



Goggle : People Video Analytics and Deep Learning Platform

Goggle แพลตฟอร์มการเรียนรู้เชิงลึกและระบบวิเคราะห์การกระทำของมนุษย์

นายปฐมพงศ์ สินธุจาม

นายศุภกร เบญจวิกรัย

นายอุกฤษฎ์ เลิศวรณาการ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมหุ่นยนต์และระบบอัตโนมัติ

สถาบันวิทยาการหุ่นยนต์ภาคนาม

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ปีการศึกษา 2562



Goggle : People Video Analytics and Deep Learning Platform

Goggle แพลตฟอร์มการเรียนรู้เชิงลึกและระบบวิเคราะห์การกระทำของมนุษย์

นายปฐมพงศ์ สินธุจาม

นายศุภกร เบญจวิกรัย

นายอุกฤษฎ์ เลิศวรรณาการ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมหุ่นยนต์และระบบอัตโนมัติ

สถาบันวิทยาการหุ่นยนต์ภาคนาม

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ปีการศึกษา 2562

Google แพลตฟอร์มการเรียนรู้เชิงลึกและระบบบวิเคราะห์การกระทำของมนุษย์

นายปฐมพงศ์ สินธุ์งาม

นายศุภกร เบญจวิกราย

นายอุกฤษฎ์ เดิศวรณาการ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาชีววิศวกรรมหุ่นยนต์และระบบอัตโนมัติ

สถาบันวิทยาการหุ่นยนต์ภาคสนาม

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ปีการศึกษา 2562

คณะกรรมการสอบวิทยานิพนธ์

ประธานกรรมการสอบวิทยานิพนธ์

(ดร.วราสินี ฉายแสงมงคล)

อาจารย์ที่ปรึกษาวิทยานิพนธ์

(ดร.วราสินี ฉายแสงมงคล)

กรรมการสอบวิทยานิพนธ์

(อ.บรรศักดิ์ ศกุลเกื้อกุลสุข)

กรรมการสอบวิทยานิพนธ์

(ดร.บุญทริกา เกษมสันติธรรม)

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ชื่อวิทยานิพนธ์	Goggle แพลตฟอร์มการเรียนรู้เชิงลึกและระบบวิเคราะห์การกระทำของมนุษย์
หน่วยกิต	6
ผู้เขียน	นายปัจมพงศ์ สินธุจัน นายศุภกร เบญจวิกรัย นายอุกฤษฎ์ เลิศวรรณการ
อาจารย์ที่ปรึกษา	ทีปรึกษาวิทยานิพนธ์หลัก ดร.วราสินี ฉายแสงมงคล
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมที่นุนยนต์และระบบอัตโนมัติ
คณะ	สถาบันวิทยาการที่นุนยนต์ภาคสนาม
ปีการศึกษา	2562

บทคัดย่อ

งานวิทยานิพนธ์นี้เป็นงานที่เกี่ยวกับการออกแบบและสร้างเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ รวมถึงระบบวิเคราะห์การกระทำของมนุษย์ โดยใช้ชื่อว่า Goggle แพลตฟอร์มการเรียนรู้เชิงลึกและระบบวิเคราะห์การกระทำของมนุษย์ ซึ่งมีจุดประสงค์เพื่อให้ผู้พัฒนาสามารถใช้งานเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ในการสร้างชุดข้อมูลสำหรับสร้างปัญญาประดิษฐ์ได้ง่ายและสะดวกขึ้น ภาพรวมของวิทยานิพนธ์จะแบ่งออกเป็นหัวข้อหลักสองส่วน โดยที่ส่วนแรกเป็นการออกแบบและสร้างแอปพลิเคชันที่ใช้ในการสร้างชุดข้อมูลจากวิดีโอ สำหรับพัฒนาโมเดลปัญญาประดิษฐ์ และส่วนที่สองเป็นใช้ชุดข้อมูลที่ได้จากแอปพลิเคชันในการพัฒนาโมเดลปัญญาประดิษฐ์สำหรับจำแนกการกระทำการกระทำของมนุษย์ เพื่อทดสอบประสิทธิภาพของชุดข้อมูล

คำสำคัญ : ระบบวิเคราะห์การกระทำการกระทำของมนุษย์ / เครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ / Goggle

กิตติกรรมประกาศ

ขอขอบพระคุณ ดร.วราสินี ฉายแสงมงคล อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก ที่ได้สละเวลามาให้คำปรึกษา ชี้แนะแนวทาง ให้ความรู้ในด้านต่างๆ ที่จำเป็นต่องานวิจัย รวมถึงการให้การสนับสนุนในเรื่องอุปกรณ์ในการทำวิจัย ช่วยตรวจสอบและแก้ไขวิทยานิพนธ์ให้เป็นไปอย่างสมบูรณ์ ตลอดจนกรุณาให้เกียรติเป็นประธานกรรมการสอบวิทยานิพนธ์

ขอขอบพระคุณอาจารย์อาจารย์ บวรศักดิ์ สกุลเกื้อกูลสุข ที่กรุณาให้เกียรติเป็นกรรมการสอบวิทยานิพนธ์ ให้คำแนะนำที่เป็นประโยชน์ต่อการวิจัย และการแก้ไขปรับปรุงงานวิจัย ตลอดจนตรวจสอบแก้วิทยานิพนธ์ให้ดำเนินไปอย่างสมบูรณ์

ขอขอบพระคุณอาจารย์ ดร.บุญทริกา เกษมสันติธรรม ที่กรุณาให้เกียรติเป็นกรรมการสอบวิทยานิพนธ์ ให้คำแนะนำที่เป็นประโยชน์ต่อการวิจัย และการแก้ไขปรับปรุงงานวิจัย ตลอดจนตรวจสอบแก้วิทยานิพนธ์ให้ดำเนินไปอย่างสมบูรณ์

ขอขอบพระคุณคณาจารย์ และบุคลากรในสถาบันวิทยาการหุ่นยนต์ภาควิชานามทุกท่าน ที่ได้ให้คำปรึกษา และช่วยเหลือด้านสถานที่พร้อมทั้งส่งอำนวยความสะดวกต่างๆ ในระหว่างการทำวิทยานิพนธ์

ขอขอบคุณนักศึกษาปริญญาตรี สถาบันวิทยาการหุ่นยนต์ภาควิชานามทุกท่าน ที่ได้ให้คำแนะนำ ถามไถ่ และเป็นกำลังใจมาโดยตลอด

และสุดท้ายนี้ ขอน้อมรำลึกถึงพระคุณบิดา มารดา และครอบครัว ที่ส่งเสริมให้กำลังใจ และให้การสนับสนุนในเรื่องต่างๆ จนกระทั้งข้าพเจ้าประสบความสำเร็จในการศึกษา

นายปฐมพงศ์ สินธุรงาม
นายศุภกร เบญจวิกรัย
นายอุกฤษฎ์ เลิศวรรณาการ

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	๑
กิตติกรรมประกาศ	๔
สารบัญ	๕
รายการรูปภาพ	๗
รายการตาราง	๘
รายการสัญลักษณ์	๙
ประมวลศัพท์และตัวย่อ	๙
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	1
1.4 ขอบเขตการดำเนินงาน	2
1.5 ขั้นตอนการดำเนินงาน	2
บทที่ 2 ทฤษฎี/การวิจัยที่เกี่ยวข้อง	4
2.1 ทฤษฎีที่เกี่ยวข้อง	5
2.1.1 Convolutional neural network	5
2.1.2 Optical flow	9
2.1.3 โมเดลปัญญาประดิษฐ์สำหรับตรวจจับวัตถุที่เกี่ยวข้อง	11
2.2 การประมวลผลวิดีโอ	14
2.2.1 การตรวจจับวัตถุ	14
2.2.2 ระบบติดตามการเคลื่อนไหวของวัตถุ	15
2.2.3 ระบบระบุตัวตนของบุคคล	17
2.2.4 ระบบจำแนกการกระทำของมนุษย์	19
2.3 ชุดข้อมูลสำหรับการประมวลผลวิดีโอ	20
2.4 โมเดลปัญญาประดิษฐ์สำหรับจำแนกการกระทำการของมนุษย์	33
2.4.1 ResNet	33
2.4.2 Inflated 3D convolutional network	35

สารบัญ (ต่อ)

เรื่อง	หน้า
2.4.3 เครื่องมือกำกับคุณลักษณะ	37
บทที่ 3 ระเบียบวิจัย	39
3.1 ความต้องการของระบบ	39
3.1.1 ความต้องการใช้งาน (functional requirements)	39
3.1.2 ความต้องการเชิงวิศวกรรม (non-functional requirements)	39
3.2 ภาพรวมระบบของเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์	40
3.3 หน้าที่ความรับผิดชอบ	41
3.4 เครื่องมือที่ใช้ในงานวิจัย	41
3.5 ภาษาที่ใช้ในการพัฒนาระบบ	42
3.6 Program library ที่ใช้ในการพัฒนาระบบและแอพพลิเคชัน	42
3.7 แผนการดำเนินงาน	42
3.8 การออกแบบหน้าต่างแอพพลิเคชันของเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์	43
3.8.1 เครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์	43
3.9 การออกแบบการทดสอบการทำงานของเครื่องมือที่ใช้	53
3.9.1 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์สำหรับการทำการตรวจสอบภาพบุคคล	53
3.10 การออกแบบการทดสอบการทำงานโดยนำตัวอย่างมาเข้ามาทดสอบ	54
3.10.1 ทดสอบประสิทธิภาพการทำงานของระบบนำตัวอย่างมาเข้ามาทดสอบ	54
3.11 การออกแบบการทดสอบการระบุตัวตนของมนุษย์	55
3.11.1 ทดสอบประสิทธิภาพการทำงานของระบบระบุตัวตนของบุคคลภายในภาพ	55
3.12 การออกแบบการทดสอบการจัดลำดับของมนุษย์	56
3.12.1 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกเทรนด์ผ่าน AVA โดยใช้ชุดข้อมูลของ AVA ในการทดสอบและเทียบผลลัพธ์กับแหล่งอ้างอิง	56
3.12.2 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกสร้างด้วย AVA และใช้ชุดข้อมูลที่ผู้วิจัยสร้างขึ้นในการทดสอบและเทียบผลลัพธ์กับแหล่งอ้างอิง	57
3.12.3 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่ถูกสร้างด้วยชุดข้อมูลที่ผู้วิจัยสร้างขึ้น และใช้ชุดข้อมูลที่ผู้วิจัยสร้างขึ้นในการทดสอบและเทียบผลลัพธ์กับแหล่งอ้างอิง	58

สารบัญ (ต่อ)

เรื่อง	หน้า
บทที่ 4 ผลการดำเนินงาน	59
4.1 เครื่องมือกำกับคุณลักษณะ	59
4.1.1 หน้าต่างแสดงผลของแอพพลิเคชัน	59
4.1.2 ผลลัพธ์การทำงานในแต่ละหน้าต่างของแอพพลิเคชัน	63
4.2 ผลการทดลองการตรวจจับวัตถุ	66
4.2.1 ข้อมูลรายละเอียดประกอบการทดสอบ	66
4.2.2 ผลทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์สำหรับการทำงาน ตรวจจับภาพบุคคล	66
4.3 ผลการทดสอบระบบติดตามตำแหน่งของมนุษย์	67
4.3.1 ข้อมูลรายละเอียดประกอบการทดสอบ	67
4.3.2 ผลทดสอบประสิทธิภาพ และความเร็วในการประมวลผล	67
4.4 ผลการทดสอบระบบระบุตัวตนของมนุษย์	68
4.4.1 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์สำหรับการระบุตัวตน ของบุคคล	68
4.5 ผลการทดสอบการจัดการกระทำของมนุษย์	71
4.5.1 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกเทรนด์ผ่าน AVA เทียบผลลัพธ์กับแหล่งอ้างอิง ได้ผลการทดลองดังตารางต่อไปนี้	71
4.5.2 ผลการทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกเทรนด์ ผ่าน AVA และ ใช้ชุดข้อมูลที่ผู้วิจัยสร้างขึ้น ในการทดสอบและเทียบผลลัพธ์กับแหล่งอ้างอิง ...	71
4.5.3 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกเทรนด์ผ่านชุด ข้อมูลสำหรับการเทรนด์ที่ผู้วิจัยสร้างขึ้น และ ใช้ชุดข้อมูลที่ผู้วิจัยสร้างขึ้น ในการทดสอบ และเทียบผลลัพธ์การทดสอบก่อนหน้า	71
เอกสารอ้างอิง	75
ภาคผนวก ก ตัวอย่างชุดข้อมูลที่ผู้วิจัยสร้างขึ้น	76

รายการรูปภาพ

รูป	หน้า
รูปที่ 1.1 แผนการดำเนินงาน.....	3
รูปที่ 2.1 ตัวอย่างโครงสร้างของ CNN ที่ใช้ในการจำแนกหมวดหมู่ของวัตถุ.....	5
รูปที่ 2.2 ตัวอย่างเครื่องเรนล และภาพที่ใช้ในการประมวลผล	5
รูปที่ 2.3 ตัวอย่างการหาผังคุณลักษณะ	6
รูปที่ 2.4 ตัวอย่างการใช้ max pooling และ average pooling กับภาพ.....	6
รูปที่ 2.5 ภาพแสดงการหา IoU ด้วยการใช้คำตอบจริงของการทำงานและกรอบสี่เหลี่ยมที่ได้จากการทำงาน.....	8
รูปที่ 2.6 ตัวอย่างการทำงานของ NMS	8
รูปที่ 2.7 ตัวอย่าง optical flow ของการเคลื่อนที่ของลูกบอล	9
รูปที่ 2.8 กระบวนการทำงานของโครงสร้างโมเดลปัญญาประดิษฐ์ของ YOLO	11
รูปที่ 2.9 โครงสร้างทั่วไปของโมเดลปัญญาประดิษฐ์ของ YOLO	11
รูปที่ 2.10 โครงสร้างโมเดลปัญญาประดิษฐ์ของ YOLO	12
รูปที่ 2.11 โครงสร้างทั่วไปของโมเดลปัญญาประดิษฐ์ของ Faster RCNN	13
รูปที่ 2.12 โครงสร้างทั่วไปของโมเดลปัญญาประดิษฐ์ SSD	14
รูปที่ 2.13 แนวคิดของระบบติดตามการเคลื่อนไหวของวัตถุ.....	16
รูปที่ 2.14 การแบ่งภาพออกเป็น 8 ส่วนของระบบระบุตัวตนของบุคคล	17
รูปที่ 2.15 ตัวอย่างการประยุกต์ใช้งานระบบจำแนกการกระทำนุษย์ ^[22]	19
รูปที่ 2.16 โครงสร้างโมเดลปัญญาประดิษฐ์ของ deep bag of frames.....	22
รูปที่ 2.17 โครงสร้าง LSTM ที่ใช้การอ้างอิงในบทความนี้.....	22
รูปที่ 2.18 หลักการของ Residual block ของ ResNet	33
รูปที่ 2.19 โครงสร้างของโมเดลปัญญาประดิษฐ์ I3D ^[16]	35
รูปที่ 2.20 โครงสร้างของโมเดลปัญญาประดิษฐ์ I3D ^[16]	35
รูปที่ 2.21 หน้าต่างการทำงานของโปรแกรม DarkLabel	37
รูปที่ 2.22 หน้าต่างการทำงานของโปรแกรม OpenLabeling	38
รูปที่ 3.1 ภาพรวมระบบของเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์.....	40
รูปที่ 3.2 กระบวนการหลักของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์.....	43
รูปที่ 3.3 หน้าต่าง Select ของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์.....	44

รายการรูปภาพ (ต่อ)

รูป	หน้า
รูปที่ 3.4 ตำแหน่งของแต่ละวิดเจ็ตในหน้าต่าง Select	45
รูปที่ 3.5 หน้าต่าง Detect ของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์	46
รูปที่ 3.6 ตำแหน่งของแต่ละวิดเจ็ตในหน้าต่าง Detect.....	47
รูปที่ 3.7 หน้าต่าง Track ของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์	48
รูปที่ 3.8 ตำแหน่งของแต่ละวิดเจ็ตในหน้าต่าง Track.....	49
รูปที่ 3.9 หน้าต่าง Label ของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์.....	50
รูปที่ 3.10 ตำแหน่งของแต่ละวิดเจ็ตในหน้าต่าง Label.....	51
รูปที่ 3.11 ตัวอย่างข้อมูลภายในไฟล์ xml.....	52
รูปที่ 4.1 รูปหน้าต่างแสดงผลของหน้าต่าง Select	59
รูปที่ 4.2 รูปหน้าต่างแสดงผลของหน้าต่าง Detect	60
รูปที่ 4.3 รูปหน้าต่างแสดงผลของหน้าต่าง Track.....	61
รูปที่ 4.4 รูปหน้าต่างแสดงผลของหน้าต่าง Label	62
รูปที่ 4.5 รูปผลลัพธ์การแยกเฟรมที่มีมนุษย์อยู่ และไม่มีมนุษย์อยู่ภายในเฟรม	63
รูปที่ 4.6 รูปคีย์เฟรมที่ถูกตีกรอบสีเหลี่ยมในส่วนที่มีมนุษย์อยู่	63
รูปที่ 4.7 รูปผลลัพธ์การทำงานของหน้าต่าง Track.....	64
รูปที่ 4.8 รูปผลลัพธ์การทำงานของหน้าต่าง Label	65
รูปที่ 4.9 ภาพตัวอย่างชุดข้อมูลสำหรับการทำงานทดลองครั้งที่ 1	68
รูปที่ 4.10 ภาพตัวอย่างชุดข้อมูลสำหรับการทำงานทดลองครั้งที่ 2.....	69
รูปที่ 4.11 ภาพตัวอย่างชุดข้อมูลสำหรับการทำงานทดลองครั้งที่ 3.....	69
รูปที่ 4.12 ภาพตัวอย่างชุดข้อมูลสำหรับการทำงานทดลองครั้งที่ 4.....	70
รูปที่ 4.13 ภาพตัวอย่างชุดข้อมูลสำหรับการทำงานทดลองครั้งที่ 5.....	70
รูปที่ 4.14 ภาพตัวอย่างชุดข้อมูลสำหรับการทำงานทดลองครั้งที่ 6.....	71
รูปที่ ก.1 รูปผลลัพธ์การทำงานของหน้าต่าง Track.....	76

รายการตาราง

ตาราง	หน้า
ตารางที่ 2.1 ตารางแสดงการเปรียบเทียบระหว่าง softmax function และ sigmoid function.....	7
ตารางที่ 2.2 ผลการทดสอบโมเดลต่างๆบนชุดข้อมูลสำหรับทดสอบของ YouTube-8M.....	25
ตารางที่ 2.3 ผลการทดลองของวิธีต่างๆบนคุณลักษณะระดับเฟรม	29
ตารางที่ 2.4 ประสิทธิภาพของโมเดล Resnet50 I3D ที่ใช้ชุดข้อมูล Kinetics และ Moments in Time ..	31
ตารางที่ 2.5 อัตราการถูกจำแนกของความผิดพลาดของชุดข้อมูลทดสอบ ImageNet.....	33
ตารางที่ 2.6 ค่าความผิดพลาดที่ได้จากการทดลองจำแนกชั้นของโมเดลปัญญาประดิษฐ์ ResNet บนชุดของข้อมูล CIFAR-10	34
ตารางที่ 2.7 ประสิทธิภาพของโมเดล I3D แบบ two-stream เมื่อใช้ข้อมูลจาก UCF-101, HMDB-51 และ Kinetics ในการสร้างและทดสอบด้วยเครื่องมือวัดผลแบบความแม่นยำจากการทำนายอันดับแรกสุด.....	36
ตารางที่ 4.1 ข้อมูลผลการทำงานของโมเดลปัญญาประดิษฐ์สำหรับการทำการทำตรวจสอบจับภาพบุคคล	66
ตารางที่ 4.2 ผลการทดสอบประสิทธิภาพของการตรวจสอบสีเหลี่ยมรายในวีดีโอ.....	67
ตารางที่ 4.3 ผลการทดสอบความแม่นยำของโมเดลปัญญาประดิษฐ์.....	68
ตารางที่ 4.4 ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 1	68
ตารางที่ 4.5 ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 2	69
ตารางที่ 4.6 ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 3	69
ตารางที่ 4.7 ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 4	70
ตารางที่ 4.8 ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 5	70
ตารางที่ 4.9 ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 6	71
ตารางที่ 4.10 ผลการทดสอบความแม่นยำของโมเดลปัญญาประดิษฐ์ที่比べผลลัพธ์กับแหล่งอ้างอิง.....	71
ตารางที่ 4.11 ผลการทดสอบความแม่นยำของโมเดลปัญญาประดิษฐ์ เมื่อใช้กับชุดข้อมูลที่ผู้วิจัยสร้างขึ้น	72
ตารางที่ 4.12 ผลการทดสอบความแม่นยำของโมเดลปัญญาประดิษฐ์ที่ผู้วิจัยสร้างขึ้น ใช้กับชุดข้อมูลที่ผู้วิจัยสร้างขึ้น	72

