 2.10



รูปที่ 2.10 ระบบการวัดอุณหภูมิโดยอาศัยการแผ่รังสีอินฟราเรดของวัตถุ [13]

ความถูกต้อง (Accuracy) ของอุณหภูมิที่วัดได้จากกล้องถ่ายภาพความร้อนขึ้นอยู่กับ

- ชนิดและลักษณะพื้นผิวของวัตถุเป้าหมายหรือวัตถุที่ต้องการวัดอุณหภูมิ

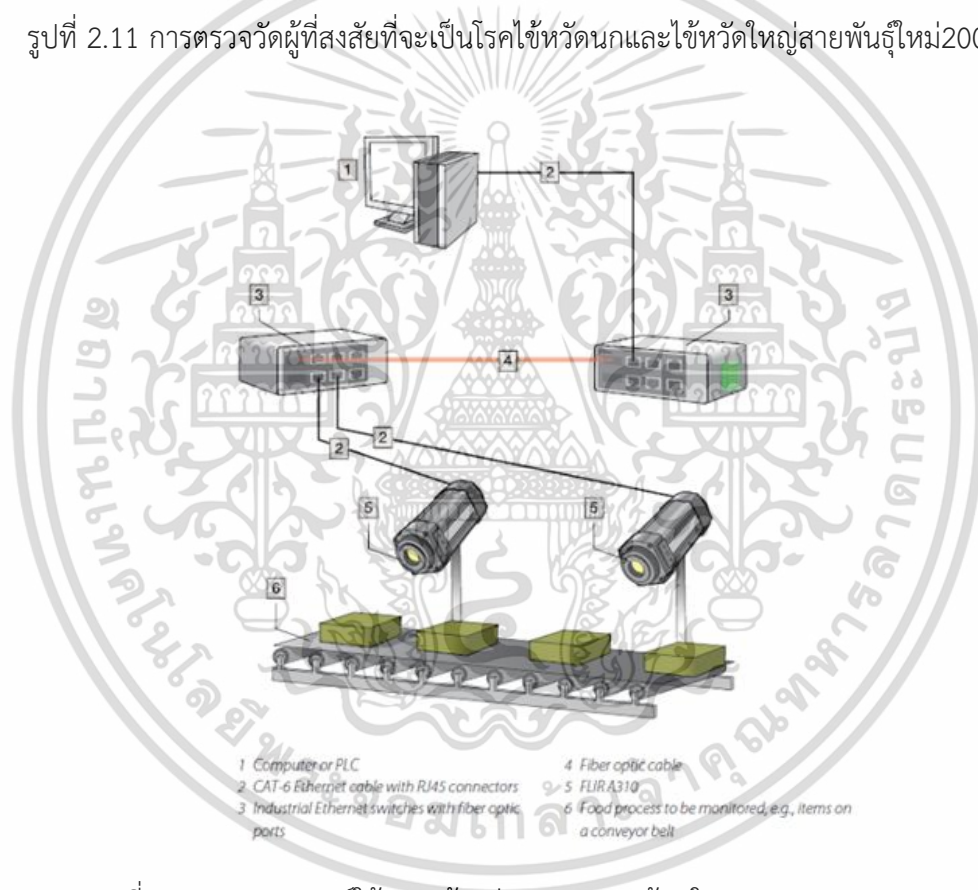
- ระยะห่างระหว่างเครื่องมือวัดและวัตถุเป้าหมาย โดยค่าความคลาดเคลื่อน (Error) อาจเกิดจากการเคลื่อนที่ของรังสีผ่านตัวกลาง เช่น อากาศที่มี ไอ ควัน ก๊าซหรือฝุ่นละอองกระจายอยู่ สิ่งต่าง ๆ เหล่านี้ดูดซับพลังงานบางส่วนจากรังสีก่อนถึงตัวเครื่องมือวัด ทำให้พลังงานที่เซนเซอร์ (Sensor) ตรวจจับได้มีค่าลดลง ปัจจัยที่ทำให้ค่าที่วัดได้คลาดเคลื่อนได้แก่พื้นที่การวัด หรือ ค่าขอบเขตการมองเห็นวัตถุ (Field of view : FOV) ของกล้องถ่ายภาพความร้อน

ตัวอย่างการประยุกต์ใช้งานกล้องถ่ายภาพความร้อน

การใช้งานของกล้องถ่ายภาพความร้อนนั้นมีมากมายทางด้านต่าง ๆ ไม่ว่าจะเป็น การตรวจวัด ใช้, จับความร้อนในอุตสาหกรรมอาหาร, อุตสาหกรรมอื่น ๆ, รวมถึงตรวจสอบการรั่วไหลของปิโตรเคมี และ ดูความร้อนในงานเครื่องกลต่าง ๆ ดังรูปที่ 2.11 ถึง 2.16



รูปที่ 2.11 การตรวจวัดผู้ที่สงสัยที่จะเป็นโรคไข้หวัดนกและไข้หวัดใหญ่สายพันธุ์ใหม่ 2009 [12]



รูปที่ 2.12 การประยุกต์ใช้งานกล้องถ่ายภาพความร้อนในอุตสาหกรรมอาหาร [13]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 การประยุกต์ใช้งานกล้องถ่ายภาพความร้อนในอุตสาหกรรมอื่นๆ [13]

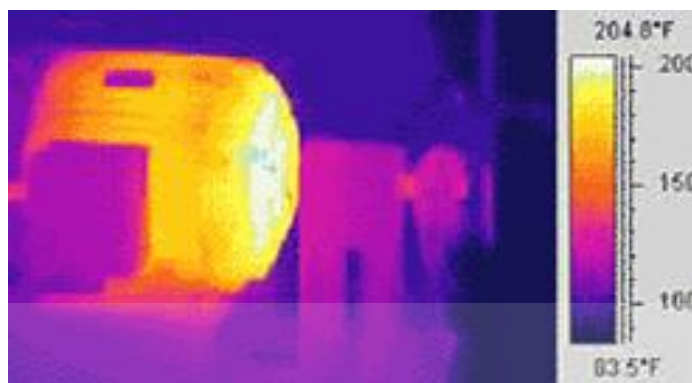


รูปที่ 2.14 การประยุกต์ใช้งานกล้องอินฟราเรดกับการตรวจสอบระบบไฟฟ้า [12]



รูปที่ 2.15 การประยุกต์ใช้งานกล้องอินฟราเรดกับอุตสาหกรรมปิโตรเคมี [12]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

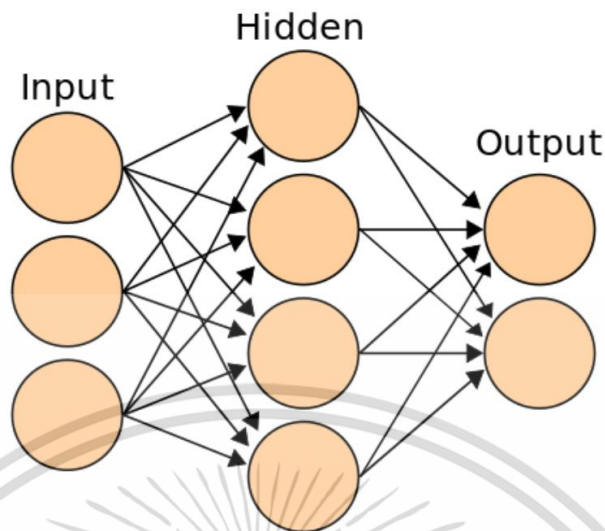


รูปที่ 2.16 การประยุกต์ใช้งานกล้องอินฟราเรดในงานเครื่องกล [12]

2.4 โครงข่ายใยประสาทเสมือน

โครงข่ายใยประสาทเสมือน (Artificial Neuron Networks) [14] คือระบบคอมพิวเตอร์จากโมเดลทางคณิตศาสตร์ เพื่อจำลองการทำงานของสมองของสัตว์ โครงข่ายใยประสาทเสมือนสามารถเรียนรู้ที่จะทำงานที่มอบหมายได้ จากการเรียนรู้ผ่านตัวอย่าง โดยไม่ถูกโปรแกรมด้วยกฎเกณฑ์ตายตัวแบบระบบอัตโนมัติ ยกตัวอย่างเช่น ในการประมวลผลภาพ คอมพิวเตอร์ที่ทำงานด้วยระบบโครงข่ายใยประสาทเทียมจะเรียนรู้การจำแนกรูปภาพแมวได้จากการให้ตัวอย่างรูปภาพที่กำกับโดยผู้เขียนโปรแกรมว่า “เป็นแมว” หรือ “ไม่เป็นแมว” จากนั้นนำผลลัพธ์ที่ได้ไปใช้ระบุภาพแมวในตัวอย่างรูปภาพอื่น ๆ โปรแกรมโครงข่ายใยประสาทเทียมสามารถแยกแยะรูปภาพแมวได้โดยปราศจากการความรู้ก่อนหน้าว่า “แมว” คืออะไร (อาทิ แมวมีขน มีหูแหลม มีเขี้ยว มีหาง) แทนที่จะใช้ความรู้ดังกล่าว โครงข่ายใยประสาทเทียมทำการระบุตัวแมวโดยอัตโนมัติด้วยการระบุลักษณะเฉพาะ จากชุดตัวอย่างที่เคยได้ประมวลผล

การประมวลผลต่าง ๆ ของโครงข่ายใยประสาทเสมือนเกิดขึ้นในหน่วยประมวลผลย่อย เรียกว่า โหนด (node) ซึ่งโหนดเป็นการจำลองลักษณะการทำงานมาจากเซลล์การส่งสัญญาณ ระหว่างโหนดที่เชื่อมต่อกัน จำลองมาจากการเชื่อมต่อของใยประสาท และแกนประสาทในระบบประสาทของสมองมนุษย์ภายในโหนด จุดเชื่อมต่อแต่ละจุด มีความคล้ายคลึงกับจุดประสานประสาท (Synapses) ในสมอง มีความสามารถในการส่งสัญญาณไปยังเซลล์ประสาทเซลล์อื่น ๆ ที่เชื่อมต่อกับมันได้ เป็นดังรูปที่ 2.17



รูปที่ 2.17 แสดงโครงข่ายประสาทเทียมที่มีการเชื่อมต่อกันผ่านกลุ่มโหนด [14]

2.4.1 โครงข่ายประสาทเทียมคอนโวลูชัน

โครงข่ายประสาทเทียมคอนโวลูชัน (Convolutional Neural Network: CNN) [15][16] ไอเดียหลักของโครงข่ายประสาทเทียมคอนโวลูชันคือการใช้ Layer ชนิดพิเศษ ที่เรียกว่า Convolution layer ซึ่งทำหน้าที่สกัดเอาส่วนต่างๆ ของภาพออกมา เช่น เส้นขอบของวัตถุต่างๆ เพื่อให้โมเดลสามารถเรียนรู้ลักษณะของภาพได้อย่างมีประสิทธิภาพและแม่นยำ โครงข่ายประสาทเทียมคอนโวลูชันจะใช้ Convolution layer มาประกอบกับ Layer ชนิดอื่น เช่น Pooling layer แล้วนำกลุ่ม Layer ดังกล่าวมาซ้อนต่อกัน โดยอาจเปลี่ยน Hyperparameter บางอย่าง เช่นขนาดของ Filter layer (ซึ่งเป็นส่วนหนึ่งของ Convolution layer) และจำนวน Channel ของ layer วิธีการนำเอาส่วนต่างๆ มาประกอบกันนี้เรียกว่าเป็นโครงสร้าง (Architecture) ของโครงข่ายประสาทเทียมคอนโวลูชันซึ่งมีหลายแบบ เช่น LeNet, AlexNet, VGG, ResNet, Inception Network เป็นต้น

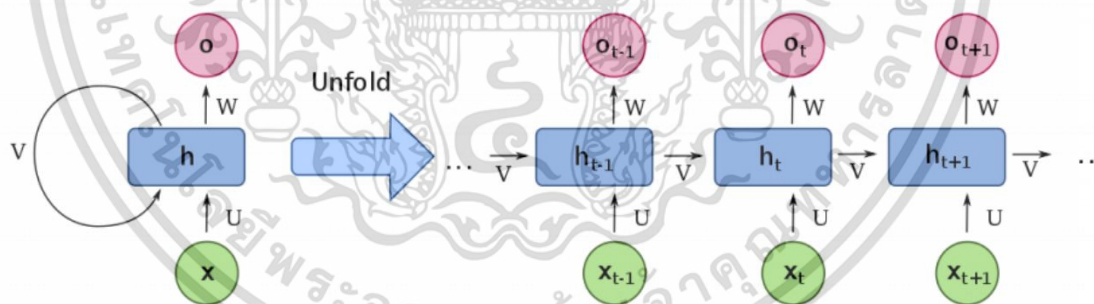
โครงข่ายประสาทเทียมคอนโวลูชันถูกนำมาใช้ในรูปแบบโมเดลที่สามารถเรียนรู้และเลือกใช้ลักษณะเด่น (Feature extraction) ของรูปภาพได้ด้วยตัวเองข้อดีคือ โมเดลจะเรียนรู้และเลือกลักษณะเด่นของภาพได้ดีกว่ามนุษย์ดังนั้นจะทำให้ ได้การประมวลผลที่แม่นยำมากกว่า ข้อแตกต่างจากโครงข่ายใยประสาทเทียม คือ คำนวณที่ใช้เป็นค่าชุดเดียวกันหมด ซึ่งเป็นหนึ่งเหตุผลว่าทำไมถึงนักวิจัยจึงเลือกใช้ โครงข่ายประสาทเทียมคอนโวลูชัน เพราะถ้าใช้โครงข่ายใยประสาทเทียมปกตินั้นโมเดลจะต้องเรียนรู้ค่าน้ำหนักจำนวนมากโดยใช้วิธีการแพร์กลับ โดยข้อมูลเข้าของโครงข่ายประสาทเทียมคอนโวลูชันจะเป็นเมทริกซ์จากการแปลงมาจากรูปภาพ

2.4.2 โครงข่ายประสาทเทียมแบบวนกลับ

โครงข่ายประสาทเทียมแบบวนกลับ (Recurrent Neural Network: RNN) [15] เป็นวิธีการที่ถูกนำมาใช้ในการวิจัยเกี่ยวกับการรู้จำเสียง (Speech recognition) และการประมวลผลภาษาธรรมชาติ (Natural Language Processing) การทำงานของ RNN คือการเอาผลลัพธ์ที่ได้จากการคำนวณย้อนกลับมาใช้เป็นข้อมูลขาเข้าอีกครั้งซึ่งมีประโยชน์อย่างมากในข้อมูลที่มีความต่อเนื่องเช่น ข้อมูลเสียงข้อความหรือแม้แต่รูปภาพเองก็ตาม

โครงข่ายประสาทเทียมแบบวนกลับเป็นฟังก์ชันที่รับข้อมูลเข้าและประมวลผลออกมาเป็นข้อมูลออกโดยโครงข่ายประสาทเทียมจะมองข้อมูลเข้าแต่ละตัวแยกกันชัดเจน ไม่ขึ้นต่อกัน และข้อมูลเข้าที่รับเข้าไปจะมีการเรียงลำดับอย่างไรก็ได้ ซึ่งโครงข่ายประสาทเทียมจะประมวลผลให้ข้อมูลออกมาเหมือนเดิมตามข้อมูลเข้า ซึ่งโครงข่ายประสาทเทียมจะมีปัญหาเกี่ยวกับข้อมูลที่เป็นลำดับ เช่น ข้อความ (ลำดับตัวอักษร), เสียง (ลำดับแรงดันอากาศ), วิดีโอ (ลำดับของภาพและเสียง) และข้อมูลที่เป็นแบบ Time Series ต่าง ๆ เป็นต้น

โครงข่ายประสาทเทียมแบบวนกลับถูกออกแบบมาเพื่อแก้ปัญหาสำหรับงานที่มีข้อมูลที่มีลำดับ โดยใช้หลักการนำสถานะภายในของโมเดล กลับมาเป็นข้อมูลขาเข้าใหม่คู่กับข้อมูลขาเข้าแบบปกติ เรียกว่าสถานะซ่อน (Hidden State) หรือสถานะภายใน (Internal State) ช่วยให้โมเดลจำรูปแบบ (Pattern) ของลำดับข้อมูลเข้า (Input Sequence) ได้ซึ่งโครงสร้างของโครงข่ายประสาทเทียมแบบวนกลับเป็นดังรูปที่ 2.18



รูปที่ 2.18 โครงสร้างของโครงข่ายประสาทเทียมแบบวนกลับ [15]

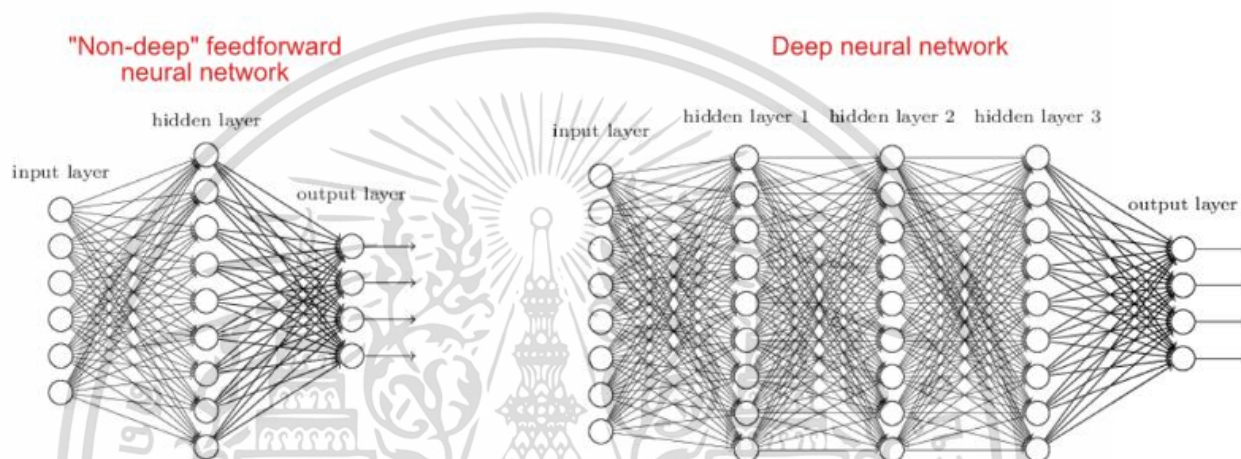
2.5 การเรียนรู้เชิงลึก (Deep Learning)

การเรียนรู้เชิงลึก [17] [18] [19] คือชุดคำสั่ง (Algorithm) ที่ถูกสร้างขึ้นมาเพื่อการเรียนรู้ของเครื่องจักรหรือเครื่องคอมพิวเตอร์ โดยชุดคำสั่งนี้จะทำให้ตัวเครื่องจักรสามารถประมวลผลข้อมูลจำนวนมาก ด้วยการจำลองโครงข่ายประสาทแบบเดียวกับในสมองของมนุษย์ เป็นวิธีการหนึ่งของ Machine Learning ที่พยายามเรียนรู้วิธีการแทนข้อมูลอย่างมีประสิทธิภาพ เช่น รูปภาพหนึ่งสามารถแทนได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวกเตอร์ของความสว่างต่อจุดพิกเซล หรือมองในระดับสูงขึ้นไป เป็นเซตของขอบของวัตถุต่างๆ หรือมองว่าเป็นพื้นที่ของรูปร่างใดๆก็ได้ การแทนความหมายดังกล่าวจะทำให้การเรียนรู้ที่จะทำงานต่างๆทำได้ง่ายขึ้น ไม่ว่าจะเป็นการเรียนรู้จดจำใบหน้าหรือการรู้จำการแสดงออกทางสีหน้า

ซึ่งจริงๆ แล้วการเรียนรู้เชิงลึกก็คือ โครงข่ายประสาทเทียม (Artificial Neural Networks) โดยเทคโนโลยีการเรียนรู้เชิงลึก และ โครงข่ายประสาทเทียมเป็นอัลกอริทึมที่ถูกสร้างขึ้นมาเพื่อ Machine Learning แต่ความแตกต่างระหว่างการเรียนรู้เชิงลึกกับโครงข่ายประสาทเทียม ก็คือระดับ hidden layer ที่ในการเรียนรู้เชิงลึกนั้นมี hidden layer มากกว่าในโครงข่ายประสาทเทียม

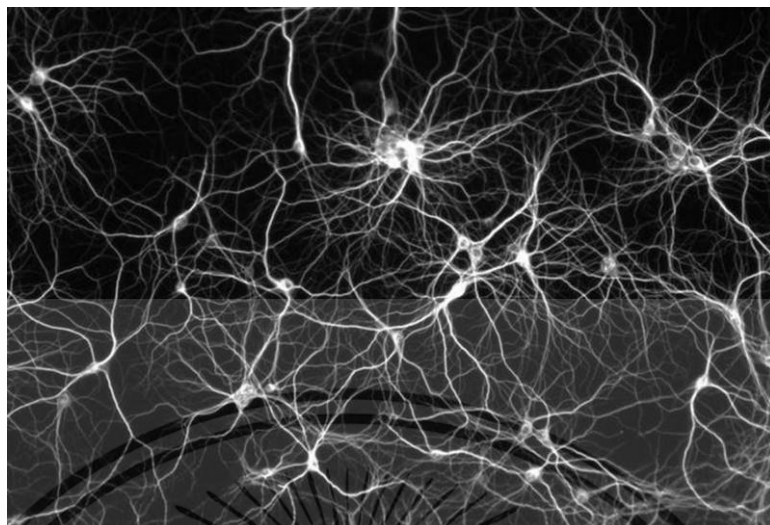


รูปที่ 2.19 ความแตกต่างระหว่าง Non-deep และ Deep neural network [19]

โดยโครงข่ายประสาทเทียม (Artificial Neural Networks) นั้นอาศัยแนวคิดและเทคนิคจากการทำงานของระบบโครงข่ายประสาทในระบบประสาทของมนุษย์ โดยจำลองการทำงานเหมือนกับกลุ่มเซลล์ประสาทที่เชื่อมโยงกันเป็นระบบประสาทที่สามารถรับรู้หลายๆ สิ่งในเวลาเดียวกัน ด้วยการประมวลผลแบบขนาน (Parallel Network) ทำให้ระบบสามารถตัดสินใจได้ใกล้เคียงกับมนุษย์

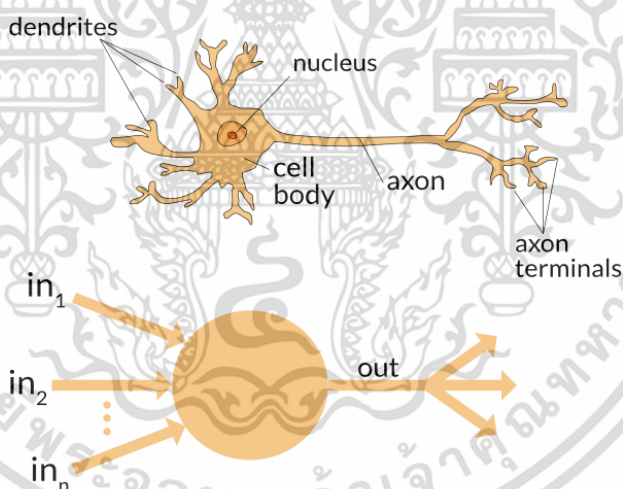
โดยหลักการของการเรียนรู้เชิงลึก ก็จะเป็นโครงข่ายประสาทเทียมที่เป็นโหนดหลายๆชั้น และจะใช้การประมวลผลแบบขนาน (Parallel Processing) ทำให้สามารถประมวลผลได้ครั้งละจำนวนมาก ช่วยให้ Machine Learning สามารถให้ผลลัพธ์ในการตัดสินใจและคาดการณ์ได้ดีมากยิ่งขึ้น เปรียบได้กับการสร้างสัญญาณการมองเห็น ที่ประกอบไปด้วยหน่วยเซลล์ต่างๆ เช่น ในเรติน่าที่เป็นตัวรับและแปลสัญญาณแสงให้กลายเป็นสัญญาณประสาทหรือกระแสประสาท ซึ่งในเรติน่าจะประกอบไปด้วย เซลล์รูปแท่ง (Rod cell) และเซลล์รูปกรวย (Cone cell) เซลล์รูปแท่งทำงานในที่มืดแสงสลัวและทำให้เกิดการเป็นรูปตัวขาวดำ ส่วนเซลล์รูปกรวยแท่งทำให้เกิดการเห็นในช่วงกลางวันเป็นรูปสี เป็นต้น

หรือเรียกได้ว่าเป็นศาสตร์แขนงหนึ่งของ Machine Learning ที่เลียนแบบระบบเซลล์ประสาทในสมองของมนุษย์ (Neural Network) ดังนั้นความสามารถของมัน ในอนาคตอาจจะเหนือมนุษย์ เนื่องจากสามารถเพิ่มพลังประมวลผลได้ไม่จำกัด



รูปที่ 2.20 การเชื่อมต่อกันของเซลล์ประสาท [17]

ถ้าลองซูมเข้าไปที่เซลล์ประสาท แต่ละตัว จะสังเกตว่า มันเชื่อมต่อกับเส้นประสาทหลายๆเส้น
ดังรูป



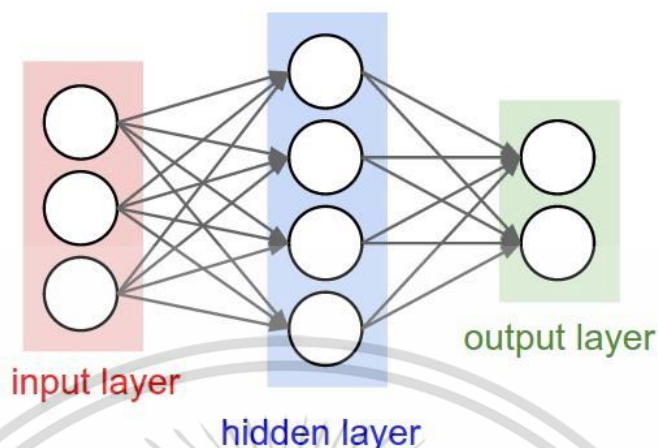
รูปที่ 2.21 การเชื่อมต่อของเซลล์ประสาท [17]

สามารถจำลองมันขึ้นมาได้ โดยมี เส้นประสาทและเซลล์ประสาท (Node) ออกมาเป็นโครงข่าย
ใยประสาทเสมือนหากคิดภาพการทำงานของดวงตาและสมอง “เมื่อดวงตาเห็นรูปร่าง รูปนอ้งหมา
ถูกนำไปประมวลผล (ซึ่งประมวลอย่างไรไม่รู้) และได้คำตอบว่า นี่คือนอ้งหมา”

ถ้าสร้างโครงข่ายใยประสาทเสมือนเพื่อจำลองจุดนี้ จะได้ 3 ส่วน

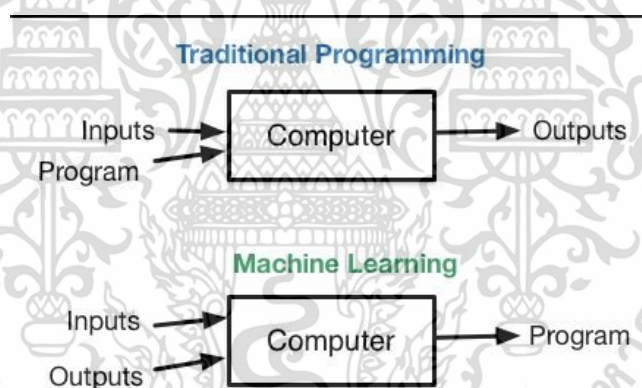
- สิ่งที่เราเห็นก็คือ input — ข้อมูลที่รับ ในที่นี้ก็คือ นอ้งหมาที่เห็น
- สิ่งในระบบเซลล์ประสาทในสมองประมวลผล เทียบได้กับ hidden layer
- สิ่งที่สมองตอบหลังประมวลผลว่านี่คือนอ้งหมา คือ output layer — ผลลัพธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 โครงข่ายใยประสาทเสมือน [17]

ดังนั้น สมอคนเปรียบได้ตั้ง Hidden Layer เชื่อมต่อระหว่าง input และ output จึงสามารถจำลองมันขึ้นมาได้ด้วยหลักการของ Machine Learning



รูปที่ 2.23 Traditional Programming & Machine learning [17]

