



โครงการวิทยาศาสตร์ สาขาคอมพิวเตอร์

เรื่อง การพัฒนาโมเดล wayaj เพื่อการตรวจจับใบหน้าและนับจำนวนคนเข้าห้องสมุด

Development of the Wayaj model for facial detection and people counting in the
library.

โดย

นางสาวจิรัชญา กัญญะพิลา

นายวายุภักษ์ สมเมือง

ครูที่ปรึกษา

นายจิรายุส อรุณเดชาชัย

รายงานฉบับนี้เป็นส่วนประกอบของโครงการวิทยาศาสตร์

สาขาคอมพิวเตอร์ ระดับชั้นมัธยมศึกษาตอนปลาย

ตามหลักสูตรโรงเรียนวิทยาศาสตร์ภูมิภาค โรงเรียนวิทยาศาสตร์จุฬารามราชวิทยาลัย เลข



โครงการวิทยาศาสตร์ สาขาคอมพิวเตอร์

เรื่อง การพัฒนาโมเดล wayaj เพื่อการตรวจจับใบหน้าและนับจำนวนคนเข้าห้องสมุด

Development of the Wayaj model for facial detection and people counting in the library.

โดย

นางสาวจิรัชญา กัญญาพิลา

นายวายุภักษ์ สมเมือง

ลงชื่อ.....ครูที่ปรึกษาหลัก

(นายจิรายุส อรุณเดชาชัย)

ลงชื่อ.....รองผู้อำนวยการ

(นายปรัชญากร ฮตมาลี)

ลงชื่อ.....ผู้อำนวยการ

(นายกิตติชัย กรวยทอง)

ชื่อโครงการ	: การพัฒนาโมเดล wayaj เพื่อการตรวจจับใบหน้าและนับจำนวนคนเข้าห้องสมุด
ชื่อผู้จัดทำ	: นางสาวจิรัชญา กัญญาพิลา นายวายุภักษ์ สมเมือง
ชื่อครูที่ปรึกษา	: นายจิรายุส อรุณเดชาชัย
สาขา	: คอมพิวเตอร์
โรงเรียน	: วิทยาศาสตร์จุฬารณราชวิทยาลัย เลย์

บทคัดย่อ

โครงการนี้มีวัตถุประสงค์เพื่อเพิ่มประสิทธิภาพในการนับและตรวจจับผู้คนเข้าห้องสมุดและรวบรวมข้อมูลทางสถิติ ทำการทดลองโดยการนำ Dataset ที่นำมาจาก Kaggle และเว็บไซต์อื่นๆที่เชื่อถือได้จำนวน 4000-5000 รูปเพื่อใช้ในการเทรนโมเดล Object detection ด้วย Neural network ผ่านการ Label โครงสร้างใบหน้าของข้อมูลใน Dataset เพื่อให้ Ai รู้จักใบหน้า และทำการทำ Preprocessing และ Data augmentation สำหรับจัดระเบียบ เพิ่มจำนวนของข้อมูล และทำการเทรนซ้ำเป็นจำนวน 50 Epochs เพื่อเพิ่มค่าความแม่นยำในการตรวจจับใบหน้า จากการทดสอบประสิทธิภาพของโปรแกรมตรวจจับคนเข้าห้องสมุด มีรายการในการทดสอบแรกคือการทดสอบประสิทธิภาพการตรวจจับกับระยะทางเป็นจำนวน 20 ครั้ง คือ ระยะห่างที่กล้องสามารถตรวจจับได้อย่างแม่นยำ ในระยะ 0.5 1 1.5 2 เมตรมีความแม่นยำในการตรวจคือ 100% 100% 100% 100% 0% ตามลำดับ และเมื่อนำโมเดล wayaj ไปทดสอบประสิทธิภาพกับโมเดลตัวอื่นๆด้วยชุดข้อมูลรูปภาพใบหน้าทั้งหมด 237 ได้ค่าความแม่นยำดังนี้ YOLOv5x ได้ความแม่นยำ 100% โมเดล cv2.frontface ได้ความแม่นยำ 90.29% และโมเดล wayaj ได้ความแม่นยำ 97.89% โดยโมเดลถือได้ว่าโมเดล wayaj มีประสิทธิภาพมากกว่าโมเดล cv2.frontface ถึง 7.6% และยังมีประสิทธิภาพใกล้เคียงกับโมเดล YOLOv5x ที่มีการตรวจจับทั้งตัวในการประมวลผลอีกด้วย

คำสำคัญ : Dataset, Kaggle, Object detection, neural network, Label, Ai, Preprocessing, Data augmentation, Epochs, Google sheet

กิตติกรรมประกาศ

โครงการฉบับนี้สำเร็จลุล่วงไปด้วยดี ขอขอบคุณ นายกิตติชัย กรวยทอง ผู้อำนวยการโรงเรียน
วิทยาศาสตร์จุฬาภรณราชวิทยาลัย เลย ที่ให้ความอนุเคราะห์ในการทำโครงการครั้งนี้รวมถึงนายจิรายุส
อรุณเดชาชัยครูที่ปรึกษาโครงการ ผู้ให้ทั้งความรู้ ความช่วยเหลือ คำชี้แนะรวมทั้งให้หลักการข้อคิดเห็นต่างๆ
อันเป็นประโยชน์อย่างยิ่งต่อการพัฒนาโครงการและขอขอบคุณ เพื่อน ๆ ที่ได้ให้ความช่วยเหลือในการทดสอบ
และแสดงความคิดเห็นต่อโครงการ

ท้ายที่สุด ขอกราบขอบพระคุณ คุณพ่อและคุณแม่ ผู้เป็นที่รักไว้ ณ ที่นี้ และทางผู้จัดทำหวังเป็นอย่าง
ยิ่งว่ารายงานโครงการฉบับนี้จะเป็นประโยชน์สำหรับท่านที่ต้องการศึกษาค้นคว้าต่อไป

คณะผู้จัดทำ

นางสาวจิรัชญา กัญญะพิลา

นายวายุภักษ์ สมเมือง

สารบัญ

	หน้า
บทคัดย่อ	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญภาพและตาราง	จ
บทที่ 1 ที่มาและความสำคัญ	
ที่มาและความสำคัญ	1
วัตถุประสงค์	1
ขอบเขตการศึกษา	1
สมมุติฐาน	1
ประโยชน์ที่คาดว่าจะได้รับ	1
นิยามเชิงปฏิบัติการ	2
บทที่ 2 เอกสารและงานวิจัยที่เกี่ยวข้อง	
Python	3
Google sheets	3
OpenCV	4
Neural network	5
Numpy	5
Github	5
JSON	6
Visual studio code	7
Gradient Descent	8
Convolutional Neural Network	9
Roboflow	11
บทที่ 3 วิธีการดำเนินการ	
วัสดุอุปกรณ์และโปรแกรมที่ใช้	13
ขั้นตอนการดำเนินงาน	13
บทที่ 4 ผลการดำเนินงาน	
การดำเนินการทดสอบประสิทธิภาพ	20
วิเคราะห์ผลการทดสอบประสิทธิภาพ	20

สารบัญ(ต่อ)

	หน้า
บทที่ 5 สรุปผลการศึกษา	
สรุปผลการศึกษา	23
อภิปรายผล	23
ข้อเสนอแนะ	24
บรรณานุกรม	25
ภาคผนวก	27

สารบัญรูปภาพและตาราง

	หน้า
บทที่ 2 เอกสารและงานวิจัยที่เกี่ยวข้อง	
รูปที่ 2.1 Python	3
รูปที่ 2.2 Google Sheets	3
รูปที่ 2.3 OpenCV	4
รูปที่ 2.4 Numpy	5
รูปที่ 2.5 Github	6
รูปที่ 2.6 รูปแบบการเก็บข้อมูลประเภท JSON object	7
รูปที่ 2.7 รูปแบบการเก็บข้อมูลประเภท array	7
รูปที่ 2.8 Visual studio code	7
รูปที่ 2.9 กราฟของค่า weight gradient descent	8
รูปที่ 2.10 ตัวกรอง 3x3	9
รูปที่ 2.11 ซ้าย: filter เลื่อนไปบน input. ขวา: ผลลัพธ์การคำนวณซึ่งออกมาเป็น Feature map.	10
รูปที่ 2.12 การ Stride และ Padding	10
รูปที่ 2.13 Max Pooling	11
รูปที่ 2.14 Roboflow	11
รูปที่ 2.15 กราฟแสดงการแปลงภาพหรือวิดีโอเป็นระดับส่วนตัวในเวลาจริงที่มีสภาพอัตราส่วนที่ดีที่สุดของYOLOv5x	12
บทที่ 3 วิธีการดำเนินการ	
รูปที่ 3.1 การ label รูปภาพก่อนนำไปเทรนโมเดล	13
รูปที่ 3.2 ตัวอย่างการทดสอบ	14
รูปที่ 3.3 ทำการทำ Preprocessing	14
รูปที่ 3.4 Auto-Orient	15
รูปที่ 3.5 Image Augmentation	15
รูปที่ 3.6 การปรับ Brightness	16
รูปที่ 3.7 การปรับ Exposure	16
รูปที่ 3.8 การ blur รูปภาพ	16
รูปที่ 3.9 เพิ่ม Noise	16
รูปที่ 3.10 Mosaic	17
รูปที่ 3.11 Bounding Box	17
รูปที่ 3.12 Bounding Box: Flip	17