รายการสัญลักษณ์

θ	เซ็ต้า
d	distance
kg	Kilogram
m^2	Square Metre

ประมวลศัพท์และตัวย่อ

AVA	Atomic Visual Actions
Artificial intelligence	ปัญญาประดิษฐ์
Machine learning model	โมเดลปัญญาประดิษฐ์
Label	คำกำกับที่บ่งบอกถึงคุณลักษณะของสิ่งที่สนใจ
Labeling	การสร้างคำกำกับคุณลักษณะ
Human action classification	การจำแนกการกระทำของมนุษย์
Video labeling	การสร้างคำกำกับคุณลักษณะภายในวิดีโอ
Video analytics	การวิเคราะห์ผลวิดีโอ
Uniform label distribution	การที่มีจำนวนตัวอย่างภายใต้คำกำกับเท่ากันทุกประเภท
KMUTT	King Mongkut's University of Technology Thonburi

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

บริษัท เพอเซปตรา ดำเนินธุรกิจเกี่ยวกับการให้บริการและคำปรึกษาเกี่ยวกับปัญญาประดิษฐ์ (artificial intelligence) เนื่องจากปัจจุบันนี้ความสามารถและประสิทธิภาพของปัญญาประดิษฐ์มีความก้าวหน้าขึ้นจนสามารถก้าวข้ามความสามารถของมนุษย์ในงานหลายประเภท ทำให้ลูกค้านั้นมีความต้องการที่จะให้ทางบริษัทสร้างปัญญาประดิษฐ์เพื่อนำไปใช้งานหรือแก้ปัญหาที่ต่างกันออกไป เช่น ใช้ปัญญาประดิษฐ์มาช่วยประเมินผลภาพจากกล้องวงจรปิด เพื่อหาบุคคลที่มีท่าทางน่าสงสัย เป็นต้น ซึ่งการจะสร้างปัญญาประดิษฐ์ที่เหมาะสมกับการแก้ปัญหาเหล่านั้น จะเป็นต้องมีชุดข้อมูล (dataset) ที่เหมาะสม บางครั้งอาจต้องใช้มนุษย์ในการสร้างขึ้นมาโดยการเก็บข้อมูลวิดีโอ และลงมือสร้างชุดข้อมูลจากวิดีโอด้วยตัวเอง หนึ่งในปัจจัยสำคัญในการพัฒนาไม้เดลปัญญาประดิษฐ์ให้มีประสิทธิภาพสูงคือจำนวนข้อมูล ซึ่งหากมีจำนวนนิวิดีโอด้วยจำนวนมาก การใช้มนุษย์ในการสร้างชุดข้อมูลนั้นอาจจะต้องใช้มนุษย์เป็นจำนวนมาก และใช้เวลานาน

ทางคณะผู้วิจัยจึงมีความต้องการที่จะออกแบบและสร้างต้นแบบของเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ (AI-assisted video labeling tool) สำหรับสร้างชุดข้อมูลจากวิดีโอ เพื่อช่วยแบ่งเบาภาระของผู้พัฒนาในการสร้างชุดข้อมูลสำหรับการพัฒนาโมเดลปัญญาประดิษฐ์ในการแก้ปัญหาที่ลูกค้าต้องการ โดยโครงการสหกิจนี้เน้นศึกษาเกี่ยวกับการวิเคราะห์และจำแนกการกระทำของมนุษย์ (human action classification) ภายใต้ในสำนักงานจากวิดีโอดังกล่าว

1.2 ວັດຖຸປະສົງຄໍ

1. เพื่อสร้างต้นแบบของเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ ที่มีนุชย์และปัญญาประดิษฐ์ทำงานร่วมกันเพื่อสร้างชุดข้อมูลในการนำไปพัฒนาปัญญาประดิษฐ์อื่นๆ ที่เหมาะสมกับปัญหาที่ต้องการ
 2. เพื่อออกแบบและสร้างต้นแบบของระบบประมวลผลวิดีโอ (video analytics) ที่สามารถตรวจจับมนุษย์และจำแนกการกระทำพื้นฐานของมนุษย์ภายในสำนักงาน ประกอบด้วย ยืน นั่ง เดิน เล่นโทรศัพท์ กินข้าว นอน โดยใช้ปัญญาประดิษฐ์
 3. เพื่อสร้างเครื่องมือที่สามารถสร้างชุดข้อมูลสำหรับการจำแนกการกระทำการของมนุษย์ให้สามารถใช้งานได้ง่าย สะดวกสบายมากขึ้น และมีประสิทธิภาพที่สูงกว่าเครื่องมือตัวอื่นในปัจจุบัน

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- เพิ่มความสอดคล้องในการสร้างஆட்டமூலสำหรับพัฒนาโนเดลปัญญาประดิษฐ์จากวิดีโอ
 - ต้นแบบระบบประมวลผลวิดีโอที่สามารถจำแนกการกระทำของมนุษย์

1.4 ขอบเขตการดำเนินงาน

1. สร้างต้นแบบของเครื่องมือที่มีกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ โดยระบบจะประกอบไปด้วยสี่ส่วนดังนี้
 - (a) หน้าต่างของเครื่องมือที่มีกำกับคุณลักษณะ (user interface)
 - (b) ระบบตรวจจับมนุษย์ในภาพ (person detection)
 - (c) ระบบติดตามการเคลื่อนไหวของมนุษย์ในวิดีโอ (person tracker)
 - (d) ระบบจำแนกการกระทำการของมนุษย์ ซึ่งประกอบไปด้วย ยืน นั่ง เดิน เล่นโทรศัพท์ กินข้าว นอน
2. ทดสอบโมเดลปัญญาประดิษฐ์สำหรับจำแนกการกระทำการของมนุษย์กับชุดข้อมูลที่ได้จากการใช้งานจริงได้หรือไม่
3. พัฒนาโมเดลปัญญาประดิษฐ์สำหรับจำแนกการกระทำการของมนุษย์ภายในสำนักงานอย่างน้อย 2 โมเดล

1.5 ขั้นตอนการดำเนินงาน

การดำเนินงานวิจัยถูกแบ่งออกเป็นสามส่วน โดยส่วนแรกคือการศึกษาเทคโนโลยีในปัจจุบันเพื่อหาความเป็นไปได้และกำหนดขอบเขตของงาน ส่วนที่สองคือออกแบบและสร้างเครื่องมือที่มีกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ เพื่อช่วยผู้พัฒนาในการสร้างชุดข้อมูล และส่วนที่สุดท้ายคือการนำชุดข้อมูลที่ได้จากการใช้เครื่องมือที่มีกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ไปพัฒนาโมเดลปัญญาประดิษฐ์สำหรับการจำแนกการกระทำการของมนุษย์ภายในสำนักงาน

ศึกษาค้นคว้าเอกสารและงานวิจัยที่เกี่ยวข้อง

1. ศึกษาเกี่ยวกับการประมวลผลวิดีโอ
2. ศึกษาเกี่ยวกับชุดข้อมูลสำหรับการประมวลผลวิดีโอ
3. ศึกษาเกี่ยวกับโมเดลปัญญาประดิษฐ์ที่ใช้ในการประมวลผลวิดีโอ
4. ศึกษาเครื่องมือที่ใช้ในการช่วยสร้างชุดข้อมูลจากวิดีโอ

เครื่องมือที่มีกำกับคุณลักษณะด้วยปัญญาประดิษฐ์

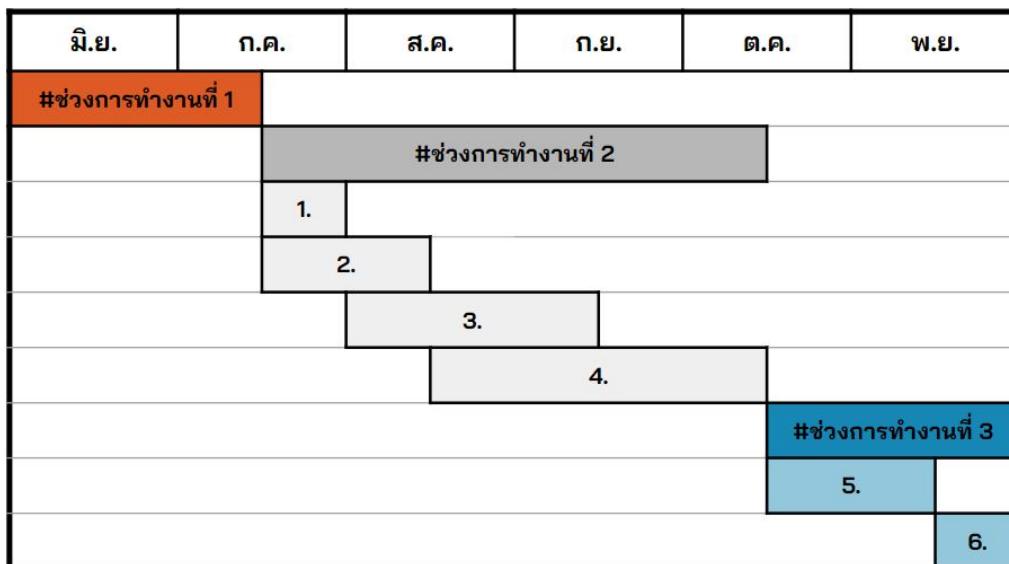
1. ออกแบบและสร้างหน้าต่างของเครื่องมือที่มีกำกับคุณลักษณะด้วยปัญญาประดิษฐ์
2. ออกแบบและสร้างระบบของเครื่องมือที่มีกำกับคุณลักษณะด้วยปัญญาประดิษฐ์
3. ทดสอบและปรับปรุงการทำงานของเครื่องมือที่มีกำกับคุณลักษณะด้วยปัญญาประดิษฐ์

โมเดลปัญญาประดิษฐ์สำหรับการจำแนกการกระทำการของมนุษย์ภายในสำนักงาน

1. สร้างชุดข้อมูลสำหรับสร้างโมเดลปัญญาประดิษฐ์จากเครื่องมือที่มีกำกับคุณลักษณะด้วยปัญญาประดิษฐ์
2. สร้างโมเดลปัญญาประดิษฐ์สำหรับการจำแนกการกระทำการของมนุษย์ภายในสำนักงาน
3. ทดสอบโมเดลปัญญาประดิษฐ์สำหรับการจำแนกการกระทำการของมนุษย์ภายในสำนักงาน

แผนการดำเนินงาน

#2562



รูปที่ 1.1: แผนการดำเนินงาน

รายละเอียดแผนดำเนินงาน

1. ช่วงการทำงานที่ 1 : ศึกษาและรวบรวมข้อมูล
 - (a) 3 มิถุนายน - 15 กรกฎาคม 2562
ศึกษาข้อมูลเกี่ยวกับการวิเคราะห์วิดีโอด้วยปัญญาประดิษฐ์และการสร้างชุดข้อมูลสำหรับจำแนกการกระทำของมนุษย์
2. ช่วงการทำงานที่ 2 : ออกแบบและพัฒนาเครื่องมือสำหรับทำกับคุณลักษณะด้วยปัญญาประดิษฐ์
 - (a) 16 กรกฎาคม - 1 สิงหาคม 2562
ออกแบบและสร้างหน้าต่างการทำงานของเครื่องมือทำกับคุณลักษณะด้วยปัญญาประดิษฐ์
 - (b) 16 กรกฎาคม - 15 สิงหาคม 2562
ออกแบบและสร้างระบบการทำงานของเครื่องมือทำกับคุณลักษณะด้วยปัญญาประดิษฐ์
 - (c) 1 สิงหาคม - 15 กันยายน 2562
รวมระบบเข้ากับหน้าต่างการทำงานของเครื่องมือทำกับคุณลักษณะด้วยปัญญาประดิษฐ์
 - (d) 16 สิงหาคม - 15 ตุลาคม 2562
ปรับปรุงระบบและหน้าต่างการทำงานของเครื่องมือทำกับคุณลักษณะด้วยปัญญาประดิษฐ์
3. ช่วงการทำงานที่ 3 : ศึกษาและรวบรวมข้อมูล
 - (a) 16 ตุลาคม - 11 พฤศจิกายน 2562
ออกแบบและสร้างระบบการทำงานของระบบวิเคราะห์การกระทำการของมนุษย์ภายในสำนักงาน
 - (b) 21 ตุลาคม - 22 พฤศจิกายน 2562
ปรับปรุงระบบการทำงานการวิเคราะห์การกระทำการของมนุษย์ภายในสำนักงาน

บทที่ 2

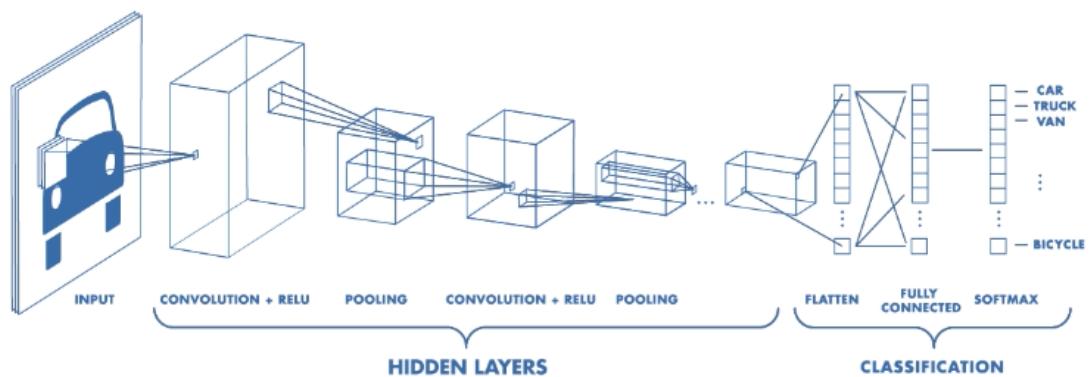
ทฤษฎี/การวิจัยที่เกี่ยวข้อง

การประมวลผลวิดีโอในปัจจุบันนั้นมีวิธีการและเทคนิคมาหลาย ผู้วิจัยจึงต้องศึกษาองค์ความรู้และงานวิจัยที่เกี่ยวข้องกับวัตถุประสงค์ของงาน เพื่อศึกษาและใช้เป็นแนวทางในการประยุกต์สำหรับสร้างเครื่องมือกำกับข้อมูลด้วยปัญญาประดิษฐ์ และโมเดลปัญญาประดิษฐ์สำหรับการจำแนกการกระทำของมนุษย์ ซึ่งหัวข้อที่ผู้วิจัยได้ไปศึกษามา มีดังต่อไปนี้

1. ทฤษฎีที่เกี่ยวข้อง
 - (a) Convolutional neural network
 - (b) Optical flow
 - (c) โมเดลปัญญาประดิษฐ์สำหรับตรวจจับวัตถุที่เกี่ยวข้องกับงานวิจัย
2. การประมวลผลวิดีโอ
 - (a) การตรวจจับวัตถุ (object detection)
 - (b) การติดตามตำแหน่งของวัตถุ (object tracker)
 - (c) การระบุตัวตนของบุคคล (person re-identification)
 - (d) การจำแนกการกระทำของมนุษย์ (human action classification)
3. โมเดลปัญญาประดิษฐ์สำหรับจำแนกการกระทำมนุษย์ (human action classification model)
4. เครื่องมือกำกับชุดข้อมูลภาพ (labeling tool)

2.1 ທາງໝາຍທີ່ເກື່ອງຈົບ

2.1.1 Convolutional neural network



รูปที่ 2.1: ตัวอย่างโครงสร้างของ CNN ที่ใช้ในการจำแนกหมวดหมู่ของวัตถุ

ตัวกรอง/เคอร์เนล (Filter/Kernel)

ตัวกรองหรือเครื่องนอล คือขั้นที่ใช้ในการสกัดคุณลักษณะของรูปภาพออกมาร้ายสีเหลี่ยมเล็กๆนิด NXN โดยที่ $N \in [1, 2, 3, \dots]$ ดังรูปที่ 2.2ก และสมมติให้ภาพที่ใช้ในการประมวลผลเป็นดังรูปที่ 2.2ข

1	0	1
0	1	0
1	0	1

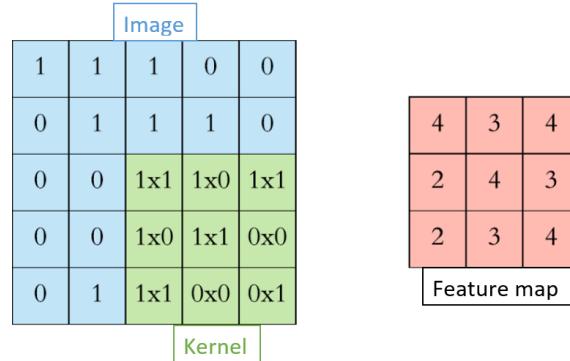
(ก) ตัวอย่างเคอร์เนลขนาด 3×3

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

(ก) ตัวอย่างภาพที่ใช้ในการประเมินผล

รูปที่ 2.2: ตัวอย่างเครื่องเนล และภาพที่ใช้ในการประเมินผล

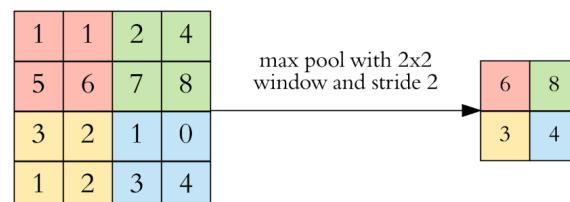
เมื่อนำเครื่องเนล (รูปที่ 2.2ก) ไปทาบกับภาพ (รูปที่ 2.2ข) แล้วคุณค่าในเครื่องเนลกับพิกเซล (pixel) ที่ทาบจะได้คุณลักษณะของช่องนั้นจากนั้นเลื่อนต่อไปจนครบทั้งรูป ซึ่งการเลื่อน (stride) นั้นขึ้นอยู่กับผู้สร้างว่าต้องการจะให้เลื่อนครั้งละกี่ช่อง และระยะการเลื่อนที่มากขึ้นจะทำให้ความสมมัติของคุณลักษณะที่ได้ออกมาแน่นอยู่ลง โดยการวางแผนเครื่องเนลเทียบบนภาพนั้นจะวางไม่ให้เกินกรอบรูป แต่ถ้าต้องการทابกับทุกพิกเซลในภาพสามารถทำได้ด้วยการให้พื้นที่ที่เกินขอบภาพไปเพ่ากับ 0 เทคนิคนี้เรียกว่า padding และคุณลักษณะที่ได้ออกมาทั้งหมดจะเรียกว่าผังคุณลักษณะ (features map) ตามรูปที่ 2.3