โดยปกติแล้ว จะเขียนโปรแกรมเพื่อให้ รับ input เข้าไปแล้วได้คำตอบ output

$$\text{input} + \text{programming} = \text{output}$$

แต่หลักการของ Machine Learning คือ การใส่ Input และ Output เข้าไปเพื่อให้ได้โปรแกรม ไม่ได้ใส่โปรแกรมเข้าไปให้มันบอกคำตอบ เหมือนกลับข้างสมการ

$$\text{input} + \text{output} = \text{programming}$$

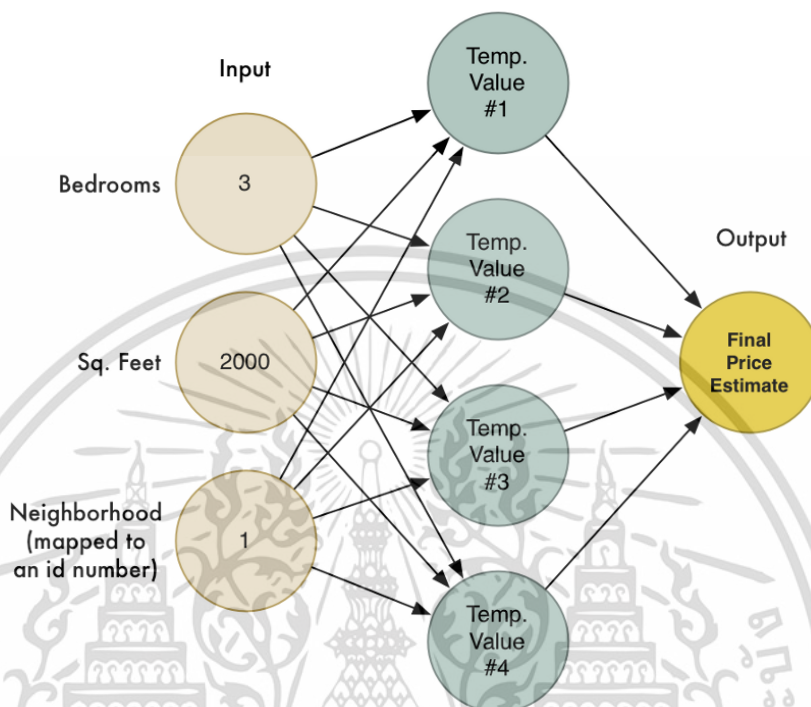
เมื่อใส่ input output เข้าไปแล้ว โมเดลก็เป็นการคิดกลับนั่นเองว่า ทำอย่างไรให้ได้คำตอบนี้

“ดังนั้น เราก็ไม่จำเป็นต้องรู้ว่าสมอทำงานอย่างไร เรามัวมันขึ้นมาอย่างมีหลักการ”

เพียงแค่ จำลองมันขึ้น ตาม concept ที่กล่าวไปด้านบน ใส่ Input และ Output เข้าไป โมเดลจะถูกทดลอง Trial and Error หาตัวเลขที่เหมาะสมสำหรับแต่ละ Node และความสัมพันธ์ของแต่ละ Node ไปเรื่อยๆ ให้มันคุณหารบวกลบกัน โดยฝั่งสมการไว้ในแต่ละวงกลม รูปที่ 2.24 จะเห็นภาพมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ว่าวงกลมชั้น Hidden Layer รับข้อมูลจาก Input มาคำนวณได้ค่าค่าหนึ่ง แล้วเก็บไว้เป็น Temp. Value ก่อนจะถูกส่งไปคิดออกมาเป็นคำตอบอีกที

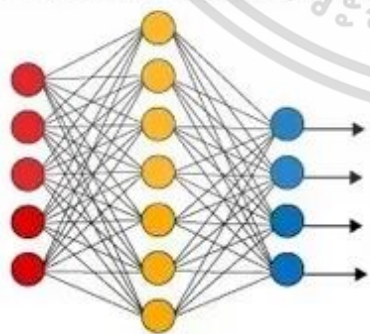


รูปที่ 2.24 ตัวอย่างของการเรียนรู้เชิงลึก [17]

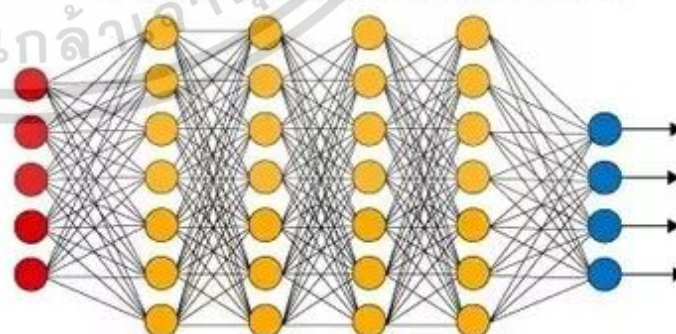
แต่ในชีวิตจริง สมองคนซับซ้อนกว่านี้มาก ดังนั้นจะต้องเพิ่ม Hidden Layer เข้าไปเยอะๆ ให้มันคิดต่อกันไปเรื่อยๆ เรียกว่าการเรียนรู้เชิงลึก

การเรียนรู้เชิงลึก คือ โครงข่ายประสาทเสมือนที่มี Hidden Layer หลายชั้นนั่นเอง เพื่อความสามารถในการคิดที่มากกว่าปกติ และสะท้อนสมองคนได้ดีขึ้น

Simple Neural Network



Deep Learning Neural Network

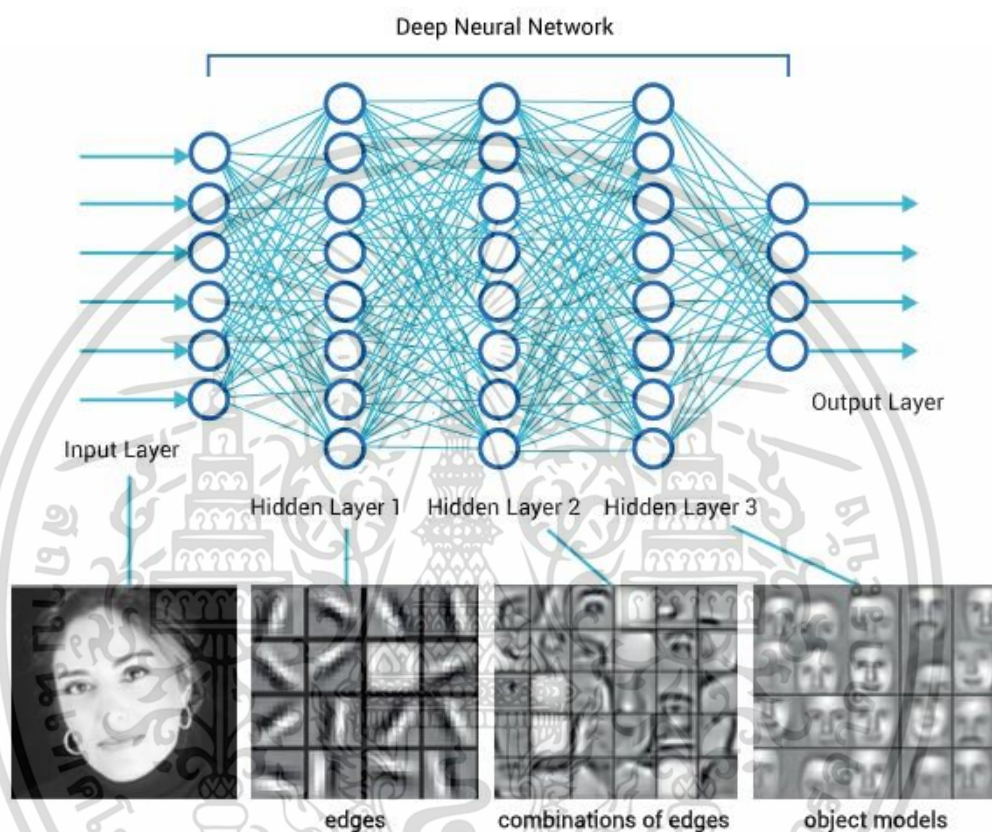


● Input Layer ● Hidden Layer ● Output Layer

รูปที่ 2.25 การเรียนรู้เชิงลึกเมื่อมีหลาย Hidden Layer [17]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อการเรียนรู้เชิงลึก มีหลาย Hidden Layer ทำให้มันสามารถคำนวณอะไรที่ซับซ้อนได้เยอะมาก และยังสามารถใช้เทคนิคต่างๆได้มากขึ้นด้วย ที่สำคัญที่สุดคือ มันค่อยๆ คิดอย่างเป็นขั้นเป็นตอนได้ ดังรูปที่ 2.26



รูปที่ 2.26 การทำงานของการเรียนรู้เชิงลึก [19]

2.5.1 ประโยชน์ของการเรียนรู้เชิงลึก [19]

เทคโนโลยีการเรียนรู้เชิงลึกทำให้ซอฟต์แวร์ด้านการจดจำภาพและเสียงของเครื่องจักรพัฒนาขึ้นมาก จนสามารถนำไปต่อยอดทำเป็นแอปพลิเคชัน อุปกรณ์ หรือบริการออนไลน์ ที่สามารถเลียนแบบความเป็นมนุษย์ มองมนุษย์คิดอะไรได้ จดจำอะไรได้ แยกแยะอะไรได้ การเรียนรู้เชิงลึกก็จำทำได้ไม่ต่างจากสมองของมนุษย์ สิ่งที่จะได้จากการเรียนรู้เชิงลึกได้แก่

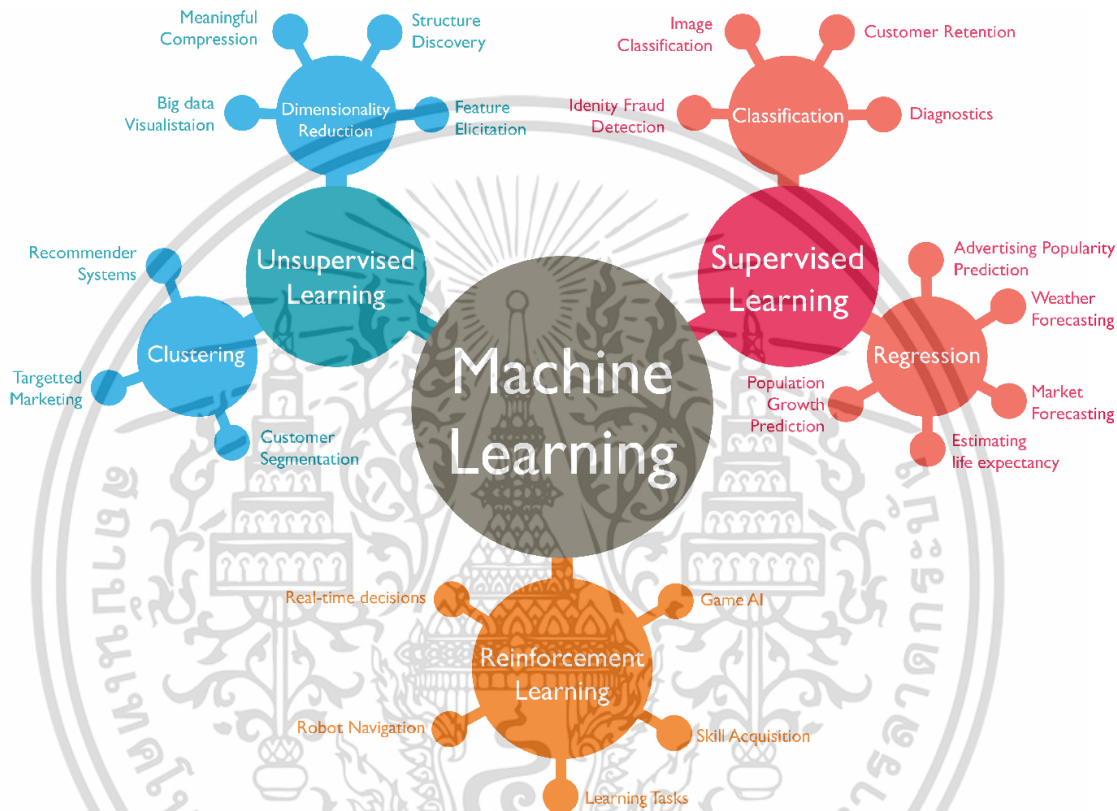
- การแยกคน : เป็นการแยกแยะใบหน้าของแต่ละคน เช่น การติดแท็กใน Google+ และ Facebook
- การแยกวัตถุที่ไม่ใช่คน : เป็นการแยกแยะสิ่งมีชีวิต เช่น แยกแมวออกจากสุนัข การแยกสิ่งไม่มีชีวิต เช่น แยกโต๊ะออกจากเก้าอี้
- แยกเสียง : เป็นการแยกแยะสำเนียงภาษาพูดให้เป็นภาษาเขียนและแปลภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แยกการแต่งกาย : เป็นการแยกได้ว่าใครแต่งตัวในสไตล์เดียวกันบ้าง แล้วเสนอข้อมูลว่ามีเสื้อผ้าแนวไหนอยู่ที่บ้าน

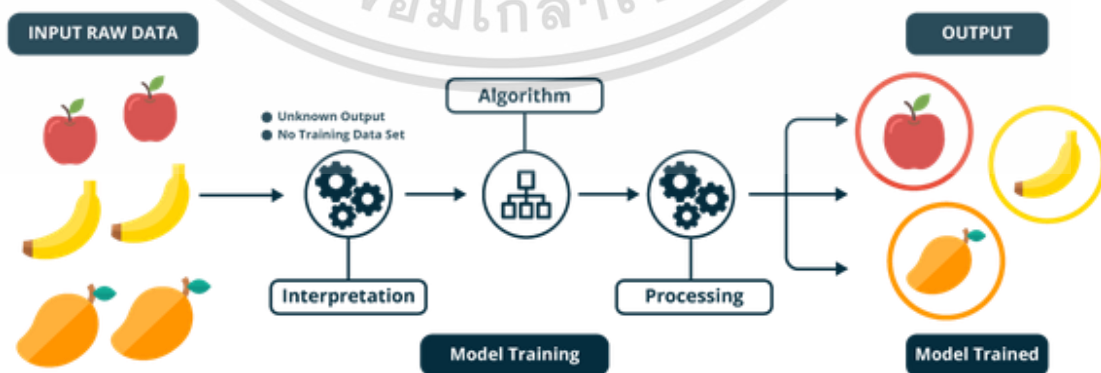
ประเภทของ Machine Learning[20]

สามารถแบ่งออกได้เป็น 3 กลุ่มใหญ่ๆคือ supervised learning , unsupervised learning และ reinforcement learning



รูปที่ 2.27 ประเภทของ Machine Learning [20]

1) เรียนรู้โดยมีข้อมูลมาสอน (Supervised learning)



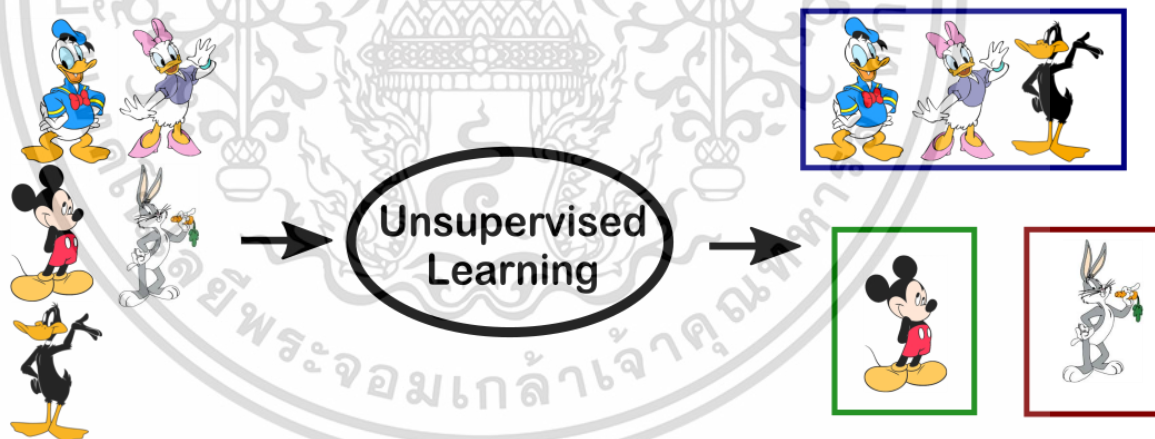
รูปที่ 2.28 การทำงานของ Supervised learning [20]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

supervised learning เป็นกลุ่มของอัลกอริทึมที่เน้นสอนคอมพิวเตอร์โดยการศึกษาจากข้อมูล ตัวอย่าง เช่น ต้องการให้คอมพิวเตอร์ แยกภาพหมาออกจากภาพสัตว์ชนิดอื่น ๆ ต้องมีข้อมูลภาพตัวอย่างของหมาป้อนให้คอมพิวเตอร์รู้ว่าถ้ารูปภาพมีลักษณะดังนี้แสดงว่าเป็นหมา หรือ ถ้าต้องการให้คอมพิวเตอร์ประมวลผลว่าคนที่มาขอสินเชื่อควรได้รับการอนุมัติหรือไม่ ต้องทำการหากลุ่มตัวอย่างเพื่อมาสอนคอมพิวเตอร์ว่าคนที่มีลักษณะแบบนี้ มีประวัติการเงินแบบนี้ควรอนุมัติหรือไม่ ถ้ามีประวัติการเงินแบบนี้ไม่ควรอนุมัติ สรุปคือมีรูปแบบการประเมินชัดเจนว่าถ้าทำงานแล้วผลลัพธ์ได้แบบนี้คือถูกหรือผิด อ้างอิงจากตัวอย่างที่ได้ทำการสอนไป ซึ่งจะเรียกผลลัพธ์ของตัวอย่างที่ใช้ในการสอนว่า label

โดยส่วนใหญ่ในการใช้ supervised learning ในชีวิตจริงมักถูกนำไปใช้แทนการทำงานแบบ rule base คือมีกฎหรือรูปแบบการทำงานที่ตายตัวหรือสามารถอธิบายเหตุผลออกมาได้อย่างชัดเจน ตัวอย่างเช่นในกรณีการขอสินเชื่อ โดยปกติแล้วแต่ละธนาคารจะมีข้อกำหนดหลักๆ ในการพิจารณาอยู่แล้วว่าถ้ามีประวัติการเงินแบบนี้ อายุประมาณนี้ เงินเดือนแบบนี้ จะสามารถปล่อยสินเชื่อในวงเงินได้ไม่เกินเท่าไร จริงๆถ้าเรามองดีๆจะพบว่าได้มีการใช้ supervised learning กันมาอย่างยาวนาน อย่างเวลาเด็กจบใหม่เข้าทำงานที่สาขาของธนาคารในส่วนงานที่เป็นงานลักษณะ rule base พนักงานธนาคารหน้าเคาน์เตอร์ พวกที่ทำงานมาก่อนหรือ senior ก็จะต้องมาอธิบายหรือสอนว่าถ้ามีลูกค้ามายื่นคำร้องว่าจะฝากเงินหรือถอนเงินจะต้องทำขั้นตอนตามนี้ 1 2 3 ตามนี้ถ้ากรอกข้อมูลแล้วได้ผลลัพธ์แบบนี้ถือว่าการฝากหรือถอนนั้นเสร็จสมบูรณ์ นี่คือนำตัวอย่างหนึ่งของการทำงานแบบ supervised learning จากคนสู่คน

2) เรียนรู้โดยไม่มีข้อมูลสอน (Unsupervised learning)



รูปที่ 2.29 แสดงการทำงานของ Unsupervised learning [20]

unsupervised learning เป็นกลุ่มอัลกอริทึมที่ไม่มี label หรือการสอนอย่างชัดเจนว่าถ้าทำงานแล้วได้ผลลัพธ์แบบนี้หมายถึงถูกหรือผิด ตัวอย่างเช่น ถ้ามีคน 100 คนในห้องแล้วทำการแบ่งกลุ่ม 100 คนนั้นเป็น 2 กลุ่ม ในกรณีนี้ไม่มีกฎชัดเจนว่าต้องแบ่งตามเพศ, อายุ, สีเสื้อหรืออื่นๆ แคแบ่งออกมาให้ได้ 2 กลุ่มซึ่งอาจจะแบ่งว่าใครอยู่กลุ่มห้องส่วนหน้าคือกลุ่ม 1 ส่วนใครอยู่ครึ่งห้องหลังคือกลุ่ม 2

จะเห็นได้ว่าการทำงานแบบ unsupervised learning จะเป็นการทำงานแบบไม่มี label หรือ กฎที่ตายตัว จากตัวอย่าง 2 ตัวอย่างข้างต้นจะเห็นว่าทั้งการแบ่งคนในห้องเป็น 2 กลุ่มหรือการลดรูป ข้อมูลจาก 24 columns มาเป็น 2 columns เกิดจากการกำหนดตัวเลขขึ้นมาเองตามที่คุณสร้างเห็นว่า เหมาะสมทั้งสิ้น

3) การเรียนรู้ตามสภาพแวดล้อม (Reinforcement learning)



รูปที่ 2.30 แสดงการทำงานของ Reinforcement learning [20]

Reinforcement learning เป็นกลุ่มอัลกอริทึมที่มีรูปแบบที่ซับซ้อนกว่า 2 แบบแรกมาก โดยหลักของ Reinforcement learning จะเป็นรูปแบบของการประมวลผล action ในทุก ๆ การเคลื่อนไหวของคอมพิวเตอร์อย่างเช่น Alphago ที่มีการคำนวณทางเลือกในแต่ละครั้งที่เดินหมาก base จากข้อมูลที่เคยเรียนรู้มา หรือ การทำงานของ robot ที่ประมวลผลเลือกกระทำ action จากสภาพแวดล้อมที่เปลี่ยนไป

ตามหลักของ Reinforcement learning จะมีการ set goal หรือ reward ของคอมพิวเตอร์เอาไว้ว่าต้องการจะให้คอมพิวเตอร์ตัวนี้ทำงานอะไรหรือแต่ละ action มีค่า reward เท่าไร จากนั้นก็ทำการให้มันเรียนรู้ตามสภาพแวดล้อมซึ่งอาจจะมาจาก sensor หรือ input ในรูปแบบต่างๆ ถ้าเกิด input เข้ามาแบบนี้แล้วเลือก action แบบนี้จะทำให้เข้าใกล้ goal หรือได้รับ reward เป็นค่าเท่าไร เพื่อหา base choice ของ action ตามสถานการณ์นั้นๆ

ตัวอย่างง่ายๆในชีวิตประจำวันที่น่าจะใกล้เคียงกับการทำงานของ reinforcement learning คือ ให้นักภาพวาดเดินเข้าห้องในบ้านตอนกลางคืนที่มีดสนิทเพื่อเดินไปเปิดไฟ โดยเปรียบเทียบการเปิดไฟเป็น goal ส่วนมือและขาเป็น sensor เพื่อรับ input จากสิ่งที่อยู่รอบตัวที่เดินก้าวเข้าไปในห้อง ทุกๆก้าวที่เดินเข้าไปมือและขาจะสัมผัสสิ่งต่างๆในห้องเพื่อทำให้รู้ว่าเดินถูกทางที่นำไปยัง switch ไฟแล้วหรือไม่ จากนั้นก็ประมวลผล action ในก้าวถัดไปว่าควรเดินต่อหรือเปลี่ยนทิศทาง

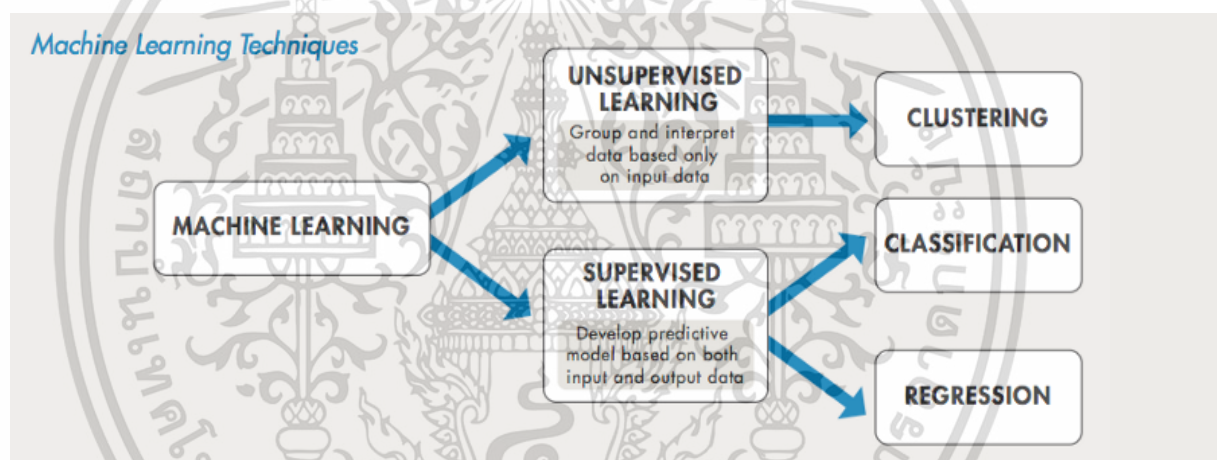
Supervised Vs Unsupervised Learning [21]

โดยจากรูปที่ 2.31 จะเป็นการอธิบายว่าความแตกต่างของ Supervised และ Unsupervised Learning คืออะไร มีการใช้งานแตกต่างกันอย่างไร

การเรียนรู้โดยมีข้อมูลมาสอน (Supervised learning) ในการเรียนรู้แบบนี้การทำนายของโมเดลจะถูกสร้างโดยอิงจากข้อมูลชุดแรกพร้อมทั้งผลลัพธ์ของการทำนายซึ่งการเรียนรู้แบบนี้จะแบ่งออกเป็น 2 ประเภท เมื่อแบ่งตามผลลัพธ์และการตอบสนอง คือ

- Regression จะเรียกว่าเป็น Regression ก็ต่อเมื่อผลลัพธ์หรือตัวแปรต่างๆเป็นตัวแปรต่อเนื่อง (continuous variable) เช่น เป็นจำนวน, ลำดับหรือตัวเลข
- Classification จะเรียกว่าเป็น Classification ก็ต่อเมื่อผลลัพธ์หรือตัวแปรต่างๆ เป็นตัวแปรไม่ต่อเนื่อง (Discrete variable) เช่น labels

การเรียนรู้โดยไม่มีข้อมูลมาสอน (Unsupervised learning) ในการเรียนรู้แบบนี้จะเป็นการหาโครงสร้างและรูปร่างของข้อมูลโดยจะเรียกว่าเป็นการ Clustering



รูปที่ 2.31 Machine Learning Techniques [21]

2.6 Confusion matrix

ในการวัดผลของการเรียนรู้เชิงลึกจำเป็นต้องมีการวัดประสิทธิภาพโมเดลก่อนว่าโมเดลนั้นมีประสิทธิภาพเพียงพอที่จะนำมาพัฒนา หรือนำไปใช้งานด้านต่างๆ ซึ่งการวัดประสิทธิภาพนั้นส่วนใหญ่จะวัดค่าจาก confusion matrix หรือดูจากค่าอื่น ๆ เช่น Accuracy Sensitivity เป็นต้น

Confusion matrix [22][23] คือ ตารางที่เป็น Error matrix ที่จะแสดงผลการทำงานของอัลกอริทึมต่าง ๆ ในการทำ classification

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

รูปที่ 2.32 ตัวอย่าง confusion matrix ขนาด 2x2 [23]

จากรูปภาพข้างต้นจะมีค่าต่าง ๆ ดังนี้

True Positive (TP) คือ สิ่งที่โปรแกรมทำนายว่า “จริง” และมีค่าเป็น “จริง”

True Negative (TN) คือ สิ่งที่โปรแกรมทำนายว่า “ไม่จริง” และมีค่า “ไม่จริง”

False Positive (FP) คือ สิ่งที่โปรแกรมทำนายว่า “จริง” แต่ มีค่าเป็น “ไม่จริง”

False Negative (FN) คือ สิ่งที่โปรแกรมทำนายว่า “ไม่จริง” แต่ มีค่าเป็น “จริง”

โดยทั่วไปแล้วจะมีตัววัดที่นิยมใช้กันในงานวิจัยและการทำงานต่าง ๆ อยู่ดังสมการต่อไปนี้

Sensitivity, recall, hit rate หรือ true positive rate (TPR)

$$TPR = \frac{TP}{P} = \frac{TP}{TP+FN} \quad (2.5)$$

Specificity, selectivity หรือ true negative rate (TNR)

$$TNR = \frac{TN}{N} = \frac{TN}{TN+FP} \quad (2.6)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Precision หรือ positive predictive value (PPV)

$$PPV = \frac{TP}{TP+FP} \quad (2.7)$$

Accuracy (ACC)

$$ACC = \frac{TP+TN}{P+N} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.8)$$

อีกหนึ่งวิธีที่นำมาใช้คือ F1-Score [24] ซึ่งเป็นค่าเฉลี่ยแบบ harmonic mean ระหว่าง precision และ recall F1-Score นั้นเป็น single metric ที่วัดความสามารถของโมเดล มีสมการดังนี้

$$F1 = 2 * \left(\frac{precision * recall}{precision + recall} \right) = \frac{2TP}{2TP+FP+FN} \quad (2.9)$$

โดยค่าแต่ละค่าไม่ว่าจะเป็น Precision, Recall หรือ F1-Score มีการนำไปใช้ที่แตกต่างกันตามความเหมาะสมของการทำโมเดล classification ไปใช้งาน

Precision หรือ positive predictive value (PPV) จะนำไปใช้ในกรณีที่ เกิดการทำนายผิด ว่าเป็น (Predict as positive) เช่น เมลสำคัญ ทำนายว่าเป็นสเปมเมล เป็นต้น

Sensitivity, recall, hit rate หรือ true positive rate (TPR) จะนำไปใช้ในกรณีที่ เกิดการทำนายผิด ว่าเป็น (Predict as negative) เช่น ติดเชื้อ HIV แต่ผลการทำนายบอกว่า ไม่ติดเชื้อ