สารบัญรูปภาพและตาราง

	หน้า
รูปที่ 3.13 Bounding Box: Crop	17
รูปที่ 3.14 Get snippet	18
รูปที่ 3.15 การเลือก Yolo v5 Pytorch	18
รูปที่ 3.16 การนำโค้ดที่ได้มาใส่ใน Google colab	18
รูปที่ 3.17 การเตรียมโปรแกรมก่อนการ Deploy	18
บทที่ 4 ผลการดำเนินงาน	
รูปที่ 4.1 แผนภูมิแสดงความแม่นยำในการตรวจจับใบหน้าของแต่ละคลาส	20
รูปที่ 4.2 กราฟแสดงความแม่นยำในการตรวจจับใบหน้าของแต่ละคลาส	21
ตารางที่ 4.1 ตารางแสดงความสัมพันธ์ระหว่างระยะห่างจากกล้องตรวจจับกับความแม่นยำของโมเดล wayaj	21
รูปที่ 4.3 กราฟแสดงความต่อเนื่องของค่าต่างๆจากการเทรน Ai	21
รูปที่ 4.4 แผนภูมิแสดงการเปรียบเทียบความแม่นยำของโมเดล wayaj เปรียบเทียบกับโมเดล cv2.frontface และ YOLOv5x กับชุดข้อมูล setA	22
รูปที่ 4.5 กราฟการเปรียบเทียบความแม่นยำของโมเดล wayaj เปรียบเทียบกับโมเดล cv2.frontface และ YOLOv5x กับรูปตัวอย่าง	22
บทที่ 5 สรุปผลการศึกษา	
รูปที่ 5.1 แผนภูมิจำนวนการตรวจจับใบหน้าของโมเดล Cv2.frontface ,wayaj, YOLOv5x จำนวน 237 รูป	23
ภาคผนวก	
รูปที่ 1 ตัวอย่างภาพในการเทรนโมเดลจาก 5662 ภาพ	28
รูปที่ 2 การทดสอบประสิทธิภาพระหว่างการใช้โมเดล wayaj และโมเดลอื่นๆ	28
รูปที่ 3 รูปตัวอย่างที่ใช้ทดสอบประสิทธิภาพระหว่างการใช้โมเดล wayaj และโมเดลอื่นๆ	28
รูปที่ 4 รูปตัวอย่างที่ใช้ทดสอบประสิทธิภาพระหว่างการใช้โมเดล wayaj และโมเดลอื่นๆ	28
รูปที่ 5 ทดสอบความแม่นยำในการตรวจจับใบหน้าตามระยะที่กำหนด	29
รูปที่ 6 การติดตั้งเพื่อทดสอบการใช้งาน	29
รูปที่ 7 การติดตั้งเพื่อทดสอบการใช้งาน	29
รูปที่ 8 คำสั่งการนำเข้าโมเดลต่างๆ	29
รูปที่ 9 การหา path ของโมเดลและตั้งค่ากล้อง	30
รูปที่ 10 ตัวอย่างโปรแกรมหลัก	30
รูปที่ 11 ตัวอย่างโปรแกรมหลัก	31

สารบัญรูปภาพและตาราง

	หน้า
รูปที่ 12 อัฟข้อมูลลงใน google sheet 31	รูปที่ 13 คิวอาร์โค้ดแสดงรายละเอียดโปรแกรมเพิ่มเติม 31
รูปที่ 13 คิวอาร์โค้ดแสดงรายละเอียดโปรแกรมเพิ่มเติม	31

บทที่ 1

บทนำ

ที่มาและความสำคัญ

ห้องสมุด เป็นแหล่งข้อมูลที่สำคัญอีกแห่งหนึ่งของโรงเรียน เนื่องจากเป็นสถานที่รวบรวมหนังสือเกี่ยวกับการเรียน ความรู้รอบตัว ความรู้เฉพาะทางและโครงการต่างๆ ทั้งบุคลากร คุณครู หรือนักเรียนต่างก็ต้องการเข้ามาศึกษา เรียนรู้ รวบรวมข้อมูลในห้องสมุดนี้ เพื่อที่จะนำไปประกอบกับกิจกรรมต่างๆ จึงทำให้ผู้คนเข้ามาใช้ห้องสมุดเป็นจำนวนมาก แต่นั่นการตรวจนับและตรวจสอบบุคคลว่ามีจำนวนการเข้าใช้ห้องสมุดเท่าใดนั้นกลับเป็นเรื่องยาก เนื่องจากต้องตรวจสอบรายชื่อและจดบันทึกผู้ที่เข้าใช้ห้องสมุดผ่านการบันทึกรายคน ซึ่งใช้เวลานานในการสรุปและรวบรวมข้อมูลดังกล่าวพร้อมทั้งการเขียนโปรแกรมคอมพิวเตอร์นับจำนวนคนจากการดึงโมเดลต่างๆ มาใช้กลับมีความแม่นยำไม่มากนัก ดังนั้นทางผู้จัดทำจึงทำการเทรนโมเดล wayaj ขึ้นมาเพื่อเพิ่มความแม่นยำในการตรวจนับใบหน้า อีกทั้งยังพัฒนาโปรแกรมที่สามารถนำข้อมูลที่รวบรวมได้ ไปใช้เป็นข้อมูลทางสถิติ เพื่อนำไปวิเคราะห์และประยุกต์ในอนาคตได้อีกด้วย

วัตถุประสงค์

1. เพื่อพัฒนาโมเดลในการตรวจนับใบหน้า
2. เพื่อเพิ่มประสิทธิภาพและความแม่นยำในการนับและตรวจนับผู้ที่เข้าใช้ห้องสมุด

ขอบเขตการศึกษา

1. รวบรวมข้อมูลการเข้าใช้ห้องสมุดของโรงเรียนวิทยาศาสตร์จุฬาภรณราชวิทยาลัย เลย
2. นำข้อมูลมาแสดงผลเป็นกราฟเพื่อให้ง่ายต่อการวิเคราะห์

สมมติฐาน

สามารถพัฒนาโมเดลตรวจนับใบหน้าเพื่อคนเข้าห้องสมุดได้อย่างแม่นยำ และมีประสิทธิภาพ

ตัวแปรที่ศึกษา

ตัวแปรต้น	จำนวนรูปภาพที่นำไปเทรน
ตัวแปรตาม	ความแม่นยำในการตรวจนับใบหน้าของโมเดล
ตัวแปรควบคุม	วิธีการเทรนโมเดล wayaj และการ label ของรูปภาพที่ใช้เทรน

ประโยชน์ที่คาดว่าจะได้รับ

1. โมเดลสำหรับตรวจจับใบหน้าที่แม่นยำ
2. ช่วยอำนวยความสะดวกในการนับและตรวจจับผู้ที่เข้าใช้ห้องสมุด
3. ศึกษาแนวโน้มจำนวนคนที่เข้าใช้ห้องสมุดได้ในแต่ละวัน

นิยามเชิงปฏิบัติการ

1. โมดูล (Modules) คือ โมดูลใน python คือกลุ่มของตัวแปร ฟังก์ชัน หรือคลาสที่ทำงานคล้ายกัน แล้วเอามารวมกันไว้ในไฟล์ไฟล์เดียว ใน python เวลาเราจะเรียกใช้โมดูลเราต้อง import โมดูลเข้ามาก่อน ถึงจะสามารถเรียกใช้งานฟังก์ชัน หรือคลาสที่อยู่ภายในโมดูลได้
2. อาร์กิวเมนต์ (Argument) คือ ค่าหรือตัวแปรที่ส่งเข้าไปในฟังก์ชันของภาษา Python ซึ่ง Argument สามารถเรียกอีกชื่อหนึ่งได้ว่า พารามิเตอร์ (Parameter)
3. อัลกอริทึม (Algorithm) คือ กระบวนการแก้ปัญหาที่อธิบายเป็นขั้นตอนอย่างชัดเจน
4. การเทรนโมเดล (training model) เปรียบเสมือนการสอนคอมพิวเตอร์ให้ “เรียนรู้” โดยเราต้องทำการต้องการ input คือ “ข้อมูล (data)” เพื่อใช้ในการตัดสินใจเลือก output ให้สอดคล้องกับข้อมูลที่เรียนมาหรือเทรนมานั่นเอง

บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

ในการเริ่มการทำงานเราต้องรู้จักภาษาที่ใช้เขียนโปรแกรม ฐานข้อมูลที่ใช้รวบรวมข้อมูล และเครื่องมือต่างๆ ที่ช่วยในการเขียนโปรแกรมตรวจจับคนเข้าห้องสมุด ได้แก่

2.1 Python

Python (ไพทอน) เป็นภาษาโปรแกรมคอมพิวเตอร์ระดับสูง โดยถูกออกแบบมาให้เป็นภาษาสคริปต์ที่อ่านง่าย ไม่เหมือนภาษาพูดของมนุษย์ มีการตัดความซับซ้อนของโครงสร้างและไวยากรณ์ของภาษาออกไปเพื่อเอาไว้แปลงเป็นชุดคำสั่ง สิ่งการใช้งานต่ออุปกรณ์เทคโนโลยีหลาย ๆ อย่าง ซึ่งการสั่งงานด้วยภาษา Python เรามักจะเขียนให้มีการทำงานแบบ Interpreter คือเป็นการแปลชุดคำสั่งทีละบรรทัด เพื่อป้อนเข้าสู่หน่วยประมวลผลให้คอมพิวเตอร์ทำงานตามที่เราต้องการ



รูปที่ 2.1 Python

2.2 Google Sheets

Google Sheets หรือเรียกย่อ ๆ ว่า Sheets เป็นซอฟต์แวร์ด้าน Spreadsheet สร้างตารางคำนวณทำงานแบบ Online บน Cloud ใช้งานได้ฟรี ทำหน้าที่คล้าย ๆ กับ Microsoft Excel เป็นตารางเป็นช่อง ๆ ใส่สูตรคำนวณได้ สามารถแชร์ให้กับคนอื่น เข้ามาทำงานร่วมกันได้ พร้อมแจ้งเตือนได้เมื่อมีการเปลี่ยนแปลงเอกสารทันที



รูปที่ 2.2 Google Sheets

2.3 OpenCV

OpenCV (Open source Computer Vision) เป็นไลบรารีฟังก์ชันการเขียนโปรแกรม (Library of Programming Functions) โดยส่วนใหญ่จะมุ่งเป้าไปที่การแสดงผลด้วยคอมพิวเตอร์แบบเรียลไทม์ (Real-Time Computer Vision) เดิมทีแล้วถูกพัฒนาโดย Intel แต่ภายหลังได้รับการสนับสนุนโดย Willow Garage ตามมาด้วย Itseez (ซึ่งต่อมาถูกเข้าซื้อโดย Intel) OpenCV เป็นไลบรารีแบบข้ามแพลตฟอร์ม (Cross-Platform) และใช้งานได้ฟรีภายใต้ลิขสิทธิ์ของ BSD แบบโอเพ่นซอร์ส (Open-Source BSD License)

OpenCV ยังสนับสนุนเฟรมเวิร์กการเรียนรู้เชิงลึก (Deep Learning Frameworks) ได้แก่ TensorFlow, Torch/PyTorch และ Caffe

2.3.1 การใช้ประโยชน์

- 2.3.1.1 ชุดเครื่องมือคุณลักษณะ 2 มิติและ 3 มิติ (2D and 3D feature toolkits)
- 2.3.1.2 การประมาณระยะในขณะเคลื่อนที่ (Egomotion Estimation)
- 2.3.1.3 ระบบรู้จำใบหน้า (Facial recognition system)
- 2.3.1.4 การจดจำท่าทาง (Gesture recognition)
- 2.3.1.5 ปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์ (Human-Computer interaction; HCI)

2.3.2 ภาษาการเขียนโปรแกรม

OpenCV ถูกเขียนขึ้นด้วยภาษา C++ มีการรองรับ Python, Java และ MATLAB/OCTAVE — API สำหรับอินเทอร์เฟซเหล่านี้สามารถพบได้ในเอกสารออนไลน์ ซึ่งมีการรวมไว้หลากหลายภาษา เช่น C#, Perl, Ch, Haskell และ Ruby ได้รับการพัฒนาเพื่อส่งเสริมการนำมาใช้งานโดยผู้ใช้ที่เพิ่มขึ้น