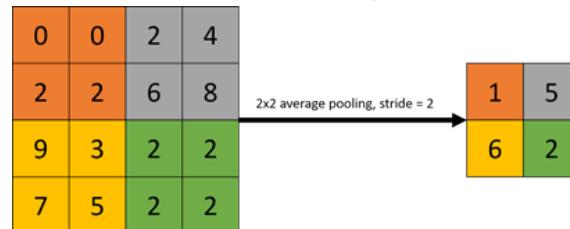
รูปที่ 2.3: ตัวอย่างการทำผังคุณลักษณะ

Pooling

Pooling คือขั้นที่สามารถลดขนาดของภาพลงเพื่อลดข้อมูลที่ไม่จำเป็นลง ซึ่งมีหลายประเภทแต่นิยมใช้มีสองประเภทได้แก่ max pooling และ average pooling โดยที่ max pooling จะใช้ในการหาค่าที่มากที่สุดในเครื่องเนลที่ทาบทอยู่ดังรูปที่ 2.4ก ในขณะที่ average pooling จะหาค่าเฉลี่ยของภายในเครื่องเนลออกม�다ังรูปที่ 2.4ข



(ก) ตัวอย่างการทำ max pooling



(ข) ตัวอย่างการทำ average pooling

รูปที่ 2.4: ตัวอย่างการใช้ max pooling และ average pooling กับภาพ

Activation function

ในแต่ละชั้นของ CNN นั้นจะมี activation function เป็นสิ่งที่กำหนดว่าผลลัพธ์ของชั้นนั้นว่าจะอยู่ในช่วงไหน เช่น -1 ถึง 1, 0 ถึง 1 ขึ้นอยู่กับฟังก์ชันที่ใช้ ซึ่งจะแบ่งออกเป็นสองแบบคือ แบบเป็นเส้นตรงและแบบไม่เป็นเส้นตรง

แบบเป็นเส้นตรงจะมีสมการของฟังก์ชันดังนี้ $f(x) = x$ โดยที่ x คือข้อมูลป้อนเข้าของฟังก์ชัน ทำให้ผลลัพธ์ที่ได้จากฟังก์ชันนี้มีค่าอยู่ในช่วง $-\inf$ ถึง \inf ส่วนในแบบไม่เป็นเส้นตรงจะมีหลายฟังก์ชัน แต่ฟังก์ชันที่เป็นที่นิยมใช้คือ Rectified Linear Unit หรือ ReLU, Leaky ReLU และ Parametric ReLU โดยทั้งสามฟังก์ชันมีสมการดังนี้

ReLU

$$f(x) = \begin{cases} 0 & \text{if } x < 0, \\ x & \text{otherwise} \end{cases}$$

Leaky ReLU

$$f(x) = \begin{cases} 0.01x & \text{if } x < 0, \\ x & \text{otherwise} \end{cases}$$

Parametric ReLU โดยที่ $\alpha \in \mathbb{R}^+$

$$f(x) = \begin{cases} \alpha x & \text{if } x < 0, \\ x & \text{otherwise} \end{cases}$$

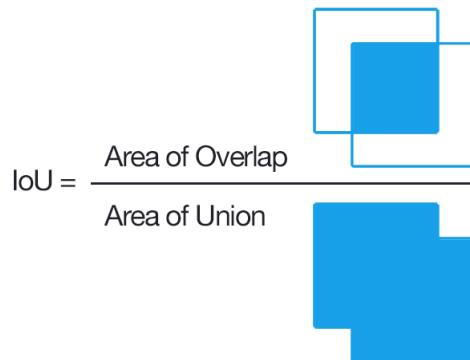
Fully connected layer

เป็นชั้นที่จะรวมคุณลักษณะทั้งหมดของชั้นก่อนหน้าให้เป็นเวกเตอร์ ก่อนที่จะนำเวกเตอร์ที่ได้ไปคำนวณด้วย activation function เพื่อหาคำตอบสำหรับการแยกหมวดหมู่ของภาพ ซึ่งฟังก์ชันที่นิยมใช้จะมีสองฟังก์ชันคือ softmax และ sigmoid (logistic) โดยทั้งสองมีความแตกต่างกันดังตารางที่ 2.1

	Softmax	Sigmoid
1	สมการคือ $f(x_i) = \frac{\exp(x_i)}{\sum_{j=0}^k \exp(x_j)}$ โดยที่ k คือจำนวนข้อมูล และ $i \in 0, 1, 2, \dots, k$	$f(x_i) = \frac{1}{1+\exp(-x_i)}$ โดยที่ $i \in 0, 1, 2, \dots, k$
2	ผลรวมของความน่าจะเป็นจะเท่ากับ 1 เสมอ	ผลรวมของความน่าจะเป็นไม่จำเป็นต้องเท่ากับ 1
3	มักใช้ในการจำแนกหมวดหมู่มากกว่าสองหมวดหมู่ขึ้นไป	มักใช้ในการจำแนกหมวดหมู่เพียงสองหมวดหมู่

ตารางที่ 2.1: ตารางแสดงการเปรียบเทียบระหว่าง softmax function และ sigmoid function

Intersection over union (IoU)



รูปที่ 2.5: ภาพแสดงการหา IoU ด้วยการใช้คำตอบจริงของการทำงานและกรอบสี่เหลี่ยมที่ได้จากการทำงาน

เป็นวิธีในการทดสอบประสิทธิภาพของการตรวจจับวัตถุ โดยค่า IoU นั้นสามารถหาได้จากการนำกรอบสี่เหลี่ยมจริงของวัตถุและกรอบสี่เหลี่ยมที่ได้จากการทำงาน มาหารอัตราส่วนระหว่างพื้นที่ที่กรอบสี่เหลี่ยมทั้งสองทับซ้อนกัน และหารด้วยพื้นที่ทั้งหมดของกรอบสี่เหลี่ยมทั้งสองรวมกัน ซึ่งสามารถเขียนในรูปสมการได้ดังนี้

$$IoU(P, G) = \frac{|P \cap G|}{|P \cup G|} \quad (2.1)$$

โดยที่

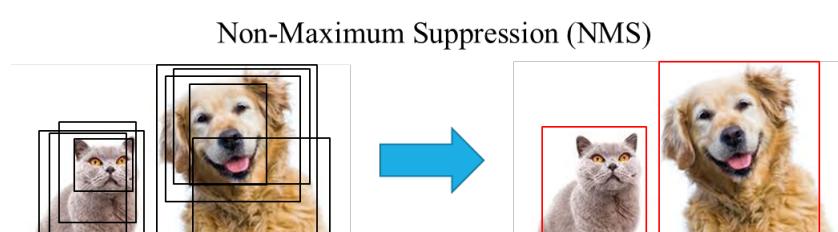
IoU = ค่าที่ใช้สำหรับวัดผลความใกล้เคียงระหว่างสองกรอบสี่เหลี่ยม

P = พื้นที่ของกรอบสี่เหลี่ยมที่ทำงานได้

G = พื้นที่ของกรอบสี่เหลี่ยมจริงของรูปภาพ

Non Maximum Suppression (NMS)

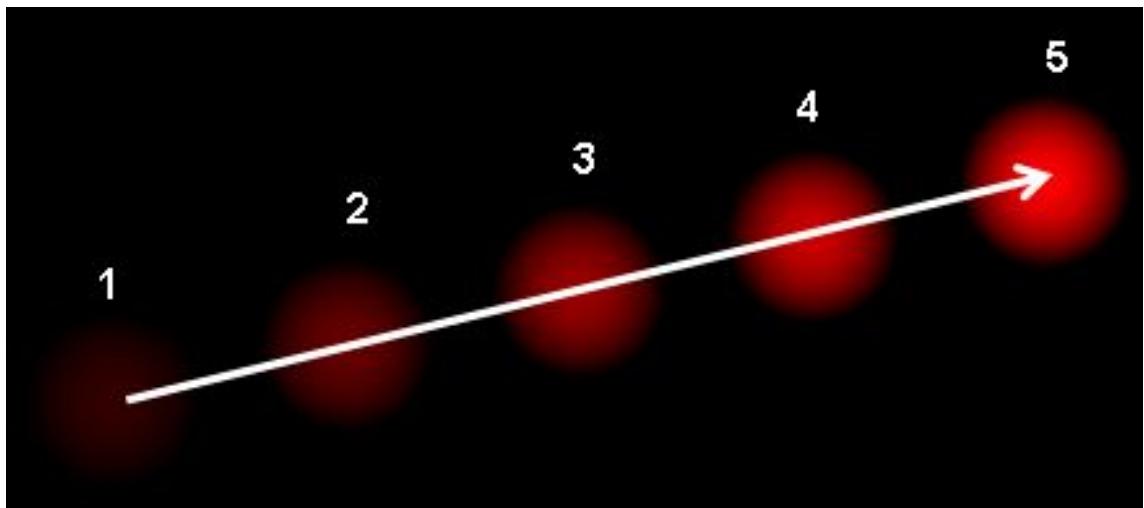
Non Maximum Suppression คือ อัลกอริทึมที่นิยมใช้เข้ามาช่วยจัดการปัญหากรอบสี่เหลี่ยมที่ซ้อนทับกันซึ่งเกิดจากการทำงานซ้ำ เพื่อให้ได้กรอบสี่เหลี่ยมที่บ่งบอกถึงตำแหน่งของวัตถุนั้นเพียงกรอบเดียว



รูปที่ 2.6: ตัวอย่างการทำงานของ NMS

2.1.2 Optical flow

Optical flow^[9] คือการแปลงการเคลื่อนที่ของวัตถุในระหว่างสองภาพซึ่งอาจจะเกิดจากการเคลื่อนที่ของวัตถุหรือตัวกล้องอุปกรณ์ในรูปแบบของเวกเตอร์ 2 มิติ โดยที่เวกเตอร์แต่ละตัวจะแสดงถึงทิศทางการเคลื่อนที่ของวัตถุหรือบุคคลระหว่างภาพตั้งรูปที่ 2.7



รูปที่ 2.7: ตัวอย่าง optical flow ของการเคลื่อนที่ของลูกบอล

จากรูปที่ 2.7 แสดงให้เห็นถึงการเคลื่อนที่ของลูกบอลในภาพที่ต่อเนื่องกัน 5 ภาพ โดยที่ลูกศรจะแสดงถึงทิศทางการเคลื่อนที่ของเวกเตอร์

การทำงานของ optical flow อุปบนสมมติฐาน 2 ประการได้แก่

1. ความเข้มพิกเซลของวัตถุจะไม่เปลี่ยนแปลงระหว่างภาพที่ต่อเนื่องกัน
2. พิกเซลที่อยู่ใกล้กันจะมีลักษณะการเคลื่อนไหวที่คล้ายกัน

เมื่อพิจารณาพิกเซล $I(x, y, t)$ จากภาพแรกจะเคลื่อนไหวเป็นระยะทาง (dx, dy) ไปยังภาพต่อไปหลังจากเวลาผ่านไปแล้ว dt ดังนั้นเนื่องจากพิกเซลเหล่านี้เหมือนกัน และความเข้มไม่มีการเปลี่ยนแปลง จึงทำให้พูดได้ว่า

$$I(x, y, t) = I(x + dx, y + dy, t + dt) \quad (2.2)$$

โดยที่

I = พิกเซลจากภายในภาพ

x = ตำแหน่งของพิกเซลในแกน x

dx = ระยะทางที่เคลื่อนที่ในแกน x

y = ตำแหน่งของพิกเซลในแกน y

dy = ระยะทางที่เคลื่อนที่ในแกน y

t = เวลา

dt = ระยะเวลาที่เปลี่ยนไประหว่างภาพ

จากนั้นใช้การประมาณค่าของ taylor series ทางฝั่งขวามือ และลบค่า common term แล้วหารด้วย dt เพื่อให้ได้สมการดังต่อไปนี้

$$f_x u + f_y v + f_t \quad (2.3)$$

$$f_x = \frac{\delta f}{\delta x}; f_y = \frac{\delta f}{\delta y} \quad (2.4)$$

$$u = \frac{\delta x}{\delta t}; v = \frac{\delta y}{\delta t} \quad (2.5)$$

โดยที่

f_x = เกรเดียน (gradient) ในแกน x

f_y = เกรเดียนในแกน y

f_t = เกรเดียนของเวลา

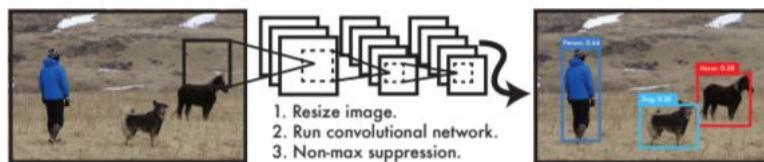
u = เวกเตอร์การเคลื่อนที่ของแกน x

v = เวกเตอร์การเคลื่อนที่ของแกน y

สมการข้างบนเรียกว่าสมการ optical flow จากสมการทำให้สามารถหา f_x และ f_y โดยเป็นเกรเดียนของภาพในแกน x และแกน y ตามลำดับ และ f_t เป็นเกรเดียนของเวลา แต่ n กับ v เป็นตัวแปรที่ไม่ทราบ ทำให้สมการนี้ไม่สามารถแก้ไขโดยมีตัวแปรที่ไม่ทราบถึง 2 ตัว จึงมีการนำวิธีการต่างๆเข้ามาใช้ในการแก้ปัญหานี้ โดยวิธีการที่นำเข้ามาใช้ในการแก้ปัญหานี้คือ dense optical flow ที่เป็นอัลกอริทึมของ Gunnar Farneback^[18] ซึ่ง optical flow เป็นหนึ่งในวิธีการสกัดข้อมูลการเคลื่อนไหวของวิดีโออุปกรณ์เพื่อใช้เป็นข้อมูลในการสร้างโมเดลปัญญาประดิษฐ์สำหรับจำแนกการกระทำหรือเหตุการณ์ในวิดีโอ

2.1.3 โมเดลปัญญาประดิษฐ์สำหรับตรวจจับวัตถุที่เกี่ยวข้อง

2.1.3.1 YOLO



รูปที่ 2.8: กระบวนการทำงานของโครงสร้างโมเดลปัญญาประดิษฐ์ของ YOLO

โครงสร้างโมเดลปัญญาประดิษฐ์ของ YOLO^[28] เป็นโครงสร้างที่มีความเร็วในการประมวลผลถึง 45 เฟรมต่อวินาที ทำให้สามารถประมวลผลแบบเรียลไทม์ได้ นอกจากนั้นยังมีความแม่นยำ mAP มากกว่าโมเดลสำหรับตรวจจับวัตถุอื่นๆถึง 2 เท่า ซึ่งเหตุผลที่โครงสร้างโมเดลปัญญาประดิษฐ์ของ YOLO เร็วกว่าโมเดลปัญญาประดิษฐ์ตัวอื่นๆ เนื่องจาก การตรวจจับวัตถุในวิธีการก่อนหน้าจะใช้วิธีท่านายกรอบสี่เหลี่ยมก่อน แล้วจึงค่อยนำกรอบสี่เหลี่ยมไปท่านายว่าเป็นหมวดหมู่อะไร ซึ่ง YOLO มีวิธีการที่ต่างออกไป คือ ท่านายตำแหน่งของกรอบสี่เหลี่ยม และท่านายว่าเป็นหมวดหมู่อะไรพร้อมกัน โดยใช้โครงข่ายประสาทแบบคอนโวลูชัน ด้วยแนวคิดนี้จึงเป็นที่มาของชื่อ YOLO หรือ you only look once

โครงสร้างของโมเดลปัญญาประดิษฐ์ของ YOLO

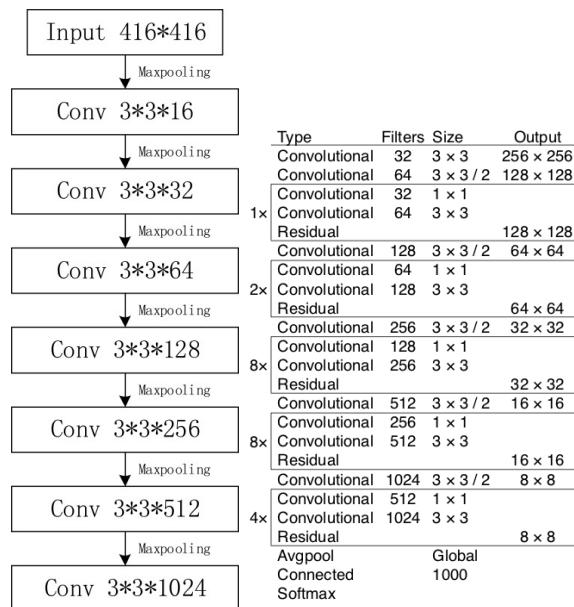


รูปที่ 2.9: โครงสร้างทั่วไปของโมเดลปัญญาประดิษฐ์ของ YOLO

จากรูปภาพที่ 2.9 จะเห็นได้ว่า YOLO ใช้โครงข่ายประสาทเที่ยมเพียงตัวเดียวซึ่งภายในโครงข่าย จะมีกระบวนการหลักๆอยู่สามอย่าง กระบวนการแรกคือการสกัดคุณลักษณะ กระบวนการนี้จะมีจำนวนชั้นของเลเยอร์ที่แตกต่างกันไปตามความลึกของการสกัดแล้วแต่โมเดล ซึ่งตัวอย่างจะเป็นดังรูปที่ 2.10 และชั้นตอนถัดมาคือการทำนายผล หลังจากที่ได้คุณลักษณะมาแล้วจะนำไปท่านายผลผ่านชั้น fully connected ซึ่งจะได้ผลลัพธ์เป็นหมวดหมู่และตำแหน่งของกรอบสี่เหลี่ยม และชั้นตอนสุดท้ายคือการทำ NMS เพื่อให้ได้ผลลัพธ์ที่ดีที่สุดออกมานะ

ซึ่งโครงสร้างโมเดลปัญญาประดิษฐ์ของ YOLO ที่ถูกใช้ในงานวิจัยนี้ประกอบไปด้วย YOLO-v3 tiny, YOLO-v3 และ YOLO-v3 spp ซึ่งทั้งสามโครงสร้างจะแตกต่างดังนี้

1. YOLO-v3 tiny ใช้ชั้น max pooling ในขั้นตอนของการลดจำนวนข้อมูลตัวอย่าง
2. YOLO-v3 ใช้ชั้น convolutional ในขั้นตอนของการลดจำนวนข้อมูลตัวอย่าง
3. YOLO-v3 spp ใช้ชั้น convolutional และคุณลักษณะที่ได้รับจาก max pooling ในขั้นตอนของการลดจำนวนข้อมูลตัวอย่าง



(ก) โครงสร้างโมเดล

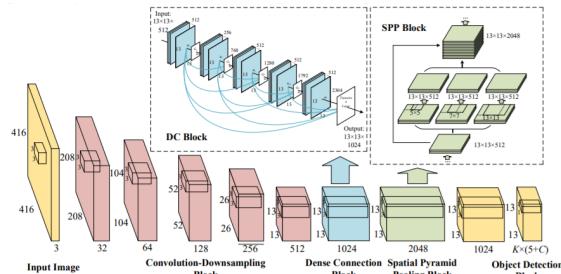
ปัญญาประดิษฐ์ของ

YOLO-v3 tiny

Table 1. Darknet-53.