F1-score เป็นค่าประสิทธิภาพโดยรวม คือ การนำค่า Precision และ Recall มาพิจารณาร่วมกัน เพื่อความแม่นยำและความครบถ้วน เช่น การใช้ค่า F1-score เพื่อวิเคราะห์หุ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเสริมการอธิบาย เพื่อให้เข้าใจมากขึ้น

		Actual	
		Spam	Ham
Prediction	Spam	20	12
	Ham	18	50

รูปที่ 2.33 ตัวอย่าง confusion matrix ทำนายอีเมลทั้งหมด N=100 [24]

ในกรณีที่เรายากทราบว่ามีเดลของเราแม่นยำขนาดไหน คือ ทายถูกต้องว่าเป็น Spam จากการพยายามทำนายทั้งหมด 100 ฉบับเราต้องใช้ Precision ก็คือ $= 20/32 = 0.625$

แต่ในกรณีที่เรายากทราบว่ามีเดลที่เราสร้างขึ้น สามารถตรวจจับ Spam ได้ถูกต้องขนาดไหน จาก Spam Email ทั้งหมด เราต้องใช้ Recall ของการตรวจจับ Spam ก็คือ $20/38 = 0.526$

ส่วนถ้าต้องการหาประสิทธิภาพของโมเดลการทำนายนี้ ที่ต้องมีทั้งการทายถูกต้องว่า email ที่เจอนั้นเป็น spam จริง ๆ และในขณะเดียวกันก็ต้องตรวจจับ spam ได้ด้วยเราก็ต้องเลือกใช้ F1-Score ซึ่งเป็นค่าเฉลี่ยทั้ง Precision และ Recall ซึ่งมีค่าเท่ากับ $2*((0.625*0.526)/(0.625+0.526)) = 0.571$

2.7 ระบบปฏิบัติการ iOS

ระบบปฏิบัติการไอโอเอส (iOS) [25] มีชื่อเดิมว่า iPhone OS เริ่มต้นด้วยการเปิดตัวของ iPhone เมื่อวันที่ 29 มิถุนายน 2550 ระบบปฏิบัติการไอโอเอส (iOS) เป็นระบบปฏิบัติการสำหรับสมาร์ตโฟน (Smartphone) ของแอปเปิล โดยเริ่มต้นพัฒนาสำหรับใช้ในโทรศัพท์ iPhone และได้พัฒนาต่อใช้สำหรับ iPod Touch และ iPad โดยระบบปฏิบัติการนี้สามารถเชื่อมต่อไปยัง AppStore สำหรับการเข้าถึงถึงแอปพลิเคชัน (Application) มากกว่า 300,000 ตัว ซึ่งมีการดาวน์โหลดไปมากกว่าห้าพันล้านครั้ง แอปเปิลได้มีการพัฒนาปรับปรุงสำหรับ iPhone, iPad และ iPod Touch ผ่านทางระบบ iTunes คือโปรแกรมฟรี สำหรับ Mac และ PC ใช้ดูหนังฟังเพลงบนคอมพิวเตอร์ รวมทั้งจัดระเบียบและ sync ทุกๆอย่าง และเป็นร้านขายความบันเทิงบนคอมพิวเตอร์, บน iPod touch, iPhone และ iPad ที่มีทุกๆอย่างสำหรับคุณ ในทุกที่และทุกเวลา พัฒนาระบบรักษาความปลอดภัยให้มีความเป็นเลิศ ซึ่งนี่คือข้อได้เปรียบ เมื่อเทียบกับคู่แข่ง

เปรียบเทียบการใช้งานระหว่างระบบปฏิบัติการ iOS กับ Android

iOS เป็นระบบปฏิบัติการที่ถูกพัฒนาโดยบริษัท Apple ซึ่งใช้ในผลิตภัณฑ์ที่บริษัท Apple เป็นผู้ผลิต อุปกรณ์สื่อสารภายใต้ยี่ห้อ Apple เช่น iPod, iPad และ iPhone



รูปที่ 2.34 สัญลักษณ์ของระบบปฏิบัติการ iOS [25]

ข้อดี คือ มี Application หลากหลาย มีบริการ App Store และโปรแกรม iTunes สนับสนุนการจัดการอุปกรณ์มีเมนูการใช้งาน รวดเร็วและเข้าใจง่าย โปรแกรม Web Browser (Safari) ตอบสนองได้รวดเร็ว

ข้อเสีย คือ ผู้ใช้งานไม่สามารถออกแบบปรับเปลี่ยนหน้าจอได้ตามความต้องการ ไม่สามารถทำงานได้พร้อมกันหลายอย่าง เช่น ไม่สามารถฟังเพลงพร้อมเปิด Web Browser เพื่อใช้งานอินเทอร์เน็ตได้

Android [26] เป็นระบบปฏิบัติการที่ถูกพัฒนาโดยบริษัท Google ซึ่งเป็นระบบปฏิบัติการแบบ Open Source ซึ่งเป็นระบบปฏิบัติการแบบเปิดหรือแบบฟรีนั่นเอง ดังนั้นผู้ผลิตโทรศัพท์เคลื่อนที่และ Tablet จึงนิยมนำ Android ไปใช้เป็น OS เช่น HTC, Samsung ในตระกูล Galaxy



รูปที่ 2.35 สัญลักษณ์ของ Android [26]

ข้อดี คือ เป็นมาตรฐานเปิดทำให้เกิดความหลากหลายและมี Application ให้เลือกใช้มากมาย และสามารถเชื่อมต่อกับบริการต่างๆของ Google ได้สะดวก เช่น Gmail, Google Talk, Google Maps และ Google Search Engine

ข้อเสีย คือ ไม่คล่องตัวเท่า iOS และการที่เป็นระบบเปิดทำให้มีอุปกรณ์ที่ใช้ระบบมีหลายยี่ห้อหลายขนาดหน้าจอทำให้ Application ต่างๆ ต้องพัฒนาออกมาสามารถใช้งานได้ เฉพาะรุ่นเท่านั้น เนื่องจากอาจติดปัญหาเรื่อง ความกว้างของหน้าจอ เป็นต้น



รูปที่ 2.36 โปรแกรม Xcode [27]

2.8 Xcode

Xcode [27] เป็นเครื่องมือสำหรับนักพัฒนาโปรแกรม และแอปพลิเคชันบนแพลตฟอร์ม OS X และ iOS บนสมาร์ตโฟนที่รู้จักกันดีอย่างแอปพลิเคชันบน iPhone สำหรับนักพัฒนาที่ต้องการจะพัฒนาแอปพลิเคชันบน iOS นั้นจำเป็นต้องมี XCode IDE ติดตั้งในเครื่องคอมพิวเตอร์ก่อน นอกจากนั้นแล้วต้องติดตั้ง iOS SDK อีกด้วยดาวน์โหลด XCode ได้ที่ <https://developer.apple.com> เมื่อทำการติดตั้งดูแล้วลองเปิดตัวโปรแกรม Xcode IDE ขึ้นมา จะเห็นหน้าต่างเริ่มต้นให้เลือกพัฒนาแอปพลิเคชัน ได้หลากหลายรูปแบบ แอปพลิเคชันที่จะพัฒนาได้นั้น จะมีตั้งแต่ Cocoa บน Mac OSX และแพลตฟอร์มของ iOS ที่ใช้พัฒนาแอปพลิเคชันบน iPhone และ iPad ซึ่งรูปแบบแอปพลิเคชันเริ่มต้น จากเดิมที่มีคือ Single View Application นั้นก็ยังใช้พัฒนาได้ปกติ เพียงแค่จะมีส่วนของ Master Detail Application เพิ่มขึ้นมาซึ่งข้อดีของ Master Detail Application บน Xcode 4.3 นั้นก็คือเส้นทางลัดที่นักพัฒนาสามารถสร้างแอปพลิเคชันได้อย่างรวดเร็วโดยไม่ต้องเขียนโปรแกรม

2.8.1 การใช้งาน Xcode แบบเบื้องต้น

1. ติดตั้งโปรแกรม Xcode ให้โหลดโปรแกรม Xcode จาก App store จากนั้นเปิดโปรแกรมขึ้นมา ทำการสร้าง Xcode project ด้วยการกดไปที่ Create a new Xcode project ดังรูปที่ 2.37



รูปที่ 2.37 คลิกที่ Create New Xcode Project [27]



รูปที่ 2.38 รูปแบบของการตั้งค่าแอปพลิเคชัน [27]

2. จากรูปที่ 2.38 ในช่อง iOS ให้เลือก Application : ด้านขวาจะให้แบบงานทั้ง 7 แบบ ดังนี้

2.1) Master-Detail Application > สำหรับออกแบบแอป Master-Detail แสดงไอเท็มรายการ
-> รายละเอียด

2.2) OpenGL Game > สำหรับสร้างแอปในรูปแบบของ Game ภาพเคลื่อนไหวที่มีชีวิต

2.3) Page-Based Application > สร้างแอปที่สามารถเปลี่ยนได้หลายๆหน้า คล้ายกับการอ่านหนังสือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

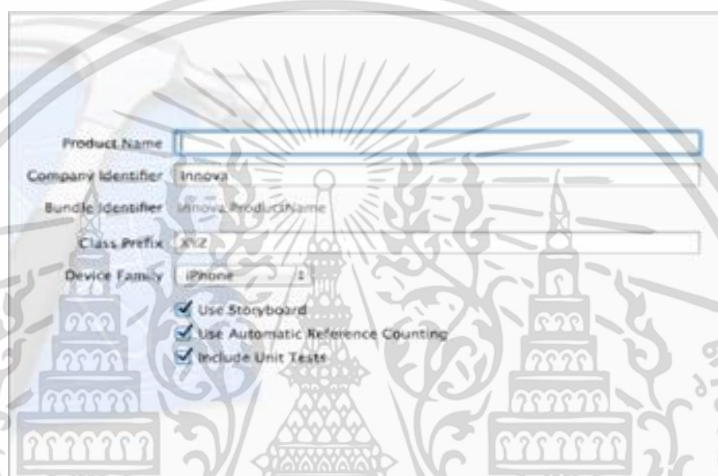
2.4) Single View Application > แอปที่จะแสดงผลทุกอย่างภายในหน้าจอเดียว

2.5) Tabbed Application > แบบที่จะแสดงผลหลายหน้าจอ โดยจะใช้ Tab เป็นตัวกำหนด

2.6) Utility Application > แบบที่จะเปลี่ยนหน้าจอโดยการ Flip View ไปอีกหน้าหนึ่ง

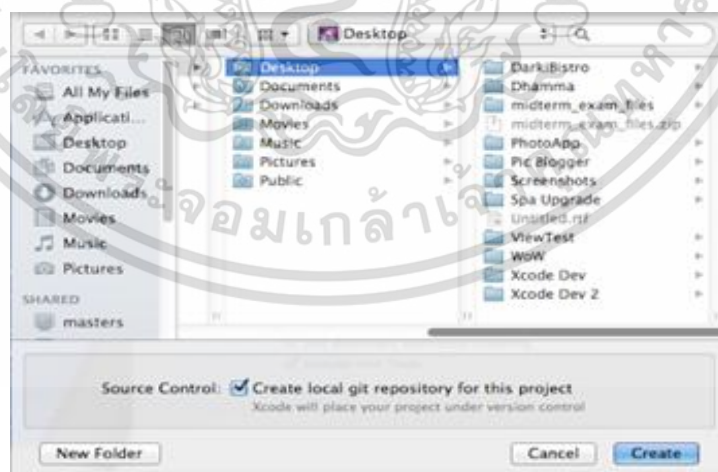
2.7) Empty Application > แบบว่าง เป็นแบบที่สามารถเลือกแบบงานที่ต้องการได้

หลังจากที่ได้แบบที่ต้องการแล้ว กด Next จากนั้นทำการตั้งชื่อแอปพลิเคชันและเลือก Device Family ดังรูปที่ 2.39



รูปที่ 2.39 ตั้งชื่อแอปพลิเคชัน [27]

3. จากนั้นเลือก Local ให้กับโปรเจกต์แล้วกด Create เพื่อสร้างโปรเจกต์ดังรูปที่ 2.40



รูปที่ 2.40 เลือกโฟลเดอร์ที่จะสร้างโปรเจกต์ [27]

4. เสร็จสิ้นการสร้าง Project

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษาที่ใช้พัฒนาบน Xcode มี 2 ภาษาคือ Swift และ Objective-C

2.8.2 Swift

ภาษาโปรแกรม “Swift” [28] เป็นภาษาโปรแกรมใหม่ที่ Apple ได้สร้างและออกแบบมาใหม่เพื่อให้ให้นักพัฒนาใช้พัฒนาโปรแกรมบน Mac OS X และ iOS โดยในอดีตจนถึงปัจจุบันภาษาที่ใช้คือ Objective-C และ Swift เป็นภาษาที่ออกแบบให้มีประสิทธิภาพสูงและง่ายต่อการพัฒนาโดยนำข้อดีของภาษาสมัยใหม่เข้ามามากมาย เช่น Type Inference, Clean Syntax, No semicolons, Closures, Generics ซึ่งคุณสมบัติที่กล่าวมาบางอย่างก็มีอยู่แล้วในภาษา Objective-C แต่ใน Swift นั้นจะสะดวกมากขึ้น ภาษา Swift ยังถูกออกแบบให้มีความปลอดภัยในการเขียนโปรแกรมมากขึ้น ทั้งนี้ไม่ใช่การเขียนโปรแกรมขณะขับรถแล้วจะไม่เกิดอุบัติเหตุบนถนน แต่เป็นความปลอดภัยในเชิงของการพัฒนา ทำให้ลดข้อผิดพลาดของโปรแกรมที่พัฒนาขึ้น ยกตัวอย่างเช่น

- ไม่อนุญาตให้มีตัวแปรที่ไม่ได้ถูกกำหนดค่าในโปรแกรม
- ไม่ต้องเขียนสัญลักษณ์ * (Asterisk) ขณะประกาศตัวแปร Pointer
- ตรวจสอบการใช้งานค่าต่ำสุดและสูงสุดของตัวเลขจำนวนเต็ม
- จะต้องเขียนวงเล็บปีกกาครอบส่วนของโปรแกรมที่อยู่ภายใต้เงื่อนไขใดๆ



รูปที่ 2.41 Swift Logo [28]

2.8.3 Objective-C

Objective-C หรือ Obj-C [29] เป็นภาษาโปรแกรมเชิงวัตถุและมีสมบัติการสะท้อน โดยแรกเริ่มภาษา Objective-C พัฒนาขึ้นจากภาษาซีโดยยังคงคุณลักษณะของภาษาซีไว้ครบทุกประการเพียงแต่เพิ่มระบบส่งข้อความ (Messaging) แบบเดียวกับภาษาสมอลทอล์กเข้าไปเท่านั้น (Objective-C runtime) ปัจจุบันภาษา Objective-C มีคุณสมบัติอื่นๆเพิ่มเติมจากการพัฒนาภาษา Objective-C 2.0 โดยบริษัทแอปเปิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษาObjective-C ส่วนอินเทอร์เฟซ (@interface) และอิมพลีเม้นเทชัน (@implementation) จะถูกแยกออกจากกัน ในทางปฏิบัติมักเก็บส่วนอินเทอร์เฟซไว้ในแฟ้ม .h และส่วนอิมพลีเม้นเทชันใน .m และปัจจุบันได้มีภาษา Swift ซึ่งรันบน iOS เช่นกันได้มีประสิทธิภาพที่ดีกว่า Objective-C แต่ว่าภาษา Objective-C มีตัวอย่างให้ศึกษาและเข้าใจง่ายกว่า ดังนั้นจึงพัฒนาเปลี่ยนจาก Objective-C ไปเป็น Swift ต่อไป



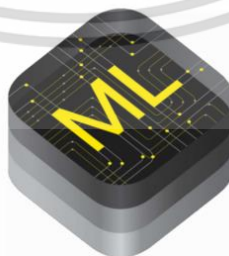
รูปที่ 2.42 Objective – C Logo [29]

2.9 Core ML

Core ML [30][31] นั้นเป็น Framework ที่ช่วยให้สามารถใช้งาน Machine Learning model ต่างๆ ภายใน Platform ของ Apple ได้นั้นเอง ทั้ง iOS, watchOS, tvOS และ macOS โดยหลักการคือ การเรียกใช้งานโมเดลที่ได้รับการสอนแล้วผ่านแอปพลิเคชัน โดยจะมีการกำหนด Input และ Output ของโมเดลเอาไว้ จากนั้นก็สามารถใช้งานได้อย่างง่าย ๆ เลย

ซึ่งการสร้างโมเดล Core ML หรือ mlmodel จะต้องใช้ coremltools ซึ่งเป็น python package ที่เอาไว้แปลงโมเดลจาก ML Framework ต่างๆ เช่น Keras, Sci-kit Learn, Caffe ให้กลายเป็น Core ML Model นั้นเอง

ปัญหา คือการสร้างโมเดลใน ML Framework ต่างๆ นั้นมีความยาก-ง่ายไม่เท่ากัน ทั้งยังมี จุดประสงค์ของแต่ละ Framework ที่ต่างกัน ขึ้นอยู่กับงานที่จะใช้ เช่น Image Classifying, Regression หรือ Tree ต่างๆ ซึ่งใช้ความรู้ด้าน Machine Learning พอสมควร บวกกับการที่ Framework เหล่านี้ นั้นใช้ C++ หรือ Python ด้วย อาจจะไม่ถนัดนัก สำหรับนักพัฒนาที่ใช้ Swift



Core ML

รูปที่ 2.43 Core ML Logo [31]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 โรคที่สามารถติดต่อได้ทางน้ำลาย

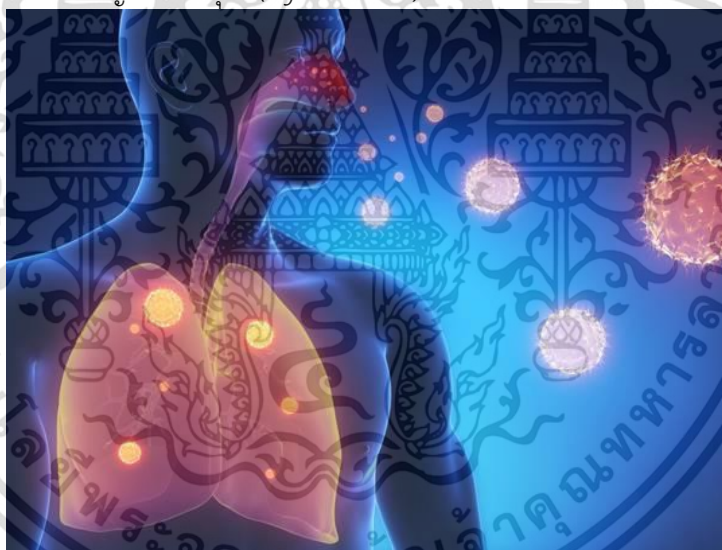
โรคที่สามารถติดต่อได้ทางน้ำลาย [32] มีด้วยกันดังนี้

2.10.1 ไข้หวัด

ไข้หวัด [32] เป็นโรคที่ติดต่อกันได้ง่ายมาก แคนอยู่ใกล้กันเป็นหวัด หรือดื่มน้ำแก้วเดียวกัน ใช้หลอดดูดน้ำร่วมกัน ก็มีสิทธิ์รับเชื้อหวัดเข้าสู่ร่างกายได้แล้ว ถึงไข้หวัดธรรมดาจะไม่ใช่วirus ที่ร้ายแรงมาก เพราะสามารถหายได้เองภายใน 5-7 วัน แต่ก็ต้องระวังกรณีรับเชื้อไข้หวัดใหญ่เข้ามา จะมีอาการรุนแรงมากกว่าและอาจเสียชีวิตได้จากโรคแทรกซ้อน ที่เคยได้ยินกันก็อย่างเช่น โรคไข้หวัดใหญ่ H1N1, โรคไข้หวัดนก มาลองเช็คว่าอาการไข้หวัดใหญ่แตกต่างจากไข้หวัดธรรมดาอย่างไร เพื่อไว้สังเกตตัวเอง

ไข้หวัดใหญ่ เป็นโรคติดเชื้อระบบทางเดินหายใจอย่างเฉียบพลัน ที่เกิดขึ้นจากเชื้อไวรัสอินฟลูเอนซา (*Influenza virus*) หรือที่รู้จักกันในชื่อของเชื้อไวรัสไข้หวัดใหญ่ โดยเชื้อไวรัสชนิดนี้มีทั้งหมด 3 สายพันธุ์ ได้แก่

- เชื้อไวรัสไข้หวัดใหญ่สายพันธุ์ A (*Influenza A*)
- เชื้อไวรัสไข้หวัดใหญ่สายพันธุ์ B (*Influenza B*)
- เชื้อไวรัสไข้หวัดใหญ่สายพันธุ์ C (*Influenza C*)



รูปที่ 2.44 ไข้หวัดใหญ่ [32]

2.10.2 คออักเสบ (Strep throat)

คออักเสบ [32] ก็เป็นอีกหนึ่งโรคที่พบได้บ่อย หากใช้หลอดดูดน้ำหรือดื่มน้ำแก้วเดียวกับคนที่ติดเชื้อแบคทีเรียกรู๊ป เอ สเตรปโตคอกคัส ในน้ำลาย หรืออาจปนเปื้อนในน้ำมูกและเสมหะ ซึ่งเมื่อได้รับเชื้อสเตรปโตคอกคัสเข้าสู่ร่างกายโดยการคลุกคลี หรือใช้ของร่วมกับคนที่ป่วยเป็นโรคคออักเสบ จะมีไข้สูง ปวดศีรษะ หนาวสั่น เบื่ออาหาร เจ็บคอ และจะเจ็บคอมากขึ้นตอนที่กลืนอาหาร ถ้าอ้าปากดูจะเห็นว่าต่อมทอนซิลแดง มีลักษณะคล้ายหนอง มีสีเหลือง ๆ ปกคลุมบนผิวของต่อมทอนซิล และหากลองจับต่อม

น้ำเหลืองที่คอจะรู้สึกว้าวม อย่งไรก็ตามโรคคออักเสบไม่ใช่โรคร้ายแรงอะไร ส่วนใหญ่หายได้เองใน 7 วัน แต่ถ้าไปหาหมอ หมอจะให้ยาปฏิชีวนะมากขึ้น ซึ่งจะช่วยให้หายเร็วขึ้นภายใน 3 วัน

2.10.3 โรคมือเท้าปาก (Hand-foot-mouth disease)



รูปที่ 2.45 โรคมือเท้าปาก [32]

โรคนี้ไม่ได้เกิดขึ้นกับเด็กๆ เท่านั้น เพราะผู้ใหญ่ก็มีสิทธิ์เป็นโรคมือเท้าปากได้ไม่ต่างกัน แค่เพียง ตุ่มน้ำแ้วเดียวกันกับเด็กหรือผู้ใหญ่ที่มีเชื้อ หรือแค่ออ จามรดใส่กัน ก็ติดเชื้อได้แล้ว

อาการที่สังเกตเห็นได้เมื่อป่วยก็คือ จะเริ่มจากมีไข้ มีแผลในปากคล้ายแผลร้อนใน และมีผื่นขึ้น เป็นจุดแดง หรือเป็นตุ่มน้ำใสที่บริเวณ ฝ่ามือ ฝ่าเท้า ลำตัว แขน ขา ส่วนใหญ่อาการจะไม่รุนแรง สามารถหายได้เอง แต่ถ้าใครป่วยก็ควรหยุดเรียน หยุดงาน พักรักษาตัวที่บ้านประมาณ 1 สัปดาห์ เพื่อจะได้ไม่ไป แพร่เชื้อให้กับคนอื่น

ทว่าหากติดเชื้อโรคมือเท้าปาก สายพันธุ์เอนเทอโรไวรัส (EV) 71 เคสนี้ค่อนข้างอันตราย เพราะเสี่ยงต่อการเกิดโรคแทรกซ้อนที่รุนแรงจนทำให้เสียชีวิตได้ ดังนั้นหากใครป่วยมือเท้าปากแล้วมีอาการซึม อ่อนแรง ชักกระตุก มือสั่น เดินเซ ต้องรีบไปพบแพทย์โดยด่วน มาเรียนรู้เรื่องโรคมือเท้าปากเพิ่มเติม เพื่อดูแลบุตรหลานและตัวคุณเองกันดีกว่า

2.10.4 คางทูม (Mumps)

โรคคางทูม [32] ก็เป็นอีกโรคหนึ่งที่พบบ่อยในเด็ก แต่ผู้ใหญ่ก็เป็นได้ โรคนี้เกิดจากเชื้อไวรัสในกลุ่มพาราไมกโซ (Paramyxovirus) ทำให้ต่อมน้ำลายที่บริเวณกกหูอักเสบ คางจึงบวมโต จึงเป็นที่มาของชื่อโรคคางทูม ติดต่อกันได้โดยตรงจากการไอ จามใส่กัน รวมทั้งสัมผัสน้ำลายของคนที่เป็นคางทูม เช่น การกินน้ำและอาหารโดยใช้ภาชนะร่วมกัน

คนที่ป่วยคางทูมจะมีไข้ ปวดศีรษะ ปวดเมื่อยเนื้อตัว เบื่ออาหาร อ่อนเพลีย และปวดในรูหูหรือ หลังหูขณะเคี้ยว กลืนอาหาร หรือแม้กระทั่งตม้มน้ำ หลังจากนั้นจะมีอาการบวมที่ข้างหูหรือขากรรไกรจนไม่สามารถเคี้ยวอาหารหรืออ้าปากได้ถนัด แต่ก็ไม่ใช่โรคที่ร้ายแรงนัก เพราะสามารถหายได้เองภายใน 7-10 วัน และเมื่อป่วยคางทูมแล้วครั้งหนึ่งก็จะมีภูมิคุ้มกันตลอดไป อย่งไรก็ดี โรคนี้สามารถป้องกันได้ตั้งแต่

ยังเด็ก โดยการฉีดวัคซีนรวมเอ็ม เอ็ม อาร์ (MMR) ซึ่งสามารถฉีดได้ตอนอายุ 1 ขวบและต้องฉีดอีกครั้ง ตอนอายุ 7 ขวบ ว่าแล้วก็ลองมาเช็กอาการของโรคคางทูมให้ละเอียดกันอีกสักนิด

2.10.5 โรคหูดข้าวสุก (Molluscum contagiosum)

หูด [32] มีอยู่หลายชนิด แต่ถ้าติดต่อกจากการใช้ของส่วนตัวร่วมกับคนอื่น ไม่ว่าจะเป็นแก้วน้ำ หลอดดูดน้ำ ผ้าเช็ดตัว รองเท้า ก็ต้องระวัง "หูดข้าวสุก" ให้ดี โรคนี้เกิดจากเชื้อไวรัส *Molluscum contagiosum virus (MCV)* ลักษณะเด่นคือ จะเป็นตุ่มเนื้อขนาดเล็ก มีสีเดียวกับผิวหนัง รูปร่างกลม ผิวหูดเรียบเป็นมันคล้ายไข่มุก มีรอยบุ๋มตรงกลางคล้ายสะดือ เวลาบีบตุ่มออกจะได้สารสีขาวขุ่นในคล้ายข้าวสุก มักขึ้นตามลำตัว ท้อง แขน ขา รักแร้ ใบหน้า ดวงตา ขาหนีบ บริเวณอวัยวะเพศ แต่จะไม่รู้สึกคันหรือเจ็บ หากมีภูมิคุ้มกันดี โรคนี้จะหายไปเองภายใน 2-9 เดือน (เฉลี่ย 2-3 เดือน) แต่หากมีโรคแทรกซ้อนอาจใช้เวลานานเป็นปี

2.10.6 โรคเริมที่ปาก (Herpes Simplex)

ตุ่มใสๆ ที่ขึ้นบริเวณริมฝีปากและทำให้รู้สึก คัน ปวดแสบปวดร้อน ที่เรียกว่า เริม [32] นั้นเกิดจากการติดเชื้อไวรัส *Herpes virus (HSV)* ซึ่งเจ้าไวรัสที่ว่านี้จะแฝงอยู่ในน้ำลาย น้ำเหลือง หรือแม้กระทั่งอสุจิ และจะเข้าสู่ร่างกายผ่านทางผิวหนัง เยื่อเมือกและเยื่อช่องปาก จากการดื่มน้ำแก้วเดียวกัน ใช้ของร่วมกัน รวมทั้งการจูบปาก

ถ้าเป็นโรคเริมคุณหายได้เองภายใน 2 สัปดาห์ แต่หากมีอาการปวดมาก มีน้ำเหลืองไหลออกมาจากตุ่มแผล ระบายเคืองตา และมีไข้สูงกว่า 38 องศาเซลเซียส ควรไปพบแพทย์เพราะนั่นแสดงว่าตุ่มเริมนั้นเกิดการติดเชื้อแล้ว

2.10.7 ไวรัสตับอักเสบ (Hepatitis)