รูปที่ 2.3 OpenCV

2.4 Neural network

เครือข่ายประสาทเทียมเป็นวิธีหนึ่งในปัญญาประดิษฐ์ที่สอนคอมพิวเตอร์ในการประมวลผลข้อมูลในลักษณะที่ได้แรงบันดาลใจมาจากสมองมนุษย์ มันเป็นส่วนหนึ่งของกระบวนการเรียนรู้ของเครื่องจักรที่เรียกว่าการเรียนรู้ลึก ซึ่งใช้โหนดหรือเซลล์ประสาทที่เชื่อมต่อกันอยู่ในโครงสร้างแบบชั้นซ้อนที่คล้ายกับสมองมนุษย์ มันสร้างระบบการปรับตัวที่คอมพิวเตอร์นำมาใช้ในการเรียนรู้จากข้อผิดพลาดและปรับปรุงอย่างต่อเนื่อง ดังนั้น เครือข่ายประสาทเทียมประพฤติให้พยายามแก้ไขปัญหาคับซ้อน เช่น สรุปเอกสารหรือรู้จำใบหน้า โดยมีความแม่นยำเพิ่มขึ้น

2.5 Numpy

เป็น library ของpythonที่สามารถจัดแจงจัดการข้อมูลให้เป็นระบบมากยิ่งขึ้นที่ถูกเขียนโดยภาษา C เลยสามารถทำให้ประมวลผลได้เร็ว และมีประสิทธิภาพ



รูปที่ 2.4 Numpy

2.6 Github

GitHub คือ website Git (version control repository) ที่อยู่บน internet มีการทำงานแบบเดียวกับ Git เลย แต่สามารถเข้าถึงข้อมูลและจัดการไปผ่าน web โดยไม่ต้องเสียเงิน หรือลงทุกตั้ง server เพื่อติดตั้ง Git เองเลย แต่ code project ทั้งหมดจะถูกแจกจ่ายให้คนอื่น ๆ สามารถเห็นได้ด้วย

2.6.1 Github มีประโยชน์อย่างไร

2.6.1.1 ตรวจสอบเวอร์ชันย้อนหลังของ source code ได้

2.6.1.2 ช่วยในการพัฒนาซอฟต์แวร์เป็นทีม

2.6.2 สถานะของ Source Code ที่เก็บอยู่ในระบบของ Git นั้นมีดังนี้

2.6.2.1 Untracked เป็นสถานะที่ Source Code ถูกเพิ่มเข้ามาใหม่และยังไม่ได้ถูกเก็บไว้ใน

ระบบของ Git

2.6.2.2 Working Directory เป็นสถานะที่กำลังมีการเปลี่ยนแปลงหรือแก้ไข Source Code หรืออาจจะเรียกสถานะนี้ว่า Modified

2.6.2.3 Staged เป็นสถานะที่ Source Code กำลังเตรียมที่จะ Commit เพื่อยืนยันการเปลี่ยนแปลงก่อนที่จะเก็บลงในสถานะ Local Repository

2.6.2.4 Local Repository เป็นสถานะที่มีการเก็บบันทึกข้อมูลการเปลี่ยนแปลงของ Source Code ลงไปที่ Git Repository ที่เป็น Local (ที่เครื่องตัวเอง)

2.6.2.5 Remote Repository เป็นสถานะที่มีการเก็บบันทึกข้อมูลการเปลี่ยนแปลงของ Source Code ลงไปที่ Git Repository ที่เป็น Hosting (ที่เครื่องเซิร์ฟเวอร์)



รูปที่ 2.5 Github

2.7 JSON

JSON เป็นข้อมูลรูปแบบ text ที่มีรูปแบบที่จะเก็บข้อมูลแบบ key, value โดยการเขียนข้อมูลชนิด JSON มีรูปแบบคือ ชื่อฟิลด์ครอบด้วยเครื่องหมาย “ (double quote), เครื่องหมาย : (colon), value แล้วครอบทั้งหมดด้วยเครื่องหมายปีกกา ตัวอย่างที่มีข้อมูล 1 อย่างจะเป็นดังนี้

2.7.1 ประเภทข้อมูลที่ JSON เก็บได้มีดังนี้

2.7.1.1 string

2.7.1.2 number

2.7.1.3 object (JSON object)

2.7.1.4 array

2.7.1.5 boolean

2.7.1.6 null

การเก็บข้อมูลประเภท JSON object ให้วางซ้อนเข้าไปอีกทีได้เช่น

```
'{"name": "Kate", "pet": {"dog": "Corgi", "cat": "Persian"}}'
```

รูปที่ 2.6 รูปแบบการเก็บข้อมูลประเภท JSON object

หรือจะเก็บข้อมูล array ก็ทำได้แบบนี้

```
'{"name": "Jothanan", "age": 28, "car": ["Tsubaru", "Honda"]}'
```

รูปที่ 2.7 รูปแบบการเก็บข้อมูลประเภท array

2.8 Visual Studio Code

วิซวลสตูดิโอโค้ด (อังกฤษ: Visual Studio Code) เป็นโปรแกรมแก้ไขซอร์สโค้ดที่พัฒนาโดยไมโครซอฟท์สำหรับ Windows, Linux และ macOS มีการสนับสนุนสำหรับการดีบั๊ก การควบคุม Git ในตัว และ GitHub การเน้นไวยากรณ์ การเติมโค้ดอัจฉริยะ ตัวอย่าง และ code refactoring มันสามารถปรับแต่งได้หลายอย่าง ให้ผู้ใช้สามารถเปลี่ยนธีม แป้นพิมพ์ลัด การตั้งค่า และติดตั้งส่วนขยายที่เพิ่มฟังก์ชันการทำงานเพิ่มเติม ซอร์สโค้ดนั้นฟรีและโอเพนซอร์สและเผยแพร่ภายใต้สิทธิ์การใช้งาน MIT ไบนารีที่คอมไพล์แล้วเป็นฟรีแวร์และฟรีสำหรับการใช้ส่วนตัวหรือเพื่อการค้า

วิซวลสตูดิโอโค้ดใช้อิเล็กทรอนิกส์เป็นเฟรมเวิร์กที่ใช้ในการปรับใช้แอปพลิเคชัน Node.js สำหรับเดสก์ท็อปที่รันบนเอ็นจิน Blink แม้ว่าจะใช้เฟรมเวิร์กอิเล็กทรอนิกส์ ซอฟต์แวร์นี้ไม่ได้ใช้อะตอม และใช้คอมโพเนนต์ตัวแก้ไขเดียวกัน (ชื่อรหัส "Monaco") กับที่ใช้ใน Azure DevOps (เดิมชื่อ Visual Studio Online และ Visual Studio Team Services)

ในการสำรวจนักพัฒนา Stack Overflow 2019 วิซวลสตูดิโอโค้ดได้รับการจัดอันดับให้เป็นเครื่องมือสำหรับนักพัฒนาด้านสิ่งแวดล้อมที่ได้รับความนิยมมากที่สุดโดย 50.7% ของผู้ตอบแบบสอบถาม 87,317 ราย อ้างว่าใช้งาน



รูปที่ 2.8 Visual Studio Code

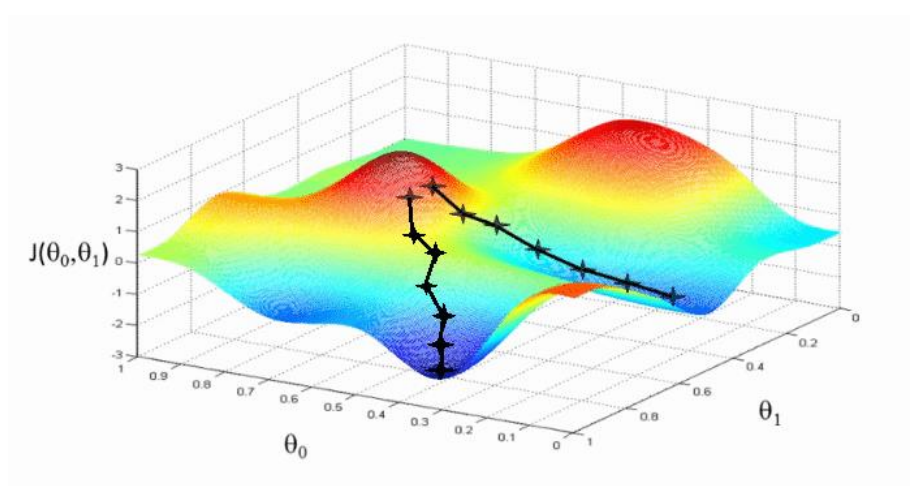
2.9 Gradient Descent

ในการเทรน Artificial Neural Network เราต้องการทราบว่าการเปลี่ยนแปลงขยับ เพิ่ม/ลด Weight หนึ่ง ๆ มีผลต่อการ เพิ่ม/ลด Loss อย่างไร โดยสมมติว่า Weight อื่น ๆ คงที่ทั้งหมด เช่นถ้าเราเพิ่ม Weight

$A + 0.0001$ แล้ว Loss ลด เราก็ลองเพิ่ม Weight A ไป แล้วลองทดสอบ ทำแบบนี้ไปทุก Weight ซ้ำไปเรื่อย ๆ จนกว่าจะได้ Loss ที่น้อยที่สุด

ในทางคณิตศาสตร์ เราไม่ต้องมาค่อย ๆ ขยับเช็คทีละ Weight เราสามารถใช้ Calculus ทำการ Diff สมการ หา Derivative ความชัน Slope ของ Weight/Loss ออกมาได้เลยว่า Loss จะเพิ่มเท่าไร เมื่อเราขยับ Weight เท่าไร Gradient Descent หมายถึงเราจะค่อย ๆ ขยับ ทุก ๆ Weight ไปทางที่ Slope ติดลบ ไปเรื่อย ๆ ให้ได้ Loss ต่ำที่สุด

ดังตัวอย่างจากคอร์ส Machine Learning ของ Andrew Ng ยกตัวอย่าง ฟังก์ชัน $J(\theta_0, \theta_1)$ แกน Z ที่มี Weight 2 ตัวคือ θ_0 แกน X และ θ_1 แกน Y เราควรที่จะเพิ่มลด θ_0, θ_1 อย่างไร ให้เราเคลื่อนจากจุดสูงสีแดง ขยับซ้าย/ขวา/หน้า/หลัง อย่างไร ให้ไปยังจุดที่ต่ำที่สุดของกราฟสีน้ำเงิน



รูปที่ 2.9 กราฟของค่า weight gradient descent

ในการเทรน Deep Learning ที่มี Weight หลายล้านตัว วิธีที่จะหาว่า Slope มีค่าเท่าไร ที่นิยมในปัจจุบันจะใช้ Multivariate Calculus หา Diff ความชันของ Slope ที่เดียวทุก Weight ไปเลย โดยจะหาทีละ Layer จากหลังมาหน้า เรียกว่า Backpropagation