(ข) โครงสร้างโมเดลปัญญาประดิษฐ์

ของ YOLO-v3

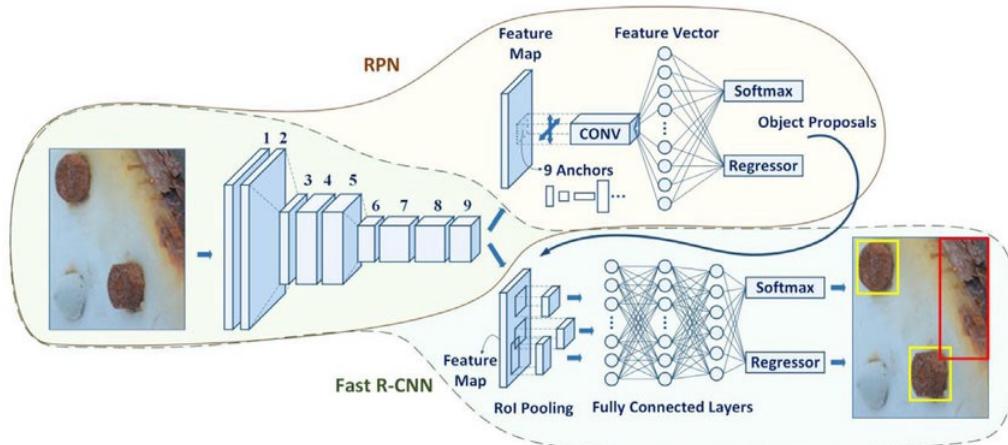


11

(ค) โครงสร้างโมเดลปัญญาประดิษฐ์ของ YOLO-v3 spp

รูปที่ 2.10: โครงสร้างโมเดลปัญญาประดิษฐ์ของ YOLO

Faster-RCNN



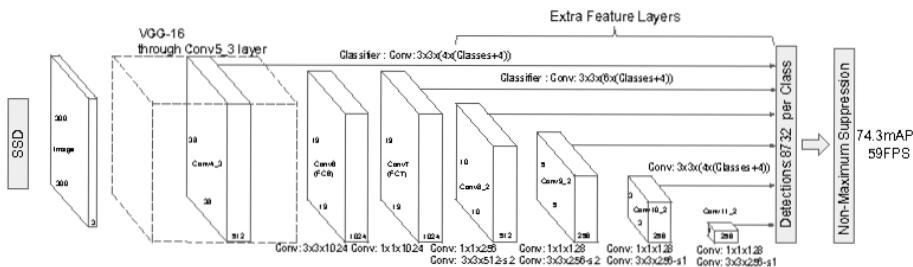
รูปที่ 2.11: โครงสร้างทั่วไปของโมเดลปัญญาประดิษฐ์ของ Faster RCNN

faster-rcnn^[30] มีการพัฒนาในการหาพื้นที่ที่สนใจ (region of interest หรือ ROI) โดยเปลี่ยนจากใช้โครงข่ายหาพื้นที่ที่สนใจแยกเฉพาะ (selective search) นำมารวมในโครงข่ายเดียวกัน ดังนั้น faster-rcnn จึงมีโครงข่ายประสาทเทียมเดียวในการทำงาน ซึ่งภายใต้โครงข่ายจะประกอบไปด้วยการทำงานหลักสามอย่าง คือ

1. การสกัดคุณลักษณะ
นำภาพเข้าชั้น convolution เพื่อการสกัดคุณลักษณะของภาพ
2. การหาพื้นที่ที่คาดว่าจะมีวัตถุอยู่
หลังจากที่ภาพผ่านการสกัดคุณลักษณะแล้ว จะถูกนำเข้าไปใน region proposal network เพื่อหาพื้นที่ที่คาดว่าจะมีวัตถุอยู่
3. การทำนายผล
ทำการ pooling คุณลักษณะของภาพและพื้นที่ที่คาดว่าจะมีวัตถุอยู่ และนำเข้าไปในชั้น full connected จะได้ผลลัพธ์เป็นหมวดหมู่ของวัตถุและตำแหน่งของกรอบสีเหลี่ยม

region proposal network (RPN)^[30] คือ โครงข่ายที่หาพื้นที่ที่คาดว่าจะมีวัตถุอยู่จะถูกใช้หลังภาพผ่านการสกัดคุณลักษณะ ซึ่ง RPN มีโครงสร้างที่มีองค์ประกอบ 2 อย่าง คือมีการบอกว่าบริเวณนั้นมีวัตถุอยู่หรือไม่ (classification layer) และสำหรับการระบุพิกัดของกรอบสีเหลี่ยมที่คาดว่าจะมีวัตถุอยู่ (regression layer) ซึ่งผลลัพธ์จะได้เป็น ROI ของวัตถุในภาพ

SSD



รูปที่ 2.12: โครงสร้างทั่วไปของโมเดลปัญญาประดิษฐ์ SSD

SSD^[30] เป็นโมเดลปัญญาประดิษฐ์ที่ใช้โครงข่ายประสาทเทียมตัวเดียวสำหรับการตรวจจับวัตถุ ซึ่งภายใต้โครงข่ายจะประกอบไปด้วยการทำงานหลัก 3 อย่าง คือ

1. การสกัดคุณลักษณะ
นำภาพผ่าน VGG-16^[11] (โมเดล CNN ชนิดหนึ่ง) เพื่อการสกัดคุณลักษณะของภาพออกมานำไปใช้ในการคำนวณ
 2. การทำนายผล
หลังจากที่ได้คุณลักษณะมาแล้วจะนำไปทำนายผลในชั้น fully connected
 3. การเลือกคัดกรองผลลัพธ์
หลังจากได้ผลลัพธ์เป็นประเภทของวัตถุ และตำแหน่งของกรอบสีเหลี่ยมจะนำไปผ่านกระบวนการ NMS เพื่อให้ได้ผลลัพธ์ที่ดีที่สุด

2.2 การประมวลผลวิดีโอ

ในส่วนของงานวิจัยนี้สิ่งที่สนใจ คือ ข้อมูลการกระทำของมนุษย์แต่ละคนภายในวิดีโอ เพื่อที่จะได้ผลลัพธ์ที่มีประสิทธิภาพออกมาเป็นข้อมูลของสิ่งที่สนใจ เช่น จำนวนคนที่เดินผ่านกล้อง หรือทิศทางการเดินของคนในวิดีโอ จึงจำเป็นต้องใช้การประมวลผลวิดีโอเพื่อที่จะสกัดสิ่งที่สนใจออกมาจากวิดีโอ ซึ่งการประมวลผลวิดีโอมีหลากหลายกระบวนการ โดยในแต่ละกระบวนการจะมีจุดประสงค์ของการทำและผลลัพธ์หลังการประมวลผลที่แตกต่างกัน ในทว่าข้อนี้จะมาอธิบายถึงกระบวนการในการประมวลผลของวิดีโอและผลลัพธ์ของกระบวนการนั้น

2.2.1 การตรวจจับวัตถุ

การตรวจจับวัตถุนั้นเป็นหนึ่งในกระบวนการประมวลผลวิดีโอ ก่อให้เกิดกระบวนการที่ผู้วิจัยจะต้องทำการระบุสิ่งที่สนใจว่า คืออะไร อยู่ที่ตำแหน่งใด การตรวจจับวัตถุก็ค้นพบเมื่อナンานมาแล้ว และในปัจจุบันนี้สามารถทำได้หลากหลายวิธี โดยภายในบทความนี้จะสรุปใจความสำคัญของวิธีการต่างในการตรวจจับวัตถุ เช่น Sliding window, Brute force search, RCNN, Fast-RCNN, Faster-RCNN, YOLO, SSD

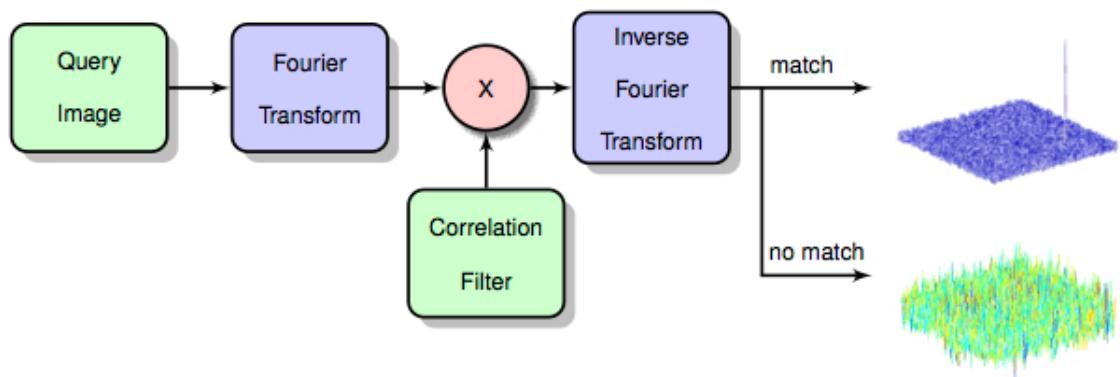
1. Sliding window^[6] วิธีการที่เปรียบเสมือนมีเครื่องเนลค่ออยู่เลื่อนไปยังแต่ละพิกเซลบนรูป ซึ่งก่อนการเลื่อนของเครื่องเนลแต่ละครั้งจะนำส่วนของภาพที่ถูกเครื่องเนลทับอยู่ไปทำนายว่าใช้วัตถุที่เราต้องการหรือไม่ จากนั้นจึงค่อยเลื่อนถัดไปจนครบทั้งภาพ
 2. Brute force search^[6] ถูกสร้างขึ้นมาเพื่อแก้ปัญหาขนาดของเครื่องเนลไม่ตรงกับขนาดของวัตถุที่อยู่ในภาพทำให้มีโอกาสที่จะไม่พบวัตถุ โดยหลักการของวิธีการนี้คือ การย่อ-ขยายภาพและนำเข้าในหลายๆ อัตราส่วน ตั้งแต่ 0.1 เท่า จนถึง 2 เท่า แต่ข้อเสียของวิธีการนี้คือ มีการคำนวนพื้นที่ซ้ำๆทำให้ใช้เวลานาน

3. RCNN^[6] ใช้อัลกอริทึม selective search เข้ามาช่วยในการเสนอพื้นที่ที่น่าจะมีวัตถุอยู่ทัดแทนการค้นหาทุกๆ ตำแหน่ง จำนวนนักนำภาพในส่วนพื้นที่นั้นไปทำนายว่า วัตถุนั้นคืออะไร กรณีที่มีพื้นที่ที่อยู่ใกล้ๆ วัตถุ ถูกเสนอเข้ามาเป็นจำนวนมากด้วย เราจะใช้ NMS ใน การเลือกพื้นที่ที่ถูกทับซ้อนมากที่สุดในบริเวณนั้น
4. Fast-RCNN^[7] จากวิธีการ RCNN แต่ละพื้นที่จะถูกนำไปสกัดคุณลักษณะ และทำนายผลที่ละพื้นที่ทำให้เสียเวลา โดย Faster-RCNN จะมีส่วนที่คล้ายกับ RCNN ในส่วนการทำ selective search เพื่อหาพื้นที่ที่น่าจะมีวัตถุเหมือนเดิม แต่ Faster-RCNN จะนำภาพไปสกัดคุณลักษณะ หลังจากที่ได้คุณลักษณะแล้วจะนำพิกัดของพื้นที่ที่น่าจะมีวัตถุ บนภาพที่ถูกสกัดคุณลักษณะแล้วไปผ่านชั้น ROI Pooling (การลดขนาดข้อมูลให้มีขนาดคงที่เพื่อเป็นข้อมูลป้อนเข้าให้กับโมเดลในการทำนายผล)
5. Faster-RCNN พัฒนาจาก Fast-RCNN โดยวิธีของ Faster-RCNN จะรวมในส่วนของ selective search และการทำงานอื่นๆ ให้อยู่ในโครงข่ายเดียวกัน สรุปคือการทำงานของโครงข่ายของ Faster-RCNN จะมีการทำงานสามอย่างคือ 1) สกัดคุณลักษณะ 2) การเสนอส่วนที่น่าจะมีวัตถุอยู่ในภาพ 3) หลังจากได้ภาพจากการสกัดคุณลักษณะ นำพิกัดของพื้นที่ที่น่าจะมีวัตถุ บนภาพที่ถูกสกัดคุณลักษณะแล้วไปผ่านชั้น ROI Pooling
6. YOLO เป็นวิธีการที่ใช้ CNN เพียงตัวเดียวทำงานภาพ โดยโครงข่ายจะแบ่งภาพออกเป็นพื้นที่ และใช้ fully connected ทำนายตำแหน่งของกรอบสี่เหลี่ยม พร้อมทั้งหมวดหมู่ของวัตถุไปพร้อมกัน
7. SSD ใช้โครงข่ายประสาทเทียมเหมือนกับ YOLO แต่การออกแบบโครงสร้างแตกต่างกัน โดยที่ SSD จะใช้ VGG-16 ในการสกัดคุณลักษณะ และใช้ Convolution layer ต่อ กันหลายชั้นเพื่อลดมิติและความละเอียดทำให้ตรวจจับวัตถุในหลายขนาด ซึ่งในแต่ละชั้นจะได้ผลลัพธ์เป็น Convolution filter จากนั้นนำ Convolution filter ไปทำนายผลต่อ

2.2.2 ระบบติดตามการเคลื่อนไหวของวัตถุ

การติดตามการเคลื่อนไหวของวัตถุ^[17] คือระบบที่ใช้สำหรับการติดตามการเคลื่อนไหวของวัตถุที่สนใจที่อยู่ในรูปภาพ โดยใช้การคำนวณทางคณิตศาสตร์ และการประมวลผลภาพ (image processing) ทำให้การประมวลผลนั้นเร็กว่าการใช้โมเดลปัญญาประดิษฐ์ ซึ่งอัลกอริทึมติดตามการเคลื่อนไหวที่นิยมใช้มีสองอัลกอริทึม คือ correlation filter และ kalman filter ซึ่งหลักการของทั้งสองอัลกอริทึมนั้นจะแตกต่างกันโดยที่ correlation filter นั้นจะใช้พิกเซลของวัตถุในการคำนวณตำแหน่งถัดไปของวัตถุ และ kalman filter จะใช้ข้อมูลการเคลื่อนไหวในการคำนวณตำแหน่งถัดไปของวัตถุ ซึ่งจากการศึกษาในบทความ "Object Tracking using Correlation, Kalman Filter and Fast Means Shift Algorithms"^[13] kalman filter มีประสิทธิภาพที่สูงนั้นจะขึ้นอยู่กับข้อมูลที่ได้จากการวัด (measurement) และความซับซ้อนในการเคลื่อนไหวของวัตถุ ในขณะที่ correlation นั้นมีประสิทธิภาพที่ด้อยกว่า kalman filter เพียงเล็กน้อยและสามารถติดตามการเคลื่อนไหวที่ซับซ้อนของวัตถุได้ดีกว่า (การเคลื่อนไหวที่ซับซ้อนหมายถึง การเคลื่อนไหวที่เกิดการเปลี่ยนทิศทางฉับพลันบ่อย) ผู้วิจัยจึงตัดสินใจเลือกใช้ correlation filter ในงานครั้นนี้

จากรูปที่ 2.13 เป็นหลักการในการติดตามการเคลื่อนไหวของวัตถุแบบ correlation filter โดยการนำรูปมาผ่านกระบวนการแปลงฟูรีเยร์ (fourier transform) และนำมาคูณกับ correlation filter ซึ่งเป็นตัวกรองที่ใช้สำหรับการหาความสัมพันธ์กับวัตถุในภาพ จากนั้นทำการแปลงฟูรีเยร์กลับ (inverse fourier transform) เพื่อตรวจสอบว่าวัตถุในภาพนั้นอยู่ที่ตำแหน่งใด โดยมีการคำนวณเริ่มจากการหา correlation filter ที่ดีที่สุดโดยใช้วิธีลดผลรวมของข้อผิดพลาดกำลังสองให้น้อยที่สุดดังนี้



รูปที่ 2.13: แนวคิดของระบบติดตามการเคลื่อนไหวของวัตถุ

$$\epsilon = \left\| \sum_{l=1}^d h^l \star f^l - g \right\|^2 + \lambda \sum_{l=1}^d \|h^l\|^2 \quad (2.6)$$

โดยที่

ϵ = ค่าความคลาดเคลื่อน

d = จำนวนมิติของผังคุณลักษณะ (feature map) ของภาพ

h = correlation filter

\star = circular correlation

f = พื้นที่สีเหลี่ยมของวัตถุที่สนใจที่ได้จากการทำผังคุณลักษณะ

g = ผลลัพธ์ correlation ที่ต้องการของ f

λ = regularization term

เมื่อพิจารณาจากรูปภาพเดียวในกรณีที่เวลา (t) เท่ากับ 1 จะสามารถจัดรูปสมการด้านบนได้ดังนี้

$$H^l = \frac{\bar{G}F^l}{\sum_{k=1}^d \bar{F}^k F^k + \lambda} \quad (2.7)$$

$$H_t^l = \frac{A_t^l}{B_t} \quad (2.8)$$

$$A_t^l = (1 - \eta)A_{t-1}^l + \eta \bar{G}_t F_t^l \quad (2.9)$$

$$B_t = (1 - \eta)B_{t-1} + \eta \sum_{k=1}^d \bar{F}_t^k F_t^k \quad (2.10)$$

โดยที่

H = correlation filter

η = อัตราการเรียนรู้

\bar{G} = g ที่ผ่านการทำ complex conjugation

F = พื้นที่สี่เหลี่ยมของวัตถุที่สนใจที่ได้จากการทำผังคุณลักษณะ

\bar{F} = f ที่ผ่านการทำ complex conjugation

t = เวลา

จากสมการที่ได้มาจะสามารถทำให้หาตำแหน่งต่อไปของวัตถุที่สนใจได้ด้วยสมการต่อไปนี้

$$y = F^{-1} \left\{ \frac{\sum_{l=1}^d \bar{A}^l Z^l}{B + \lambda} \right\} \quad (2.11)$$

โดยที่

y = correlation score

F^{-1} = การแปลงฟูรีเยอร์ผกผันแบบ逆 (inverse discrete fourier transform)

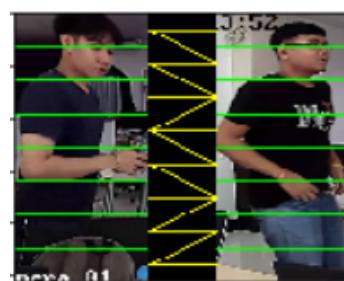
\bar{A}^l = A^l ที่ผ่านการทำ complex conjugation

Z = พื้นที่สี่เหลี่ยมของวัตถุที่สนใจที่ได้จากการหาผังคุณลักษณะของภาพใหม่

โดยค่าของ y ที่ได้ออกมาจะทำให้รู้ถึงตำแหน่งของวัตถุที่สนใจได้ ณ ตำแหน่งที่ y มีค่าสูงสุด

2.2.3 ระบบบุตัวตนของบุคคล

ระบบบุตัวตนของบุคคล^[25] คือการระบุตัวตนของบุคคลภายในวิดีโอหรือระหว่างสองภาพ ซึ่งสามารถนำมาประยุกต์ใช้ในด้านของการรักษาความปลอดภัย หรือการตามหาบุคคล ซึ่งการระบุตัวตนของบุคคลนั้น เป็นปัญหาที่ท้าทาย เนื่องจากคุณลักษณะทั่วไปของบุคคลในภาพไม่เพียงพอต่อการระบุบุคคลภายในภาพว่า เป็นบุคคลคนเดียวกันได้ ซึ่งวิธีการที่ใช้ในการระบุตัวตนของบุคคลเรียกว่า Dynamically Matching Local Information (DMLI) ที่สามารถจัดแนวรายละเอียดข้อมูลของภาพและเพิ่มประสิทธิภาพให้สูงขึ้น ถึงแม้ว่า DMLI นั้นจะไม่ใช่วิธีการที่มีประสิทธิภาพสูงสุดแต่มีประสิทธิภาพใกล้เคียงกับโมเดลอื่นๆ แต่ผู้วิจัยสามารถนำวิธีนี้มาประยุกต์เข้ากับงานวิจัยครั้งนี้ได้สะดวกที่สุด จึงนำวิธีการนี้มาใช้สำหรับงานวิจัยครั้งนี้