ไวรัสตับอักเสบ [32] มีหลายสายพันธุ์ ทั้งไวรัสตับอักเสบ เอ บี ซี ไปจนถึงไวรัสตับอักเสบ จี แต่ไวรัสตับอักเสบที่ติดต่อกันได้ทางอาหารและน้ำดื่มก็คือไวรัสตับอักเสบ เอ (Hepatitis A) และไวรัสตับอักเสบ อี (Hepatitis E) อาการของไวรัสตับอักเสบ เอ ที่สามารถสังเกตได้ก็คือ มีไข้ ตัวร้อน อ่อนเพลีย เบื่ออาหาร คลื่นไส้ อาเจียน ปวดท้องบริเวณชายโครงขวา ท้องร่วง ปัสสาวะจะเริ่มมีสีเข้ม อุจจาระสีซีด และมีอาการตัวเหลือง ตาเหลืองที่เรียกว่าดีซ่าน

ส่วนอาการของไวรัสตับอักเสบ อี นั้นก็จะเหมือนกับไวรัสตับอักเสบ เอ เพียงแต่จะมีอาการดีซ่านให้เห็นชัดเจนกว่า และถ้าเป็นโรคไวรัสตับอักเสบ เอและอี กว่าจะหายดีก็ต้องใช้เวลาประมาณ 1 เดือนเลยทีเดียว

2.10.8 คอตีบ (Diphtheriae)

ถ้ามีไข้ต่ำๆ เจ็บคอ เบื่ออาหาร และมีอาการร่วมกับไอเสียงดังก้อง กลืนอาหารลำบาก อาจหายใจติดขัด หอบ ซีฟจรเต้นเร็ว ในรายที่มีอาการรุนแรง พิษจากคอตีบจะทำให้เกิดเนื้อตายเป็นแผ่นฝ้าในลำคอหรือหลอดลม ทำให้หายใจไม่ออก เสียชีวิตได้เลย ดังนั้นอย่านิ่งนอนใจเด็ดขาด เพราะโรคนี้แพร่

จากอีกคนหนึ่งไปสู่อีกคนหนึ่งได้ด้วยการไอ จาม การใช้สิ่งของร่วมกัน หรือสัมผัสกับเสมหะของคนที่เป็นโรคคออักเสบหรือคนที่ติดเชื้อคออักเสบ [32]

หากไม่เคยฉีดวัคซีนป้องกันโรคคออักเสบมาก่อน ก็มีโอกาเสี่ยงที่จะป่วยโรคนี้เหมือนกันหากใช้หลอดดูดน้ำร่วมกับคนเป็นคออักเสบ หรืออยู่ใกล้เด็กทารกที่ยังไม่เคยฉีดวัคซีน ดังนั้นเพื่อป้องกันโรคนี้ก็สามารถไปฉีดวัคซีนรวมคออักเสบและบาดทะยักชนิดสำหรับเด็กโตและผู้ใหญ่ (Tdap) ไปได้ สำหรับเด็กสามารถฉีดวัคซีนป้องกันโรคคออักเสบได้ตั้งแต่อายุ 2-3 เดือน

2.10.9 เยื่อหุ้มสมองอักเสบ (Meningitis)

โรคที่มีอาการรุนแรงถึงขั้นหูหนวก ตาบอด เป็นอัมพาตหรืออัมพาตจนถึงแก่ชีวิตได้ อย่าง "เยื่อหุ้มสมองอักเสบ"[32] ก็มีสาเหตุหนึ่งมาจากการสัมผัสและใช้สิ่งของร่วมกับคนป่วยเช่นกัน ไม่ว่าจะเป็นหลอดดูดน้ำ แก้วน้ำ ข้อน้ำ โดยโรคนี้ส่วนใหญ่เกิดจากการติดเชื้อแบคทีเรีย ไวรัส และรา เมื่อติดเชื้อจะมีไข้สูง ปวดศีรษะรุนแรง คอแข็ง คลื่นไส้ อาเจียน ชัก ตาสู้แสงไม่ได้ หากรักษาไม่ทันอาจจะมีอาการรุนแรงดังที่บอกไปข้างต้น ทว่ายังมีโรคอื่น ๆ ที่มีอาการคล้ายกับโรคเยื่อหุ้มสมองอักเสบอยู่บ้าง

2.10.10 ไข้กาหลังแอ่น (Meningococcal meningitis)

เชื้อที่ทำให้เกิดโรคไข้กาหลังแอ่น [32] คือ *Neisseria meningitidis* ซึ่งเป็นโรคติดเชื้อแบคทีเรียที่มีความรุนแรงมาก ที่น่ากลัวคือสามารถติดต่อกันได้ผ่านการหายใจ ไอ จาม แม้กระทั่งสัมผัสน้ำมูก น้ำลาย เสมหะของผู้ที่เป็นพาหะหรือคนที่เป็นโรคก็เสี่ยงติดเชื้อได้แล้ว โดยเชื้อจะผ่านเข้าไปทางลำคอแล้วเข้าไปในกระแสเลือด บางคนติดเชื้อแล้วไม่มีอาการก็เป็นเพียงพาหะเท่านั้น

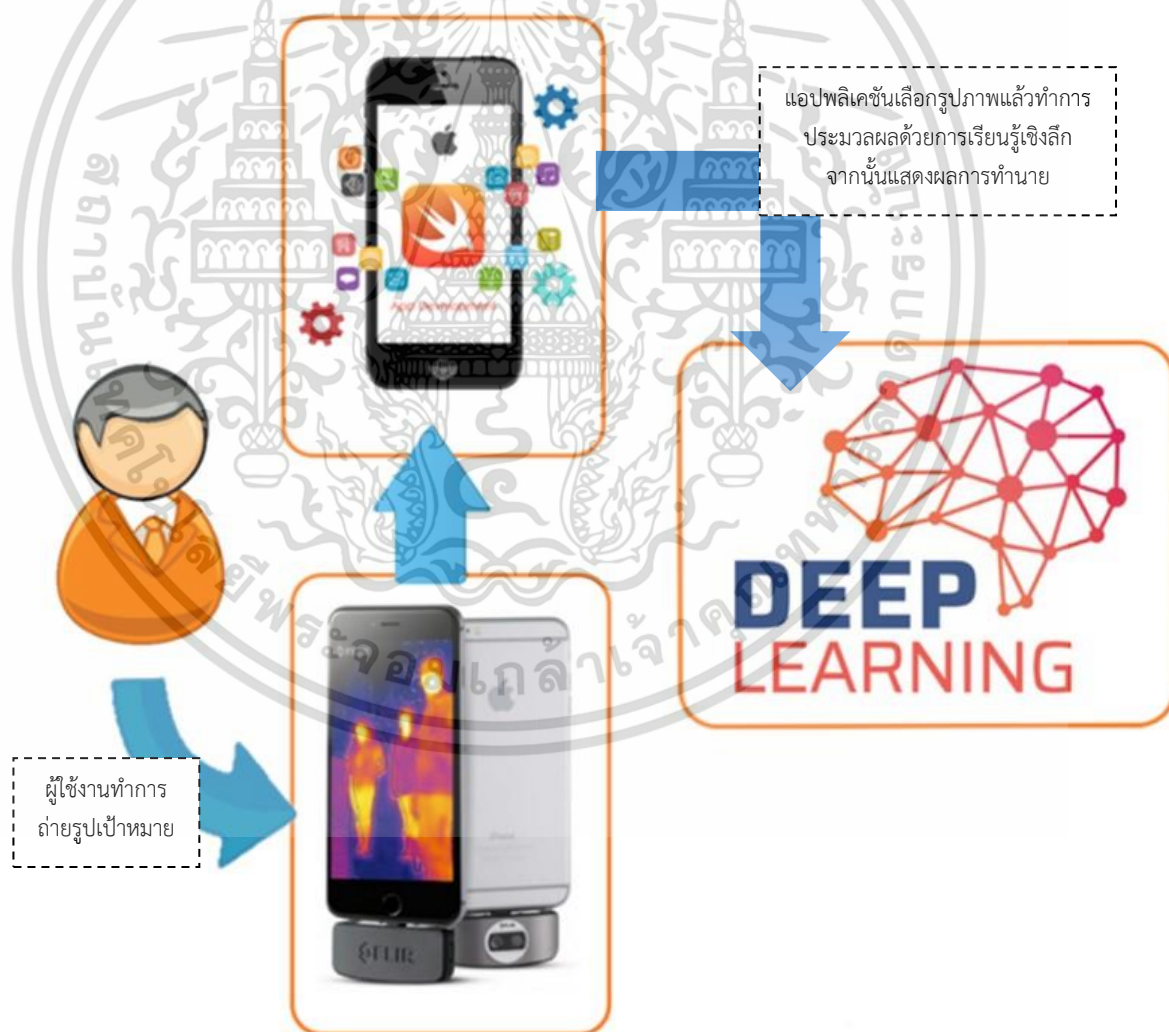
แต่หากเชื้อเล็ดรอดเข้าสู่กระแสโลหิตหรือระบบประสาทส่วนกลางได้ อาการของโรคจะรุนแรงขึ้น เริ่มจากมีอาการคล้ายไข้หวัด เจ็บคอ ไอ ปวดศีรษะ มีไข้สูง จากนั้นจะมีผื่นขึ้นตามตัว ลักษณะคล้ายจุดเลือดออก และเป็นแฉกคล้ายดาวกระจาย ซึ่งเป็นลักษณะเฉพาะของโรค

หากเชื้อไปที่เยื่อหุ้มสมองจะทำให้เกิดโรคเยื่อหุ้มสมองอักเสบ มีอาการปวดศีรษะรุนแรง อาเจียน คอแข็ง ไม่ค่อยรู้สึกตัว สับสน อาจมีอาการชักเกร็ง หลังแอ่นแข็งได้ ซึ่งนี่เป็นที่มาของชื่อโรคไข้กาหลังแอ่นนั่นเอง ในรายที่รุนแรงจะมีเลือดออกในลำไส้และต่อมหมวกไต รวมทั้งภาวะที่เลือดจับตัวกันเป็นลิ่มทั่วร่างกาย (Waterhouse-Friderichsen Syndrome) จนเกิดอาการช็อกและเสียชีวิตในที่สุด โดยอาการทั้งหมดนี้จะเกิดขึ้นภายในระยะเวลาอันสั้น อาจเสียชีวิตได้ภายใน 48 ชั่วโมง

บทที่ 3

การดำเนินงาน

การดำเนินการวิจัยจะแบ่งออกเป็น 2 ส่วน ส่วนแรกคือการเก็บข้อมูลภาพของจากกล้องอินฟราเรด ส่วนที่ 2 ก็คือการออกแบบอัลกอริทึมของโปรแกรมประมวลผลด้วยเทคโนโลยีการเรียนรู้เชิงลึก และการออกแบบแอปพลิเคชัน iOS โดยหลักการทำงานของโปรแกรมประมวลผลด้วยเทคโนโลยีการเรียนรู้เชิงลึกดังกล่าวมีการทำงานด้วยกัน 3 ขั้นตอนดังรูปที่ 3.1 ขั้นแรกผู้ใช้งาน (User) ทำการถ่ายรูปเป้าหมายจากนั้น ขั้นตอนต่อไปทำการเข้าแอปพลิเคชัน Binary classification หรือ 4 level of classification แล้วเลือกภาพที่ต้องการทำนาย ขั้นตอนที่ 3 แอปพลิเคชันจะทำการประมวลผลรูปภาพด้วยการเรียนรู้เชิงลึกแล้วแสดงผลการทำนายออกมาทางแอปพลิเคชัน



รูปที่ 3.1 แผนภาพการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 การเก็บข้อมูลภาพ

ทางผู้จัดทำได้ใช้กล้องอินฟราเรด FLIR ONE Pro ซึ่งเป็นกล้องที่มีรายละเอียดคร่าว ๆ ดังตาราง

Specification	FLIR ONE Pro
Certifications	MFi (iOS version), RoHS, CE/FCC, CEC-BC, BN61233
Operating temperature	0 °C – 35 °C (32 °F to 95 °F)
Size	Width 68mm Height 34mm Dept 14 mm
Weight	36.5 g
Thermal resolution	160 x 120 px. With 17 µm pixel size
Visual resolution	1440 x 1080 px.
HFOV/VFOV	55° ± 1° / 43° ± 1°
Frame rate	8.7 Hz
Focus	Fixed 15cm - infinity

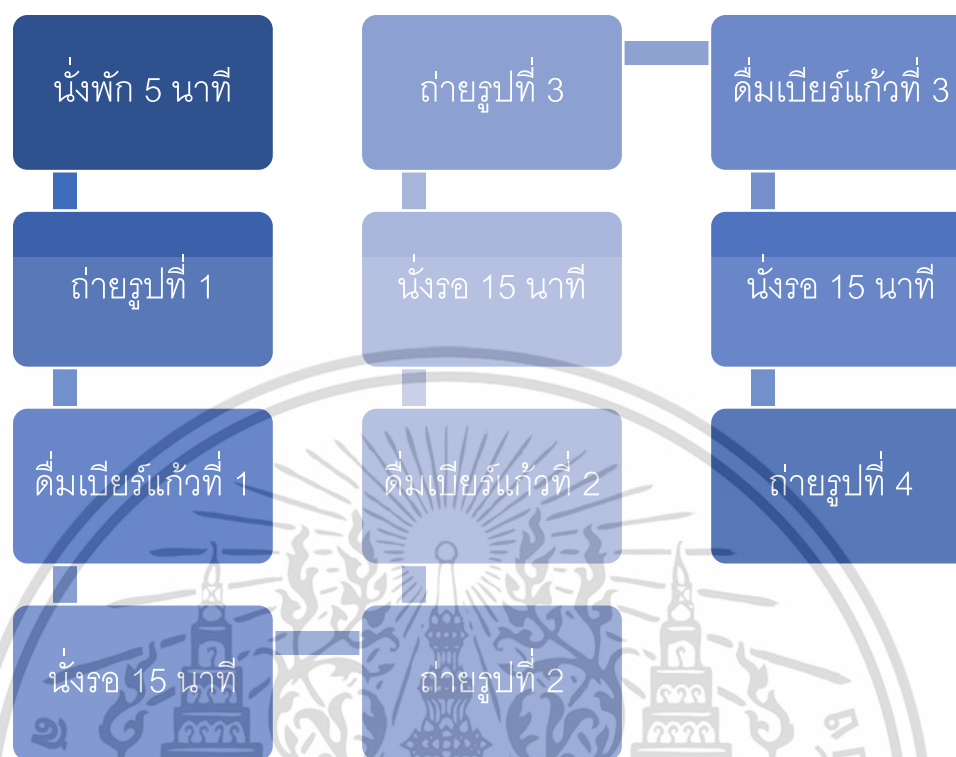
ตารางที่ 3.1 แสดงรายละเอียดต่าง ๆ ของกล้องอินฟราเรด FLIR ONE Pro โหมด video คร่าว ๆ

โดยกล้องอินฟราเรดดังกล่าวเชื่อมต่อกับแอปพลิเคชันบน iOS ใช้ในการถ่ายรู้อาสาสมัครประมาณ 50 คน โดยถ่ายรูปจำนวน 4 รูปต่ออาสาสมัคร 1 คน ซึ่งมีวิธีการเก็บข้อมูลรูปภาพดังแสดงในรูปที่ 3.2 รวมเป็นทั้งหมด 200 รูป และหลังจากนั้นก็ถ่ายรู้อาสาสมัครเฉพาะภาพที่ไม่ได้ดื่มแอลกอฮอล์ ในแต่ละท่าทางเพื่อเพิ่มปริมาณข้อมูลอีกประมาณ 50 รูป โดยการเก็บข้อมูลครั้งนี้อาสาสมัครดื่มเบียร์ทั้งหมด 3 แก้วแต่ละแก้วมีปริมาณ 200 มิลลิลิตรซึ่งมีแอลกอฮอล์ปริมาณ 6.4% โดยปริมาตร ซึ่งนับเป็นปริมาณ 1 ดื่มมาตรฐานหรือ มีปริมาณแอลกอฮอล์ 10 กรัม ตัวอย่างอาสาสมัครเพศชายน้ำหนัก 55 กิโลกรัม ดื่มเบียร์ 1 แก้ว ปริมาณ 200 มิลลิลิตรจะมีปริมาณแอลกอฮอล์ในเลือด (BAC) ประมาณ 0.026% ดังนั้นหากอาสาสมัครดังกล่าว ดื่มเบียร์ไป 2 แก้วจะมีปริมาณแอลกอฮอล์ในเลือดปริมาณ 0.052% ซึ่งจะเกินกฎหมายควบคุม

ขั้นตอนการเก็บภาพจะเป็นไปตามรูปที่ 3.2 โดยที่เริ่มจากให้อาสาสมัครนั่งพัก 5 นาทีเพื่อให้ร่างกายทำงานเป็นปกติหลังจากนั้นจะเป็นการถ่ายรูป ในการเก็บข้อมูลครั้งนี้จะให้อาสาสมัครดื่ม เบียร์ที่มีแอลกอฮอล์ 6.4% โดยปริมาตร ปริมาณ 200 มิลลิลิตรโดยไม่มีน้ำแข็งและต้องดื่มให้หมดภายใน 5 นาที

- ทำการถ่ายรูปด้วยกล้องอินฟราเรดเพื่อเก็บเป็นรูปภาพกลุ่มที่ 1 คือไม่ได้ดื่มแอลกอฮอล์ แล้วทำการดื่มแอลกอฮอล์แก้วที่ 1
- นั่งรอ 15 นาทีเพื่อให้มีการดูดซึมแอลกอฮอล์ จากนั้นทำการถ่ายรูปครั้งที่ 2 เพื่อเก็บเป็นรูปภาพกลุ่มที่ 2 คือ ดื่มแอลกอฮอล์ 1 แก้ว แล้วทำการดื่มแอลกอฮอล์แก้วที่ 2
- นั่งรอ 15 นาทีเพื่อให้มีการดูดซึมแอลกอฮอล์ จากนั้นทำการถ่ายรูปครั้งที่ 3 เพื่อเก็บเป็นรูปภาพกลุ่มที่ 3 คือ ดื่มแอลกอฮอล์ 2 แก้ว แล้วทำการดื่มแอลกอฮอล์แก้วที่ 3
- นั่งรอ 15 นาทีเพื่อให้มีการดูดซึมแอลกอฮอล์ จากนั้นทำการถ่ายรูปครั้งที่ 4 เพื่อเก็บเป็นรูปภาพกลุ่มที่ 4 คือ ดื่มแอลกอฮอล์ 3 แก้ว เป็นกลุ่มสุดท้ายของรูปภาพทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 ขั้นตอนการเก็บข้อมูลภาพ

ในการถ่ายรูปจากกล้องอินฟราเรดจะมีการใช้ filter contrast และทำการยึดกล้องมือถือไว้กับขาตั้งกล้องเพื่อไม่ให้เกิดการขยับตำแหน่งของรูปภาพ ซึ่งในการเก็บภาพจะถ่ายรูปภาพอาสาสมัครทั้งหมดภายในห้องที่มีเครื่องปรับอากาศ อุณหภูมิประมาณ 26 องศาเซลเซียส โดยตัวอย่างในการถ่ายรูปอินฟราเรดเป็นไปตามรูปที่ 3.3



รูปที่ 3.3 วิธีการเก็บข้อมูลภาพ

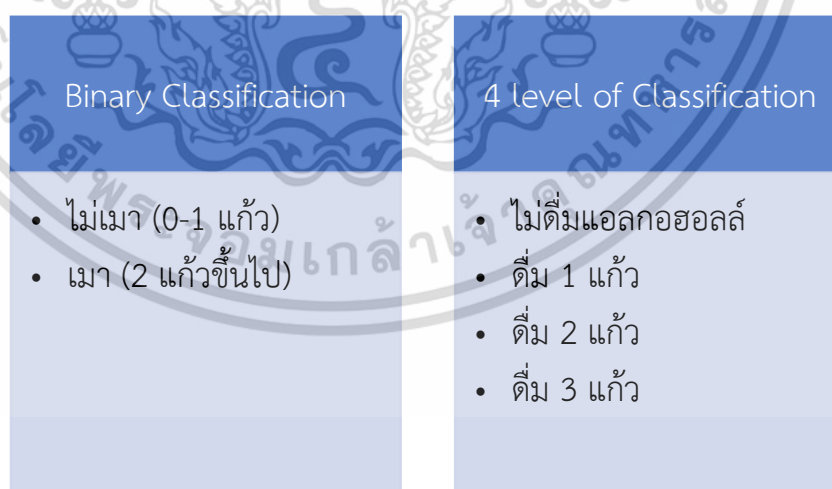
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบอัลกอริทึมของโปรแกรมประมวลผล

ในส่วนของการประมวลผลภาพนั้น โไลบรารีหลักที่ใช้ในการประมวลผล คือ TensorFlow เป็นไลบรารีสำหรับใช้พัฒนา Machine Learning เป็น Open source (เขียนด้วย Python) ที่พัฒนาโดย Google มีความยืดหยุ่นและประมวลผลได้เร็ว เพราะสามารถดึงกำลังในการประมวลผลจากการ์ดจอ (Graphics Processing unit หรือ GPU) มาใช้ได้อย่างมีประสิทธิภาพมีขั้นตอนการทำงานดังนี้

3.2.1 การเตรียมข้อมูล

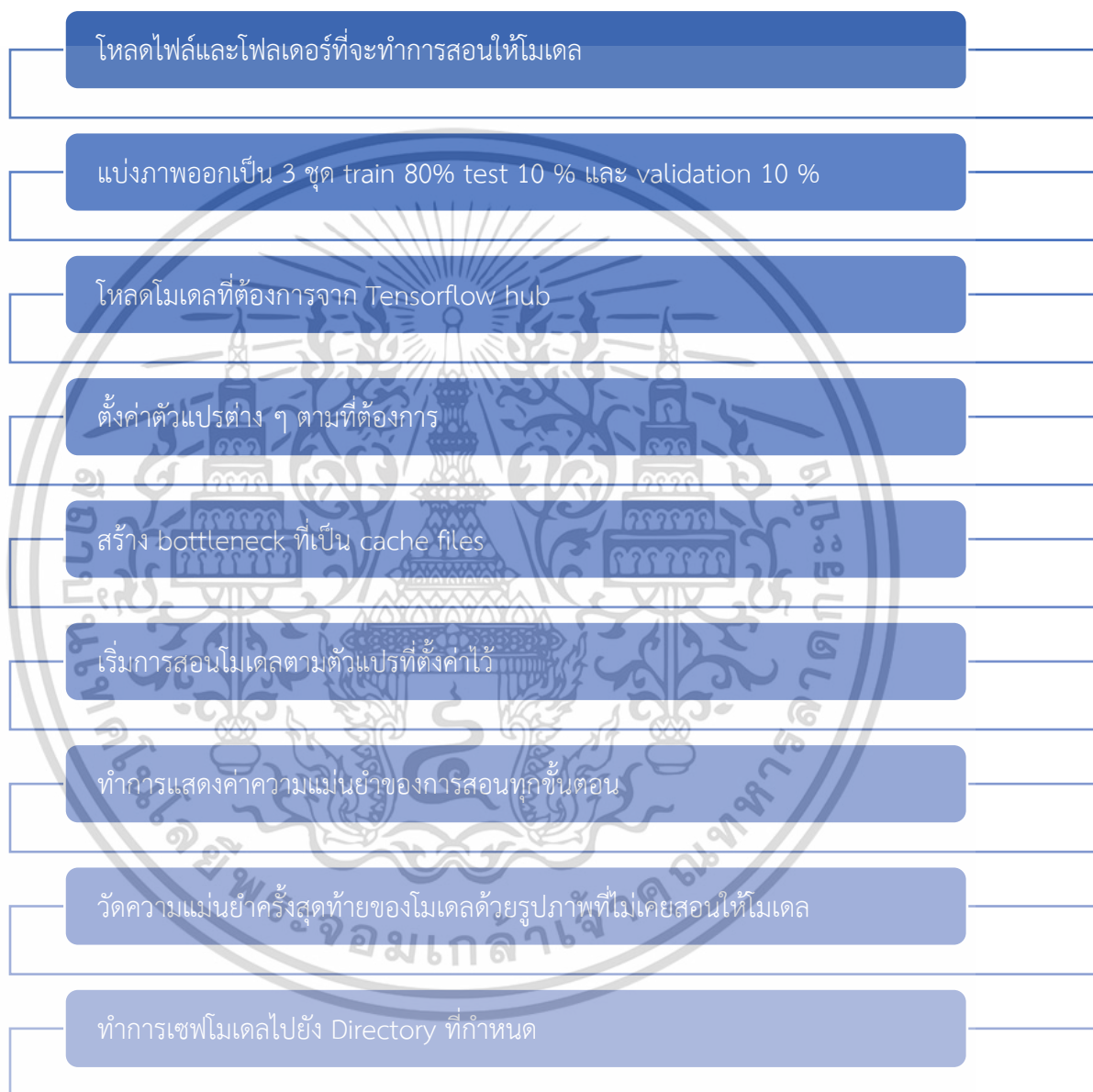
ในการเตรียมข้อมูลเพื่อใช้ในการสร้างโมเดลเพื่อตรวจวัดปริมาณแอลกอฮอล์ จะต้องมียข้อมูลซึ่งเป็นข้อมูลที่รู้ค่าตอบอยู่แล้ว ในที่นี้คือรูปภาพของอาสาสมัครเป็นข้อมูล input และปริมาณแอลกอฮอล์ที่ดื่ม ซึ่งใช้เป็นข้อมูล output และไฟล์ภาพที่ได้จะต้องมีสกุลไฟล์เป็น jpg, jpeg หรือ png จากนั้นนำข้อมูลที่ได้จากอาสาสมัครทั้ง 50 คน มาคัดแยกประเภทเพื่อเตรียมสำหรับการสอนโมเดล โดยจะมีการแยกประเภทเป็น 2 แอปพลิเคชัน ในส่วน Binary Classification จะแยกรูปภาพเป็น เมา กับ ไม่เมา ซึ่งจากปริมาณแอลกอฮอล์ในเลือด (BAC) ที่คำนวณจาก Widmark formula จะเห็นได้ว่าหากอาสาสมัครเพศชายน้ำหนัก 55 กิโลกรัม ดื่มเบียร์ 1 แก้ว ปริมาณ 200 มิลลิลิตรจะมีปริมาณแอลกอฮอล์ในเลือด (BAC) ประมาณ 0.026% ดังนั้นหากอาสาสมัครดังกล่าว ดื่มเบียร์ไป 2 แก้วจะมีปริมาณแอลกอฮอล์ในเลือดปริมาณ 0.052% ซึ่งจะเกินกฎหมายควบคุม ดังนั้นการแบ่งรูปประเภทไม่เมาจึงประกอบไปด้วยรูปภาพของอาสาสมัครที่ไม่ดื่มแอลกอฮอล์ไปจนถึงดื่มแอลกอฮอล์ 1 แก้ว ส่วนรูปประเภทเมาประกอบไปด้วยรูปภาพของคน que ดื่มแอลกอฮอล์มากกว่า 400 มิลลิลิตร (2 แก้ว) อีกแอปพลิเคชันหนึ่งคือ 4 level of classification จะแบ่งประเภทรูปออกเป็น 4 ประเภทตามจำนวนแอลกอฮอล์ที่ดื่ม คือ ไม่ดื่มแอลกอฮอล์, ดื่ม 1 แก้ว, ดื่ม 2 แก้ว และ ดื่ม 3 แก้ว



รูปที่ 3.4 การแบ่งประเภทของรูปภาพ

3.2.2 การเตรียมและสอนโมเดลที่จะใช้

ทางผู้จัดทำได้ใช้ TensorFlow ในการสอนโมเดลโดยใช้ TensorFlow hub เป็นไลบรารีที่อาศัยหลักการ Transfer learning ซึ่งเป็นวิธี Machine Learning โดยที่นำโมเดลที่ถูกพัฒนาแล้วกลับมาใช้ใหม่ให้เป็นโมเดลเริ่มต้นสำหรับงานต่อไป โดยโปรแกรมการสอนโมเดลใหม่มีดังนี้



รูปที่ 3.5 ขั้นตอนการสอนโมเดล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งผู้จัดทำได้สร้างโมเดลขึ้นมา 2 โมเดล โมเดลแรกคือโมเดล Binary Classification ที่ทำการแยกระหว่างคนปกติ (ไม่ดื่มแอลกอฮอล์และดื่มแอลกอฮอล์ 1 แก้ว) กับคนที่มีอาการมินเมา (ดื่มแอลกอฮอล์ 2 และ 3 แก้ว) ใช้ transfer learning มาจากโมเดล EfficientNetB0. โดยมีตัวแปรในการสอนโมเดล คร่าวๆดังนี้

- 4,000 training steps
- 0.01 learning rate
- Evaluate step interval = 10
- Train batch size = 100
- Test batch size = 1
- Validation batch size = 100

ได้ค่าความแม่นยำของโมเดลประมาณ 78.57% และโมเดลที่สองที่สร้างขึ้นก็คือโมเดล 4 level of Classification ที่ทำการแยกปริมาณแอลกอฮอล์ที่ดื่ม (200 มิลลิลิตรต่อแก้ว) ตั้งแต่ไม่ดื่มแอลกอฮอล์, ดื่ม 1 แก้ว, ดื่ม 2 แก้ว และ ดื่ม 3 แก้ว ซึ่งได้ Transfer learning มาจากโมเดล mobilenet_v1_025_224 โดยมีตัวแปรในการสอนโมเดลคร่าว ๆ ดังนี้