2.10 Convolutional Neural Network

Convolutional Neural Network (CNN) หรือ โครงข่ายประสาทแบบคอนโวลูชัน เป็นโครงข่ายประสาทเทียมหนึ่งในกลุ่ม bio-inspired โดยที่ CNN จะจำลองการมองเห็นของมนุษย์ที่มองพื้นที่เป็นที้อยๆ และนำกลุ่มของพื้นที่ย่อยๆ มาผสานกัน เพื่อดูว่าสิ่งที่เห็นอยู่เป็นอะไรกันแน่ การมองพื้นที่ย่อยของมนุษย์จะมีการแยกคุณลักษณะ (feature) ของพื้นที่ย่อยนั้น เช่น ลายเส้น และการตัดกันของสี ซึ่งการที่มนุษย์รู้ว่าพื้นที่ตรงนี้เป็นเส้นตรงหรือสีตัดกัน เพราะมนุษย์ดูทั้งจุดที่สนใจและบริเวณรอบ ๆ ประกอบกัน

Feature Extraction

แนวคิดของ CNN นั้นค่อนข้างเป็นแนวคิดที่ดีมาก แต่สิ่งที่ซับซ้อนของมันคือระบบการคำนวณที่สอดคล้องกับ Concept ของมันเองและต้องมีคณิตศาสตร์มารองรับ โดยการคำนวณตามแนวคิดนี้ใช้หลักการเดียวกันกับ คอนโวลูชันเชิงพื้นที่ (Spatial Convolution) ในการทำงานด้าน Image Processing

การคำนวณนี้จะเริ่มจากการกำหนดค่าใน **ตัวกรอง (filter) หรือ เคอร์เนล (kernel)** ที่ช่วยดึงคุณลักษณะที่ใช้ในการรู้จำวัตถุออก โดยปกติตัวกรอง/เคอร์เนลอันหนึ่งจะดึงคุณลักษณะที่สนใจออกมาได้หนึ่งอย่าง เราจึงจำเป็นต้องตัวกรองหลายตัวกรองด้วย เพื่อหาคุณลักษณะทางพื้นที่ที่หลากหลายประกอบกัน

2.10.1 ลักษณะของ Filter

สำหรับ Filter ของภาพดิจิทัลนั้น โดยปกติแล้วจะเป็นตารางสองมิติที่มีขนาดตามพื้นที่ย่อยๆ ที่เราอยากพิจารณา

สมมุติว่าถ้าเราต้องการหาเส้นตรงทะแยงสีขาว ตัวกรองของเราอาจจะอยู่ในลักษณะนี้

1	-1	-1
-1	1	-1
-1	-1	1

รูปที่ 2.10 ตัวกรอง 3x3

ตัวกรอง 3x3 สำหรับหาเส้นตรงทะแยงสีขาว

ตำแหน่งตรงกลางที่มีกรอบสีฟ้าคือ Anchor ที่เอาไว้หาบนพิกเซลของภาพข้อมูลเข้า ตัวกรองจะถูกหาบนพิกเซลแรกๆของภาพข้อมูลเข้า จากนั้นจะถูกลื่อนไปหาบน

บนพิกเซลอื่นในภาพทีละพิกเซลจนครบทุกพิกเซลในภาพ เราอาจจะไม่หาบนพิกเซลที่อยู่ใกล้กรอบภาพ เพราะตัวกรองจะล้นออกไปนอกภาพ เมื่อเราเลื่อนตัวกรองไปเรื่อยๆจนครบทุกพิกเซลที่สามารถเลื่อนได้ในภาพ สิ่งที่เราได้นั้นจะเป็นสิ่งที่เรียกว่า *ฟังก์ชันลักษณะ (feature map)*

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

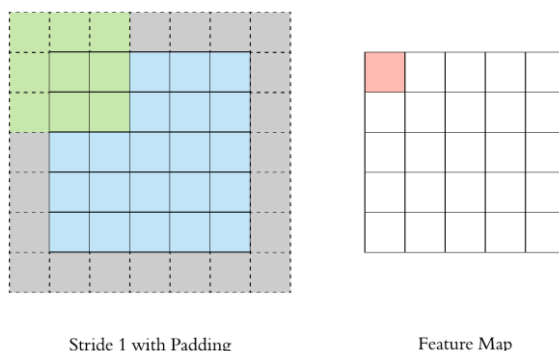
4		

รูปที่ 2.11 ซ้าย: filter เลื่อนไปบน input. ขวา: ผลลัพธ์การคำนวณซึ่งออกมาเป็น Feature map.

2.10.2) Stride และ Padding

Stride เป็นตัวกำหนดว่าเราจะเลื่อนตัวกรอง (filter) ไปด้วย Step เท่าไร เราสามารถกำหนดค่าของ Stride ให้มากขึ้นก็ได้ ถ้าเราต้องการให้การคำนวณหาคุณลักษณะมีพื้นที่ที่ทับซ้อนกันน้อยขึ้น แต่อย่างไรก็ตามการกำหนดค่าของ Stride ที่มากขึ้นจะทำให้เราได้ฟังก์ชันลักษณะ (feature map) ที่มีขนาดเล็กลง

Padding จากรูปด้านล่างเราจะเห็นว่าพื้นที่สีเทารอบๆ Input พื้นที่เหล่านี้เป็นพื้นที่ที่เราเพิ่มเข้าไป โดยอาจจะเพิ่ม 0 หรือค่าต่างๆเข้าไป เพื่อให้เวลาในการทำ CNN นั้น Feature Map ที่ได้ยังคงมีขนาดเท่ากับ Input



รูปที่

รูปที่ 2.12 การ Stride และ Padding

บางปัญหา Input ที่อยู่ตามขอบภาพอาจมีความสำคัญที่ส่งผลต่อการตัดสินใจบางอย่าง เราจึงจำเป็นต้องเก็บคุณลักษณะตามขอบของรูปภาพไว้ด้วย

2.10.3 Max Pooling

ก่อนอื่นเรามองลองดูหนึ่งในปัญหาของการทำ CNN กันก่อน สมมติเราใช้ CNN ด้วยขนาดตัวกรอง 3x3 พิกเซล แต่เรารู้ดีว่าเวลาเรามองภาพแล้วเราตอบได้ว่ามันคืออะไร เพราะเรามองไปในบริเวณที่กว้างกว่านั้น



รูปที่ 2.13 Max Pooling

จากภาพ จะเห็นว่าทำให้รูปภาพมีขนาดสเกลที่เล็กลง แต่เราก็ยังสามารถมองออกว่ามันคือเครื่องปั้นดินเผา แสดงว่า เราจำแนกวัตถุขึ้นนี้ที่ความละเอียดต่ำลง แต่เรากำลังทำ CNN ที่ความละเอียดสูง

2.11 Roboflow

เว็บสำหรับการหา Dataset Labelภาพและทำการเทรนโมเดลสำหรับนำไปเทรนโมเดลโดยการทำ Transfer Learning

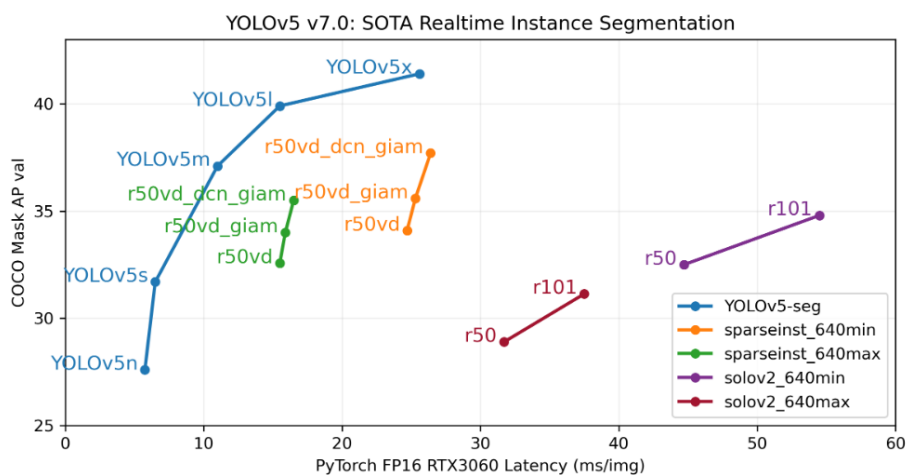


รูปที่ 2.14 Roboflow

2.12 YOLOv5x และ CV2.frontface

YOLOv5 คือโมเดลในตระกูลของโมเดลคอมพิวเตอร์วิชั่นที่เรียกว่า "You Only Look Once" (YOLO) โมเดล. YOLOv5 ถูกใช้งานอย่างแพร่หลายในการตรวจจับวัตถุ. YOLOv5 มีทั้งหมด 4 รุ่นหลักคือ small (s), medium (m), large (l), และ extra large (x) แต่ละรุ่นมีการเสนออัตราความแม่นยำที่สูงขึ้นเรื่อยๆ. แต่ละรุ่นยังใช้เวลาในการฝึกอบรมที่แตกต่างกันด้วย

CV2.frontface ไบบริารีแบบโอเพ่นซอร์สของ OpenCV ให้ที่เตรียมไว้สำหรับตรวจจับใบหน้า



รูปที่ 2.15 กราฟแสดงการแปลงภาพหรือวิดีโอเป็นระดับส่วนตัวในเวลาจริงที่มีสภาพอัตราส่วนที่ดีที่สุดของ YOLOv5x

บทที่ 3

วิธีดำเนินการ

3.1 วัสดุอุปกรณ์และโปรแกรมที่ใช้

3.2.1 VSCode

3.2.2 Google Sheets

3.2.3 Github

3.2.4 Stack Overflow

3.2.5 roboflow

3.2.6 คอมพิวเตอร์ 1 เครื่อง

3.3 ขั้นตอนการดำเนินงาน

3.3.1 ดาวน์โหลด Dataset เพื่อใช้สำหรับการ Train

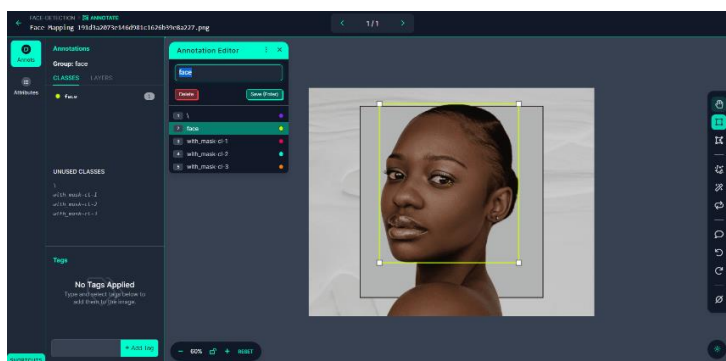
โดยทำการโหลด Dataset ใน Kaggle ซึ่งเป็นเว็บสำหรับการทำ Ai และรวบรวม Dataset