รูปที่ 2.14: การแบ่งภาพออกเป็น 8 ส่วนของระบบบุตัวตนของบุคคล

การทำงานของระบบบุตัวตนของบุคคลจะเริ่มจากการแบ่งภาพออกเป็น 8 ส่วนและนำคุณลักษณะของภาพมาผ่านกระบวนการ normalization เพื่อลดความซ้ำซ้อนของข้อมูล และนำมาระบบเพื่อบรรยากาศแตกต่าง

ของคุณลักษณะของภาพ หลังจากนั้นหาค่าเฉลี่ยของความแตกต่างของมา โดยค่าที่ได้ออกมาจะเรียกว่า original distance ถ้าค่าที่ออกมากไปลักษณะของบุคคลในภาพทั้งสองเป็นบุคคลเดียวกัน และใช้การกำหนดเกณฑ์ของ original distance สำหรับระบุตัวตนของคนในภาพว่าเป็นคนเดียวกันหรือไม่

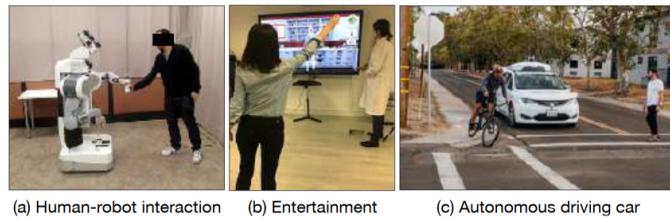
โดยชุดข้อมูลที่นำมาใช้สำหรับการทำโมเดลปัญญาประดิษฐ์ได้แก่

1. Market1501 เป็นชุดข้อมูลที่เก็บข้อมูลภาพของบุคคลโดยใช้กล้องจำนวนหกตัว ถ่ายภาพบุคคลที่ด้านหน้าของซุปเปอร์มาร์เก็ตในมหาวิทยาลัย Tsinghua
2. DukeMTMCReID เป็นชุดข้อมูลที่เก็บข้อมูลภาพของบุคคลโดยใช้กล้องจำนวนแปดตัว ถ่ายภาพบุคคลที่วิทยาเขตของมหาวิทยาลัย Duke ซึ่งมีการเก็บภาพมากถึงสองล้านภาพของนักศึกษาสองพันคน
3. CUHK-03 เป็นชุดข้อมูลที่เก็บภาพของบุคคลที่มหาวิทยาลัยที่ชื่องกง
4. MSMT17 เป็นชุดข้อมูลที่เก็บข้อมูลภาพของบุคคลโดยใช้กล้องจำนวนสิบห้าตัว โดยที่กล้องแต่ละตัวจะไม่ได้ตั้งอยู่สถานที่เดียวกัน และเก็บข้อมูลที่ในวันที่มีสภาพอากาศต่างกัน

โดยทุกชุดข้อมูลจะใช้โครงสร้าง (architecture) ResNet50 ในการสร้างโมเดลปัญญาประดิษฐ์ และทดสอบด้วยวิธี Global+DMLI คือการนำคุณลักษณะทั่วไปของภาพมาจัดแนวรายละเอียดของข้อมูล ดังรูปที่ 2.14 และนำไปเข้าโมเดลปัญญาประดิษฐ์เพื่อคำนวนหาค่า rank1 และ mAP โดยที่ค่า rank1 หมายถึงค่าเบอร์เซ็นต์ความมั่นใจสูงสุดของโมเดลปัญญาประดิษฐ์ที่ทำนายอกรามากถูกต้อง และค่า mAP คือการหาค่าเฉลี่ยความแม่นยำในแต่ละหมวดหมู่ ซึ่งสามารถดูค่า rank1 และ mAP ของโมเดลปัญญาประดิษฐ์สำหรับการทำระบุตัวตนของบุคคลได้ในหัวข้อที่ 4.4.1

2.2.4 ระบบจำแนกการกระทำของมนุษย์

ระบบจำแนกการกระทำเป็นหัวข้อที่มีการให้ความสนใจอย่างมากสำหรับการทำระบบประมวลผลวิดีโอในปัจจุบัน เนื่องจากระบบประมวลผลวิดีโอนั้นสามารถใช้งานได้หลากหลายสถานการณ์ เช่น ใช้สอดส่องการจราจรบนท้องถนน วิเคราะห์พฤติกรรมการเคลื่อนที่ของลูกค้าภายในห้าง การตามหาคนหายหรือพัสดุทางภัยในอาคาร เป็นต้น ซึ่งการที่ระบบสามารถจำแนกการกระทำของมนุษย์ภายในวิดีโอด้วยความสามารถเพิ่มความสามารถของระบบประมวลผลวิดีโอด้วย เช่น สามารถแจ้งเตือนเมื่อพบบุคคลที่มีพฤติกรรมน่าสงสัยในวิดีโอด้วย เพื่อไม่ให้เกิดเหตุการณ์อันตรายขึ้น เป็นต้น



รูปที่ 2.15: ตัวอย่างการประยุกต์ใช้งานระบบจำแนกการกระทำมนุษย์^[22]

การสร้างโมเดลปัญญาประดิษฐ์สำหรับจำแนกการกระทำนั้น จำเป็นต้องมีชุดข้อมูลที่เหมาะสมกับการกระทำที่สนใจ และโครงสร้างโมเดลปัญญาประดิษฐ์ที่เหมาะสม ซึ่งในปัจจุบันนั้นมีชุดข้อมูลสาธารณะที่สามารถนำมาใช้งานได้หลากหลายชุดข้อมูล เช่น YouTube-8M ชุดข้อมูลสำหรับการประมวลผลวิดีโอที่มีขนาดใหญ่ที่สุดและมีจำนวนคำจำกัดมากที่สุด, AVA ชุดข้อมูลของการกระทำที่มาจากการขับเพียงเล็กน้อย, Sports-1M ชุดข้อมูลที่เกี่ยวข้องกับกิจกรรมกีฬาต่างๆของมนุษย์ เป็นต้น ซึ่งชุดข้อมูลที่ผู้วิจัยเลือกนำมาศึกษาได้แก่ YouTube-8M, AVA และ Moment in Time โดยแต่ละชุดข้อมูลจะมีความแตกต่างกันในหลายๆด้านแต่จะมีสิ่งที่เหมือนกัน คือ เป็นชุดข้อมูลสำหรับการวิเคราะห์วิดีโอที่สนใจในการกระทำและกิจกรรมของมนุษย์ โดยจะกล่าวถึงความแตกต่างในด้านต่างๆ เช่น เป้าหมายของแต่ละชุดข้อมูล วิธีการเก็บข้อมูลสำหรับชุดข้อมูล วิธีการสร้างคำจำกัดคุณลักษณะ และรายละเอียดของชุดข้อมูล จากนั้นจะสรุปข้อมูลของแต่ละชุดข้อมูลในหัวข้อที่ 2.3

2.3 ชุดข้อมูลสำหรับการประมวลผลวิดีโอ

ในปัจจุบันมีชุดข้อมูลมากมายถูกสร้างขึ้นมาสำหรับใช้สร้างโมเดลสำหรับแก้ปัญหาในด้านต่างๆ เช่น การตรวจจับวัตถุภายในภาพ การจำจำใบหน้าบุคคลภายในภาพ การจำแนกการกระทำของมนุษย์ เป็นต้น ซึ่งสิ่งที่ทำให้โมเดลปัญญาประดิษฐ์นั้นมีประสิทธิภาพสูงคือ จำนวนของข้อมูล โดยในปัจจุบันปัญหาด้านการทำความเข้าใจภาพด้วยปัญญาประดิษฐ์ (image understanding) สามารถพัฒนาให้มีประสิทธิภาพสูงนั้นเนื่องจากมีจำนวนข้อมูลที่มากมาก ในขณะที่ปัญหาด้านการทำความเข้าใจวิดีโอด้วยปัญญาประดิษฐ์ (video understanding) นั้นกำลังมีการให้ความสนใจเพิ่มขึ้นเรื่อยๆ ในช่วงระยะเวลาหลายปีที่ผ่านมา ในหัวข้อนี้จะพูดถึงการศึกษาชุดข้อมูลที่ใช้สำหรับการทำความเข้าใจวิดีโอ โดยจะมุ่งเน้นไปที่การจำแนกการกระทำการของมนุษย์เป็นหลัก

ชุดข้อมูล YouTube-8M

YouTube-8M คือชุดข้อมูลวิดีโอที่มีจำนวนวิดีโอยอดเยี่ยมที่สุดถึง 8 ล้านวิดีโอ (พ.ศ. 2559) โดยมีจุดมุ่งหมายหลักในการจำแนกสาระสำคัญของวิดีโอ (video theme) ด้วยคำสั้นๆ เช่น ถ้าวิดีโอนั้นมีมนุษย์กำลังปั่นจักรยานบนถนนดินริมหน้าผาชุดข้อมูลนี้จะกำกับวิดีโอนี้ว่า mountain biking ซึ่งทำให้ YouTube-8M แตกต่างจากชุดข้อมูลวิดีโອื่นๆ ส่วนใหญ่ที่จะเน้นการกระทำหรือกิจกรรมของมนุษย์ ซึ่งข้อมูลโดยสรุปของชุดข้อมูลมีดังนี้

1. รายละเอียดของชุดข้อมูล

- (a) เป้าหมายของชุดข้อมูล : เพื่อจำแนกสาระสำคัญของวิดีโอ
- (b) จำนวนของวิดีโอ : 8,264,650 วิดีโอ
- (c) ความยาวเฉลี่ยของแต่ละวิดีโอ : 229.6 วินาที
- (d) จำนวนของหมวดหมู่ของคำกำกับคุณลักษณะ : 4,800 หมวดหมู่
- (e) กฎในการรวบรวมวิดีโอดังนี้
 - i. ทุกคำกำกับต้องเป็นรูปธรรม
 - ii. ในแต่ละคำกำกับต้องมีจำนวนวิดีโอมากกว่า 200 วิดีโอ
 - iii. ความยาวของวิดีอottต้องอยู่ระหว่าง 120 - 500 วินาที

หลังจากได้กฎในการรวบรวมวิดีโอล้วน ขั้นตอนต่อไปคือการสร้างคำศัพท์ที่ใช้ในการค้นหาข้อมูลวิดีโอดังใน YouTube

(f) ขั้นตอนในการสร้างคำศัพท์มีดังนี้

- i. กำหนดบัญชีขาว (whitelist) ของคำกำกับที่เป็นรูปธรรมมา 25 ชนิด เช่น กีฬา เป็นต้น
- ii. กำหนดบัญชีดำ (blacklist) ของคำกำกับที่คิดว่าไม่เป็นรูปธรรมไว้ เช่น software เป็นต้น
- iii. รวบรวมคำกำกับที่มีอยู่ในรายการที่อนุญาตอย่างน้อยหนึ่งคำ และต้องไม่มีอยู่ในบัญชีดำ ซึ่งจะทำให้ได้คำกำกับที่ต้องการมาประมาณ 50,000 คำ
- iv. จักนั้นใช้ผู้ประเมินจำนวนสามคน ในการคัดคำกำกับที่คิดว่าเป็นรูปธรรม และสามารถจดจำหรือเข้าใจได้ยากโดยไม่ต้องเขียนชัญในด้านนั้น ซึ่งผู้ประเมิน ก็จะมีคำถามว่า “มันยกขนาดไหนถึงจะระบุได้ว่ามีคำกำกับดังกล่าวอยู่ในรูปหรือวิดีโอ โดยใช้เพียงแค่การมองเท่านั้น?” โดยแบ่งเป็นระดับดังนี้
 - A. บุคคลทั่วไปสามารถเข้าใจได้ (1)
 - B. บุคคลทั่วไปที่ผ่านการอ่านบทความที่เกี่ยวข้องมาแล้วสามารถเข้าใจได้ (2)
 - C. ต้องเขียนชัญในด้านใดซักด้านจึงจะเข้าใจได้ (3)
 - D. เป็นไปไม่ได้ ถ้าไม่มีความรู้ที่ไม่เป็นรูปธรรม (4)
 - E. ไม่เป็นรูปธรรม (5)
- v. หลังจากคำน้ำหนึ่งบันและการให้คะแนน จะทำการเก็บไว้เฉพาะคำกำกับที่มีคะแนนเฉลี่ยมากที่สุดอยู่ที่ประมาณ 2.5 คะแนนหรือต่ำกว่าเท่านั้น

- vi. ทำให้สุดท้ายเหลือเพียงประมาณ 10,000 คำที่สามารถใช้ได้
- vii. หลังจากได้คำกำกับที่คิดว่าเป็นรูปธรรมแล้วก็นำไปค้นหาและรวบรวมด้วย YouTube annotation system โดยมีขั้นตอนดังนี้
 - A. สุ่มเลือกวิดีโอมาก 10 ล้านวิดีโอพร้อมกับคำกำกับของวิดีโอด้วยใช้กฎที่กำหนดไว้
 - B. ทำให้เหลือจำนวนวิดีโอยู่ 8,264,650 วิดีโอ
 - C. แยกออกเป็นสามส่วนคือ ชุดข้อมูลสำหรับสร้างโมเดล (train set) ชุดข้อมูลสำหรับตรวจสอบ (validate set) และชุดข้อมูลสำหรับทดสอบ (test set) ในอัตราส่วน 70:20:10 ตามลำดับ

2. โมเดลปัญญาประดิษฐ์

(a) การเตรียมข้อมูล

- i. คุณลักษณะระดับเฟรม : ต้องทำการลดขนาดของข้อมูลลง เนื่องจากข้อมูลมีขนาดใหญ่มาก ทำให้ใช้เวลาในการประมวลผลนาน ซึ่งกระบวนการนี้จะมีการลดความเร็วเฟรมต่อวินาที เวลา เทอร์ของคุณลักษณะ (feature vector) และแปลงข้อมูลจาก 32 บิต ให้เป็น 8 บิต
- ii. คุณลักษณะระดับวิดีโอ : การแยกเวกเตอร์คุณลักษณะระดับวิดีโອอกจากคุณลักษณะระดับเฟรมซึ่งการทำแบบนี้ทำให้ได้ประโยชน์สามข้อ คือโมเดลที่ว่าไปที่ไม่ใช้โครงข่ายประสาทเทียม สามารถนำไปใช้งานได้ ขนาดข้อมูลเล็กลง และเหมาะสมกับการนำไปสร้างโมเดลในขอบเขตอื่นมากขึ้น

(b) โมเดลปัญญาประดิษฐ์ที่ใช้ในการทดสอบชุดข้อมูลแบบที่เป็นคุณลักษณะระดับเฟรม

i. one vs all logistic regression classifier + average pooling

สร้างโมเดลปัญญาประดิษฐ์ของทุกคำกำกับแยกกัน จะได้โมเดลปัญญาประดิษฐ์ 4800 โมเดล ซึ่งในการทำนายผลจะใช้การเฉลี่ยความน่าจะเป็นของแต่ละคำกำกับจากทุกๆเฟรมในวิดีโอด้วยคำกำกับที่มีความน่าจะเป็นมากที่สุดจะเป็นคำตอบของการทำนาย โดยมีสมการคำนวณความน่าจะเป็นเฉลี่ยของแต่ละคำกำกับดังนี้

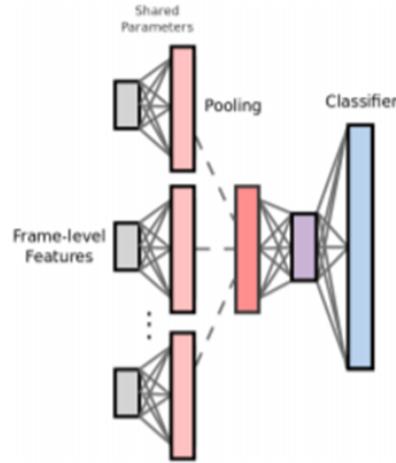
$$p_v(e|X_{1:F_v}^v) = \frac{1}{F_v} \sum_{j=1}^{F_v} p(e|X_j^v) \quad (2.12)$$

โดยที่

v	= วิดีโอที่ใช้ในการทำนายผล
e	= คำกำกับ
F_v	= จำนวนเฟรมสูงสุดของวิดีโอ v
$p_v(e X_{1:F_v}^v)$	= ความน่าจะเป็นของคำกำกับ e บนวิดีโอ v

ii. Deep bag of frames

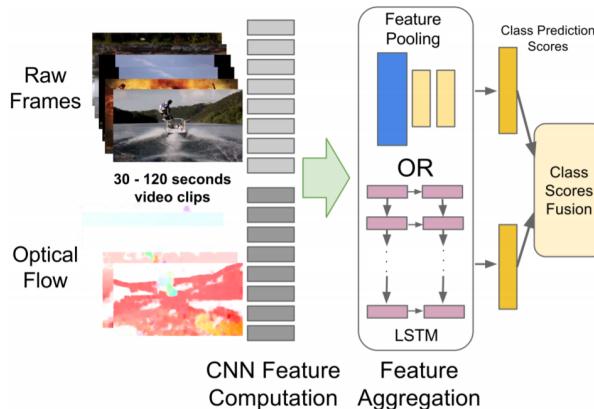
มีหลักการเหมือนกับ deep bag of words^[24] คือการแยกคุณลักษณะของเฟรม (extract frame feature) ที่โมเดลคิดว่าสำคัญอกรมาทำนายผล ซึ่งโครงสร้างโมเดลปัญญาประดิษฐ์ เป็นดังนี้ดังรูปที่ 2.16 โดยที่จะสุมหญิบ 20 เฟรมของวิดีโอมามาผ่าน ReLU activation function จากนั้นทำการ batch normalization ก่อนจะใช้ max pooling ในการรวมคุณลักษณะที่ได้ให้เป็นคุณลักษณะระดับวิดีโอ สุดท้ายใช้ softmax ในการจำแนกว่าเป็นคำกำกับใด



รูปที่ 2.16: โครงสร้างโมเดลปัญญาประดิษฐ์ของ deep bag of frames

iii. Long short-term memory (LSTM)

โมเดล LSTM ที่ใช้ในบทความนิ้นมีการอ้างอิงโครงสร้างมาจากบทความ "Beyond Short Snippets: Deep Networks for Video Classification"^[34] ซึ่งมีโครงสร้างดังรูปที่ 2.17 แต่



รูปที่ 2.17: โครงสร้าง LSTM ที่ใช้การอ้างอิงในบทความนี้

เนื่องจากข้อมูลของ YouTube-8M นั้นไม่สามารถเข้าถึงเฟรมวิดีโอติด (raw video frame) ได้จึงทำให้สามารถใช้ได้เพียงชั้นของ LSTM และ softmax เท่านั้น ซึ่งจากการทดลองพบว่า การใช้ LSTM 2 ชั้นที่มี hidden unit 1024 หน่วย นั้นมีประสิทธิภาพมากที่สุด