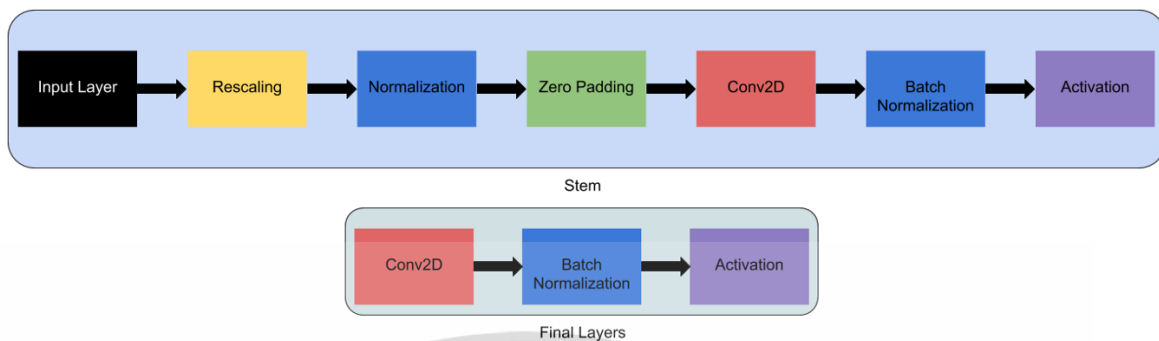
- 4000 training steps
- 0.01 learning rate
- Evaluate step interval = 10
- Train batch size = 100
- Test batch size = 1
- Validation batch size = 100

ได้ค่าความแม่นยำของโมเดลประมาณ 57.4 %

หมายเหตุ : โปรแกรมภาษา python และวิธีลงไลบรารีอยู่ในภาคผนวก

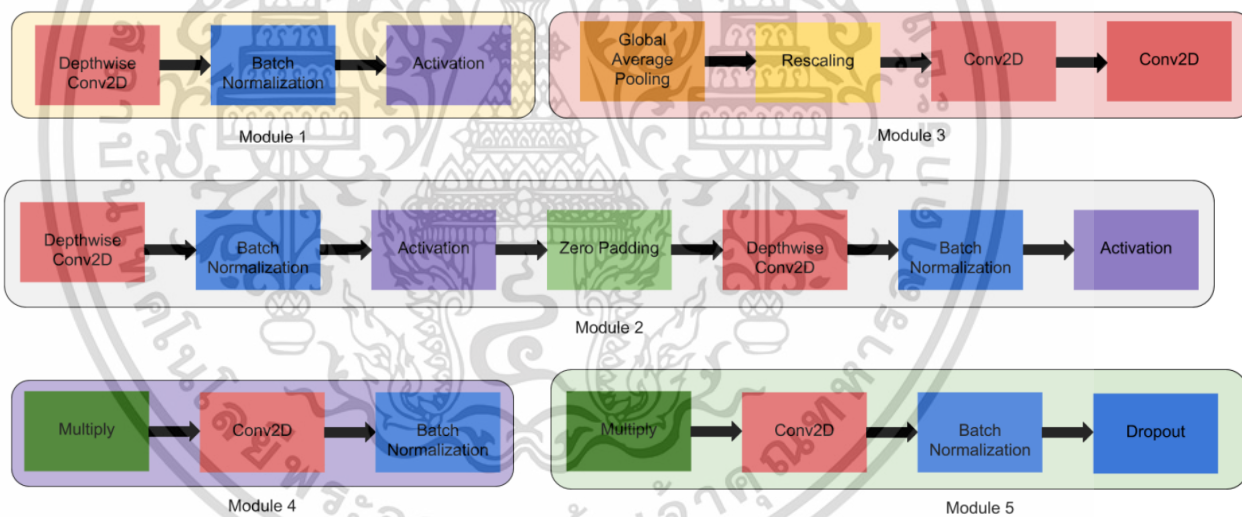
3.2.3 โครงสร้างของโมเดลที่ใช้

โมเดล EfficientNetB0 [37] เป็นโมเดลที่สร้างขึ้นโดยมีพื้นฐานที่เหมือนกับโมเดลอื่น ๆ คือมี Stem และ Final layers ดังรูปที่ 3.6



รูปที่ 3.6 แสดงโครงสร้างของ Stem และ Final Layers ของ EfficientNet [37]

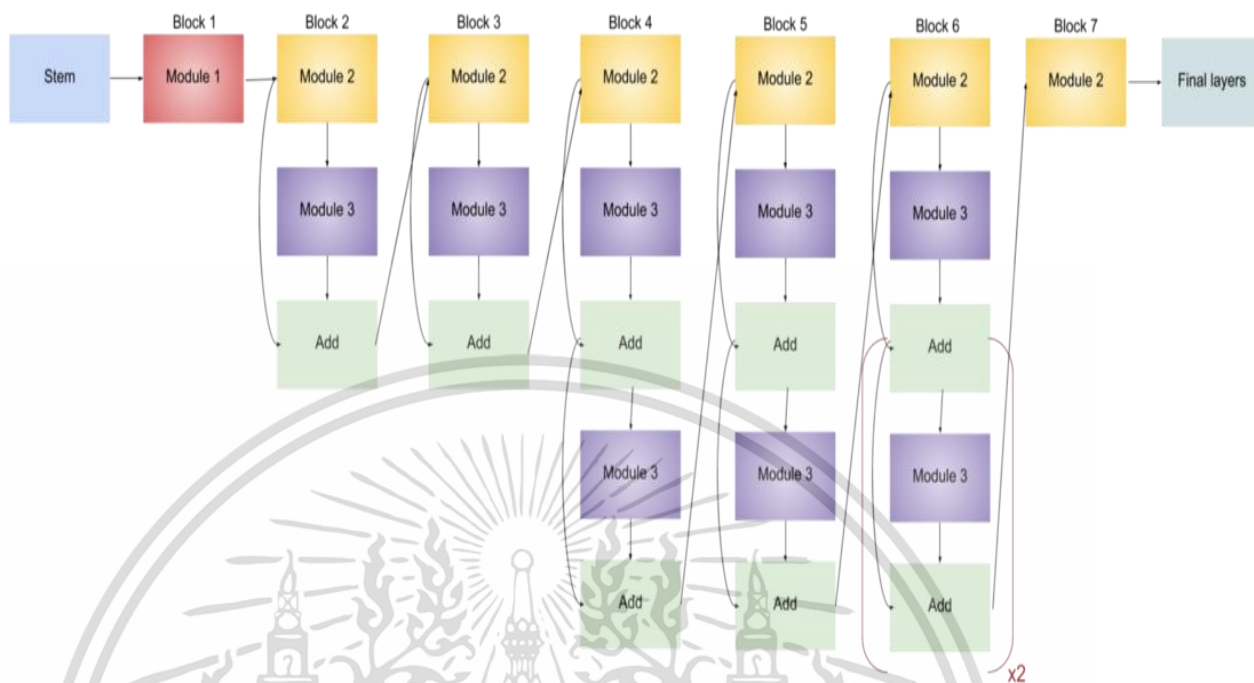
และ EfficientNet จะมี Module อีกจำนวน 5 Module ที่มีโครงสร้างแตกต่างกันตามรูปที่ 3.7



รูปที่ 3.7 แสดงโครงสร้างของ Module ทั้ง 5 ของ EfficientNet [37]

หลังจากนั้นนำ Module ต่าง ๆ มาเรียงกันซึ่งจะมีโครงสร้างต่างกันไปโดย EfficientNetB0 จะมีโครงสร้างดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดงโครงสร้างของโมเดล EfficientNetB0 โดยที่ x2 หมายถึง Module ภายในวงเล็บนี้เข้ามาซ้ำ 2 รอบ [37]

3.2.4 การตรวจสอบความถูกต้องและการใช้งานของโมเดล

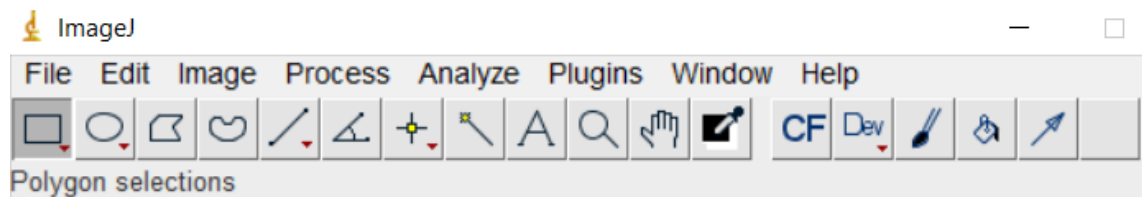
หลังจากที่ทำการเทรนโมเดลเรียบร้อยแล้ว ต้องมีการทดสอบโมเดลดังกล่าวและแสดงผลออกมาในรูปของตัวแปรต่าง ๆ ในวิทยานิพนธ์นี้จะแสดงผลออกมาด้วย confusion matrix ซึ่งจะใช้ภาษา Python และใช้ไลบรารี Pandas และ seaborn โดยการใช้ confusion matrix จะใช้กับชุดข้อมูลที่แยกออกมาเป็น validate set จำนวน 10% ของรูปภาพทั้งหมดซึ่ง จากนั้นนำ confusion matrix ของแต่ละโมเดลมาเปรียบเทียบเพื่อเลือกโมเดลที่มีความเหมาะสมมากที่สุด

3.2.5 การ FLIP ภาพและปรับพารามิเตอร์ต่าง ๆ

เนื่องจากจำนวนภาพทั้งหมดมีจำนวนที่น้อยดังนั้นเราจึงหาวิธีการเพิ่ม Dataset ให้มากยิ่งขึ้นด้วยการ FLIP ภาพซึ่งจะทำให้ข้อมูลสำหรับการสอนโมเดลมีจำนวนที่มากขึ้น โดยการพลิกกลับภาพนั้นจะใช้ ImageJ ซึ่งเป็นโปรแกรมที่สามารถใช้งาน image processing รวมถึงการ Analyze ต่าง ๆ ได้ซึ่งสามารถดาวน์โหลดได้จาก [38] ในงานวิจัยนี้จะใช้ ImageJ เพื่อทำการ Horizontal flip โดยมีวิธีการดังต่อไปนี้

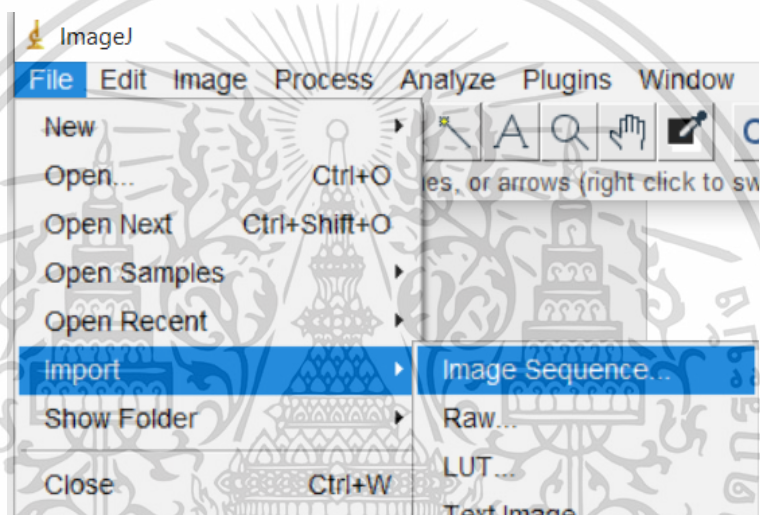
1. ทำการเปิด ImageJ ขึ้นมา โปรแกรมจะแสดงหน้าต่างดังรูปที่ 3.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 หน้าตาโปรแกรม ImageJ

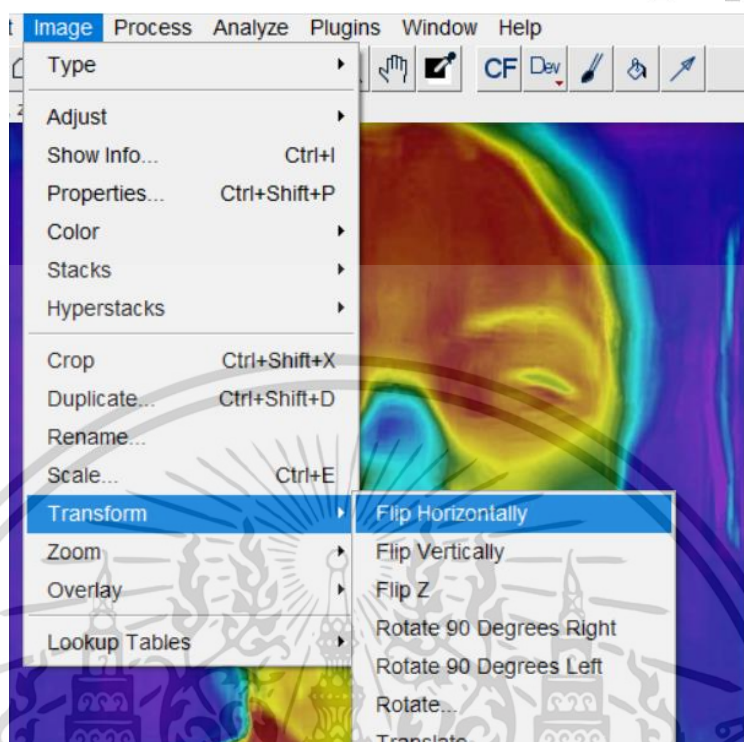
2. ทำการ import Image sequence ที่ต้องการ Horizontal Flip



รูปที่ 3.10 แสดงการ Import Image Sequence ใน ImageJ

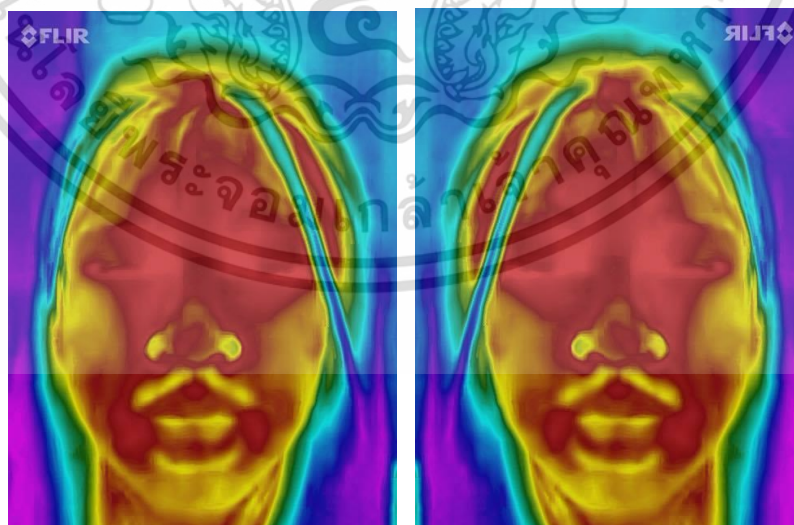
3. ทำการ Horizontal Flip จากคำสั่ง Image → transform

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 แสดงวิธีการ Flip Horizontally ในโปรแกรม ImageJ

ผลลัพธ์ในการ Horizontal flip จะเป็นดังรูปที่ 3.12 ซึ่งเปรียบเทียบระหว่างรูปก่อนพลิกกลับและหลังจากพลิกกลับแล้ว



รูปที่ 3.12 แสดงตัวอย่างรูปภาพที่ทำการพลิกกลับ โดยภาพซ้ายเป็นภาพตั้งต้น ภาพขวาเป็นภาพที่พลิกกลับแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากวิธีการพลิกกลับรูปภาพแล้ว ยังทำการปรับพารามิเตอร์ต่าง ๆ ในการสอนโมเดลการเรียนรู้เชิงลึก เพื่อให้มีค่าความถูกต้อง รวมถึงเหมาะสมต่องานนี้ ซึ่งพารามิเตอร์ต่าง ๆ เช่น Learning rate, training steps, Evaluate step interval, Number of epochs, Batch size

3.2.6 แปลงโมเดลเพื่อนำไปใช้ในแอปพลิเคชัน

โมเดลที่ได้หลังจากการเทรน จะเป็นไฟล์ 2 ไฟล์ output.pb และ output.txt ซึ่งไม่สามารถนำไปใช้กับ smartphone ได้ ดังนั้นจึงจำเป็นต้องทำการแปลงไฟล์ให้เป็นไฟล์ mlmodel ซึ่ง TensorFlow มีไลบรารีที่สามารถแปลงไฟล์ไปเป็น mlmodel ได้ซึ่งชื่อว่า tfcoreml ใช้กับภาษา python โดยใส่ input ดังนี้

- ที่อยู่ของโมเดลที่ต้องการจะแปลงเป็น mlmodel
- ชื่อไฟล์ mlmodel ที่ต้องการ
- ชนิดและขนาดของ input

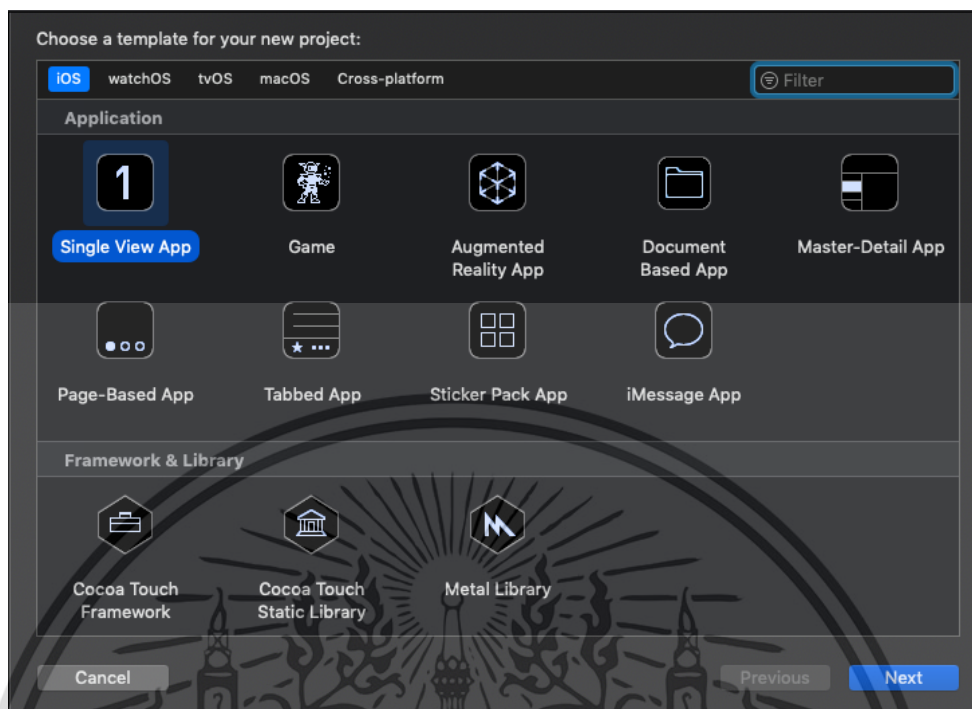
หมายเหตุ โปรแกรมภาษา python และวิธีลงไลบรารีอยู่ในภาคผนวก

3.2.7 สร้าง Machine Learning บน iOS ด้วย Core ML

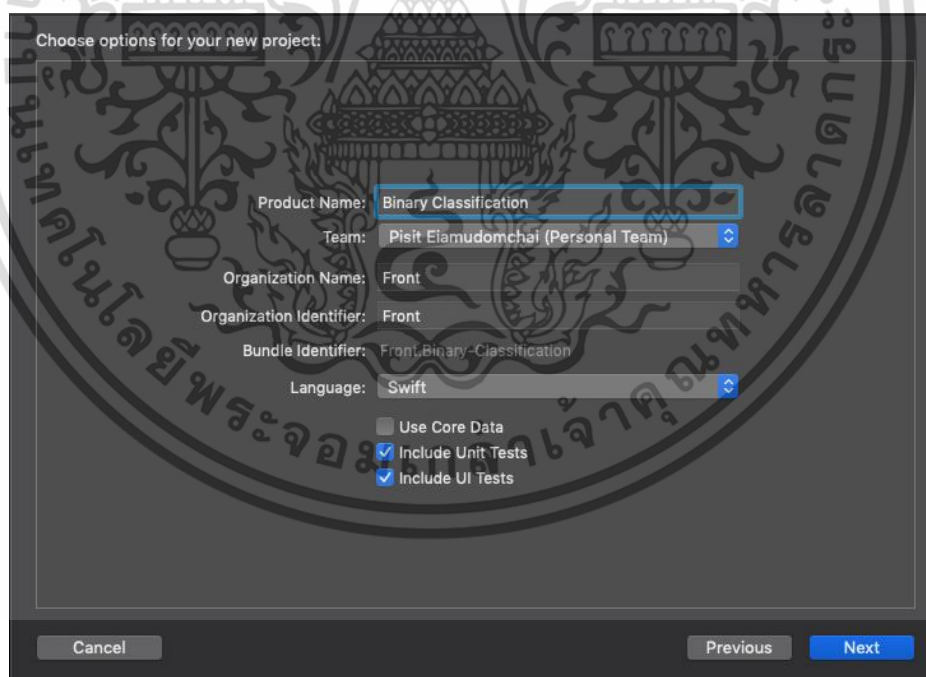
เป็นการทำให้สามารถรัน machine learning ไว้บน Mobile Device ได้ แต่โมเดลที่จะใช้จะต้องอยู่ใน format ของ Core ML ด้วยนั่นคือมีสกุลไฟล์ mlmodel ซึ่งได้ทำการสอนและแปลงโมเดลไว้แล้ว โดยการเขียนแอปพลิเคชัน จะใช้ภาษา Swift 5 ด้วยโปรแกรม Xcode 10.2

ซึ่งผู้จัดทำต้องการสร้าง 2 แอปพลิเคชัน คือ (1) แอปพลิเคชัน Binary Classification (2) แอปพลิเคชัน 4 level of Classification มีขั้นตอนดังนี้

1. Download Xcode 10.2 จาก <https://developer.apple.com/download/> และ install
2. สร้าง project



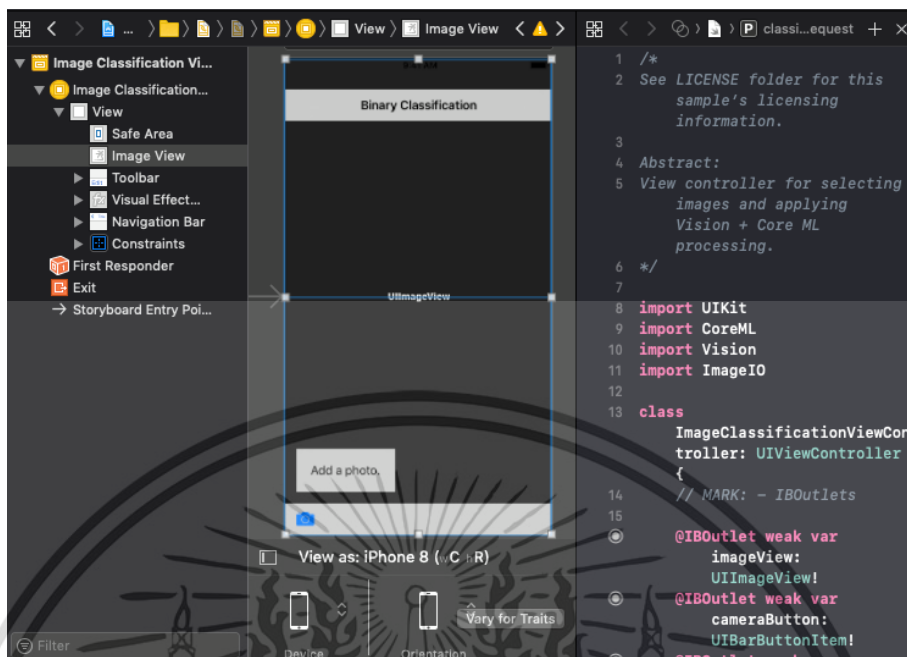
รูปที่ 3.13 เลือก Single View App



รูปที่ 3.14 ตั้งชื่อและเลือกภาษาที่ใช้เป็น Swift

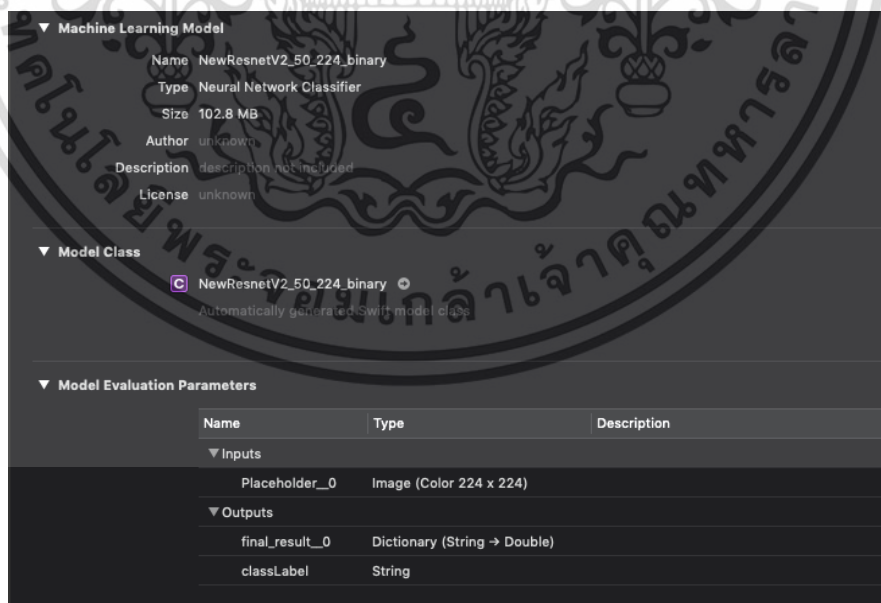
3. วาด View ใน Storyboard โดยมี UIImageView, UIBarButtonItem และ UILabel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 วาด View ใน Storyboard

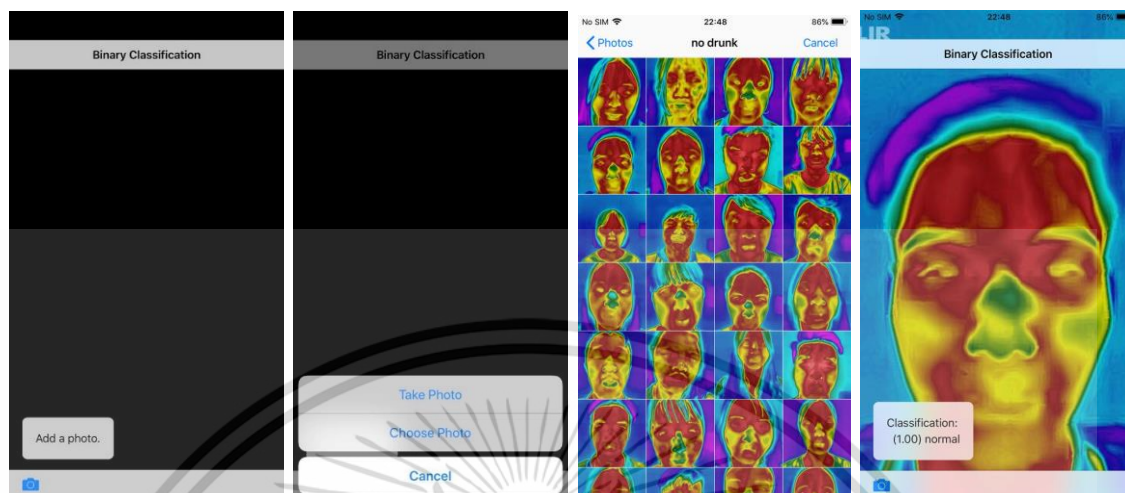
4. ทำการลากไฟล์ .mlmodel ที่ต้องการใช้เข้าไปใน project ดังรูป โดยโมเดล ResnetV2 ที่ train มา นั้น รับ input เป็น image color (RGB) ขนาด 224x224 ส่วน Output จะเป็นชื่อ Class ของรูปที่โมเดล ทำนายได้



รูปที่ 3.16 รายละเอียดของ ResnetV2.mlmodel ที่ Xcode อ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ติดตั้งโปรแกรมลงบน iOS ได้ดังรูป



รูปที่ 3.17 แต่ละหน้าของแอปพลิเคชันเมื่อติดตั้งลงบน iOS

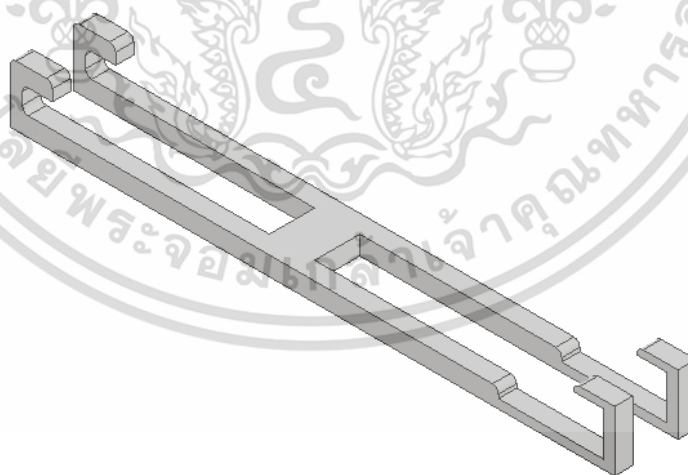
6. ทั้งสองแอปพลิเคชันมีขั้นตอนการสร้างเหมือนกัน แต่เปลี่ยนโมเดลที่จะใช้ในการทำนายของแต่ละแอปพลิเคชัน