3.3.2 Train Ai ขั้นตอนการเทรน Ai เพื่อใช้ในการตรวจจับใบหน้าแบ่งได้เป็นขั้นตอนดังนี้

3.3.2.1. นำรูปภาพใบหน้าที่ได้มาจากDataset เข้าไปยังเว็บ Roboflow เพื่อทำการ Label รูปภาพ โดยทำการติกรอบเฉพาะใบหน้าเพื่อให้ AI นั้นรู้จักใบหน้าและโครงสร้างของใบหน้า และนำไปจัดไว้ในคลาสของ face และทำการทำซ้ำไปจนครบจำนวนรูปภาพของใบหน้าในDataset

3.3.2.2. นำเข้ารูปภาพที่มาสแงและการบวกรวนจากสิ่งต่างๆเช่น มือ ใบไม้ แว่นตา หมวก หรือแสงสะท้อน เพื่อให้ AI มีความยืดหยุ่นในสถานะการต่างๆโดยทำการจัดไว้ในคลาส face

3.3.2.3. นำรูปภาพที่หันข้าง และรูปภาพที่เบลอมานในDataset เดียวกันและมาทำการ Label ภาพสำหรับในกรณีที่เกิดการเดินหันข้างเข้าห้องสมุดหรือการที่เคลื่อนที่เร็วจนภาพเบลอ



รูปที่ 3.1 การ label รูปภาพก่อนนำไปเทรนโมเดล

3.3.2.4. ทำการแบ่งชุดข้อมูลเป็น 3 ชุดข้อมูล

3.3.2.4.1 Test ในชุดนี้จะมีการนำข้อมูลมาทำการเทรนเพื่อให้ข้อมูลนั้นได้รู้จักใบหน้า และ สิ่งรบกวนต่างๆที่มามีคบบังใบหน้า

3.3.2.4.2 Validation ในชุดนี้จะเป็นการนำ โมเดลที่ได้มาจากการเทรน มาทดสอบหา Metrics เพื่อหาว่าโมเดลทำงานได้ดีแค่ไหนและเลือกโมเดลที่ดีที่สุดมา

3.3.2.4.3 Train ในชุดนี้จะเป็นการนำโมเดลที่ผ่านขั้นตอน Validation เพื่อทดสอบว่าโมเดล นั้นทำงานได้ดีแค่ไหนกับข้อมูลที่ไม่เคยเห็นมาก่อน



รูปที่ 3.2 ตัวอย่างการทดสอบ

3.3.2.5. ทำการทำ Preprocessing เพื่อทำให้รูปนั้นอยู่ในลักษณะเดียวกัน

3

Preprocessing

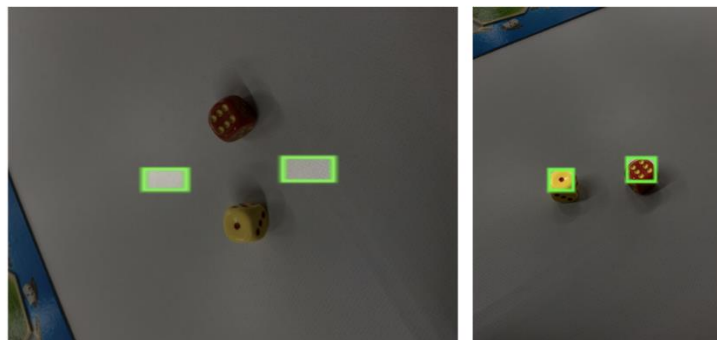
Decrease training time and increase performance by applying image transformations to all images in this dataset.

Auto-Orient	Edit	×
Resize Stretch to 640×640	Edit	×
Auto-Adjust Contrast Using Histogram Equalization	Edit	×
Grayscale	Edit	×
<div style="display: flex; align-items: center; justify-content: center;"> <div style="background-color: #4a4a9a; color: white; border-radius: 50%; width: 20px; height: 20px; display: flex; align-items: center; justify-content: center; margin-right: 5px;">+</div> Add Preprocessing Step </div>		

Continue

รูปที่ 3.3 ทำการทำ Preprocessing

3.3.2.5.1 ทำ Auto-Orient เพื่อแก้ไขปัญหามือหมุนภาพแต่ตำแหน่งกรอบตรวจจับยังอยู่ตำแหน่งเดิมของเฟรมไม่ขยับตามวัตถุในภาพที่ถูกหมุน



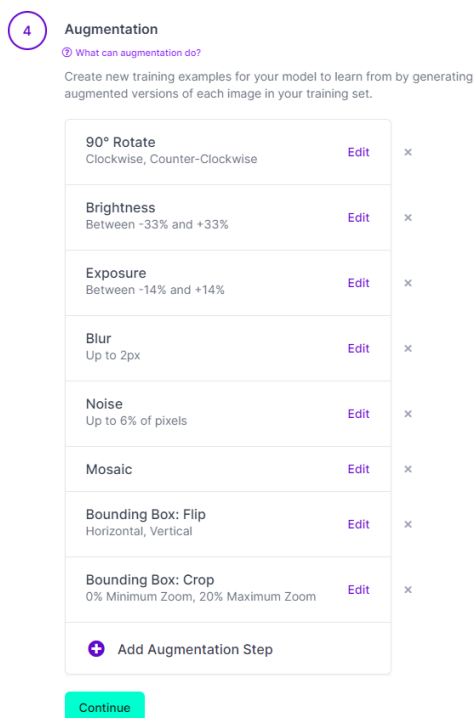
รูปที่ 3.4 Auto-Orient 13

3.3.2.5.2 Resize เพื่อทำให้ภาพทั้งหมดนั้นอยู่ในขนาดที่เท่ากันเพื่อทำให้ไม่เกิดปัญหาในการทำ Max pooling โดยทำการ Resize เป็น 640*640 pixel

3.3.2.5.3 Auto-Adjust Contrast เพื่อให้โมเดลตรวจจับขอบได้ดีขึ้นสำหรับการตรวจจับใบหน้า

3.3.2.5.4 Grayscale เพื่อทำให้ภาพทั้งหมดเป็นภาพที่เป็นสีขาวดำทำให้สามารถเทรนได้รวดเร็วกว่าจากการทำภาพสี

3.3.2.6. Image Augmentation เป็นการจำลองรูปภาพในสถานการณ์ต่างๆ สำหรับการเทรนโมเดลทำให้โมเดลมีความยืดหยุ่นเพิ่มขึ้นและเพิ่มจำนวนภาพ



รูปที่ 3.5 Image Augmentation

3.3.2.6.1 หมุนภาพ 90° เป็นการทดลองกรณีกล้องเกิดการขยับหรือเคลื่อนไหว และในขั้นตอนนี้ยังสามารถช่วยในการต่อเนื่องในการตรวจจับได้อีกด้วย

3.3.2.6.2 ปรับ Brightness เพื่อจำลองภาพในสถานการณ์ที่กล้องที่ใช้ตรวจจับตั้งอยู่ในบริเวณที่มีแสงรบกวนมากเกินไปหรือน้อยเกินไปจนส่งผลกระทบต่อใบหน้า แต่แสงที่ถูกปรับจะถูกปรับเฉพาะบริเวณใบหน้าเท่านั้น

3.3.2.6.3 ปรับ Exposure จำลองการมีแสงมารบกวนกล้องตรวจจับ โดยการปรับค่าคล้ายการปรับ Brightness แต่จะมีการปรับแสงของทั้งภาพต่างจากการปรับ Brightness ที่ปรับแสงบริเวณใบหน้าเพียงจุดเดียว

3.3.2.6.4 เพิ่ม Blur จำลองกล้องที่มีความคมชัดต่ำหรือภาพที่ได้รับมีความคมชัดไม่มากนัก เพื่อให้โมเดลยังสามารถตรวจจับใบหน้าได้

3.3.2.6.5 เพิ่ม Noise จำลองการเกิดสิ่งรบกวนหรือมีการบดบังใบหน้า



รูปที่3.6 การปรับ Brightness



รูปที่3.7 การปรับ Exposure



รูปที่3.8 การ blur รูปภาพ



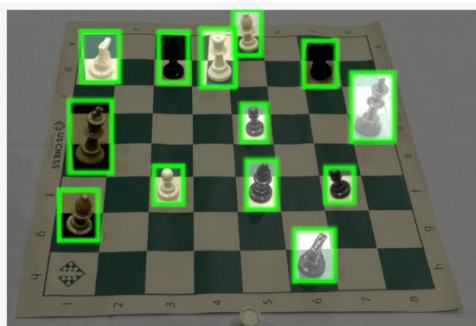
รูปที่3.9 เพิ่ม Noise

3.3.2.6.6 Mosaic เป็นการสุ่มรูปมาวางติดกันเป็นจำนวน 4 รูปเพื่อจำลองกรณีในภาพมีจำนวนที่ต้องตรวจจับจำนวนมากและไม่สามารถตรวจได้บางจุดและเป็นทั้งการ Random Crops ไปในตัวเพื่อช่วยการตรวจจับในพื้นที่ขนาดเล็ก และมีการสุ่ม Class ทำให้ยืดหยุ่นในการตรวจจับที่หลายๆ Class



รูปที่3.10 Mosaic

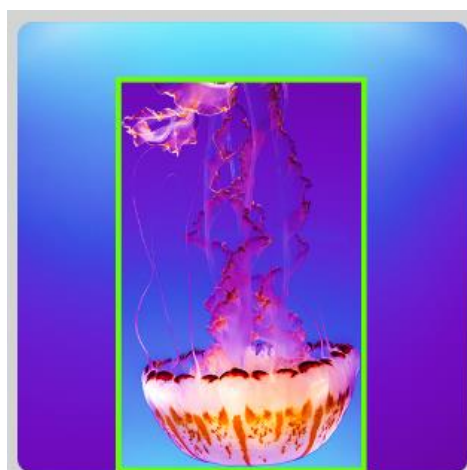
6.7 Bounding Box เพื่อทำการ Augmentation ในกรอบของรูปที่ถูก Label ในขอบเขตของกรอบที่ Label เพื่อเพิ่มจำนวนภาพและเพิ่มความแม่นยำในการตรวจจับ