(c) โมเดลปัญญาประดิษฐ์ที่ใช้ในการทดสอบชุดข้อมูลแบบที่เป็นคุณลักษณะระดับวิดีโอ

i. Logistic regression

สร้างโมเดลปัญญาประดิษฐ์ของทุกคำกำกับแยกกัน จะได้โมเดลปัญญาประดิษฐ์ 4800 โมเดล โดยที่พารามิเตอร์ Θ (parameter) ของแต่ละโมเดลหาได้จาก

$$\sum_{i=1}^N L(y_{i,e}, \sigma(w_e X_i)) \quad (2.13)$$

$$L(y_{i,e}, \sigma(w_e X_i)) = y_{i,e} \log(\sigma(w_e X_i)) + (1 - y_{i,e}) \log(1 - \sigma(w_e X_i)) \quad (2.14)$$

$$p(e|X) = \sigma(w_e^T X_i) \quad (2.15)$$

$$\sigma(z) = \frac{1}{1 + \exp(-z)} \quad (2.16)$$

โดยที่

N = จำนวนวิดีโอทั้งหมด

X = คุณลักษณะระดับวิดีโอ

$y_{i,e}$ = คำตอบของคำกำกับ e ในวิดีโอที่ i

w_e = weight ของคำกำกับ e

$p(e|X)$ = ความน่าจะเป็นของคำกำกับ e ของคุณลักษณะระดับวิดีโอ X

- ii. Support vector machine (SVM) สร้างโมเดลปัญญาประดิษฐ์ของทุกคำกำกับแยกกันทำให้จะได้โมเดล SVM 4800 โมเดล โดยได้ใช้ค่า -1 ถึง 1 ในการแสดงถึงคำกำกับด้านลบ (negative label) และคำกำกับค้านบวก (positive label) ตามลำดับ และใช้ hinge loss ในการคำนวณหา loss (L) ซึ่งสมการจะเป็นดังนี้

$$L(y, \hat{y}) = \max(0, b - (2y - 1)\hat{y}) \quad (2.17)$$

โดยที่

y = คือคำตอบของการทำนาย โดยสามารถเป็น 0 หรือ 1 เท่านั้น

\hat{y} = คำผลการทำนายโดยจะมีค่าอยู่ในช่วง -1 ถึง 1

b = Hinge-loss พารามิเตอร์

- iii. Mixture of Expert (MoE) Mixture of experts ที่ถูกนำมาใช้ในการอ้างอิงในบทความนิยมจากบทความ "Hierarchical mixtures of experts and the EM algorithm"^[20] โดยการทำนายความน่าจะเป็นว่าคำกำกับ e ในวิดีโอ X ด้วยสมการ

$$p(e|X) = \sum_{h \in H_e} p(h|X) \sigma(u_h X) \quad (2.18)$$

ซึ่ง $p(h|X)$ สามารถเขียนได้ในรูปสมการดังนี้

$$p(h|X) = \frac{\exp(w_h X)}{1 + \sum_{h' \in H_e} \exp(w'_h X)} \quad (2.19)$$

โดยที่

H_e = hidden state หรือ expert ของโมเดล

X = วิดีโอที่ใช้ในการทำนายผล

u_h = logistic weight ของ expert h

w_h = softmax weight ของ expert h

$p(h|X) = |H_e| + 1$ ที่ผ่าน softmax function

ให้ชุดข้อมูลสำหรับใช้ในการสร้างโมเดล $(x_i, g_i)_{i=1\dots N}$ โดยที่ x_i คือเวกเตอร์คุณลักษณะ (feature vector) g_i คือคำตอบของการทำนายซึ่งสามารถเป็นได้เพียง 0 และ 1 เท่านั้น

และ N คือจำนวนวิดีโอสูงสุด ซึ่งสมการ log-loss ระหว่างผลการทำนายกับคำตอบของการทำนายคือ

$$L(p, g) = -g \log p - (1 - g) \log(1 - p) \quad (2.20)$$

ซึ่งสามารถเขียนในรูปอนุพันธ์ของ softmax weight และ logistic weight ได้ดังนี้

$$\frac{\partial L[p_{y|x,g}]}{\partial w_h} = x \frac{p_{h|x}(p_{y|h,x} - p_{y|x})(P_{y|x} - g)}{p_{y|x}(1 - p_{y|x})} \quad (2.21)$$

$$\frac{\partial L[p_{y|x,g}]}{\partial u_h} = x \frac{p_{h|x}p_{y|h,x}(1 - p_{y|x})(P_{y|x} - g)}{p_{y|x}(1 - p_{y|x})} \quad (2.22)$$

(d) เครื่องมือที่ใช้วัดผลสำหรับงานวิจัยนี้ คือ

- i. Mean Average Precision (mAP) ในแต่ละคำทำกับได้ทำการปัดเศษแนบทั้ง 10^{-4} แล้วเรียงลำดับคะแนนทั้งหมดที่ไม่ใช่ 0 จากนั้นให้ค่า τ เป็นค่าแบ่งเกณฑ์ (threshold) โดยที่มี $P(\tau)$ คือ precision และ $R(\tau)$ คือ recall ซึ่งหาได้จาก

$$P(\tau) = \frac{\sum_{t \in T} \mathbb{I}(y_t \geq \tau) g_t}{\sum_{t \in T} \mathbb{I}(y_t \geq \tau)} \quad (2.23)$$

$$R(\tau) = \frac{\sum_{t \in T} \mathbb{I}(y_t \geq \tau) g_t}{\sum_{t \in T} g_t} \quad (2.24)$$

โดยที่

y_t = ค่าความน่าจะเป็นในการทำนาย ซึ่งมีค่าอยู่ในช่วง 0 ถึง 1

g_t = คำตอบของการทำนาย โดยจะมีค่าเป็นได้แค่ 0 และ 1 เท่านั้น

$\mathbb{I}(y_t \geq \tau)$ = พังก์ชันซึ่งจะมีค่าเป็น 1 ถ้าหาก y_t มีค่ามากกว่า τ นอกเหนือจากนั้นจะมีค่าเป็น 0

สามารถหา average precision ได้จากการนี้

$$AP = \sum_{j=1}^{10000} P(\tau_j) [R(\tau_j) - R(\tau_j + 1)] \quad (2.25)$$

โดยที่ $\tau = \frac{j}{10000}$

ii. Hit@k

เหมือนกันกับ Top@k คือการจัดลำดับความน่าจะเป็นของแต่ละคำกำกับจำนวน k อันดับแรก ถ้าหากมีคำกำกับที่ถูกต้องอยู่ในลำดับเหล่านั้น จะถือว่าการทำนายถูกต้อง ซึ่งสามารถเขียนเป็นสมการได้ดังนี้

$$\frac{1}{|V|} \sum_{v \in V} V_e \mathbb{I}(rank_{v,e} \leq k) \quad (2.26)$$

โดยที่

$$\begin{aligned} V &= วิดีโอที่ใช้ในการทดสอบทั้งหมด \\ G_v &= คำตอบของวิดีโอ v \\ rank_{v,e} &= อันดับของคำตอบที่ถูกต้อง e ของวิดีโอ v ที่ได้จากการทำนาย \\ k &= อันดับที่ใช้เป็นเกณฑ์ \end{aligned}$$

iii. Precision at equal recall rate (PERR)

สำหรับแต่ละวิดีโอจะดูความแม่นยำของผลการทำนาย k อันดับแรก โดยที่ k คือจำนวนคำตอบทั้งหมดของวิดีโอนั้น จากนั้นเฉลี่ยค่าเหล่านั้นด้วยจำนวนวิดีโอทั้งหมด สามารถเขียนได้ในรูปสมการดังนี้ โดยใช้ตัวแปรเดียวกันกับของ Hit@k

$$\frac{1}{|V : |G_v| > 0|} \sum_{v \in V : |G_v| > 0} \left[\frac{1}{|G_v|} \sum_{e \in G_v} \mathbb{I}(rank_{v,e} \leq |G_v|) \right] \quad (2.27)$$

(e) ประสิทธิภาพของโมเดลปัญญาประดิษฐ์ที่ใช้ชุดข้อมูลของ YouTube-8M ในการสร้างเทียบกับชุดข้อมูลสำหรับทดสอบของ YouTube-8M

Input features	Modeling approach	mAP	Hit@1	PERR
Frame-level	Logistic + average	11.0	50.8	42.2
	Deep bag of frames	26.9	62.7	55.1
	LSTM	26.6	64.5	57.3
Video-level	SVM	17.0	56.3	47.9
	Logistic regression	28.1	60.5	53.0
	Mixture-of-2-experts	30.0	63.3	55.8

ตารางที่ 2.2: ผลการทดสอบโมเดลต่างๆบนชุดข้อมูลสำหรับทดสอบของ YouTube-8M

- (f) โมเดลปัญญาประดิษฐ์ที่ใช้ชุดข้อมูลของ YouTube-8M ในการสร้างแล้วปรับโมเดลด้วยชุดข้อมูลของ Sports-1M เมื่อนำไปทดสอบกับชุดข้อมูลของ Sports-1M พบร่วงประสิทธิภาพเมื่อทดสอบด้วย Hit@1 และ Hit@5 สูงสุดอยู่ที่ 65.7% และ 86.2% ตามลำดับ ซึ่งใกล้เคียงกับสถิติสูงสุดในตอนนั้นที่ทำไว้ 73.0% และ 91.0% (พ.ศ. 2559) โดยที่ใช้เพียงคุณลักษณะจากการเดรียมข้อมูลไว้แล้ว เมื่อเทียบกับสถิติเก่าที่มีการใช้ optical flow เข้ามาช่วย จึงเป็นการพิสูจน์ให้เห็นว่าจำนวนข้อมูลนั้นมีผลต่อการพัฒนาโมเดลปัญญาประดิษฐ์
- (g) โมเดลปัญญาประดิษฐ์ที่ใช้ชุดข้อมูลของ YouTube-8M ในการสร้างแล้วปรับโมเดลด้วยชุดข้อมูลของ ActivityNet เมื่อนำไปทดสอบกับชุดข้อมูลของ ActivityNet พบร่วงประสิทธิภาพเมื่อทดสอบด้วย mAP สูงสุดอยู่ที่ 77.6% ในขณะที่สถิติเดิมทำไว้เพียง 53.8% เป็นการพิสูจน์ว่าชุดข้อมูลนี้นั้นมีความครอบคลุม (generalize) พอดีจะนำไปใช้กับงานประเภทนี้ในหมวดอื่นๆ เนื่องจากว่าจำนวนข้อมูลที่มีขนาดใหญ่และมีความหลากหลาย
- (h) ปัญหาที่พบ
เนื่องจากว่า YouTube-8M นั้นมีจำนวนข้อมูลที่เยอะมาก ทำให้ไม่สามารถตรวจสอบความถูกต้องของชุดข้อมูลได้ทั้งหมดว่ามีความถูกต้องมากน้อยขนาดไหน ทำให้อาจเกิดข้อผิดพลาดได้ (ปัจจุบันปี 2019 YouTube-8M ได้มีการตรวจสอบข้อมูลอีกครั้ง เพื่อเพิ่มประสิทธิภาพของชุดข้อมูลซึ่งทำให้ปัจจุบันจำนวนข้อมูล และจำนวนคำจำกัดน้อยลงจากข้อมูลที่ใช้อ้างอิงในบทความข้างต้นที่ได้กล่าวมา)

ชุดข้อมูล Atomic visual action (AVA)

AVA^[12] คือ ชุดข้อมูลที่รวมวิดีโอที่มีความยาว 15 นาที ถูกแบ่งด้วยความถี่ 1 hz (900 keyframes) จากในภาพยนต์โดยใช้การกระทำของมนุษย์เป็นศูนย์กลาง เพื่อใช้สำหรับสร้างโมเดลที่เข้าใจกรรมของมนุษย์ในวิดีโอด้วยคำจำกัดความรู้สึกที่ต้องการ ซึ่งข้อดีของ AVA คือ ชุดข้อมูลจะมีคำจำกัดความรู้สึกที่ต้องการ ซึ่งสามารถแบ่งได้เป็นสามหมวดหมู่คือ ท่าทาง (Pose), ปฏิสัมพันธ์กับวัตถุ (Interaction with object) และปฏิสัมพันธ์กับบุคคล (Interaction with people) ซึ่งสามารถมีคำจำกัดได้มากถึง 7 คำจำกัด

1. รายละเอียดชุดข้อมูล

(a) ขั้นตอนการเก็บข้อมูลสำหรับการทำชุดข้อมูลมีขั้นตอนการทำ 5 ขั้นดังนี้

- i. การสร้างคำศัพท์การกระทำจะมีหลักการ 3 ข้อในการรวบรวมคำศัพท์ดังนี้
 - A. เก็บรวบรวมคำศัพท์ที่นำไปที่เกิดขึ้นในชีวิตประจำวัน
 - B. จะต้องมีเอกสารฉากรายชื่อของนักแสดงที่มีชื่อเสียง ซึ่งจะมีความหลากหลายของเชื้อชาติ รวมกันอยู่ วิดีโอด้วยที่ถูกคัดเลือกจะมีเกณฑ์ดังนี้
 - A. วิดีโอด้วยอยู่ในหมวด ภาพยนต์ และละครโทรทัศน์
 - B. วิดีโอด้วยต้องมีความยาวมากกว่า 30 นาที
 - C. เผยแพร่มาแล้วเป็นระยะเวลาน้อยกว่า 1 ปี
 - D. มีจำนวนยอดคนดูมากกว่า 1,000 ครั้ง
 - E. ลงทะเบียนวิดีโอบางประเภท เป็นภาพขาว-ดำ มีความละเอียดต่ำ การ์ตูน หรือวิดีโอกลเมอร์
 - ii. การสร้างตัวอย่างที่ใช้สำหรับฝึกอบรม ภาพยนต์ และละครโทรทัศน์ ให้สามารถจำแนกได้แม่นยำ ตามที่ต้องการ ซึ่งจะต้องมีขั้นตอนดังนี้
 - A. สร้างกรอบสี่เหลี่ยมโดยใช้โมเดลปัญญาประดิษฐ์ faster RCNN สำหรับการตรวจจับมนุษย์
 - B. ใช้มนุษย์ในการตรวจสอบและแก้ไขกรอบสี่เหลี่ยมที่ผิดพลาด
 - iii. การสร้างกรอบสี่เหลี่ยมโดยใช้โมเดลปัญญาประดิษฐ์ faster RCNN สำหรับการตรวจจับมนุษย์ ทำการติดตามตำแหน่งของบุคคลที่อยู่ในช่วงเวลาเดียวกันด้วยใช้วิธีการแทร็คโดยใช้คำนึงมนุษย์ เป็นศูนย์กลาง โดยการคำนวณค่าความใกล้เคียงกันระหว่างบุคคล โดยใช้ person embedding (ใช้โครงข่ายประสาทเทียมในการหาไฟเจอร์ขั้นสูงและใช้เมทริกซ์ในการหาความสัมพันธ์ ของแต่ละคน) จากนั้นจะใช้อัลกอริทึม Hungarian distance (อัลกอริทึมสำหรับการหาข้อเสนอที่ดีที่สุด) ในการหาตัวเลือกคู่ของกรอบสี่เหลี่ยมที่ดีที่สุด
 - iv. การติดตามตำแหน่งของบุคคล

ทำการติดตามตำแหน่งของบุคคลที่อยู่ในช่วงเวลาเดียวกันด้วยใช้วิธีการแทร็คโดยใช้คำนึงมนุษย์ เป็นศูนย์กลาง โดยการคำนวณค่าความใกล้เคียงกันระหว่างบุคคล โดยใช้ person embedding (ใช้โครงข่ายประสาทเทียมในการหาไฟเจอร์ขั้นสูงและใช้เมทริกซ์ในการหาความสัมพันธ์ ของแต่ละคน) จากนั้นจะใช้อัลกอริทึม Hungarian distance (อัลกอริทึมสำหรับการหาข้อเสนอที่ดีที่สุด) ในการหาตัวเลือกคู่ของกรอบสี่เหลี่ยมที่ดีที่สุด
 - v. การสร้างคำจำกัดความรู้สึก

การสร้างคำจำกัดของกรรมการทำจะถูกสร้างขึ้นโดยมนุษย์ ซึ่งผู้วิจัยจะใช้โปรแกรมสำหรับช่วยเหลือในการสร้างคำจำกัดความรู้สึก โดยสามารถกำหนดคำจำกัดของกรรมการทำได้สูงถึง 7 คำต่อ 1 กรอบสี่เหลี่ยม นอกจากนั้นสามารถตั้งสถานะเนื้อหาที่ไม่เหมาะสม หรือ

กรอบสีเหลี่ยมที่ผิดพลาดได้อีกด้วย ซึ่งในทางปฏิบัติเพื่อลดโอกาสที่จะเกิดข้อผิดพลาด จึงแบ่งขั้นตอนในการสร้างคำกำกับออกเป็น 2 ขั้นตอนดังนี้

- A. สร้างข้อเสนอสำหรับคำกำกับของการกระทำ
- B. ข้อเสนอจะถูกตรวจสอบข้อเสนอที่ได้จากขั้นตอนแรก ซึ่งจะใช้มุขย์ในการตรวจสอบ 3 คน โดยคำกำกับจะต้องถูกตรวจสอบด้วยผู้ตรวจสอบอย่างน้อย 2 คน จึงจะถูกยืนยันว่าเป็นคำกำกับหลัก

2. โมเดลปัญญาประดิษฐ์

- (a) โมเดลปัญญาประดิษฐ์ที่งานวิจัยนี้ใช้ คือ two stream variant ซึ่งจะทำการประมวลผลทั้ง RGB flow และ optical flow โดยเป็นโครงสร้างของ faster RCNN ที่นำ Inception network เข้ามาใช้
- (b) เครื่องมือที่ใช้วัดผลสำหรับงานวิจัยนี้ คือค่า IoU และ 3D IoUs
 - i. ค่า IoU คือค่าที่ใช้วัดความสอดคล้องระหว่างสองกรอบสีเหลี่ยม(กรอบสีเหลี่ยมจริงของเฟรม และ กรอบสีเหลี่ยมที่ทำนายขึ้นมา) ซึ่งใช้สำหรับการวัดผลระดับเฟรม
 - ii. ค่า 3D IoUs คือค่าที่ใช้วัดความสอดคล้องระหว่างกรอบสีเหลี่ยมภายใน 2 วิดีโอ ซึ่งใช้สำหรับการวัดผลระดับวิดีโอ โดยเทียบกันระหว่างกรอบสีเหลี่ยมจริงในช่วงของเฟรมที่ต่อกัน (ground-truth tubes) และ กรอบสีเหลี่ยมที่ทำนายขึ้นมาในช่วงของเฟรมที่ต่อกัน (linked detection tubes)
- (c) ประสิทธิภาพของโมเดลปัญญาประดิษฐ์ในปัจจุบัน
ข้อมูลโมเดลปัญญาประดิษฐ์ที่นำมาทดสอบ
 - i. Actionness^[23] เป็นการหาความน่าจะเป็นของการกระทำ โดยใช้โครงสร้างของ hybrid fully convolutional network (HFCN) hybrid fully เป็นโครงสร้างที่ประกอบด้วยโครงข่ายประสาทเทียม 2 ชนิด คือ
 - A. Appearance-FCN (A-FCN) คือ โครงข่ายประสาทเทียมที่นำมาใช้แสดงลักษณะของวัตถุ(ตำแหน่งวัตถุ, ความตื้นลึกวัตถุ) ที่ปรากฏบนรูป RGB1
 - B. MotionFCN (M-FCN) คือ โครงข่ายประสาทเทียมที่แยกการเคลื่อนไหวจากข้อมูลของ optical flow
 - ii. Peng without MR, Peng with MR (Multi-region two-stream R-CNN)^[33] เป็นโมเดลปัญญาประดิษฐ์ที่ใช้สำหรับตรวจสอบวิดีโອิชีวิตจริง ซึ่งพื้นฐานของโมเดลนี้เป็น Faster R-CNN โดยโมเดลนี้มีกระบวนการ 3 กระบวนการคือ
 - A. สร้างข้อเสนอพื้นที่ที่มีการเคลื่อนไหว
 - B. สะสม Optical flow จากเฟรมหลายเฟรม เพื่อนำไปปรับปรุงการตรวจจับการกระทำ
 - C. นำพื้นที่หลายๆส่วนมาวิเคราะห์ผ่านโมเดล Faster R-CNN
 - iii. ACT Action Tubelet Detector^[32] เป็นการระบุตำแหน่งของการกระทำที่มีระยะเวลาสั้นๆ ซึ่งใช้วิธีการตรวจจับระดับเฟรม และ ใช้การติดตามตำแหน่งในการเชื่อมระหว่างเฟรม ปัจจุบันไบยังเฟรมถัดไป. ACT ถูกสร้างต่อจาก SSD framework และ ใช้คอนโวลูชันในการสกัดคุณลักษณะในแต่ละเฟรมซึ่งการคิดคะแนนและความน่าจะเป็นของหมวดหมู่จะคิดจาก การนำคุณลักษณะเรียงต่อกัน และ หาข้อมูลจากลำดับข้อมูลนั้น