หมายเหตุ : โปรแกรมที่ใช้อยู่ใน ภาคผนวก

3.3 ฮาร์ดแวร์

ใช้โปรแกรม Autodesk Inventor ในการออกแบบตัวยึดระหว่างโทรศัพท์และกล้องอินฟราเรด

ดังรูป



รูปที่ 3.18 CAD ตัวยึดระหว่างโทรศัพท์และกล้องอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

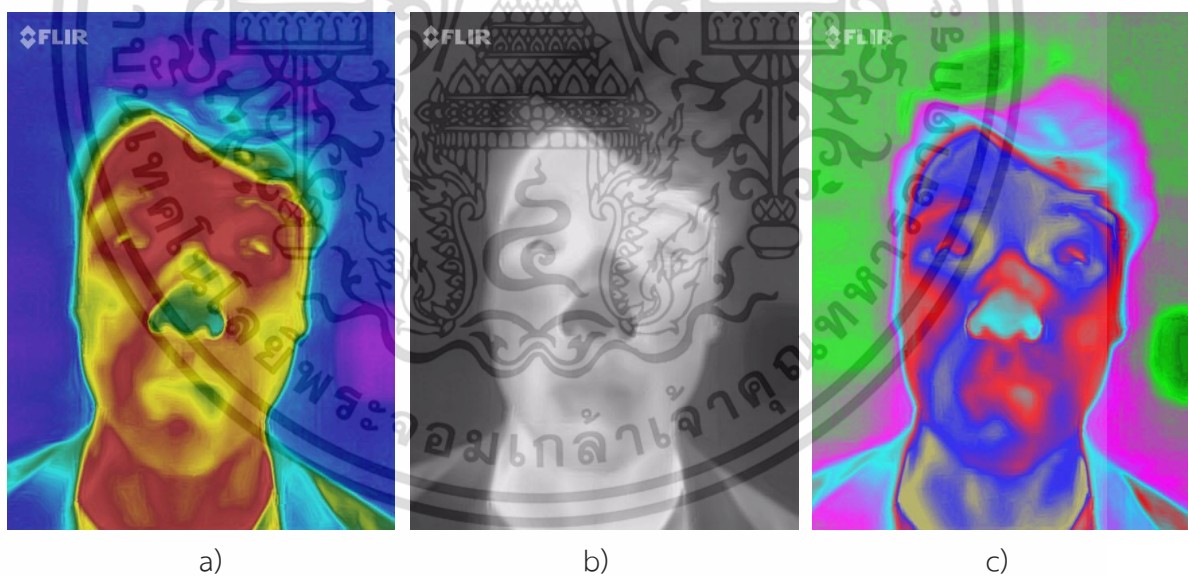
บทที่ 4

ผลการทดลองและอภิปรายผล

เทคโนโลยีการเรียนรู้เชิงลึกในการตรวจวัดปริมาณแอลกอฮอล์ด้วยกล้องอินฟราเรด นั้นเป็นระบบที่ทำนายบุคคลที่ดื่มแอลกอฮอล์และไม่ดื่มแอลกอฮอล์

4.1 การเก็บข้อมูลภาพ

จากการเก็บข้อมูลภาพ ได้รูปอินฟราเรดทั้งหมดประมาณ 200 รูป แบ่งเป็น 4 กลุ่ม คือ ไม่ดื่มแอลกอฮอล์ 50 รูป, ดื่ม 1 แก้ว 50 รูป, ดื่ม 2 แก้ว 50 รูป และดื่ม 3 แก้ว 50 รูป และมีการเก็บภาพคนที่ไม่ดื่มแอลกอฮอล์เพิ่มอีก สุดท้ายได้ภาพ ไม่ดื่มแอลกอฮอล์ 164 รูป, ดื่ม 1 แก้ว 35 รูป, ดื่ม 2 แก้ว 38 รูป, และดื่ม 3 แก้ว 38 รูป รวมทั้งหมดประมาณ 275 รูป ที่มีบางรูปหายไปเนื่องจากเราได้คัดแยกรูปที่ไม่สามารถใช้งานได้เช่น บางรูปใส่แว่นตอนถ่ายภาพ ใบหน้าอยู่ห่างจากกล้องเกินไป ใบหน้าอยู่ใกล้จากกล้องเกินไป หรือ เลือกโหมดในการถ่ายรูปผิด รูปอินฟราเรดทั้งหมดจะมีขนาด 480x640 pixels และมี resolution 96 dpi จากนั้นทำการเลือกรูปภาพที่โดยเลือกใช้ filter contrast, grey และ antarctic ของกล้องอินฟราเรด แสดงดังรูป



รูปที่ 4.1 a) filter contrast, b) filter grey, c) filter antarctic

จากการเก็บข้อมูลภาพทั้ง 3 filter ทางผู้จัดทำเลือกแค่ filter contrast เท่านั้นที่นำไปสอนโมเดลเนื่องจาก เมื่อสังเกตภาพโดยรวมแล้วจะเห็นความแตกต่างระหว่างภาพคนไม่ดื่มแอลกอฮอล์และคนดื่มแอลกอฮอล์ของ filter contrast มากที่สุด และเมื่อลองนำภาพทั้ง 3 filter ไปสอนโมเดลแล้วพบว่า filter contrast ก็มีค่าความแม่นยำมากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลจากการสอนโมเดลและการปรับพารามิเตอร์ต่าง ๆ

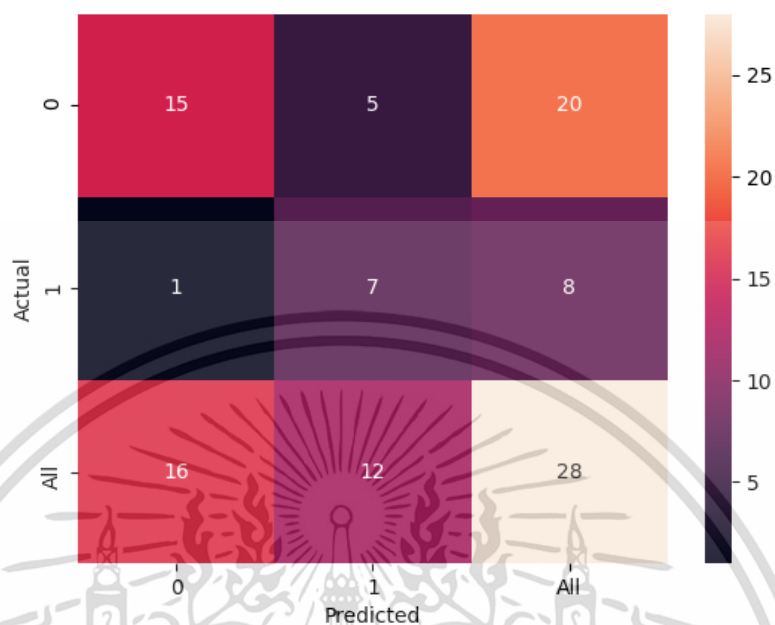
การสอนโมเดลในงานวิจัยนี้จะทำการสอนด้วยข้อมูล 90% และอีก 10% จะนำไปทดสอบโมเดล โดยจากรูปภาพทั้งหมด 275 รูปนั้น จะเป็นรูปภาพที่นำไปสอน 247 รูปและเป็นรูปที่นำไปทดสอบ 28 รูป โดยรูปทดสอบนั้นจะคละกันตามจำนวนประเภทที่มีอย่างละ 10% ผลการทดสอบโมเดลสำหรับ binary classification นั้นจะแสดงให้เห็นดังตารางที่ 4.1

โมเดลสำหรับ Binary classification

ชื่อโมเดล	ขนาดรูปภาพ	ความแม่นยำ
EfficientNetB0	224x224	78.57 %
mobilenet_v1_75_192	192x192	71.43 %
mobilenet_v1_75_224	224x224	71.43 %
EfficientNet-Lite4	300x300	71.43 %
mobilenet_v2_050_224	224x224	71.00 %

ตารางที่ 4.1 แสดงค่าความแม่นยำ 5 อันดับสูงสุดของโมเดลสำหรับ Binary classification

สำหรับโมเดล Binary Classification จะเลือกโมเดล EfficientNetB0 ซึ่งมีความแม่นยำ 78.57 % มีขนาดไฟล์ประมาณ 15.6 MB ในการสร้างโมเดล Binary Classification แยก output ออกเป็น 2 ประเภทคือ เมา กับ ไม่เมาโดยแบ่งให้ไม่เมาคือ Normal ซึ่งประกอบไปด้วยรูปภาพที่ไม่ดื่มแอลกอฮอล์ กับดื่มแอลกอฮอล์ 1 แก้วซึ่งจะมี BAC ไม่เกินกฎหมายกำหนด ส่วนอีกประเภทคือ เมา Drunk ประกอบไปด้วยรูปภาพที่ดื่มแอลกอฮอล์ 2 แก้วขึ้นไป ค่าความแม่นยำที่ได้จะวัดจาก confusion matrix ของการทดสอบโมเดลด้วยข้อมูลชุดทดสอบ 28 รูปที่ได้ทำการคัดแยกออกไปก่อนการสอนโมเดล โดยจะมีรายละเอียดดังรูป



รูปที่ 4.2 แสดง confusion matrix ของโมเดล EfficientNetB0 โดยที่ 0 คือไม่เมา และ 1 คือเมา

โดยจากผลลัพธ์ข้างต้นสามารถคำนวณแล้วเขียนออกมาเป็นค่าตัวแปรต่าง ๆ ได้เช่นค่า True positive, True negative, False positive, False negative, sensitivity, F1-score เป็นต้น ดังรูป

```

population: 28
P: 8
N: 20
PositiveTest: 12
NegativeTest: 16
TP: 7
TN: 15
FP: 5
FN: 1
TPR: 0.875
TNR: 0.75
PPV: 0.5833333333333334
NPV: 0.9375
FPR: 0.25
FDR: 0.4166666666666667
FNR: 0.125
ACC: 0.7857142857142857
F1_score: 0.7
MCC: 0.570544330734548
informedness: 0.625
markedness: 0.5208333333333335
prevalence: 0.2857142857142857
LRP: 3.5
LRN: 0.16666666666666666
DOR: 21.0
FOR: 0.0625

```

รูปที่ 4.3 แสดงค่าตัวแปรวัดผลต่าง ๆ ที่ได้จากโมเดล EfficientNetB0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งหากข้อมูลรูปภาพที่ใช้สอนแต่ละประเภทมีมากขึ้นทำให้ค่าความแม่นยำสูงขึ้นตาม แต่ค่าความแม่นยำ 78.57 % ก็ยังไม่เป็นที่น่าพอใจ ดังนั้นถ้าหากว่าเพิ่มจำนวนข้อมูลรูปภาพอีกก็จะสามารถเพิ่มความแม่นยำให้มากขึ้นอีกได้ โดยโมเดล EfficientNetB0 มีจำนวนเลเยอร์ทั้งหมด 967 เลเยอร์ ดังนี้

Placeholder : 1
 Sub : 1
 RealDiv : 1
 PlaceholderWithDefault : 1
 MatMul : 1
 AddV2 : 1
 Softmax : 1
 Add : 9
 DepthwiseConv2dNative : 16
 Mean : 17
 BiasAdd : 32
 FusedBatchNormV3 : 49
 Conv2D : 65
 Sigmoid : 65
 Mul : 66
 Identity : 311
 Const : 331

โมเดลสำหรับ 4Level of classification

ชื่อโมเดล	ขนาดรูปภาพ	ความแม่นยำ
mobilenet_v1_025_224	224x224	57.14 %
mobilenet_v1_75_224	224x224	50.0 %
mobilenet_v1_025_128	128x128	50.0 %
EfficientNet-Lite0	224x224	46.42 %
EfficientNet-Lite4	300x300	46.42 %

ตารางที่ 4.2 แสดงค่าความแม่นยำ 5 อันดับสูงสุดของโมเดลสำหรับ 4level of Classification

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการสอนโมเดลจะเห็นได้ว่าโมเดลที่มีความแม่นยำมากที่สุดนั้นคือโมเดล mobilenet_v1_025_224 ซึ่งมีค่าความแม่นยำอยู่ที่ 57.14 % และมีขนาดไฟล์ประมาณ 1.03 MB ค่าความแม่นยำที่ได้จะวัดจาก confusion matrix ของการทดสอบโมเดลด้วยข้อมูลชุดทดสอบ 28 รูปที่ได้ทำการคัดแยกออกไปก่อนการสอนโมเดล โดยจะมีรายละเอียดดังรูป



รูปที่ 4.4 แสดง confusion matrix ของโมเดล mobilenet_v1_025_224 โดยเลขในทั้งแกนตั้งและนอนแทนจำนวนแก้วของแอลกอฮอล์ เช่น 0 แก้ว, 1 แก้ว เป็นต้น

และจากการทดสอบโมเดลก็สามารถคำนวณตัวแปรในการวัดผลต่าง ๆ ออกมาได้ดังรูป

Class Statistics:				
Classes	0	1	2	3
Population	28	28	28	28
P: Condition positive	15	4	4	5
N: Condition negative	13	24	24	23
Test outcome positive	15	5	5	3
Test outcome negative	13	23	23	25
TP: True Positive	11	1	2	2
TN: True Negative	9	20	21	22
FP: False Positive	4	4	3	1
FN: False Negative	4	3	2	3
TPR: (Sensitivity, hit rate, recall)	0.733333	0.25	0.5	0.4
TNR=SPC: (Specificity)	0.692308	0.833333	0.875	0.956522
PPV: Pos Pred Value (Precision)	0.733333	0.2	0.4	0.666667
NPV: Neg Pred Value	0.692308	0.869565	0.913043	0.88
FPR: False-out	0.307692	0.166667	0.125	0.0434783
FDR: False Discovery Rate	0.266667	0.8	0.6	0.333333
FNR: Miss Rate	0.266667	0.75	0.5	0.6
ACC: Accuracy	0.714286	0.75	0.821429	0.857143
F1 score	0.733333	0.222222	0.444444	0.5
MCC: Matthews correlation coefficient	0.425641	0.0761387	0.342624	0.441473
Informedness	0.425641	0.0833333	0.375	0.356522
Markedness	0.425641	0.0695652	0.313043	0.546667
Prevalence	0.535714	0.142857	0.142857	0.178571
LR+: Positive likelihood ratio	2.38333	1.5	4	9.2
LR-: Negative likelihood ratio	0.385185	0.9	0.571429	0.627273
DOR: Diagnostic odds ratio	6.1875	1.66667	7	14.6667
FOR: False omission rate	0.307692	0.130435	0.0869565	0.12

รูปที่ 4.5 แสดงค่าตัวแปรวัดผลต่าง ๆ ที่ได้จากโมเดล mobilenet_v1_025_244

โดยโมเดล mobilenet_v1_025_244 มีจำนวนเลเยอร์ทั้งหมด 863 เลเยอร์ดังนี้

Placeholder : 1

AvgPool : 1

Squeeze : 1

PlaceholderWithDefault : 1

MatMul : 1

AddV2 : 1

Softmax : 1

FakeQuantWithMinMaxArgs : 2

Reshape : 13

DepthwiseConv2dNative : 13

Conv2D : 14

Rsqrt : 27

Relu6 : 27

Sub : 28

Add : 54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FakeQuantWithMinMaxVars : 56

Mul : 82

Identity : 250

Const : 291

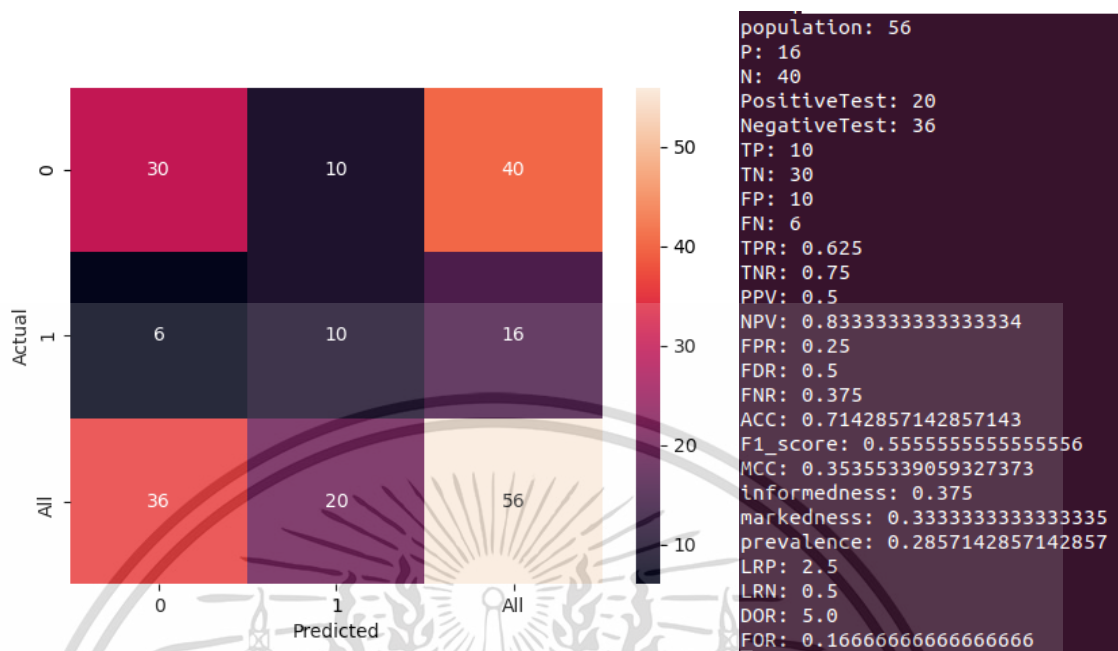
จากผลการทดสอบโมเดลข้างต้นจะสังเกตได้ว่าความแม่นยำที่ไม่สูงมากคาดว่าเนื่องจากหลายปัจจัยการสร้างโมเดล 4 level of Classification เป็นโมเดลที่แยก output ออกเป็น 4 ประเภทจำเป็นที่จะต้องใช้อ้อมรูปภาพที่เยอะ โดยรูปทั้งหมดที่มีประมาณ 275 รูป ถ้าแบ่งการสอนออกเป็น 4 ประเภท จะเห็นว่าแต่ละคลาสนั้นมีจำนวนข้อมูลประมาณ 35 รูป ยกเว้นคนที่ไม่ดื่มแอลกอฮอล์ที่มีรูป 160 รูป จากนั้นรูปทั้งหมดยังต้องแบ่งออกเป็น 4 ส่วน Test set 10% validation 10% unseen test set อีก 10% จึงเหลือรูปที่ใช้สอนโมเดลจริง ๆ แค่ 70 % ดังนั้นค่าความแม่นยำจึงไม่สูงมาก รวมถึง unseen test set นั้นมีจำนวนรูปภาพแค่เพียง 28 รูปเท่านั้น แต่หากสังเกตค่า sensitivity หรือ True positive rate จะเห็นว่าค่าของคลาส 0 แก้วนั้นค่อนข้างดีในขณะที่คลาส 1, 2 และ 3 นั้นมีค่าน้อยมากซึ่งเป็นเพราะจำนวนข้อมูลที่ใช้ทดสอบโมเดลของคลาส 1 ถึง 3 นั้นมีจำนวนที่น้อยเมื่อเทียบกับคลาส 0 ซึ่งมีถึง 160 รูป ดังนั้นทางผู้จัดทำคิดว่าหากมีการเก็บข้อมูลรูปภาพเพิ่มมากขึ้นจะสามารถที่จะเพิ่มประสิทธิภาพของโมเดลขึ้นได้

จากผลการสอนโมเดลข้างต้นจึงรู้ถึงปัญหาของจำนวนรูปภาพซึ่งมีจำนวนที่น้อย จึงเกิดการทดสอบการพลิกกลับของรูปภาพหรือ Horizontal Flip ซึ่งเริ่มต้นจากแยกรูปภาพ Test set ออกไปก่อน 20% ประมาณ 56 รูป (N=56) จากนั้นก็นำรูปภาพสำหรับสอนโมเดลไปทำการ Flip ในโปรแกรม ImageJ ผลลัพธ์ที่ได้จะมีจำนวนรูปที่ใช้สอนโมเดลทั้งหมด 438 รูป (N=438) จากนั้นเลือกโมเดลที่ดีที่สุดจากผลการทดสอบข้างต้นมา 3 โมเดลแล้วทำการสอนโมเดลใหม่โดยใช้รูปภาพที่ Flip แล้วจากนั้นนำไปทดสอบโมเดลด้วยชุดรูปภาพ Test set (N=56) เพื่อดูประสิทธิภาพของโมเดลที่ได้ผลลัพธ์ที่ได้เป็นดังตารางที่ 4.3 และแสดง confusion matrix กับค่าตัวแปรวัดผลต่าง ๆ ของโมเดลที่มีความแม่นยำมากที่สุดในรูปแบบที่ 4.6

โมเดลสำหรับ Binary classification ที่มีการ FLIP ข้อมูลแล้ว

ชื่อโมเดล	ขนาดรูปภาพ	ความแม่นยำ
EfficientNetB0	224x224	71.43 %
mobilenet_v1_75_192	192x,192	69.64 %
EfficientNet-Lite4	300x300	64.28 %

ตารางที่ 4.3 แสดงค่าความแม่นยำสูงสุด 3 อันดับของ Binary classification โมเดลที่ทำการสอนกับข้อมูลที่ FLIP แล้ว



รูปที่ 4.6 แสดง confusion matrix และค่าตัวแปรวัดผลต่าง ๆ ของโมเดล EfficientNetB0 ที่สอนโดยรูปภาพที่ FLIP แล้ว โดยที่ 0 คือไม่เมา และ 1 คือเมา

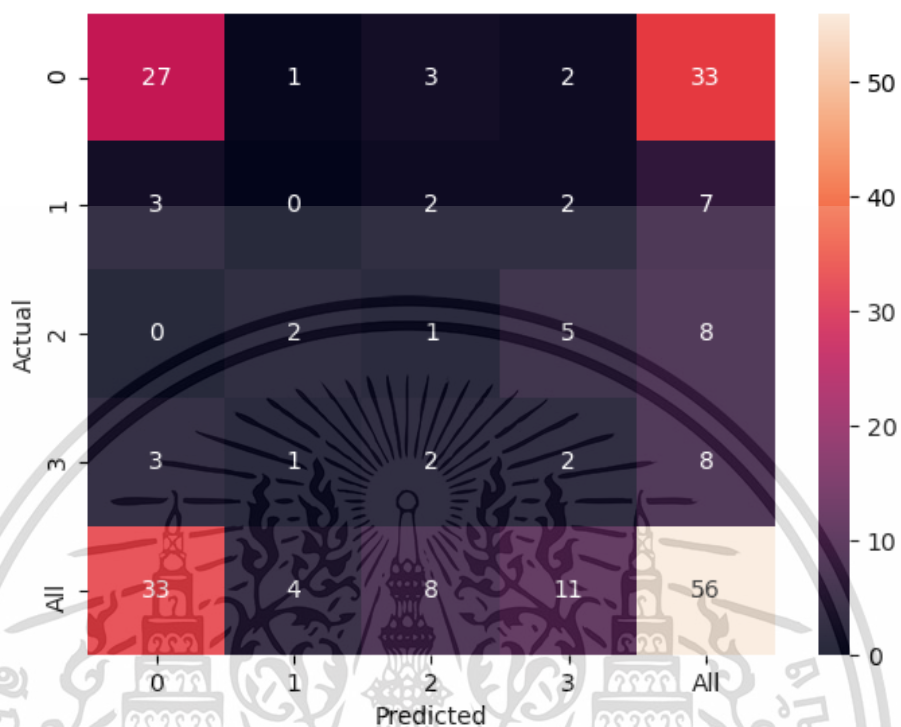
จากนั้นทำการทดสอบกับโมเดล 4 level of classification จึงได้ผลความแม่นยำดังตารางที่ 4.4 และ confusion matrix กับตัวแปรวัดผลต่าง ๆ ของโมเดลที่มีความแม่นยำมากที่สุดในรูปที่ 4.7 และ 4.8 ตามลำดับ

โมเดลสำหรับ 4Level of classification ที่มีการ FLIP ข้อมูลแล้ว

ชื่อโมเดล	ขนาดรูปภาพ	ความแม่นยำ
mobilenet_v1_025_224	224x224	44.46 %
mobilenet_v1_75_224	224x224	53.57 %
mobilenet_v1_025_128	128x128	41.07 %

ตารางที่ 4.4 แสดงค่าความแม่นยำสูงสุด 3 อันดับของ 4Level of classification โมเดลที่ทำการสอนกับข้อมูลที่ FLIP แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

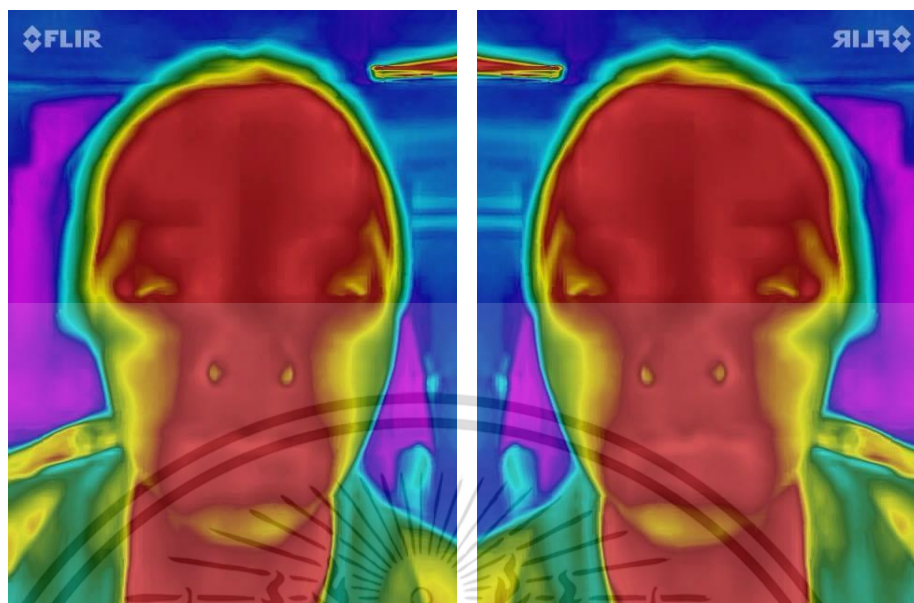


รูปที่ 4.7 แสดง confusion matrix ของโมเดล mobilenet_v1_075_244 ที่สอนโดยรูปภาพที่ FLIP แล้วโดยเลขในทั้งแกนนตั้งและนอนแทนจำนวนแก้วของแอลกอฮอล์ เช่น 0 แก้ว, 1 แก้ว เป็นต้น

Class Statistics:				
Classes	0	1	2	3
Population	56	56	56	56
P: Condition positive	33	7	8	8
N: Condition negative	23	49	48	48
Test outcome positive	33	4	8	11
Test outcome negative	23	52	48	45
TP: True Positive	27	0	1	2
TN: True Negative	17	45	41	39
FP: False Positive	6	4	7	9
FN: False Negative	6	7	7	6
TPR: (Sensitivity, hit rate, recall)	0.818182	0	0.125	0.25
TNR=SPC: (Specificity)	0.73913	0.918367	0.854167	0.8125
PPV: Pos Pred Value (Precision)	0.818182	0	0.125	0.181818
NPV: Neg Pred Value	0.73913	0.865385	0.854167	0.866667
FPR: False-out	0.26087	0.0816327	0.145833	0.1875
FDR: False Discovery Rate	0.181818	1	0.875	0.818182
FNR: Miss Rate	0.181818	1	0.875	0.75
ACC: Accuracy	0.785714	0.803571	0.75	0.732143
F1 score	0.818182	0	0.125	0.210526
MCC: Matthews correlation coefficient	0.557312	-0.104828	-0.0208333	0.0550482
Informedness	0.557312	-0.0816327	-0.0208333	0.0625
Markedness	0.557312	-0.134615	-0.0208333	0.0484848
Prevalence	0.589286	0.125	0.142857	0.142857
LR+: Positive likelihood ratio	3.13636	0	0.857143	1.33333
LR-: Negative likelihood ratio	0.245989	1.08889	1.02439	0.923077
DOR: Diagnostic odds ratio	12.75	0	0.836735	1.44444
FOR: False omission rate	0.26087	0.134615	0.145833	0.133333

รูปที่ 4.8 แสดงค่าตัวแปรวัดผลต่าง ๆ ที่ได้จากโมเดล mobilenet_v1_075_244 ที่สอนโดยรูปภาพที่ FLIP แล้ว