รูปที่3.11 Bounding Box

6.7.1 Bounding Box: Flip เป็นการกลับด้านรูปภาพที่ถูก Label เพื่อเพิ่มจำนวนรูปภาพและเพิ่มสถานการณ์ให้โมเดลนั้นทน

6.7.2 Bounding Box: Crop เพื่อทำการ Crop ภาพที่ Label แล้วเพื่อเพิ่มจำนวนของภาพและยังเพิ่มความแม่นยำในการตรวจจับ



รูปที่3.12 Bounding Box: Flip



รูปที่3.13 Bounding Box: Crop

3.3.2.11. ทำการ Deploy โมเดลออกมาเพื่อนำมาใช้งานต่อ และตั้งชื่อว่า wayaj

3.3.2.12. เพิ่ม Dataset เพื่อให้โมเดลนั้นมีความเสถียรมากขึ้น โดยทำการนำโมเดลตัวก่อนมาทำการ Label ชุดข้อมูลเพื่อมาทำการทำโมเดลตัวใหม่

3.3.3. เขียนโปรแกรมใน VSCode

3.3.3.1 นำโมดูลเข้าในโปรแกรม

3.3.1.1 ดาวน์โหลดโมดูลที่จำเป็นต่อการเขียนโปรแกรม

3.3.1.2 ดาวน์โหลด Opencv ลงใน VSCode

3.3.1.3 ดาวน์โหลด Imutils ลงใน VSCode

3.3.1.4 ดาวน์โหลด Datetime ลงใน VSCode

3.3.1.5 ดาวน์โหลด Gspread ลงใน VSCode

3.3.1.6 ดาวน์โหลด Oauth2clinet ลงใน VSCode

3.3.3.2 นำเข้าโมเดล wayaj.pt ที่ทำการเทรนไว้เข้าในโปรแกรม

3.3.3.3 เขียนคำสั่งนำเข้าโมดูลต่างๆ ที่จำเป็นต่อการเขียนโปรแกรม

3.3.3.4 เขียนโปรแกรมใน VSCode

3.3.3.4.1 เขียนหา path ของโมเดล

3.3.3.4.2 เขียนโปรแกรมในการเรียกใช้การใช้งานกล้องที่ใช้ตรวจจับ โดยใช้

`cv2.VideoCapture(0)`

3.3.3.4.3 กำหนดค่าตัวแปรต่างๆ และสร้างฟังก์ชันเพื่อนำไปใช้ในการแสดงผล และคำนวณ

ค่าต่างๆ

3.3.3.4.4 เขียนส่วนแสดงผลภาพวิดีโอจากกล้อง

3.3.3.4.5 เขียนอัลกอริทึมในการตรวจจับ และนับจำนวนคนเข้าห้องสมุด

3.3.3.4.6 เชื่อมและส่งข้อมูลไปที่ Database

บทที่ 4

ผลการดำเนินงาน

การทดสอบประสิทธิภาพโปรแกรมการตรวจจับคนเข้าห้องสมุดนี้ คณะผู้จัดทำได้ดำเนินการทดสอบตามแผนที่ได้ดำเนินการไว้ดังนี้

4.1 การดำเนินการทดสอบประสิทธิภาพ

4.1.1 ทดสอบความแม่นยำในการตรวจจับใบหน้ากับระยะห่างจากกล้องตรวจจับ

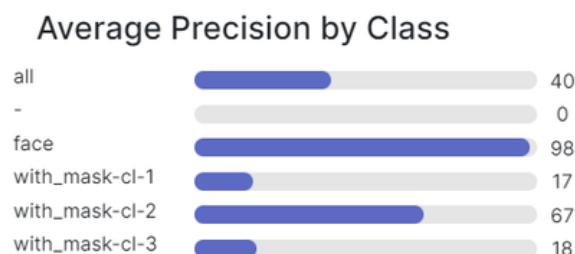
4.1.1.1 นำเครื่องคอมพิวเตอร์โน้ตบุ๊กจัดตั้งไว้บริเวณทางเข้าออกห้องสมุด

4.1.1.2 จัดบันทึกจำนวนผู้ใช้งานห้องสมุด เพื่อนำไปเปรียบเทียบกับโปรแกรมนับจำนวนที่ติดตั้งไว้ว่ามีความแม่นยำเท่าใดทดสอบ 20 ครั้ง ต่อ 1 ระยะ

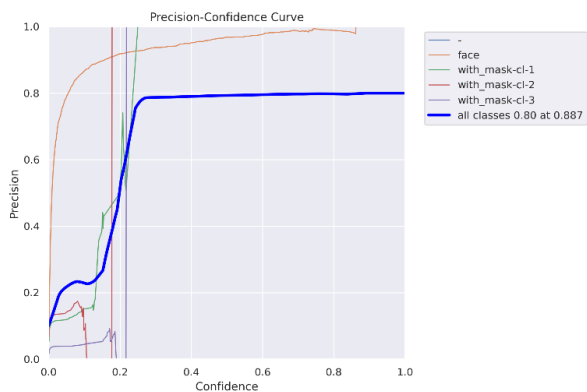
4.1.2 ทดสอบความแม่นยำด้วยชุดข้อมูล setA เป็นรูปภาพใบหน้าทั้งหมด 217 รูปภาพโดยและรูปภาพตัวอย่าง 20 รูปในโปรแกรมที่ใช้โมเดล wayaj และโมเดลอื่นๆที่ได้นำมาเปรียบเทียบ ได้แก่ โมเดล cv2.frontface และ YOLOv5x โดยรูปภาพเพิ่มเติมที่ใช้มีรายละเอียดดังนี้ รูปใบหน้าตรงปกติ 4 รูป รูปที่มีการนำสีมาตกแต่งใบหน้า 4 รูป มีวัตถุบดบังใบหน้า 3 รูป รูปมุมข้าง 5 รูป ย้อนแสง 4 รูป รวมเป็น 237 รูป โดยรูปภาพทั้งหมดเป็นภาพที่ไม่เคยถูกเทรนลงในโมเดล wayaj ทั้งสิ้น

4.2 วิเคราะห์ผลการทดสอบประสิทธิภาพ

4.2.1 กราฟความแม่นยำในการตรวจจับใบหน้าของแต่ละคลาสในโมเดล wayaj จากการวิเคราะห์ของเว็บไซต์ roboflow ประกอบด้วยคลาส -, face, with_mask-cl-1, with_mask-cl-2, with_mask-cl-3 โดยคลาสหลักที่ใช้มีเพียงแค่คลาส face ที่มีความแม่นยำถึง 98 %



รูปที่ 4.1 แผนภูมิแสดงความแม่นยำในการตรวจจับใบหน้าของแต่ละคลาส



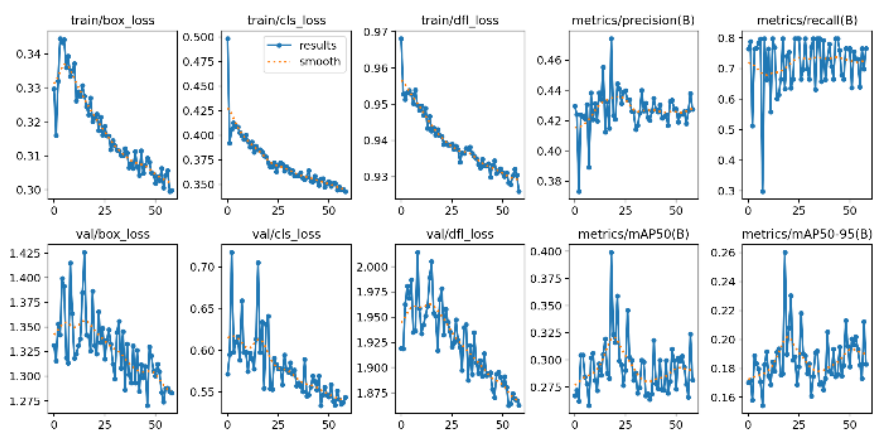
รูปที่ 4.2 กราฟแสดงความแม่นยำในการตรวจจับใบหน้าของแต่ละคลาส

4.2.2 ตารางแสดงความสัมพันธ์ระหว่างระยะห่างจากกล้องตรวจจับกับความแม่นยำของโมเดล wayaj ในการทดสอบ 20 ครั้ง ต่อ 1 ระยะ

ระยะ(เมตร)	ค่าความแม่นยำ
0.5	100%
1.0	100%
1.5	100%
2.0	100%
3.0	0%

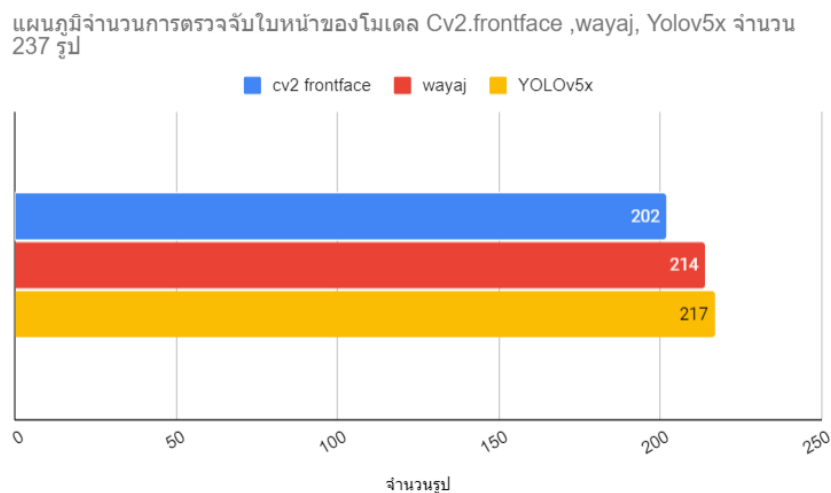
ตารางที่ 4.1 ตารางแสดงความสัมพันธ์ระหว่างระยะห่างจากกล้องตรวจจับกับความแม่นยำของโมเดล wayaj

4.2.2 กราฟแสดงความต่อเนื่องของค่าต่างๆจากการเทรน Ai



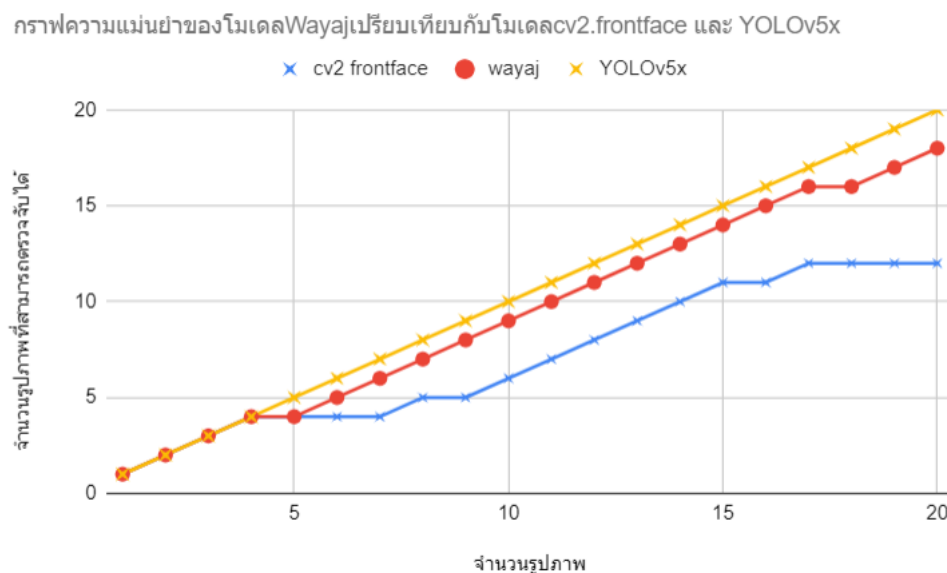
รูปที่ 4.3 กราฟแสดงความต่อเนื่องของค่าต่างๆจากการเทรน Ai