จากการทดสอบการเทียบโมเดลปัญญาประดิษฐ์ของงานวิจัยนี้และวิธีการอื่นๆ โดยนำไปทดสอบกับชุดข้อมูลวิดีโอ JHMDB และ UCF101-24 ได้ผลลัพธ์ออกมาดังนี้

Frame-mAP	JHMDB (mAP)	UCF101-24 (mAP)
Actionness	39.9	-
Peng w/o MR	56.9	64.8
Peng w/ MR	58.5	65.7
ACT	65.7	69.5
2 stream(Our approach)	73.3	76.3

ตารางที่ 2.3: ผลการทดลองของวิธีต่างๆบนคุณลักษณะระดับเฟรม

- (d) ปัญหาที่พบ ในปัจจุบันยังไม่มีโมเดลปัญญาประดิษฐ์ที่ทดสอบด้วยชุดข้อมูล AVA และได้ผลการทำงานที่ดี เนื่องจากชุดข้อมูลนี้สนใจการกระทำของมนุษย์ที่มีรายละเอียดเล็กๆน้อยๆ ทำให้ยากต่อการทำนายสำหรับโมเดลปัญญาประดิษฐ์

ចុះថ្ងៃខែឆ្នាំ ទៅការណ៍ពីការបង្កើតរឹងរាល់

Moments in time^[27] คือชุดข้อมูลที่ใช้มนุษย์ในการกำกับข้อมูลทั้งหมดให้กับวิดีโอดิสต์แล้วและมีจำนวนกิจกรรมหรือการกระทำต่างกัน 339 หมวดหมู่ โดยแต่ละวิดีโอมีความยาวอยู่ที่ 3 วินาที เนื่องจากเป็นเวลาเฉลี่ยที่มนุษย์ใช้ในการเข้าใจกับเหตุการณ์ที่เกิดขึ้น (human working memory) รูปแบบของชุดข้อมูลจะมีอยู่ทั้งหมดด้วย 3 รูปแบบ ได้แก่ ภาพนิ่ง (spatial) เสียง (auditory) และการเคลื่อนไหว (temporal) นอกจากนี้ชุดข้อมูลนี้นั้นไม่รวมรวมเพียงแค่การกระทำของมนุษย์เท่านั้น ยังรวมไปถึง สัตว์ สิ่งของ และปรากฏการณ์ธรรมชาติ ทำให้ชุดข้อมูลนี้เป็นการท้าทายรูปแบบใหม่ เพราะด้วยข้อมูลที่มีความซับซ้อนมากขึ้น

1. รายละเอียดชุดข้อมูล

- (a) เป้าหมายของชุดข้อมูล : สนใจเหตุการณ์ที่เกิดขึ้นในวิดีโอ เช่น การกระทำของคนหรือสัตว์เหตุการณ์ และประวัติการณ์ธรรมชาติ

(b) จำนวนของวิดีโอ : มากกว่า 1,000,000 วิดีโอ

(c) ความยาวเฉลี่ยของแต่ละวิดีโอ : 3 วินาที

(d) จำนวนของหมวดหมู่ : 339 หมวดหมู่

(e) วิธีการเก็บรวบรวมข้อมูล

 - เริ่มจากการรวบรวมคำที่ใช้อยู่ทั่วไปในชีวิตประจำวันมา 4,500 คำจาก VerbNet^[29] เว็บไซต์ที่เก็บรวบรวมคำกริยาภาษาอังกฤษขนาดใหญ่ จากนั้นนำมาแบ่งกลุ่มคำที่มีความหมายใกล้เคียงกันโดยใช้คุณลักษณะจาก Propbank^[35] และ FrameNet^[14] โดยเก็บข้อมูลเป็นแบบเวกเตอร์คุณลักษณะฐานสอง (binary feature vector) ซึ่งถ้าคำใดมีความเกี่ยวข้องกันทางคุณลักษณะก็จะให้ค่าเป็น 1 ถ้าไม่เกี่ยวข้องกันจะให้ค่าเป็น 0 จากนั้นจึงใช้วิธี k-means clustering ในการแบ่งกลุ่ม เมื่อแบ่งกลุ่มแล้วจากนั้นจะเลือกคำจากในแต่ละกลุ่มนั้น โดยคำที่เลือกมานั้นจะเป็นคำที่ใช้บ่อยที่สุดในกลุ่มนั้น และลบคำนั้นออกจากกลุ่มอื่นๆทั้งหมด (ค่าหนึ่งสามารถอยู่ได้หลายกลุ่ม) จากนั้นจะทำการวนการนี้ไปเรื่อยๆ แต่คำที่เลือกมาจะต้องไม่มีความหมายคลุมเครือ หรือเป็นสิ่งที่ไม่สามารถมองเห็นหรือได้ยินได้ และต้องไม่มีความหมายเหมือนกับคำที่เคยเลือกมาก่อน จนสุดท้ายแล้วได้ออกมาที่ 339 หมวดหมู่
 - ต่อมาทำการหาชุดข้อมูลวิดีโอด้วยจะตัดออกมาเพียง 3 วินาทีที่เกี่ยวข้องกับคำใน 339 หมวดหมู่ที่เลือกมาจากวิดีโอแหล่งต่างกัน 10 แหล่ง การตัดวิดีโอนั้นจะไม่ใช้พวาก Video2Gif (ไม่เดลที่ระบุตำแหน่งของสิ่งที่นำเสนอด้วยวิดีโอ) เพราะจะทำให้เกิดอคติขึ้นจะเกิดขึ้นตอนสร้างไม่เดล ดังนั้นจึงใช้มนูษย์ในการตัดวิดีโอ จากนั้นจะทำการส่งข้อมูลของคำ และวิดีโอที่ตัดไปยัง Amazon Mechanical Turk (AMT หรือตลาดแรงงาน) เพื่อทำการสร้างคำ做起กับโดยพนักงานของ AMT ทำให้ได้ 64 วิดีโอที่เกี่ยวข้องกับคำหนึ่ง และอีก 10 วิดีโอที่มีคำกำกับอยู่แล้ว โดยวิดีโอด้วยคำกำกับอยู่แล้วนั้นถูกันนักงานของ AMT ตอบเหมือนกันเกิน 90% ถึงจะนำเข้าไปรวมกับชุดข้อมูลส่วนอีก 64 วิดีโอ ถ้าเป็นชุดข้อมูลสำหรับสร้างไม่เดลจะต้องผ่านพนักงานของ AMT อย่างน้อย 3 ครั้ง และต้องมีคำกำกับเหมือนกัน 75% ขึ้นไปถึงจะถือว่าเป็นคำกำกับที่ถูกต้อง ถ้าเป็นชุดข้อมูลสำหรับตรวจสอบ และชุดข้อมูลสำหรับทดสอบจะต้องผ่านพนักงานของ AMT อย่างน้อย 4 ครั้ง และต้องมีคำกำกับเหมือนกัน 85% ขึ้นไป เหตุผลที่ไม่ตั้งเกณฑ์ไว้ที่ 100% เพราะจะทำให้วิดีโอนั้นยากเกินไปที่จะทำให้สามารถจำการกระทำได้

2. การเตรียมข้อมูล

- (a) ชุดข้อมูลสำหรับสร้างโมเดลจะมี 802,264 วิดีโอ และมีวิดีโອนแต่ละหมวดหมู่อยู่ที่ 500 ถึง 5,000 วิดีโอ
- (b) ชุดข้อมูลสำหรับตรวจสอบความต่อเนื่องจะมี 33,900 วิดีโอ และมีวิดีโອนแต่ละหมวดหมู่อยู่ที่ 100 วิดีโอ
- (c) แยกเฟรม RGB ออกมาจากวิดีโอ และทำการเปลี่ยนขนาดให้เป็น 340×256 pixel
- (d) ใช้อัลกอริทึม TVL1 optical flow จาก OpenCV เพื่อผลข้อมูลrgbawanที่จะเกิดขึ้น
- (e) ทำการแปลงค่าที่อยู่ใน optical flow ให้เป็นเลขจำนวนเต็มเพื่อทำให้การคำนวนนั้นเร็วขึ้น
- (f) ปรับค่า displacement ใน optical flow ให้ค่าสูงสุดเป็น 15 ต่ำสุดเป็น 0 และทำการปรับขนาดให้เป็นช่วง 0 - 255
- (g) เก็บข้อมูลอกมาในรูปแบบของภาพขาวดำเพื่อลดพื้นที่ในการเก็บข้อมูล
- (h) แก้ปัญหาเรื่องการเคลื่อนไหวของกล้องด้วยการนำค่าเฉลี่ยของเวกเตอร์ไปลบกับ displacement
- (i) สุดท้ายจะเป็นสุ่มตัดภาพอกมาเพื่อเพิ่มจำนวนข้อมูล

3. โมเดลปัญญาประดิษฐ์

- (a) ในงานวิจัยนี้มีการทดสอบโมเดลปัญญาประดิษฐ์หลายรูปแบบ โดยโมเดลปัญญาประดิษฐ์ที่มีประสิทธิภาพการทำงานที่ดีที่สุด 5 ลำดับแรกมีดังนี้
 - i. SVM มีรูปแบบข้อมูลที่ป้อนเข้า คือ เฟรมที่ต่อเนื่อง (spatial) + เฟรมเดี่ยว (temporal) + ข้อมูลเสียง (auditory)
 - ii. I3D มีรูปแบบข้อมูลที่ป้อนเข้า คือ เฟรมที่ต่อเนื่อง + เฟรมเดี่ยว
 - iii. TRN-Multiscale มีรูปแบบข้อมูลป้อนเข้า คือ เฟรมที่ต่อเนื่อง + เฟรมเดี่ยว
 - iv. TSN-2stream มีรูปแบบข้อมูลป้อนเข้า คือ เฟรมที่ต่อเนื่อง + เฟรมเดี่ยว
 - v. ResNet50-ImageNet มีรูปแบบข้อมูลป้อนเข้า คือ เฟรมที่ต่อเนื่อง
- (b) เครื่องมือที่ใช้วัดผลงานวิจัยนี้
 - i. Classification accuracy Top-1, Top-5
- (c) ประสิทธิภาพของโมเดลปัญญาประดิษฐ์ในปัจจุบัน
 - i. ทำการทดสอบด้วยวิธี cross dataset transfer โดยการนำโมเดล ResNet50 I3D ที่สร้างด้วยชุดข้อมูล Kinetics และ Moments in Time แล้วนำหัวทั้ง 2 โมเดลไปทดสอบกับชุดข้อมูลอื่น โดยจะปรับอัตราความถี่ของเฟรม (frame rate) ของวิดีโอให้เป็น 5 fps

Pretrained	Fine-Tuned		
	UCF-101	HMDB-51	Something Something
Kinetics	Top-1 : 92.6	Top-1 : 62.0	Top-1 : 48.6
	Top-5 : 99.2	Top-5 : 88.2	Top-5 : 77.9
Moments	Top-1 : 91.9	Top-1 : 65.9	Top-1 : 50.0
	Top-5 : 98.6	Top-5 : 89.3	Top-5 : 78.8

ตารางที่ 2.4: ประสิทธิภาพของโมเดล Resnet50 I3D ที่ใช้ชุดข้อมูล Kinetics และ Moments in Time

- ii. จะเห็นได้ว่า Kinetics ให้ผลลัพธ์ที่ดีกว่าใน UCF-101 เพราะว่ามีหมวดหมู่ที่ตรงกันอยู่หลายอย่าง ในขณะที่ HMDB-51 นั้นมีการรวมข้อมูลจากหลายแหล่ง และมีจำนวนหมวดหมู่ที่หลากหลายจึงทำให้มีความไม่คล้ายเดียงกับตัวข้อมูลของ Moments in Time ดังนั้นจึงเทียบผลลัพธ์จาก Something Something ซึ่งจะทำให้เห็นว่า Moments in Time มีประสิทธิภาพที่ดีกว่าและวิดีโอที่มีความยาวมากกว่า 3 วินาทีจะไม่ส่งผลกระทบกับประสิทธิภาพของ Moments in Time

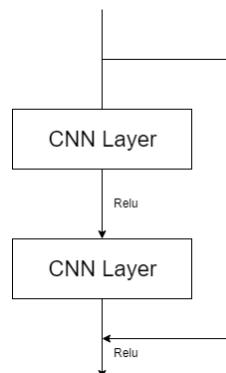
4. ปัญหาที่พบ

ผลลัพธ์จากการทำนายด้วยโมเดลถ้าผ่านภาพที่มีรายละเอียดเยอะจะทำให้การ ทำนายโอกาสผิดนั้นค่อนข้างสูง ซึ่งปัญหานี้สามารถทำให้เกิดน้อยลงด้วยการนำวิธี class activation mapping (CAM) จะเป็นการเน้นภาพในส่วนที่มีข้อมูลมากที่สุดและทำนายผลออกมา แต่ก็ยังมีจุดที่เป็นปัญหาอยู่ เช่น การกระทำที่เกิดขึ้นเร็วมากจะทำให้การทำนายนั้นมีโอกาสผิดสูงขึ้น

2.4 โมเดลปัญญาประดิษฐ์สำหรับจำแนกการกระทำมนุษย์

2.4.1 ResNet

ในการสร้างโมเดลปัญญาประดิษฐ์นั้นการใช้จำนวนขั้นเยือนนั้นจะทำให้ได้คุณลักษณะของข้อมูลที่อกรมา เยอะตามไปด้วย แต่การที่คุณลักษณะของข้อมูลเหล่านี้ไม่ได้หมายความว่าโมเดลปัญญาประดิษฐ์จะให้ประสิทธิภาพ ที่ดีเสมอไป ซึ่งสามารถแก้ปัญหานี้ได้โดยใช้ residual network (ResNet) ซึ่งเป็น CNN ประเภทหนึ่ง ที่ส่วนใหญ่ จะนำมาใช้กับข้อมูลที่เป็นภาพ เช่น การจดจำวัตถุ เป็นต้น โดย ResNet นี้จะสามารถทำการข้ามชั้นที่ไม่จำเป็น ได้ การข้ามชั้นที่ไม่จำเป็นจะช่วยลดเวลาที่ใช้ในการสร้างโมเดลปัญญาประดิษฐ์ และทำให้ประสิทธิภาพของโมเดล ปัญญาประดิษฐ์ดีขึ้น



รูปที่ 2.18: หลักการของ Residual block ของ ResNet

การทดลองโมเดลปัญญาประดิษฐ์ ResNet ด้วยการทำจำแนกภาพโดยใช้ชุดข้อมูลทดสอบ ImageNet ที่มี หมวดหมู่มากกว่า 1,000 หมวดหมู่ มาเทียบกับโมเดลปัญญาประดิษฐ์ทั่วไป (plain model) ที่จำนวนชั้น 18 ชั้น และ 34 ชั้น โดยโครงสร้างพื้นฐานของโมเดลปัญญาประดิษฐ์ ResNet และโมเดลปัญญาประดิษฐ์ทั่วไปเหมือนกัน ซึ่ง ผลลัพธ์อัตราrror อย่างของความผิดพลาดจะได้ออกมาตามตารางที่ 2.5

จำนวนชั้นของโมเดลปัญญาประดิษฐ์	Training error	
	plain	ResNet
18	27.94	27.88
34	28.54	25.03

ตารางที่ 2.5: อัตราrror อย่างของความผิดพลาดของชุดข้อมูลทดสอบ ImageNet

จากตาราง 2.5 จะเห็นได้ว่าโมเดลปัญญาประดิษฐ์ทั่วไป 34 ชั้นมีค่าอัตราrror อย่างของความผิดพลาดสูงกว่า โมเดลปัญญาประดิษฐ์ ResNet ได้อย่างชัดเจน ในขณะที่โมเดลปัญญาประดิษฐ์ทั่วไปจะมีอัตราrror อย่างของความ ผิดพลาดสูงขึ้นเมื่อเทียบกันระหว่าง 18 ชั้นและ 34 ชั้น

ต่อมาจะนำโมเดลปัญญาประดิษฐ์ ResNet มาทดสอบกับชุดข้อมูล CIFAR-10 ซึ่งเป็นชุดข้อมูลที่มีรูปสำหรับใช้สร้างโมเดลปัญญาประดิษฐ์ 50,000 รูป รูปสำหรับทดสอบ 10,000 รูป และมีจำนวนหมวดหมู่ทั้งหมด 10 หมวดหมู่ โดยจะมีการออกแบบของจำนวนชั้นของโมเดลปัญญาประดิษฐ์ ResNet ตามจำนวนของชั้น convolution ที่มีผังคุณลักษณะเท่ากัน 6 ชั้นติดกันและการข้ามชั้นทีละ 2 ชั้น จึงทำให้ได้รูปแบบการคิดชั้นดังนี้ $6n + 2$ สำหรับการทดสอบจะให้ค่า $n = [3, 5, 7, 9, 200]$ ดังตารางต่อไปนี้

โมเดลปัญญาประดิษฐ์	จำนวนชั้น	Training error
ResNet	20	8.75
ResNet	32	7.51
ResNet	44	7.17
ResNet	56	6.97
ResNet	110	6.43
ResNet	1202	7.93

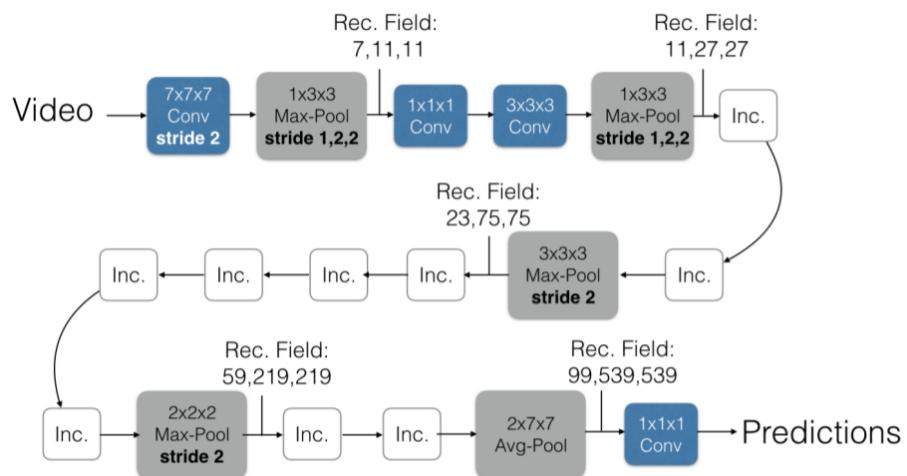
ตารางที่ 2.6: ค่าความผิดพลาดที่ได้จากการทดลองจำนวนชั้นของโมเดลปัญญาประดิษฐ์ ResNet บนชุดของข้อมูล CIFAR-10

จากตาราง 2.6 จะเห็นได้ว่าที่โมเดลปัญญาประดิษฐ์ ResNet ที่มีจำนวนชั้น 1,202 นั้นมีค่าความผิดพลาดเกิดขึ้นมากกว่าจำนวนชั้น 110 ซึ่งอาจจะเป็นไปได้ว่าขนาดของโมเดลปัญญาประดิษฐ์ ResNet ที่มีจำนวนชั้น 1,202 นั้นมากเกินไปสำหรับชุดข้อมูลขนาดเล็กนี้

2.4.2 Inflated 3D convolutional network

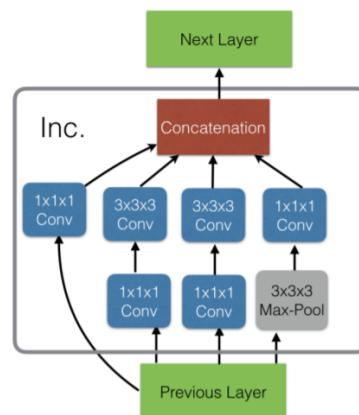
ในการพัฒนาโมเดลปัญญาประดิษฐ์สำหรับจำแนกการกระทำของมนุษย์นั้นมีพื้นฐานมาจากการจำแนกวัตถุ หมายถึงการใช้รูปภาพหนึ่งรูปในการประมวลผลและทำนายอุปกรณ์ว่าภายในรูปนั้นมีบริบทการกระทำอย่างไร โดยไม่ได้คำนึงถึงข้อมูลเชิงต่อเนื่อง (spatio-temporal information) จากบทความ "Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset"^[16] นั้นได้พัฒนาโครงสร้างของโมเดลปัญญาประดิษฐ์ที่มีประสิทธิภาพในการประมวลผลภาพเคลื่อนไหวได้ชื่อว่า I3D หรือ inflated 3D-convolution network โดยโครงสร้างพื้นฐานของ I3D นั้นมาจากการสร้างของ Inception-v1^[4] ที่ถูกพัฒนาโดย Google ซึ่งเป็นโครงสร้างที่มีประสิทธิภาพสูงในการจำแนกวัตถุในรูปภาพ แล้ว I3D นั้นได้ทำการขยายมิติของเครื่องเรนาลจาก 2 มิติ เป็น 3 มิติ เพื่อให้โมเดลปัญญาประดิษฐ์สามารถเรียนรู้ข้อมูลเชิงต่อเนื่องได้

Inflated Inception-V1



รูปที่ 2.19: โครงสร้างของโมเดลปัญญาประดิษฐ์ I3D^[16]

Inception Module (Inc.)