จากผลลัพธ์ที่แสดงทั้ง Binary classification และ 4Level of classification จะเห็นได้ว่าหลังจากสอนด้วยข้อมูลที่มีการ Flip แล้วนั้นไม่มีการเพิ่มขึ้นอย่างมีนัยยะสำคัญของประสิทธิภาพของโมเดล ซึ่งอาจเป็นเพราะความแตกต่างของรูปภาพปกติกับรูปภาพที่ทำการ Flip แล้วนั้นไม่มีความแตกต่างอย่างมีนัยยะสำคัญบางรูปแบบไม่มีความแตกต่างกันเลยดังนั้นในงานวิจัยนี้การ Flip รูปภาพไม่มีผลต่อประสิทธิภาพของโมเดล



รูปที่ 4.9 แสดงตัวอย่างของรูปภาพ (a) รูปภาพก่อน Flip (b) รูปภาพหลัง Flip ซึ่งไม่มีความแตกต่างที่ส่งผลต่อประสิทธิภาพของโมเดล

จากนั้นทำการทดสอบการปรับ hyperparameter ต่าง ๆ ของการสอนโมเดล เช่น Learning rate, Training step และ Train batch size โดยทดสอบกับโมเดล EfficientNetB0 ของ Binary classification ซึ่งได้ผลลัพธ์ดังตาราง 4.5, 4.6 และ 4.7

Learning rate	Training step	Train batch size	Accuracy
0.1	4000	100	75%
0.01	4000	100	78.57
0.001	4000	100	75%
0.0001	4000	100	75%

ตารางที่ 4.5 แสดงผลการปรับ Learning rate โดยค่าอื่น ๆ คงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Learning rate	Training step	Train batch size	Accuracy
0.01	500	100	75%
0.01	1000	100	75%
0.01	2500	100	75%
0.01	6000	100	75%
0.01	8000	100	75%

ตารางที่ 4.6 แสดงผลการปรับ Training step โดยค่าอื่น ๆ คงที่

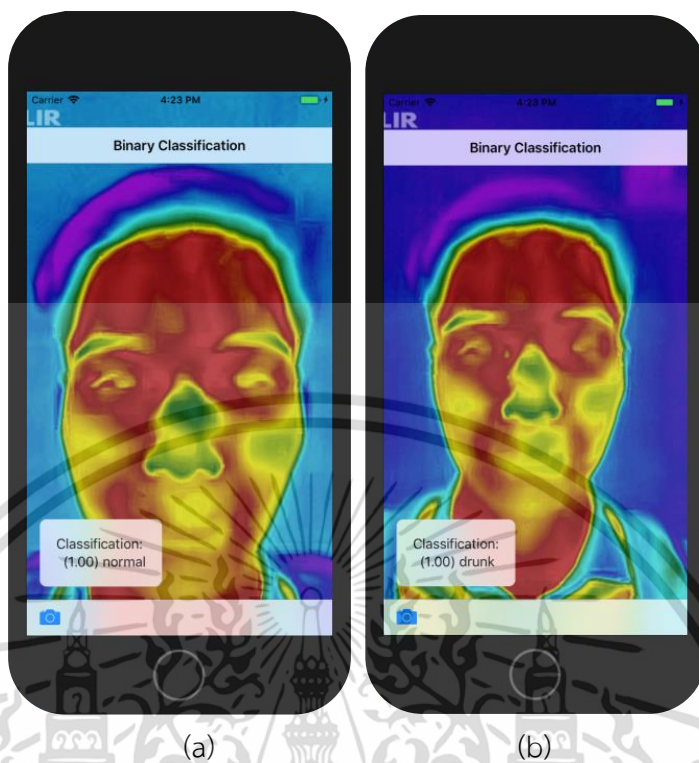
Learning rate	Training step	Train batch size	Accuracy
0.01	4000	2	78.57%
0.01	4000	10	78.57%
0.01	4000	50	78.57%
0.01	4000	150	78.57%
0.01	4000	200	75%

ตารางที่ 4.7 แสดงผลการปรับ Train batch size โดยค่าอื่น ๆ คงที่

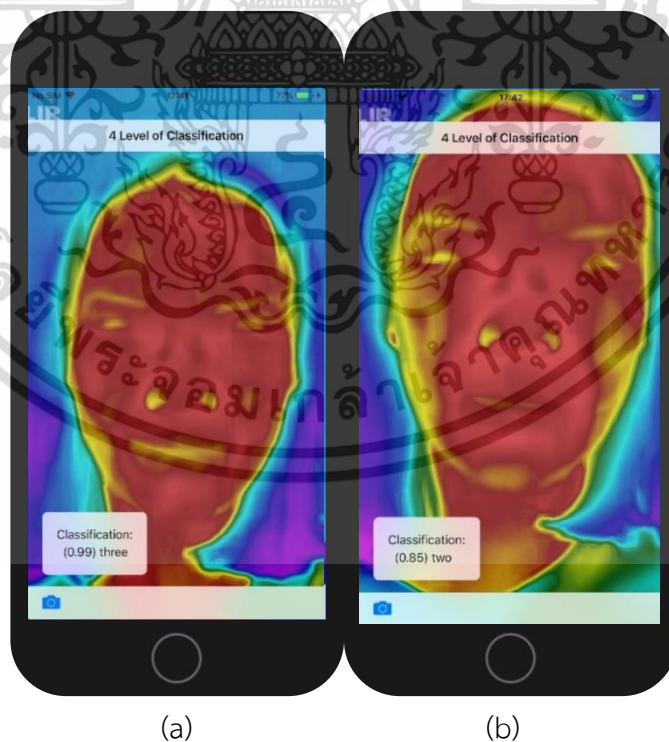
จากผลการปรับ hyperparameter ต่าง ๆ แสดงให้เห็นว่าค่าที่เหมาะสมกับชุดข้อมูลของเรามี Learning rate ที่ 0.01 ส่วน Training step เป็น 4000 ส่วน Train batch size สามารถใช้ช่วง 100 ได้

4.3 ผลการทดสอบการประมวลผลของ CoreML บน iOS

แอปพลิเคชันมีสองแอปพลิเคชัน, Binary Classification, 4 level of Classification มีตัวอย่างการทำงานดังรูปที่ 4.10 และ 4.11 ตามลำดับโดยภายในแอปพลิเคชันจะแสดงชื่อแอปพลิเคชัน รูปภาพที่ทำการคัดแยกประเภท ผลของการทำนายพร้อมทั้งค่าความน่าจะเป็น และปุ่มสำหรับเลือกรูปภาพเพื่อนำมาทำนาย



รูปที่ 4.10 การทำงานของแอปพลิเคชัน Binary Classification (a) แสดงผลลัพธ์ไม่เมา (b)แสดงผลลัพธ์เมา



รูปที่ 4.11 การทำงานของแอปพลิเคชัน 4 Level of Classification (a)แสดงผลลัพธ์ 3 แก้ว (b) แสดงผลลัพธ์ 2 แก้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การประกอบฮาร์ดแวร์

หลังจากทำการออกแบบตัวยึดโทรศัพท์กับกล้องอินฟราเรดก็ทำการพิมพ์แบบสามมิติออกมาแล้วนำไปประกอบกับโทรศัพท์เพื่อป้องกันไม่ให้โทรศัพท์กับกล้องอินฟราเรดหลุดออกจากกันดังรูปที่ 4.12, 4.13 และ 4.14



รูปที่ 4.12 มุมด้านหลังเมื่อประกอบตัวยึด



รูปที่ 4.13 มุมด้านหน้าเมื่อประกอบตัวยึด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 มุมด้านข้างเมื่อประกอบตัวยึด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผล

5.1 สรุปผลการวิจัย

จากการวิจัยผลการทดลองเริ่มจากข้อมูลรูปภาพอินฟราเรดทั้งหมด 275 รูปแบ่งเป็น ไม่ต้ม แอลกอฮอล์ 164 รูป, ต้ม 1 แก้ว 35 รูป, ต้ม 2 แก้ว 38 รูป, และต้ม 3 แก้ว 38 รูป ซึ่งหากระยะห่างของหน้าคนกับกล้องห่างกันเกินไปก็จะเกิดความผิดพลาดของข้อมูลเนื่องจากรังสีอินฟราเรดจะมีการเปลี่ยนแปลงไปส่วนด้านโปรแกรมที่ทำนายการต้มแอลกอฮอล์ โดยมุ่งเน้นให้โมเดลสามารถทำนายได้ว่าอาสาสมัครดื่มเบียร์ไปกี่แก้ว เช่น 1 แก้ว, 2 แก้ว, 3 แก้ว หรือไม่ได้ดื่มเลยเป็นหลัก ซึ่งจาก Widmark formula จะสามารถคำนวณปริมาณแอลกอฮอล์ในเลือดหรือ BAC ได้โดยประมาณคือ 1 แก้ว ประมาณ 0.026% แอลกอฮอล์ประมาณ 10 กรัม, 2 แก้วก็จะเป็นปริมาณ BAC 0.052% โดยประมาณ และยังมีอีกหนึ่งโมเดลที่สามารถทำนายได้ว่าอาสาสมัครเมาหรือไม่เมา ซึ่งวิธีการสร้างโมเดลการเรียนรู้เชิงลึกประกอบไปด้วยขั้นตอนการสอน ขั้นตอนการตรวจสอบ และขั้นตอนการทดสอบ จากการทดลองเมื่อนำโมเดลที่ถูกสอนแล้วไปผ่านขั้นตอนการสอน และขั้นตอนการตรวจสอบ แล้วพบว่าโมเดล EfficientNetB0 คือโมเดลที่มีความแม่นยำมากที่สุด สำหรับ Binary Classification ที่แยกระหว่างไม่เมา (0-1 แก้ว) กับ เมา (2-3) แก้ว ซึ่งมีค่าความแม่นยำอยู่ที่ 78.57% ส่วนโมเดลที่มีความแม่นยำมากที่สุด สำหรับ 4 level of Classification ที่แยกระหว่างไม่ดื่มแอลกอฮอล์, ต้ม 1 แก้ว, 2 แก้ว และ 3 แก้วซึ่งคือโมเดลที่ชื่อว่า mobilenet_v1_025_224 มีค่าความแม่นยำอยู่ที่ 57.14%

จากนั้นทำการเพิ่มจำนวนข้อมูลรูปภาพอินฟราเรดด้วยการพลิกกลับซึ่งทำให้ได้รูปสำหรับสอนโมเดลทั้งหมด 438 รูป แต่ว่าหลังจากทำการสอนโมเดลด้วยชุดข้อมูลดังกล่าวก็ไม่พบการเพิ่มขึ้นอย่างมีนัยยะสำคัญของประสิทธิภาพของโมเดล ซึ่งเกิดเนื่องจากการพลิกกลับรูปภาพไม่ได้มีความแตกต่างกันมากนัก คล้ายกับเป็นรูปที่สมมาตรกันจึงไม่มีผลอย่างมีนัยยะสำคัญกับประสิทธิภาพของโมเดล อีกทั้งทำการทดสอบการปรับแต่งตัวแปรในการสอนโมเดลด้วยค่าต่าง ๆ สุดท้ายได้ค่าดังนี้ Learning rate = 0.01, Training step = 4000 ส่วน Train batch size สามารถใช้ช่วงค่าประมาณ 100 ได้

การทำระบบประมวลผลของ CoreML บน iOS มีทั้งหมด 2 แอปพลิเคชัน คือ Binary Classification และ 4 Level of Classification โดยเลือกใช้โมเดลที่มีค่าความแม่นยำสูงที่สุดจากการใช้ Transfer learning ผ่านไลบรารี TensorFlow และแปลงโมเดลเป็นโมเดลที่รองรับ CoreML ซึ่งแอปพลิเคชัน Binary Classification ใช้โมเดล EfficientNetB0 ส่วนแอปพลิเคชัน 4 Level of Classification ใช้โมเดล mobilenet_v1_025_224 เมื่อทำการติดตั้งแอปพลิเคชันลงบน iOS แล้ว แอปพลิเคชันสามารถใช้งานได้ปกติ มีการประมวลผลรวดเร็ว แต่เมื่อทดลองใช้งานแอปพลิเคชันแล้วพบว่าผลที่ได้ไม่ค่อยมีความแม่นยำ เนื่องจากมีรูปในการสอนโมเดลน้อยเกินไป

แต่แอปพลิเคชันไม่สามารถถ่ายรูปจากกล้องอินฟราเรดได้เนื่องจาก API สำหรับการเรียกใช้งานกล้องอินฟราเรด FLIR One Pro รองรับแค่ภาษา Objective-C ส่วนในการเขียนแอปพลิเคชันทำนายการตี้มเปียร์จำเป็นต้องใช้ Framework ของ CoreML ซึ่งรองรับเฉพาะภาษา Swift และใช้ได้แค่ใน iOS 12.0 ขึ้นไป

อย่างไรก็ตามเทคโนโลยีการเรียนรู้เชิงลึกบนแอปพลิเคชันสำหรับการตรวจวัดแอลกอฮอล์ด้วยกล้องอินฟราเรดนั้นก็ยังมีข้อจำกัดต่าง ๆ อยู่บางอย่างก็คือ แอปพลิเคชันนี้แยกประเภทออกเป็นการตี้มเปียร์จำนวนกี่แก้วหรือว่าใช้ทำนายว่าเมาหรือไม่เมา แม้ว่าจะสามารถคำนวณปริมาณ BAC ได้ว่า 1 แก้วประมาณ 0.026% แต่ก็ยังขึ้นอยู่กับเพศ น้ำหนักและอัตราการเผาผลาญของแต่ละบุคคล และเนื่องจากเทคโนโลยีนี้ใช้กล้องอินฟราเรดเป็นตัวถ่ายภาพซึ่งจะอ้างอิงถึงรังสีความร้อนที่แผ่ออกมาจากใบหน้าของบุคคลดังนั้นหากว่าบุคคลนั้นมีไข้ ตัวร้อนหรือ ออกกำลังกาย อยู่กลางแจ้ง ก็อาจจะทำให้การใช้งานโมเดลการเรียนรู้เชิงลึกไม่มีประสิทธิภาพเท่าที่ควร อีกทั้งคนที่สุขภาพร่างกายแข็งแรงทั่วไปนั้นไม่ค่อยมีความแตกต่างมากนักในการตี้มแอลกอฮอล์ 1, 2 หรือ 3 แก้ว ซึ่งทำให้การทำนายของโมเดลนั้นเป็นไปได้ยากสังเกตได้จากประสิทธิภาพของโมเดล 4Level of classification นั้นน้อยกว่าเมื่อเทียบกับ Binary classification และสุดท้ายก็คือข้อมูลรูปภาพอินฟราเรดทั้งหมดนั้นเก็บจากบุคคลสัญชาติไทยอายุประมาณ 20-24 ปี ซึ่งหากนำโมเดลนี้ไปใช้กับคนยุโรปหรือคนอเมริกาอาจไม่ได้ประสิทธิภาพเท่าที่กล่าวมา

5.2 ข้อเสนอแนะ

จากข้อจำกัดของกล้องอินฟราเรดหาคะยะห่างระหว่างใบหน้าและกล้องต่างกันข้อมูลที่ได้อาจจะเปลี่ยนไปจึงมีข้อเสนอแนะดังนี้

1. สร้างชุดคล้ายกล้องในการเก็บข้อมูลเพื่อให้ไม่มีแสงจากภายนอกเข้ามารบกวนรังสีอินฟราเรด
 2. เพิ่มกรอบในแอปพลิเคชันการถ่ายรูปเพื่อให้วางใบหน้าในระยะและบริเวณที่ใกล้เคียงกัน ไม่มากไม่น้อยเกินไป
 3. ควรทำการทดสอบ Repeatability เพื่อดู feature อื่นๆ ที่อาจเกี่ยวข้องกับการทำนายของโมเดลวิธีการ เช่น ให้อาสาสมัครคนเดิม ตีมีเปียร์ 1 แก้ว แล้วถ่ายรูป เป็นเวลา 10 วันเพื่อดูว่าแต่ละวันมีความแตกต่างกันอย่างไร มี feature อะไรที่มีผลกระทบอีกบ้าง
 4. ควรที่จะเพิ่ม feature อื่น ๆ มาใส่ในโมเดลไม่ว่าจะเป็น สถานที่ที่ถ่าย, อุณหภูมิร่างกายโดยปกติ, อุณหภูมิของสภาพแวดล้อม, อายุ หรือ เพศ ของอาสาสมัคร เป็นต้น
- ส่วนปัญหาที่เกิดจากโมเดลการทำนายทั้ง Binary และ 4Level of classification ที่ทางผู้จัดทำพัฒนามีประสิทธิภาพที่น้อยจึงต้องเพิ่มประสิทธิภาพของโมเดลมากขึ้นสามารถทำได้โดยดังนี้

1. ควรเพิ่มจำนวนอาสาสมัคร เพื่อให้มีฐานข้อมูลในการสอนโมเดลมากขึ้นและทำให้โมเดลรู้จักข้อมูลที่มีความหลากหลาย
2. ในขั้นตอนการเก็บข้อมูลภาพของอาสาสมัคร ควรถ่ายรูปภาพหลากหลายมุมและหลายรูปในเวลาเดียวกันเพื่อให้ได้ฐานข้อมูลมากขึ้น
3. ทำการสร้างโมเดลซึ่งมีความเฉพาะกับข้อมูลที่ใช้ให้มากที่สุด

ในส่วนของแอปพลิเคชันก็มีข้อเสนอแนะโดยที่แอปพลิเคชันยังไม่สามารถรองรับภาษา Swift ได้ทั้งหมดจึงควรเขียนแอปพลิเคชันในส่วนของการถ่ายรูปให้ใช้ภาษา Swift ได้ และควรที่จะปรับแต่งแอปพลิเคชันให้ใช้ได้ง่ายขึ้นกว่าเดิม

บรรณานุกรม

1. ยอดสถิติเมาแล้วขับ 3 วันพุ่ง 4,452 คดี [ออนไลน์]. 2562 [ค้นเมื่อ 16 มิถุนายน 2564]. จาก www.bangkokbiznews.com/news/detail/860193#:~:text=เมื่อเปรียบเทียบสถิติคดีที่,คิดเป็นร้อยละ 51.22
2. Drunk-spotting With an Infrared Camera [ออนไลน์]. 2555 [ค้นเมื่อ 27 มีนาคม 2564]. จาก <https://www.technologyreview.com/s/429640/drunk-spotting-with-an-infrared-camera/>.
3. How thermal cameras could be used to catch drink drivers: Algorithm spots changes in the face caused by an 'alcohol flush' [ออนไลน์]. 2557 [ค้นเมื่อ 1 มีนาคม 2564]. จาก <http://www.dailymail.co.uk/sciencetech/article-3062494/How-thermal-cameras-used-catch-drink-drivers-Algorithm-spots-changes-face-caused-alcohol-flush.html>.
4. Gade, R. and T.B. Moeslund, Thermal cameras and applications: a survey. Mach. Vision Appl., 2014. 25(1): p. 245–262.
5. Koukiou, G., G. Panagopoulos, and V. Anastassopoulos. Drunk person identification using thermal infrared images. in 2009 16th International Conference on Digital Signal Processing. 2009.
6. Hermosilla G, Verdugo JL, Farias G, Vera E, Pizarro F, Machuca M. Face Recognition and Drunk Classification Using Infrared Face Images. Journal of Sensors. 2018;2018:5813514.
7. What Is BAC? [ออนไลน์]. [ค้นเมื่อ 1 เมษายน 2564]. จาก <https://alcohol.stanford.edu/alcohol-drug-info/buzz-buzz/what-bac>
8. Guide to Standard Drinks [ออนไลน์]. 2559 [ค้นเมื่อ 1 เมษายน 2564]. จาก https://www.alcohol.org.nz/sites/default/files/documents/AL437%20Guide%20to%20Standard%20Drinks_APR%202016.pdf
9. How to Calculate Blood Alcohol Content (Widmark Formula) [ออนไลน์]. 2562 [ค้นเมื่อ 1 เมษายน 2564]. จาก [https://www.wikihow.com/Calculate-Blood-Alcohol-Content-\(Widmark-Formula\)](https://www.wikihow.com/Calculate-Blood-Alcohol-Content-(Widmark-Formula))
10. The Widmark Formula and Calculating Your BAC Level [ออนไลน์]. [ค้นเมื่อ 1 เมษายน 2564]. จาก <https://gambonelaw.com/faqs/the-widmark-formula-and-calculating-your-bac-level/>

11. Blood alcohol [ออนไลน์]. 2564 [ค้นเมื่อ 1 เมษายน 2564]. จาก https://en.wikipedia.org/wiki/Blood_alcohol_content
12. Infrared Knowledge [ออนไลน์]. 2560 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <http://www.flir.in.th/th/infrared-knowledge2.html>.
13. อินฟราเรด [ออนไลน์]. 2560 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <https://th.wikipedia.org/wiki/อินฟราเรด>.
14. โครงข่ายประสาทเทียม [ออนไลน์]. [ค้นเมื่อ 13 มิถุนายน 2564]. จาก <https://th.wikipedia.org/wiki/โครงข่ายประสาทเทียม>
15. โครงข่ายประสาทเทียมอัจฉริยะ (Artificial Neuron Network) [ออนไลน์]. [ค้นเมื่อ 13 มิถุนายน 2564]. จาก <https://csit.nu.ac.th/kraisak/ds/ds/chapter07/Chapter07.pdf>
16. Convolutional Neural Network [ออนไลน์]. 2563 [ค้นเมื่อ 13 มิถุนายน 2564]. จาก <https://guopai.github.io/ml-blog19.html>
17. Deep Learning คืออะไร [ออนไลน์]. 2560 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <https://blog.finnomena.com/deep-learning-คืออะไร-อาชีพไหนจะตกงานบ้าง-499c250784a1>.
18. Deep Learning (ดีพี เลินนิ่ง) [ออนไลน์]. 2561 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <http://www.mindphp.com/บทความ/240-ai-machine-learning/5278-deep-learning.html>
19. Deep Learning คืออะไร [ออนไลน์]. 2561 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <https://medium.com/@athivvat/deep-learning-คืออะไร-785e16d01773>.
20. ประเภทของ Machine Learning [ออนไลน์]. 2561 [ค้นเมื่อ 2 มีนาคม 2564]. จาก <https://medium.com/tencent-thailand/ประเภทของ-machine-learning-f3159fee7b56>.
21. Supervised Vs Unsupervised Learning [ออนไลน์]. 2561 [ค้นเมื่อ 5 มีนาคม 2564]. จาก <https://medium.com/@nattaponra/ml2-supervised-vs-unsupervised-learning-จะไหวหรือไม่-ไหว-aae9aa6f142b>.
22. Confusion Matrix Explained [ออนไลน์]. 2561 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <https://datarockie.com/2018/04/30/confusion-matrix-explained/>

23. วัดประสิทธิภาพ Model จาก Confusion Matrix [ออนไลน์]. 2562 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <https://medium.com/@cheng3374/วัดประสิทธิภาพ-model-จาก-confusion-matrix-69d391bcd48>
24. Confusion Matrix เครื่องมือสำคัญในการประเมินผลสัมฤทธิ์ของการทำนาย ในMachine learning [ออนไลน์]. 2562 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <https://medium.com/@pagongatchalee/confusion-matrix-เครื่องมือสำคัญในการประเมินผลสัมฤทธิ์ของการทำนาย-ในmachine-learning-fba6e3f9508c>
25. ระบบปฏิบัติการ IOS [ออนไลน์]. 2561 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <https://beerkung.wordpress.com/ระบบปฏิบัตการรุ่นล่าสุด/ระบบปฏิบัติการ-ios/>.
26. ระบบปฏิบัติการ ANDROID [ออนไลน์]. 2561 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <https://beerkung.wordpress.com/ระบบปฏิบัตการรุ่นล่าสุด/ระบบปฏิบัติการ-android/>.
27. iOS Developer ตอนที่ 1 รู้จักกับ Xcode IDE [ออนไลน์]. 2555 [ค้นเมื่อ 7 มีนาคม 2564]. จาก <http://www.daydev.com/developer/s6-programming-language/ios-developer-chapter-1-xcode.html>
28. รู้จักกับ Swift [ออนไลน์]. 2557 [ค้นเมื่อ 10 เมษายน 2564]. จาก <https://www.macthai.com/2014/06/07/intruduction-to-swift-programming-language-from-apple/>.
29. ภาษาอ็อบเจกทีฟ-ซี [ออนไลน์]. 2561 [ค้นเมื่อ 30 มีนาคม 2564]. จาก <https://th.wikipedia.org/wiki/ภาษาอ็อบเจกทีฟ-ซี>.
30. Core ML and Vision: Machine Learning in iOS 11 Tutorial [ออนไลน์]. 2560 [ค้นเมื่อ 1 มีนาคม 2564]. จาก <https://www.raywenderlich.com/577-core-ml-and-vision-machine-learning-in-ios-11-tutorial>.
31. Get Ready for Core ML 2 [ออนไลน์]. 2560 [ค้นเมื่อ 2 มีนาคม 2564]. จาก <https://developer.apple.com/machine-learning/>.
32. โรคเสี่ยงติดต่อทางน้ำลาย แคะใช้หลอดดูดน้ำร่วมกัน [ออนไลน์]. 2561 [ค้นเมื่อ 12 มกราคม 2564]. จาก <https://health.kapook.com/view166231.html>
33. FLIR ONE Pro [ออนไลน์]. 2560 [ค้นเมื่อ 1 เมษายน 2564]. จาก <https://www.flir.com/products/flir-one-pro/>.