4.2.3 กราฟการเปรียบเทียบความแม่นยำของโมเดล wayaj เปรียบเทียบกับโมเดลcv2.frontface และ YOLOv5x กับชุดข้อมูล setA



รูปที่ 4.4 แผนภูมิแสดงการเปรียบเทียบความแม่นยำของโมเดล wayaj เปรียบเทียบกับโมเดล cv2.frontface และ YOLOv5x กับชุดข้อมูล setA

4.2.3 กราฟการเปรียบเทียบความแม่นยำของโมเดล wayaj เปรียบเทียบกับโมเดล cv2.frontface และ YOLOv5x กับรูปตัวอย่าง 20 รูป



รูปที่ 4.5 กราฟการเปรียบเทียบความแม่นยำของโมเดล wayaj เปรียบเทียบกับโมเดล cv2.frontface และ YOLOv5x กับรูปตัวอย่าง

บทที่ 5

สรุปผลการศึกษา

5.1 สรุปผลการศึกษา

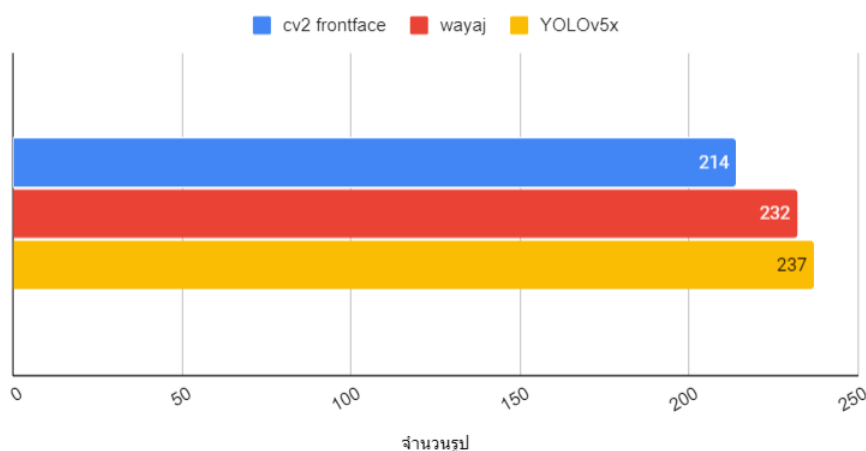
การสร้างโปรแกรมตรวจจับคนเข้าห้องสมุดนั้น มีวัตถุประสงค์เพื่อช่วยอำนวยความสะดวกในการตรวจจับคนเข้าห้องสมุด โดยใช้ตัวโปรแกรมนั้นตรวจจับแทนผู้ดูแลห้องสมุด

จากการทดสอบประสิทธิภาพของโปรแกรมตรวจจับคนเข้าห้องสมุด มีรายการในการทดสอบประสิทธิภาพ คือ ระยะห่างที่กล้องสามารถตรวจจับได้อย่างแม่นยำ ซึ่งระยะห่างที่กล้องสามารถตรวจจับได้อย่างมีประสิทธิภาพนั้นคือ ระยะ 1 ถึง 2 เมตร ห่างจากกล้อง โดยเมื่อทำการเปรียบเทียบกับโมเดลอื่นๆ ด้วยชุดข้อมูล setA 217 รูป และเพิ่มความหลากหลายของข้อมูลทางผู้จัดทำจึงเพิ่มการตรวจสอบเพิ่มด้วยชุดตัวอย่างเดียวกันทั้งหมด 20 รูป รวมเป็น 237 รูป ได้ค่าความแม่นยำดังนี้ YOLOv5x ได้ความแม่นยำ 100% โมเดล cv2.frontface ได้ความแม่นยำ 90.29% และโมเดล wayaj ได้ความแม่นยำ 97.89%

5.2 อภิปรายผล

จากการทดสอบประสิทธิภาพของโปรแกรมตรวจจับคนเข้าห้องสมุด มีรายการในการทดสอบแรกคือการทดสอบประสิทธิภาพการตรวจจับกับระยะทางเป็นจำนวน 20 ครั้ง คือ ระยะห่างที่กล้องสามารถตรวจจับได้อย่างแม่นยำ ได้แก่ระยะ 1 ถึง 2 เมตร ห่างจากกล้อง และเมื่อนำโมเดล wayaj ไปทดสอบประสิทธิภาพกับโมเดลตัวอื่นๆ ได้ค่าความแม่นยำดังนี้ YOLOv5x ได้ความแม่นยำ 100% โมเดล cv2.frontface ได้ความแม่นยำ 90.29% และโมเดล wayaj ได้ความแม่นยำ 97.89% โดยโมเดลถือได้ว่าโมเดล wayaj มีประสิทธิภาพมากกว่าโมเดล cv2.frontface ถึง 7.6% และยังมีประสิทธิภาพใกล้เคียงกับโมเดล YOLOv5x ที่มีการตรวจจับทั้งตัวในการประมวลผลอีกด้วย

แผนภูมิจำนวนการตรวจจับใบหน้าของโมเดล Cv2.frontface ,wayaj, YOLOv5x จำนวน 237 รูป



รูปที่ 5.1 แผนภูมิจำนวนการตรวจจับใบหน้าของโมเดล Cv2.frontface ,wayaj, YOLOv5x จำนวน 237 รูป

5.3 ข้อเสนอแนะ

จากการเขียนโปรแกรมเมื่อนำมาใช้จริงไม่สามารถใช้ Face recognition เพื่อเพิ่มความแม่นยำได้ เนื่องจากฮาร์ดแวร์มีประสิทธิภาพไม่ถึงทำให้ต้องตัดการทำ Face recognition ทิ้งไป และต้องการงบประมาณเพิ่มเติมเพื่อเพิ่ม GPU สำหรับการขยายฐานข้อมูลในการเพิ่มประสิทธิภาพของโมเดล

บรรณานุกรม

What Is Object Detection? [ออนไลน์]. สืบค้นเมื่อ 13 กรกฎาคม 2566. เข้าถึงจาก:
https://www.mathworks.com/discovery/objectdetection.html?fbclid=IwAR09LK3QQi634DZNLZP86v7U8U2WyeUDJ_f3C_YgBjvRoWzCpxWMC1MdxgU

Face Detection and Recognition Algorithm in Digital Image Based on Computer Vision Sensor. [ออนไลน์]. สืบค้นเมื่อ 13 กรกฎาคม 2566. เข้าถึงจาก:

https://www.hindawi.com/journals/js/2021/4796768/?fbclid=IwAR0y6x_iRRoMih2LJZ4m4yYkakTBJZBkQ9Nt0t33m9wb1Y9SYtkiOohPddU

การอ่านข้อมูล จาก Google Sheets ด้วย Python. [ออนไลน์]. สืบค้นเมื่อ 13 กรกฎาคม 2566. เข้าถึงจาก: <https://www.mindphp.com/developer/tips-python/7749-google-sheet-api.html?fbclid=IwAR3A0Pu7LeaExr0lCv3ZyOvOUce9mbI0SoUzbyzyQplXFrOqGoBgBPqeFk>

Python connect Google sheets by Google API. [ออนไลน์]. สืบค้นเมื่อ 13 กรกฎาคม 2566. เข้าถึงจาก: <https://dev.to/ksupdev/python-connect-google-sheets-by-google-api-2g3c?fbclid=IwAR1hxTLE63L-zXTxYLUC0JfZo6--JmoNME5nmSEpG6exAb1rH-M0a0y--d0>

What is neural network? [ออนไลน์]. สืบค้นเมื่อ 13 กรกฎาคม 2566. เข้าถึงจาก: <https://aws.amazon.com/what-is/neural-network/#:~:text=A%20neural%20network%20is%20a,that%20resembles%20the%20human%20brain.>

Gradient Descent คืออะไร อะไรคือ การเคลื่อนลงตามความชัน, Stochastic Gradient Descent (SGD) คืออะไร [ออนไลน์]. สืบค้นเมื่อ 13 กรกฎาคม 2566. เข้าถึงจาก: <https://www.bualabs.com/archives/631/what-is-gradient-descent-in-deep-learning-what-is-stochastic-gradient-descent-sgd-optimization-ep-1/>

Convolutional Neural Network [ออนไลน์]. สืบค้นเมื่อ 14 กรกฎาคม 2566. เข้าถึงจาก: <https://medium.com/@natthawatphongchit/%E0%B8%A1%E0%B8%B2%E0%B8%A5%E0%B8%AD%E0%B8%87%E0%B8%94%E0%B8%B9%E0%B8%A7%E0%B8%B4%E0%B8%98%E0%B8%B5%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%84%E0%B8%B4%E0%B8%94%E0%B8%82%E0%B8%AD%E0%B8%87-cnn-%E0%B8%81%E0%B8%B1%E0%B8%99-e3f5d73eebaa>

Roboflow [ออนไลน์]. สืบค้นเมื่อ 14 กรกฎาคม 2566. เข้าถึงจาก: <https://www.crunchbase.com/organization/roboflow>

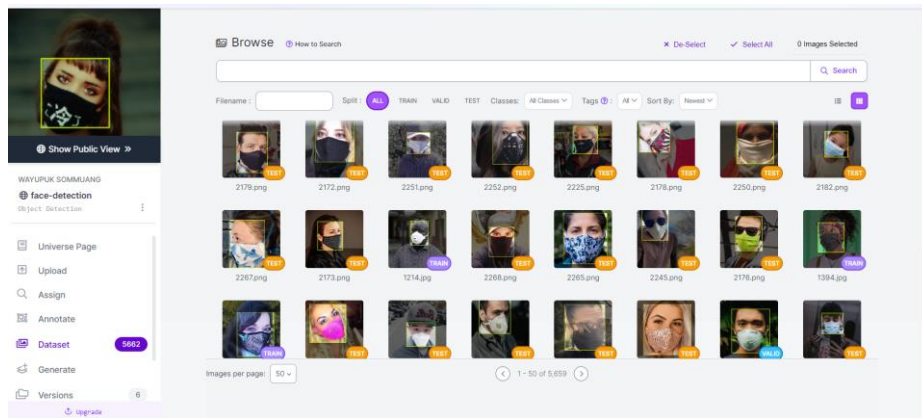
What is YOLOv5? A Guide for Beginners. [ออนไลน์]. สืบค้นเมื่อ 14 กรกฎาคม 2566.