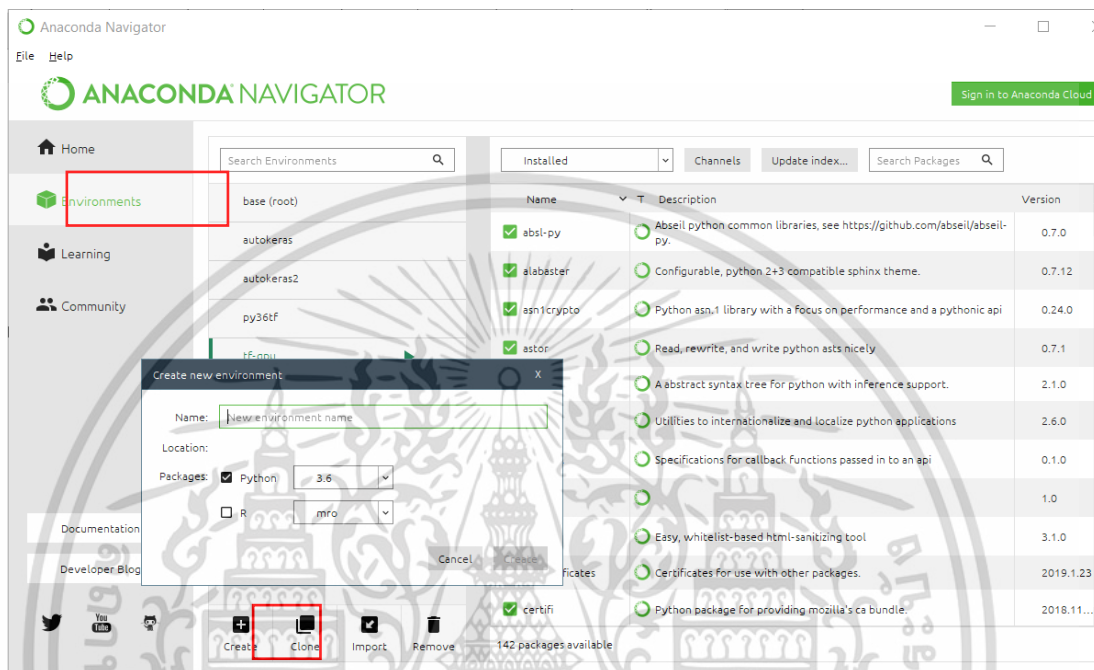
34. [Create ML] สร้างโมเดล Machine Learning (Core ML 2) แบบง่ายๆด้วย Create ML จาก Apple [ออนไลน์]. 2560 [ค้นเมื่อ 10 เมษายน 2564]. จาก <https://medium.com/@kajornsakp/create-ml-สร้าง-machine-learning-โมเดล-core-ml-แบบ-ง่ายๆด้วย-create-ml-จาก-apple-2a7b4ec4d2b5>.
35. สร้าง Machine Learning บน iOS ด้วย Core ML และ ML Kit [ออนไลน์]. 2560 [ค้นเมื่อ 10 เมษายน 2564]. จาก <https://medium.com/scoutoutlimited/สร้าง-machine-learning-บน-ios-ด้วย-core-ml-และ-ml-kit-856575b8a837>.
36. ทำความรู้จักกับ Core ML — Machine Learning บน iOS 11 [ออนไลน์]. 2559 [ค้นเมื่อ 10 เมษายน 2564]. จาก <https://blog.finnomena.com/ทำความรู้จักกับ-core-ml-machine-learning-บน-ios-11-1a94c2f8d7a9>.
37. Complete Architectural Details of all EfficientNet Models [ออนไลน์]. 2563 [ค้นเมื่อ 13 กรกฎาคม 2564]. จาก <https://towardsdatascience.com/complete-architectural-details-of-all-efficientnet-models-5fd5b736142>
38. Image Processing and Analysis in Java: ImageJ program [ออนไลน์]. [ค้นเมื่อ 16 พฤษภาคม 2564]. จาก <https://imagej.nih.gov/ij/>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการลงไลบรารีในการสอนโมเดล

1. ดาวน์โหลด Anaconda
2. ทำการสร้าง Environment โดยกดเข้าไปที่ Environments → Create



ตั้งชื่อและเลือก version ของ python แล้วกด create

3. กดเข้าหน้า terminal แล้วทำการตรวจสอบ version ของ python ว่าตรงตามที่ต้องการหรือเปล่าด้วยคำสั่ง `python --version`

```

C:\WINDOWS\system32\cmd.exe
(tf-gpu) C:\Users\FrontMe>python --version
Python 3.6.8 :: Anaconda, Inc.
(tf-gpu) C:\Users\FrontMe>
  
```

4. ลง Tensorflow - GPU ด้วยคำสั่ง `conda install tensorflow-gpu` ซึ่งคำสั่ง `conda` จะเป็นการลงไลบรารีที่เกี่ยวข้องกับไลบรารีหลักให้ด้วย
5. ลงไลบรารีอื่นๆดังนี้

Bottleneck==1.3.2

importlib-metadata==4.4.0

Keras-Applications==1.0.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Keras-Preprocessing==1.1.2

kiwisolver @ file:///tmp/build/80754af9/kiwisolver_1612282414123/work

Markdown==3.3.4

matplotlib @ file:///tmp/build/80754af9/matplotlib-suite_1613407855456/work

mkl-fft==1.3.0

mkl-random @ file:///tmp/build/80754af9/mkl_random_1618853974840/work

mkl-service==2.3.0

numpy @ file:///tmp/build/80754af9/numpy_and_numpy_base_1620831194891/work

pandas==0.23.4

pandas-ml==0.6.1

tensorboard==1.15.0

tensorflow==1.15.0

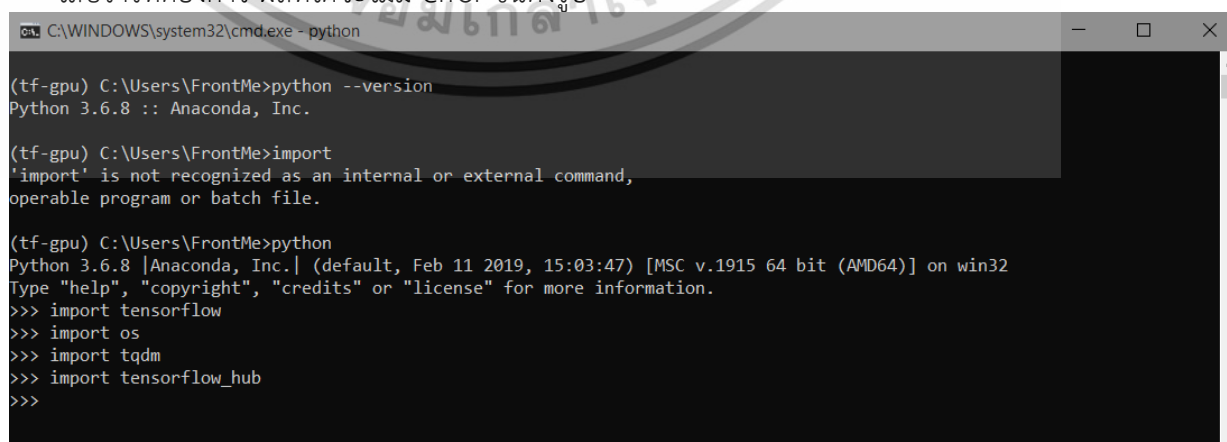
tensorflow-estimator==1.15.1

tensorflow-hub==0.12.0

termcolor==1.1.0

tqdm==4.61.0

6. ทำการเช็คที่สามารถใช้ไลบรารีข้างต้นได้หรือไหมด้วยการพิมพ์ python แล้วตามด้วย import ไลบรารีที่ต้องการ ผลที่ได้จะไม่มี error ขึ้นดังรูป



```

C:\WINDOWS\system32\cmd.exe - python
(tf-gpu) C:\Users\FrontMe>python --version
Python 3.6.8 :: Anaconda, Inc.

(tf-gpu) C:\Users\FrontMe>import
'import' is not recognized as an internal or external command,
operable program or batch file.

(tf-gpu) C:\Users\FrontMe>python
Python 3.6.8 [Anaconda, Inc.] (default, Feb 11 2019, 15:03:47) [MSC v.1915 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow
>>> import os
>>> import tqdm
>>> import tensorflow_hub
>>>
  
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมภาษา python ที่ใช้ใน

https://github.com/tensorflow/hub/tree/master/examples/image_retraining

วิธีการลงไลบรารีในการแปลงโมเดลเพื่อนำไปใช้ในแอปพลิเคชัน

ใช้ไลบรารี tfcoreml ซึ่งมี dependencies ที่ต้องการดังนี้

- tensorflow >= 1.5.0
- coremltools >= 0.8
- numpy >= 1.6.2
- protobuf >= 3.1.0
- six >= 1.10.0

จาก dependencies จะเห็นว่ามีไลบรารีบางตัวที่ version ไม่ตรงกับที่เคยลงไว้ จึงจำเป็นต้องสร้าง Environment ขึ้นใหม่เพื่อลงเฉพาะไลบรารีที่เกี่ยวข้อง

โปรแกรมภาษา python ที่ใช้ดังนี้

```
import tfcoreml as tf_converter
tf_converter.convert(tf_model_path = '/home/thorlabs/python/pangfront/RetrainIR/Forbinary/MODEL FOR 012/ResnetV2_50/ResnetV2_50 same/output_graph.pb',
                    mlmodel_path = '/home/thorlabs/python/pangfront/RetrainIR/Forbinary/MODEL FOR 012/ResnetV2_50/ResnetV2_50_224_binary2.mlmodel',
                    output_feature_names = ['final_result:0'],
                    image_input_names = 'Placeholder__0',
                    input_name_shape_dict = {"image": [1,224,224,3]},
                    class_labels = '/home/thorlabs/python/pangfront/RetrainIR/Forbinary/MODEL FOR 012/ResnetV2_50/ResnetV2_50 same/output_labels.txt',
                    image_scale= 2.0/255.0)
```

tf_model_path คือ directory ของไฟล์ pb

mlmodel_path คือ directory ของไฟล์ mlmodel ที่ต้องการจะ save

output_feature_names คือ ชื่อของ output

image_input_names คือ ชื่อของ input

input_name_shape_dict คือ ชนิดและขนาดของรูปภาพ input

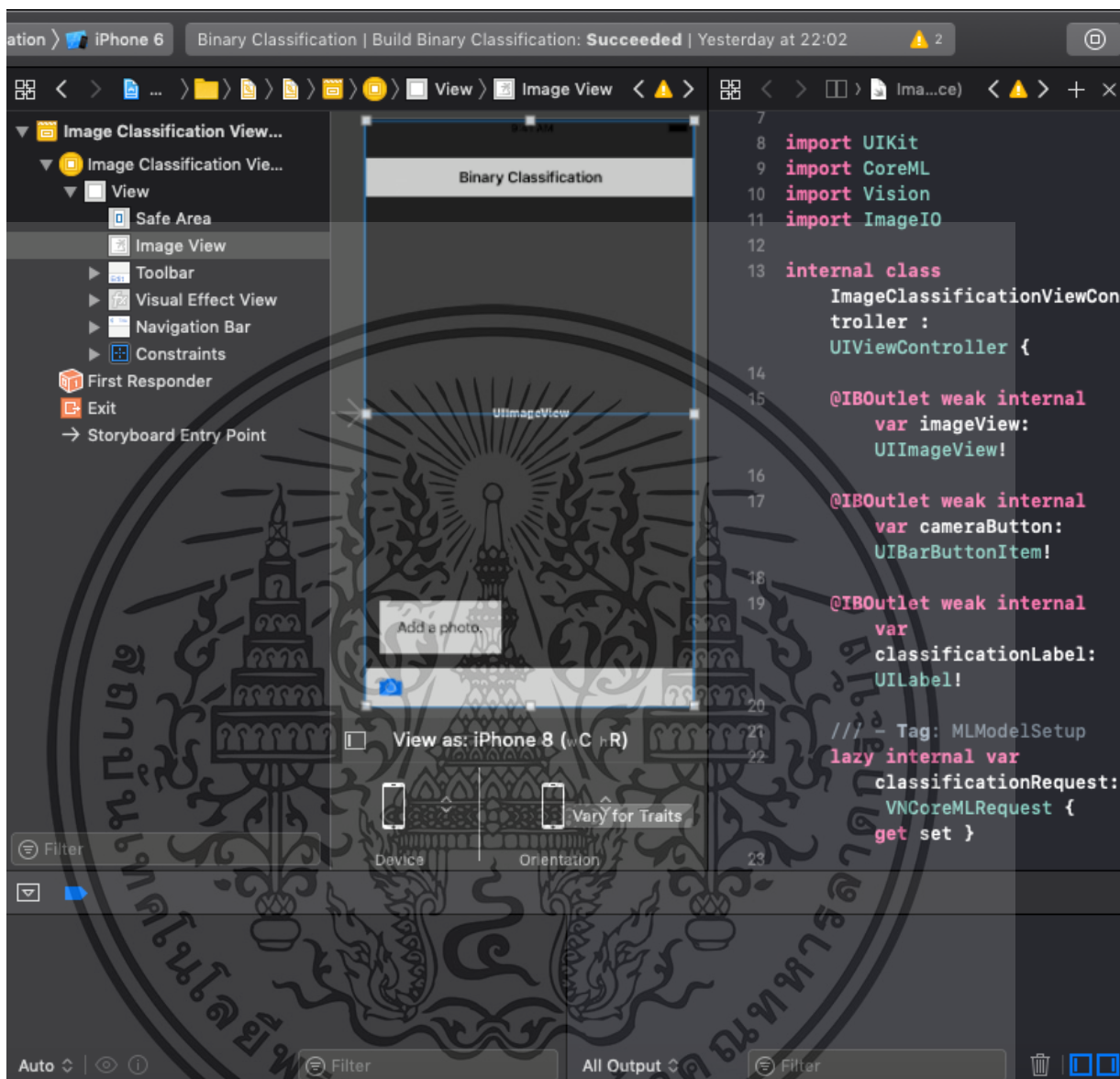
class_labels คือ directory ของไฟล์ output.txt

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดคำสั่งที่เขียนแอปพลิเคชัน Binary Classification



Storyboard

Code

/*

See LICENSE folder for this sample's licensing information.

Abstract:

View controller for selecting images and applying Vision + Core ML processing.

*/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

import UIKit
import CoreML
import Vision
import ImageIO

class ImageClassificationViewController: UIViewController {
// MARK: - IBOutlets

@IBOutlet weak var imageView: UIImageView!
@IBOutlet weak var cameraButton: UIBarButtonItem!
@IBOutlet weak var classificationLabel: UILabel!

// MARK: - Image Classification

/// - Tag: MLModelSetup
lazy var classificationRequest: VNCoreMLRequest = {
do {
/*
Use the Swift class `MobileNet` Core ML generates from the model.
To use a different Core ML classifier model, add it to the project
and replace `MobileNet` with that model's generated Swift class.
*/
let model = try VNCoreMLModel(for: NewResnetV2_50_224_binary().model)
let request = VNCoreMLRequest(model: model, completionHandler: { [weak self]
request, error in
self?.processClassifications(for: request, error: error)
})
request.imageCropAndScaleOption = .centerCrop
return request
} catch {
fatalError("Failed to load Vision ML model: \(error)")
}
}()

/// - Tag: PerformRequests
func updateClassifications(for image: UIImage) {
classificationLabel.text = "Classifying..."

```

```

let orientation = CGImagePropertyOrientation(image.imageOrientation)
guard let cilImage = CIImage(image: image) else { fatalError("Unable to create
\\(CIImage.self) from \\(image).") }

DispatchQueue.global(qos: .userInitiated).async {

    let handler = VNImageRequestHandler(cilImage: cilImage, orientation:
orientation)
    do {
    try handler.perform([self.classificationRequest])
    } catch {
    /*
    This handler catches general image processing errors. The
    `classificationRequest`'s
    completion handler `processClassifications(_:error:)` catches errors specific
    to processing that request.
    */
    print("Failed to perform classification.\\n\\(error.localizedDescription)")
    }
    }

}

// Updates the UI with the results of the classification.
// - Tag: ProcessClassifications
func processClassifications(for request: VNRequest, error: Error?) {
    DispatchQueue.main.async {
        guard let results = request.results else {
            self.classificationLabel.text = "Unable to classify
image.\\n\\(error!.localizedDescription)"
            return
        }
        // The `results` will always be `VNClassificationObservation`s, as specified
        by the Core ML model in this project.
        let classifications = results as! [VNClassificationObservation]

        if classifications.isEmpty {
            self.classificationLabel.text = "Nothing recognized."
        } else {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// Display top classifications ranked by confidence in the UI.
let topClassifications = classifications.prefix(1)
let descriptions = topClassifications.map { classification in
// Formats the classification for display; e.g. "(0.37) cliff, drop, drop-off".
return String(format: "(%.2f) %@", classification.confidence,
classification.identifier)
}
self.classificationLabel.text = "Classification:\n" +
descriptions.joined(separator: "\n")
}
}
}
// MARK: - Photo Actions
@IBAction func takePicture() {
// Show options for the source picker only if the camera is available.
guard UIImagePickerController.isSourceTypeAvailable(.camera) else {
presentPhotoPicker(sourceType: .photoLibrary)
return
}
let photoSourcePicker = UIAlertController()
let takePhoto = UIAlertAction(title: "Take Photo", style: .default) { [unowned self]
_ in
self.presentPhotoPicker(sourceType: .camera)
}
let choosePhoto = UIAlertAction(title: "Choose Photo", style: .default) { [unowned
self] _ in
self.presentPhotoPicker(sourceType: .photoLibrary)
}
photoSourcePicker.addAction(takePhoto)
photoSourcePicker.addAction(choosePhoto)
photoSourcePicker.addAction(UIAlertAction(title: "Cancel", style: .cancel, handler:
nil))
present(photoSourcePicker, animated: true)
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

func presentPhotoPicker(sourceType: UIImagePickerControllerSourceType) {
    let picker = UIImagePickerController()
    picker.delegate = self
    picker.sourceType = sourceType
    present(picker, animated: true)
}
}

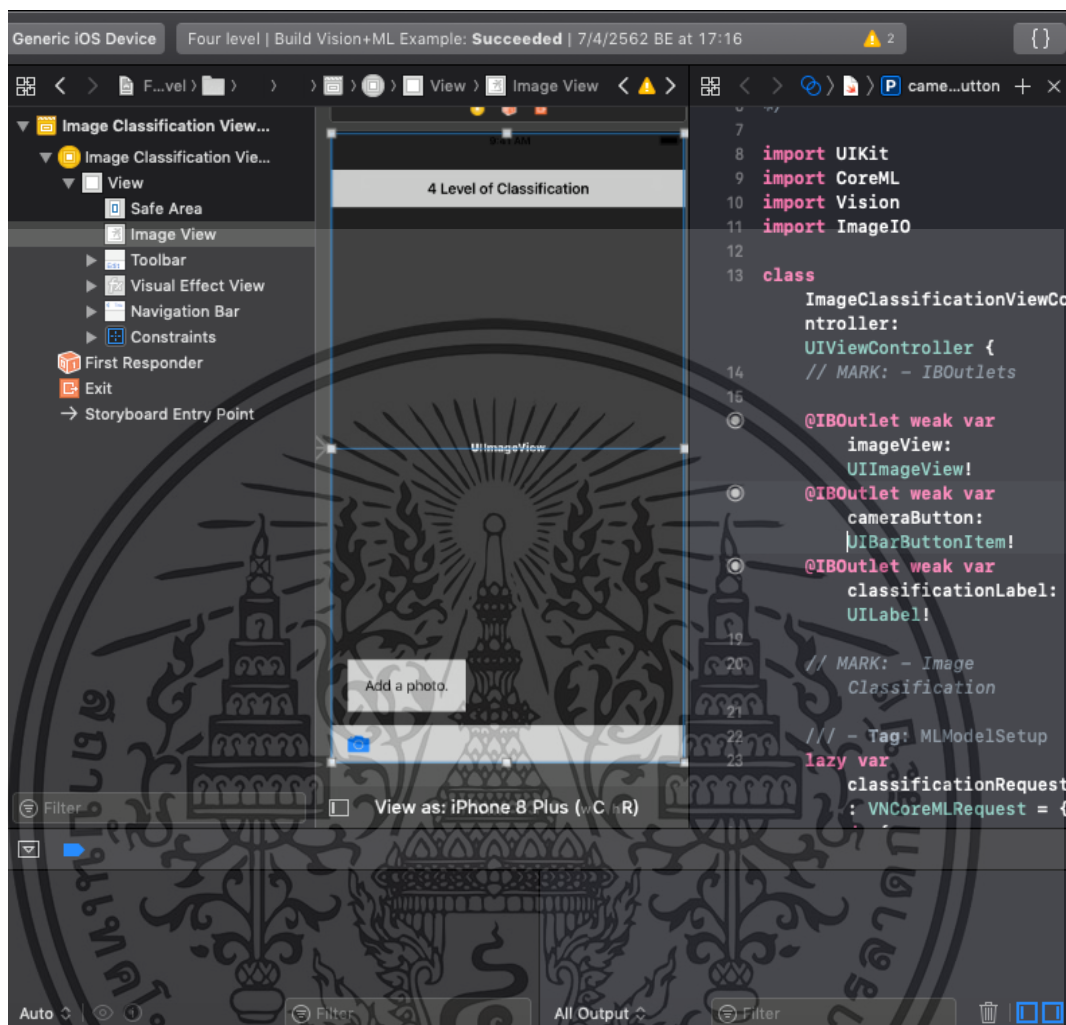
extension ImageClassificationViewController: UIImagePickerControllerDelegate,
    UINavigationControllerDelegate {
    // MARK: - Handling Image Picker Selection
    func imagePickerController(_ picker: UIImagePickerController,
        didFinishPickingMediaWithInfo info: [String: Any]) {
        picker.dismiss(animated: true)
        // We always expect `imagePickerController(:didFinishPickingMediaWithInfo:)` to
        // supply the original image.
        let image = info[UIImagePickerControllerOriginalImage] as! UIImage
        imageView.image = image
        updateClassifications(for: image)
    }
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดคำสั่งที่เขียนแอปพลิเคชัน 4 Level of Classification



Storyboard

Code

```
/*
```

See LICENSE folder for this sample's licensing information.

Abstract:

View controller for selecting images and applying Vision + Core ML processing.

```
*/
```

```
import UIKit
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

import CoreML

import Vision

import ImageIO

class ImageClassificationViewController: UIViewController {

    // MARK: - IBOutlets

    @IBOutlet weak var imageView: UIImageView!

    @IBOutlet weak var cameraButton: UIBarButtonItem!

    @IBOutlet weak var classificationLabel: UILabel!

    // MARK: - Image Classification
    /// - Tag: MLModelSetup
    lazy var classificationRequest: VNCoreMLRequest = {
        do {
            /*
             Use the Swift class `MobileNet` Core ML generates from the
             model.

             To use a different Core ML classifier model, add it to the project
             and replace `MobileNet` with that model's generated Swift class.
            */
            let model = try VNCoreMLModel(for: nasnet_4classify().model)

            let request = VNCoreMLRequest(model: model,
                completionHandler: { [weak self] request, error in
                    self?.processClassifications(for: request, error: error)
                })

            request.imageCropAndScaleOption = .centerCrop

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        return request
    } catch {
        fatalError("Failed to load Vision ML model: \(error)")
    }
}()

/// - Tag: PerformRequests

func updateClassifications(for image: UIImage) {
    classificationLabel.text = "Classifying..."

    let orientation = UIImagePropertyOrientation(image.imageOrientation)
    guard let cilImage = UIImage(image: image) else { fatalError("Unable to create
    \(UIImage.self) from \(image).") }

    DispatchQueue.global(qos: .userInitiated).async {
        let handler = VNImageRequestHandler(cilImage: cilImage, orientation: orientation)
        do {
            try handler.perform([self.classificationRequest])
        } catch {
            /*
            This handler catches general image processing errors. The
            `classificationRequest`s
            completion handler `processClassifications(_:error:)` catches errors specific
            to processing that request.
            */
            print("Failed to perform classification.\n\(error.localizedDescription)")
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
}

/// Updates the UI with the results of the classification.

/// - Tag: ProcessClassifications

func processClassifications(for request: VNRequest, error: Error?) {
    DispatchQueue.main.async {
        guard let results = request.results else {
            self.classificationLabel.text = "Unable to classify
image.\n\(error!.localizedDescription)"
            return
        }
        // The `results` will always be `VNClassificationObservation`s, as specified by the
        Core ML model in this project.
        let classifications = results as! [VNClassificationObservation]
        if classifications.isEmpty {
            self.classificationLabel.text = "Nothing recognized."
        } else {
            // Display top classifications ranked by confidence in the UI.
            let topClassifications = classifications.prefix(1)
            let descriptions = topClassifications.map { classification in
                // Formats the classification for display; e.g. "(0.37) cliff, drop, drop-off".
                return String(format: "(%.2f) %@", classification.confidence, classification.identifier)
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

self.classificationLabel.text = "Classification:\n" + descriptions.joined(separator:
"\n")
}
}
}

// MARK: - Photo Actions

@IBAction func takePicture() {
    // Show options for the source picker only if the camera is available.
    guard UIImagePickerController.isSourceTypeAvailable(.camera) else {
        presentPhotoPicker(sourceType: .photoLibrary)
    }
    return
}

let photoSourcePicker = UIAlertController()
let takePhoto = UIAlertAction(title: "Take Photo", style: .default) { [unowned self]
_in
    self.presentPhotoPicker(sourceType: .camera)
}

let choosePhoto = UIAlertAction(title: "Choose Photo", style: .default) { [unowned
self] _ in
    self.presentPhotoPicker(sourceType: .photoLibrary)
}

photoSourcePicker.addAction(takePhoto)
photoSourcePicker.addAction(choosePhoto)
photoSourcePicker.addAction(UIAlertAction(title: "Cancel", style: .cancel, handler:
nil))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        present(photoSourcePicker, animated: true)
    }

    func presentPhotoPicker(sourceType: UIImagePickerControllerSourceType) {

        let picker = UIImagePickerController()

        picker.delegate = self

        picker.sourceType = sourceType

        present(picker, animated: true)
    }
}

extension ImageClassificationViewController: UIImagePickerControllerDelegate,
    UINavigationControllerDelegate {
    // MARK: - Handling Image Picker Selection
    func imagePickerController(_ picker: UIImagePickerController,
        didFinishPickingMediaWithInfo info: [String: Any]) {
        picker.dismiss(animated: true)

        // We always expect
        `imagePickerController(:didFinishPickingMediaWithInfo:)` to supply the
        original image.

        let image = info[UIImagePickerControllerOriginalImage] as! UIImage

        imageView.image = image

        updateClassifications(for: image)
    }
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



FLIR ONE PRO

The FLIR ONE Pro gives you the power to find invisible problems faster than ever. Combining a higher-resolution thermal sensor able to measure temperatures up to 400 °C (752 °F) with powerful measurement tools and report generation capability, the FLIR ONE Pro will work as hard as you do. Its revolutionary VividIR™ image processing lets you see more details and provide your customers with proof that you solved their problem right the first time. The updated design includes the revolutionary OneFit™ adjustable connector to fit your phone, without taking the phone out of its compatible protective case. An improved FLIR ONE app lets you measure multiple temperatures or regions of interest at once and stream to your smartwatch for remote viewing. Whether you're inspecting electrical panels, looking for HVAC problems, or finding water damage, the new FLIR ONE Pro is a tool no serious professional should be without.

VividIR IMAGE PROCESSING

See It & Solve It - Sharpest Mobile Thermal Imaging Performance Lets You Detect Problems with Precision and Accuracy, then Document Your Fix for the Customer

- Most advanced image resolution enhancement detects the thermal details you need to find problems fast
- With 160 x 120 thermal resolution, FLIR ONE Pro uses FLIR's highest resolution micro thermal camera and can measure temperatures as high as 400 °C (752 °F)
- FLIR MSX® embosses visible edges from the 1440 x 1080 HD camera onto thermal imagery to create a sharper, easier to understand picture

OneFit CONNECTOR

Leave Your Case On - Adjustable Connector Means You Don't Have to Choose Between Thermal Vision and Safeguarding Your Device when Using Compatible Protective Cases

- Adjust length of USB-C and Lightning connector up to an additional 4 mm
- Reversible connectors for Android and iOS
- Secure the FLIR ONE to your mobile device while keeping your phone safe

HARD-WORKING APP

Work Like a Pro - Work-Based Features Include Advanced Capabilities for More Professional Problem Solving and Functionality

- Use multiple real-time spot meters and regions of interest
- Access real-time thermal tips and tricks in the FLIR ONE app followed by professional reporting through FLIR Tools
- See around corners and in awkward spaces by connecting to your Apple Watch or Android smartwatch

www.flir.com/flirone



The World's **Sixth Sense**®

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Specifications

General		FLIR One Pro	
Certifications	MFi (iOS version), RoHS, CE/FCC, CEC-BC, EN61233		
Operating temperature	0 °C – 35 °C (32 °F to 95 °F), battery charging 0 °C to 30 °C (32 °F to 86 °F)		
Non-operating temperature	-20 °C to 60 °C (-4 °F to 140 °F)		
Size	68mm W x 34mm H x 14mm D (2.7in x 1.3in x .6in)		
Weight	36.5g		
Mechanical shock	Drop from 1.8m (5.9ft)		
Video			
Thermal and visual cameras with MSX			
Thermal sensor	Pixel size 12µM, 8 – 14µM spectral range		
Thermal resolution	160x120		
Visual resolution	1440x1080		
HFOV / VFOV	55° ± 1° / 43° ± 1°		
Frame rate	8.7Hz		
Focus	Fixed 15cm – Infinity		
Radiometry			
Scene dynamic range	-20 °C to 400 °C (-4 °F to 752 °F)		
Accuracy	±3 °C (5.4 °F) or ±5%, typical Percent of the difference between ambient and scene temperature. Applicable 60s after start-up when the unit is within 15 °C to 35 °C (59 °F to 95 °F) and the scene is within 5 °C to 120 °C (41 °F to 248 °F)		
Thermal sensitivity (MRTD)	150mK		
Emissivity settings	Matte: 95%, Semi-Matte: 80%, Semi-Glossy: 60%, Glossy: 30% Reflected background temperature is 22 °C (72 °F)		
Shutter	Automatic/Manual		
Power			
Battery life	Approximately 1h		
Battery charge time	40min		
Interfaces			
Video	Male Lightning (iOS), Male USB-C (Android)		
Charging	Female USB-C (5V/1A)		
App			
Video and still image display/capture	Saved as 1440x1080		
File formats	Photo – radiometric jpeg Video – MPEG-4 (file format MOV (iOS), MP4 (Android))		
Capture modes	Video, Photo, Time lapse		
Palettes	Gray (white hot), Hottest, Coldest, Iron, Rainbow, Contrast, Arctic, Lava and Wheel.		
Spot meter	Off / °C / °F. Resolution 0.1 °C / 0.1 °F		
Adjustable MSX distance	0.3m – Infinity		
Battery charge monitor	0 – 100%		

CORPORATE HEADQUARTERS
FLIR Systems, Inc.
27700 SW Parkway Ave.
Wilsonville, OR 97070
PH: +1 877.773.3547

SANTA BARBARA
FLIR Systems, Inc.
6769 Hollister Ave.
Goleta, CA 93117
PH: +1 805.690.6600

www.flir.com
NASDAQ: FLIR

Equipment described herein is subject to US export regulations and may require a license prior to export. Diversion contrary to US law is prohibited. Imagery for illustration purposes only. Specifications are subject to change without notice. ©2017 FLIR Systems, Inc. All rights reserved. 06/06/17

17-1746-OEM-FLIROne_Pro

CHINA
FLIR Systems Co., Ltd
Room 502, West Wing, Hanwei Building
No. 7 Guanghua Ave.
Chaoyang District, Beijing 100004, China
Phone: +86 10-59797755

EUROPE
FLIR Systems, Inc.
Luxemburgstraat 2
2321 Meer
Belgium
PH: +32 (0) 3665 5100

ประวัติผู้เขียน

ชื่อ-นามสกุล นายพิสิษฐ์ เอี่ยมอุดมชัย

วัน เดือน ปีเกิด 6 สิงหาคม 2539

ที่อยู่ 110 หมู่บ้านสุภาวาลัย ซอยร่มเกล้า 19/2 ถนนร่มเกล้า แขวงคลองสามประเวศ
เขตลาดกระบัง จังหวัดกรุงเทพมหานคร 10520

ประวัติการศึกษา 2562 วิศวกรรมศาสตรบัณฑิต สาขาชีวการแพทย์ (เกียรตินิยมอันดับ 1)
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ความชำนาญเฉพาะด้าน

- 1.) การเขียนโปรแกรมไมโครคอนโทรลเลอร์
- 2.) การเขียนโปรแกรม python
- 3.) เขียนแบบสามมิติ
- 4.) การเขียนโปรแกรมการเรียนรู้เชิงลึก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้