เข้าถึงจาก: <https://blog.roboflow.com/yolov5-improvements-and-evaluation/>

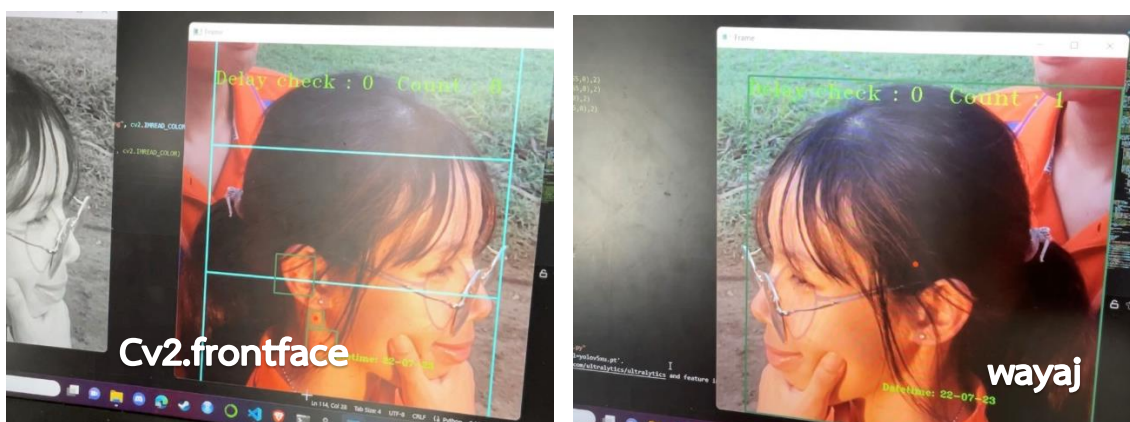
Face Detection using Haar Cascade Classifier in OpenCV and Python. [ออนไลน์].

สืบค้นเมื่อ 14 กรกฎาคม 2566. <https://medium.com/@rhugwedanshedge/face-detection-using-haar-cascade-classifier-in-opencv-and-python-d6318d0e189a>

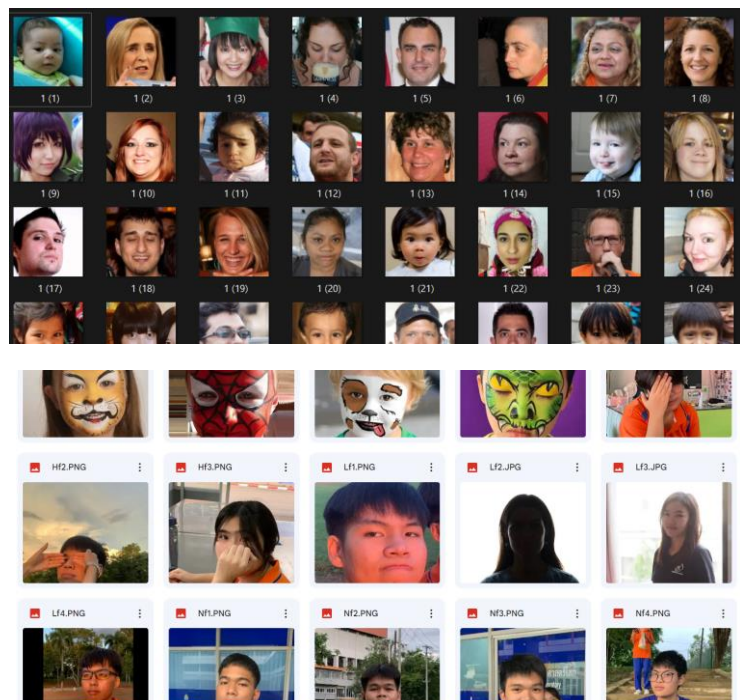
ภาคผนวก



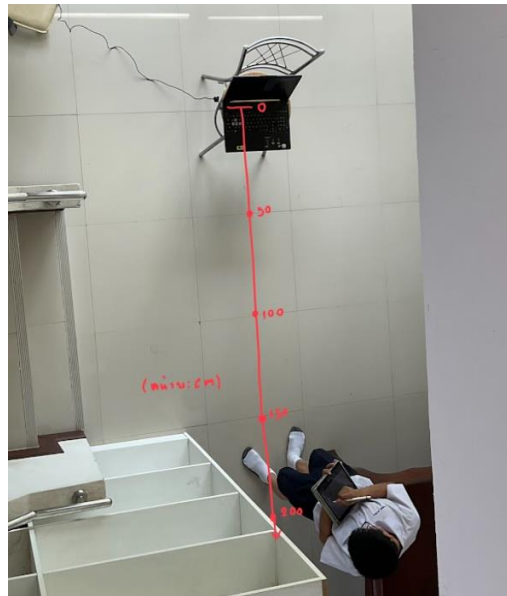
รูปที่ 1 ตัวอย่างภาพในการเทรนโมเดลจาก 5662 ภาพ 28



รูปที่ 2 การทดสอบประสิทธิภาพระหว่างการใช้โมเดล wayaj และโมเดลอื่นๆ



รูปที่ 3 และ 4 รูปตัวอย่างที่ใช้ทดสอบประสิทธิภาพระหว่างการใช้โมเดล wayaj และโมเดลอื่นๆ



รูปที่ 5 ทดสอบความแม่นยำในการตรวจจับใบหน้าตามระยะที่กำหนด



รูปที่ 6 และ 7 การติดตั้งเพื่อทดสอบการใช้งาน

```

1  import cv2
2  import argparse
3  import imutils
4  import datetime
5  import time
6  from threading import Thread
7  import gspread
8  from oauth2client.service_account import ServiceAccountCredentials
9  import json
10 import numpy as np
11 import tensorflow.lite as tflite
12 import tensorflow as tf
13 from ultralytics import YOLO
14 # myfacedetector-001.xml
15 import torch

```

รูปที่ 8 คำสั่งการนำเข้าโมดูลต่างๆ

```

19 def load_model(model_path):
20     r"""Load TFLite model, returns a Interpreter instance."""
21     interpreter = tf.lite.Interpreter(model_path=model_path)
22     interpreter.allocate_tensors()
23     return interpreter
24 class WebcamVideoStream:
25     def __init__(self, src=0):
26         # initialize the video camera stream and read the first frame
27         # from the stream
28         self.stream = cv2.VideoCapture(src)
29         (self.grabbed, self.frame) = self.stream.read()
30         # initialize the variable used to indicate if the thread should
31         # be stopped
32         self.stopped = False
33
34     def start(self):
35         # start the thread to read frames from the video stream
36         Thread(target=self.update, args=()).start()
37         return self
38     def update(self):
39         # keep looping infinitely until the thread is stopped
40         while True:
41             # if the thread indicator variable is set, stop the thread
42             if self.stopped:
43                 return
44             # otherwise, read the next frame from the stream
45             (self.grabbed, self.frame) = self.stream.read()
46     def read(self):
47         # return the frame most recently read
48         return self.frame
49     def stop(self):
50         # indicate that the thread should be stopped
51         self.stopped = True
52
53 CAMERA_WIDTH = 640
54 CAMERA_HEIGHT = 480

```

รูปที่ 9 การหา path ของโมเดลและตั้งค่ากล้อง

```

# Putting the boxes and labels on the image
if 553 > ((x+(w))/2) > 49 and 0 < ((y+(h))/2) < 480:
    if y > 480:
        y = 480
    if y < 0:
        y = 0
    if x > 553:
        x = 553
    if x < 49:
        x = 49
    cv2.rectangle(frame, (x,y), (w,h), (70,100,20), 2)
    center = center_point(x,y,w-x,h-y)
    cv2.circle(frame, (center), 4, (0,0,255), -1)
    countp = center_point(x,y,w,h)
    detect.append(countp)
    detect_count.append([x,y,w,h])
    print(countp)
    print(type(detect))
    CPaf = len(detect_count)

```

```

cv2.rectangle(frame, (x,y), (w,h), (70,100,20), 2)
center = center_point(x,y,w-x,h-y)
cv2.circle(frame, (center), 4, (0,0,255), -1)
countp = center_point(x,y,w,h)
detect.append(countp)
detect_count.append([x,y,w,h])
print(countp)
print(type(detect))
CPaf = len(detect_count)
for (x,y) in detect :
    if 480 > y > 0 and 553 > x > 49:
        if len(detect_count) > 0 :
            m=m
            print("detc"+str(len(detect_count)))
        else:
            CPaf = 0
            print(len(detect_count))
            print("hello ma boi")
            print("CPaf:"+str(type(CPaf)))
            if CPbf == CPaf :
                print("hekwwek[eeihwge")
                break
        else :
            if CPaf == 0 :
                CPbf = 0
                print("CPBF=0")
            else :
                if CPaf > CPbf:
                    count = count + int(CPaf-CPbf)
                    n+=1
                    CPbf = CPaf
                else:
                    m=m
print(detect_count)

```

รูปที่ 10 และ 11 ตัวอย่างโปรแกรมหลัก

```

249 actualday = dayl[0]
250 print(actualday)
251 actualday +=1
252 print(month)
253 scopes = [
254     'https://www.googleapis.com/auth/spreadsheets',
255     'https://www.googleapis.com/auth/drive'
256 ]
257 keys = "C:\VScode\PYTHON\keys.json"
258 credentials = ServiceAccountCredentials.from_json_keyfile_name(keys, scopes) #access the json key you downloaded earlier
259 file = gspread.authorize(credentials) # authenticate the JSON key with gspread
260 sheet = file.open("project") #open sheet
261 sheettt = sheet.sheet1 #replace sheet_name with the name that corresponds to yours, e.g, it can be sheet1
262 # dayn = int(sheettt.acell('D1').value)
263 sheettt.update_acell('A'+ str(actualday),str(maintime))
264
265 sheettt.update_acell('C'+ str(actualday),str(endtime))
266
267 if int(sheettt.acell('B'+ str(actualday)).value) > 0:
268     sheettt.update_acell('B'+ str(actualday),str(count+int(sheettt.acell("B"+ str(actualday)).value)))
269 else:
270     sheettt.update_acell('B'+ str(actualday),str(count))
271 ch = "-choose(WEDDAY)"
272 ab = "A"+str(actualday)+",2)"
273 daya = ",","mon","tue","wed","thu","fri","sat","sun"
274 alll = ch + ab + daya
275 sheettt.update_acell('O'+str(actualday),alll)
276 # dayn +=1
277 sheettt.update_acell('D1',str(actualday))
278
279 print("success")

```

รูปที่ 12 อัปเดตข้อมูลลงใน google sheet



รูปที่ 13 คิวอาร์โค้ดแสดงรายละเอียดโปรแกรมเพิ่มเติม