รูปที่ 2.20: โครงสร้างของโมเดลปัญญาประดิษฐ์ I3D^[16]

จากรูปที่ 2.19 ในส่วนของชั้น Inception (Inc.) จะมีลักษณะโครงสร้างดังรูปที่ 2.20 เนื่องจากว่าการที่โมเดลนั้นซับชั้อนมากขึ้นก็ต้องใช้ทรัพยากรในการประมวลผลมากขึ้น Google จึงออกแบบโครงสร้างที่สามารถลดความซับซ้อนของโมเดลลงด้วยการใช้เครอร์เนลขนาด 1×1 ในเครอร์เนล 2 มิติ ($1 \times 1 \times 1$ ใน 3 มิติ) เพื่อลดจำนวน channel ของเครอร์เนลลง ตัวอย่างเช่น หากในชั้นก่อนหน้าได้ผลลัพธ์ที่มีขนาด $14 \times 14 \times 480$ (480 คือจำนวน channel ของเครอร์เนล) หากในชั้นถัดไปใช้เครอร์เนลที่มีขนาด $5 \times 5 \times 48$ จะทำให้มีจำนวนพารามิเตอร์ถึง $(14 \times 14 \times 480) \times (5 \times 5 \times 48) = 112.9$ ล้าน แต่ถ้าหากใช้เครอร์เนลขนาด $1 \times 1 \times 16$ มาคั่นระหว่างสองชั้นนี้จะทำให้จำนวนพารามิเตอร์ถูกย่อเป็น $(14 \times 14 \times 480) \times (1 \times 1 \times 16) = 1.5$ ล้าน และผลลัพธ์ของชั้นนี้จะมีขนาด $14 \times 14 \times 16$ ก่อนจะนำไปคำนวณในชั้นถัดไป $(14 \times 14 \times 16) \times (5 \times 5 \times 48) = 3.8$ ล้าน เมื่อนำจำนวนพารามิเตอร์รวมกันจะได้พารามิเตอร์เพียง $3.8 + 1.5 = 5.3$ ล้านเท่านั้น ซึ่งน้อยกว่าการใช้เครอร์เนลขนาด $5 \times 5 \times 48$ โดยตรง ซึ่งทำให้การพัฒนาโมเดลนั้นเป็นไปได้เร็วขึ้นมาก ทั้งยังสามารถลดปัญหาการเกิด overfit ได้ด้วย ประสิทธิภาพของโมเดล I3D แบบ two-stream เมื่อเทียบกับ long-short term memory (LSTM), 3D-convolution network, two-stream และ 3D-fused โดยใช้เครื่องมือในการวัดผลคือ Top@1 accuracy ตามตารางที่ 2.7

Architecture	UCF-101			HMDB-51			Kinetics		
	RGB	Flow	RGB + Flow	RGB	Flow	RGB + Flow	RGB	Flow	RGB + Flow
LSTM	81.0	–	–	36.0	–	–	63.3	–	–
3D-ConvNet	51.6	–	–	24.3	–	–	56.1	–	–
Two-Stream	83.6	85.6	91.2	43.2	56.3	58.3	62.2	52.4	65.6
3D-Fused	83.2	85.8	89.3	49.2	55.5	56.8	–	–	67.2
Two-Stream I3D	84.5	90.6	93.4	49.8	61.9	66.4	71.1	63.4	74.2

ตารางที่ 2.7: ประสิทธิภาพของโมเดล I3D แบบ two-stream เมื่อใช้ข้อมูลจาก UCF-101, HMDB-51 และ Kinetics ในการสร้างและทดสอบด้วยเครื่องมือวัดผลแบบความแม่นยำจากการทำงานอยันต์บีแรกสุด

2.4.3 เครื่องมือกำกับคุณลักษณะ

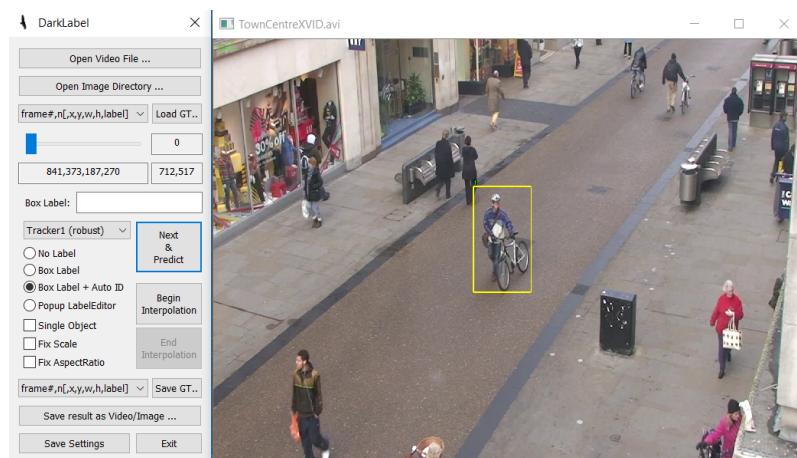
จากการค้นคว้าหาเครื่องมือในการสร้างคำกำกับข้อมูลเพื่อใช้เป็นแนวทางในการออกแบบเครื่องมือกำกับข้อมูลด้วยปัญญาประดิษฐ์ พบรเครื่องมือที่เปิดให้ใช้งานสาธารณะ (open source) 2 โปรแกรม คือ DarkLabel และ OpenLabeling โดยสรุปข้อมูลสำคัญได้ดังนี้

โปรแกรม DarkLabel^[3]

เป็นโปรแกรมที่ช่วยในการทำนายคำกำกับและบันทึกในรูปแบบต่างๆ รองรับข้อมูลป้อนเข้าในรูปแบบไฟล์วิดีโอ avi, mp4 หรือกลุ่มรูปภาพ มีขั้นตอนการสร้างคำกำกับดังนี้

1. สร้างกรอบสี่เหลี่ยมครอบบริเวณวัตถุที่สนใจโดยใช้มนุษย์เป็นคนสร้าง
2. กดปุ่ม Next และ Predict อย่างต่อเนื่อง เพื่อทำนายตำแหน่งต่อไปของกรอบสี่เหลี่ยมในเฟรมถัดๆไป จนกระทั่งการเกิดข้อผิดพลาด
3. ลบกรอบสี่เหลี่ยมที่พลาด และเริ่มทำขั้นตอนที่ 1 ใหม่ อีกรังสีจักรบทุกเฟรมในวิดีโอ

หลังจากที่ผู้วิจัยได้ทดลองใช้โปรแกรม DarkLabel พบร่วมกับโปรแกรมมีการทำงานส่วนใหญ่ใช้มนุษย์ในการทำด้วยตัวเอง ซึ่งทำให้ใช้เวลาในการทำงาน



รูปที่ 2.21: หน้าต่างการทำงานของโปรแกรม DarkLabel

โปรแกรม OpenLabeling^[8]

เป็นโปรแกรมที่ช่วยในการสร้างคำจำกัด โดยโปรแกรมจะมีการทำงานอยู่ 2 รูปแบบการทำงาน คือแบบทำด้วยตัวเอง (Mode Manual) และแบบอัตโนมัติ (Mode Auto) ซึ่งมีการทำงานแยกกันอย่างชัดเจน

1. การทำงานแบบอัตโนมัติ

หลังจากป้อนวิดีโอเข้าไปในโปรแกรมแล้วมีขั้นตอนการสร้างคำจำกัดดังนี้

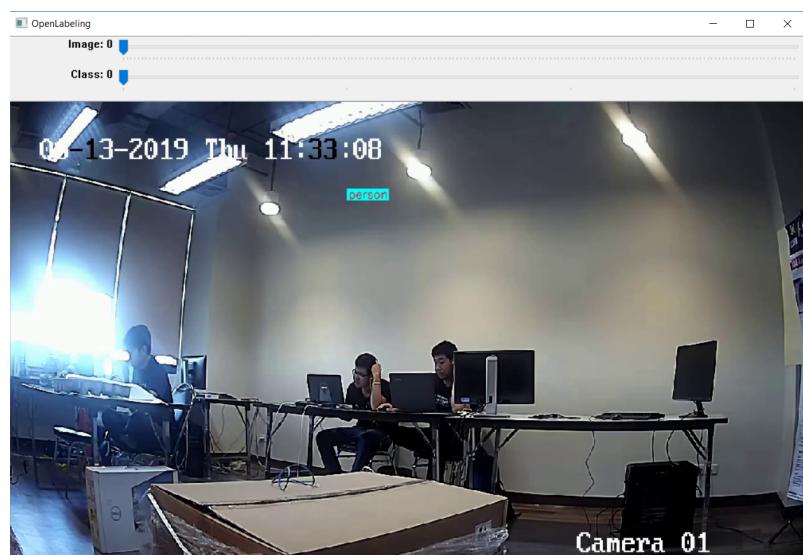
- (a) โปรแกรมจะทำงานอัตโนมัติโดยใช้โมเดลปัญญาประดิษฐ์ในการนำทางคีย์เฟรม (keyframe) และนำทางตำแหน่งต่อไปของกรอบสี่เหลี่ยมในเฟรมถัดไปด้วยอัลกอริทึมที่ใช้การคำนวนคณิตศาสตร์และการประมวลผลภาพในภาพที่เหลือ ผลลัพธ์ที่ได้คือรูปภาพและไฟล์คำจำกัดภาพ

2. การทำงานแบบทำด้วยตัวเอง

หลังจากป้อนวิดีโอเข้าไปในโปรแกรมแล้วมีขั้นตอนการสร้างคำจำกัดดังนี้

- (a) สร้างกรอบสี่เหลี่ยมขึ้นมาโดยใช้มนุษย์เป็นคนสร้าง
- (b) กดปุ่มเพื่อนำทางตำแหน่งต่อไปของกรอบสี่เหลี่ยมในเฟรมถัดไป จนกระทั่งเกิดข้อผิดพลาด
- (c) ลบกรอบสี่เหลี่ยมที่พลาด และเริ่มทำขั้นตอนที่ 1 อีกครั้งจนครบทุกเฟรมในวิดีโอ

หลังจากที่ได้ทดลองใช้โปรแกรม OpenLabeling ทั้ง 2 รูปแบบการทำงานแล้วพบว่า การทำงานแบบอัตโนมัติไม่สามารถปรับแก้ไขสิ่งใดในระหว่างกระบวนการนั้น ทำให้หากเกิดกรณีที่โมเดลนำทางกรอบสี่เหลี่ยมพลาดจะไม่สามารถแก้ไขได้ และการทำงานแบบทำด้วยตัวเองผู้ใช้งานจะต้องสร้างกรอบสี่เหลี่ยมขึ้นมาเอง



รูปที่ 2.22: หน้าต่างการทำงานของโปรแกรม OpenLabeling

บทที่ 3

ระเบียบวิธีวิจัย

ในการทำโครงการวิจัยเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ จะมีการทำงานหลากหลายส่วนมาทำงานร่วมกัน ซึ่งต้องมีระเบียบวิธีวิจัยอธิบายถึงขั้นตอนการดำเนินงานตั้งแต่เริ่มศึกษาข้อมูลจนไปถึงสิ้นสุดกระบวนการวิจัย

3.1 ความต้องการของระบบ

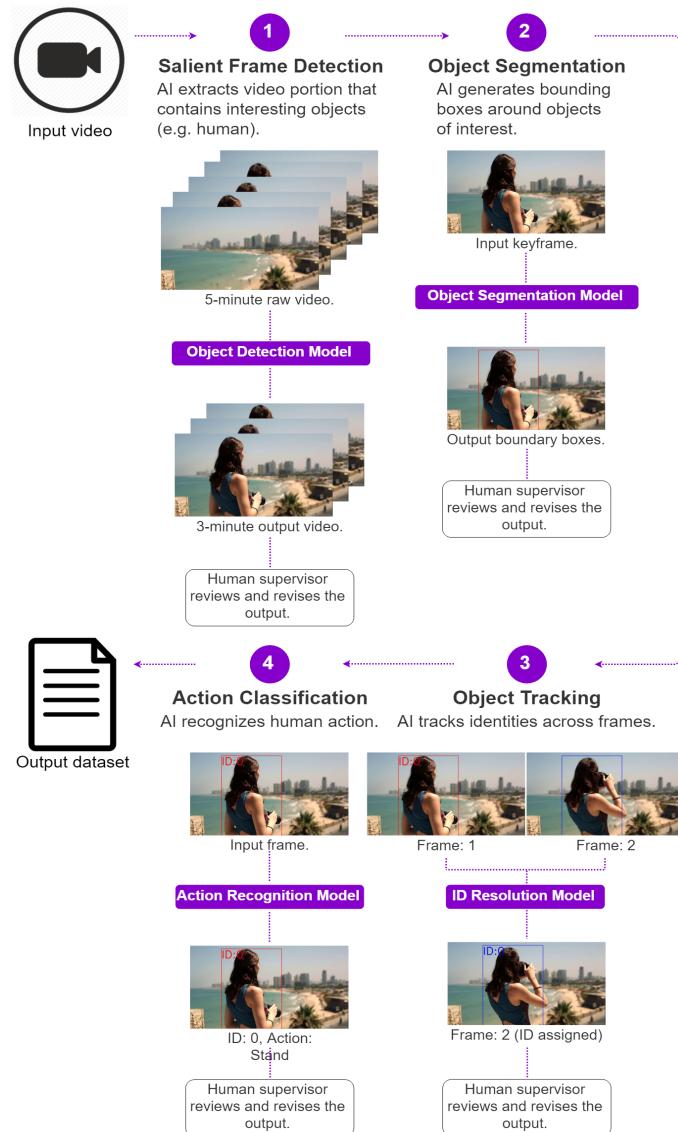
3.1.1 ความต้องการใช้งาน (functional requirements)

1. เครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ต้องสามารถตัดวิดีโอช่วงเวลาที่ไม่มีมนุษย์อยู่ออกได้ อัตโนมัติโดยใช้ปัญญาประดิษฐ์
2. เครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์สามารถระบุตำแหน่งมนุษย์แต่ละคนในวิดีโอด้วย การกระทำของมนุษย์ในวิดีโอได้ โดยการกระทำที่กำหนดจะประกอบไปด้วย ยืน นั่ง นอน เล่นโทรศัพท์ เดิน กินข้าว พูดคุย
3. ชุดข้อมูลที่ได้จากการเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ต้องสามารถนำไปใช้ในการพัฒนาโมเดล ปัญญาประดิษฐ์ต่อได้
4. สร้างระบบต้นแบบของเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ที่มีมนุษย์สามารถทำงานร่วมกับ ปัญญาประดิษฐ์ได้
5. ระบบวิเคราะห์การกระทำมนุษย์ต้องสามารถนำวิดีโอมาวิเคราะห์ข้อมูลการกระทำและตำแหน่งของ มนุษย์แต่ละคน แล้วนำข้อมูลเหล่านั้นไปสร้างรายงานที่มีคำกำกับอุปกรณ์ได้ โดยรายละเอียดรายงาน จะมีดังนี้
 - (a) เวลา (time stamp)
 - (b) การกระทำ ซึ่งประกอบไปด้วย ยืน นั่ง เดิน เล่นโทรศัพท์ กินข้าว นอน
 - (c) ตำแหน่ง โดยจะบอกในลักษณะของกรอบสี่เหลี่ยมครอบพื้นที่ที่มนุษย์คนนั้นๆอยู่

3.1.2 ความต้องการเชิงวิศวกรรม (non-functional requirements)

1. สร้างเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์โดยใช้ภาษาเพthon
2. ความละเอียดอย่างต่ำของวิดีโอต้องมากกว่า 640×480 (กว้าง x สูง)
3. วิดีโожะต้องมีอัตราเฟรมต่อวินาที (fps) อย่างต่ำ 10 เฟรมต่อวินาที

3.2 ภาพรวมระบบของเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์



รูปที่ 3.1: ภาพรวมระบบของเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์

3.3 หน้าที่ความรับผิดชอบ

ปฐมพงศ์ สินธุ์งาม สร้างและทดสอบโมเดลปัญญาประดิษฐ์สำหรับจดจำการกระทำมนุษย์ 3D รวมถึงออกแบบและสร้างระบบ Tracker

ศุภกร เบญจวิกรัย รวมฟังก์ชันและระบบต่างๆของเครื่องมือ รวมถึงออกแบบและสร้างระบบ Select และ Detect

อุกฤษฎ์ เลิศวรรณาการ สร้างและทดสอบโมเดลปัญญาประดิษฐ์สำหรับจดจำการกระทำมนุษย์ Resnet-50 รวมถึงออกแบบและสร้างระบบ Person ReID

3.4 เครื่องมือที่ใช้ในงานวิจัย

ในหัวข้อนี้จะกล่าวถึงซอฟต์แวร์ ภาษา และ program library ที่ใช้ในการพัฒนาระบบ รวมถึงข้อมูลจำเพาะของคอมพิวเตอร์ที่ใช้ในการพัฒนาระบบ

Pycharm community 2017.1.2

เป็นโปรแกรมໄ่วยใช้สำหรับเขียนและแก้ไขโค้ดซึ่งข้อดีของโปรแกรมนี้ คือ มีคุณสมบัติต่างๆที่สามารถอำนวยความสะดวกในการเขียนโปรแกรมได้ เช่น syntax highlighting, auto-completion ฯลฯ และสามารถประมวลผล (compile) โปรแกรมทดสอบแอพพลิเคชันได้

Jupyter 2017.1.2

เป็นโปรแกรมสำหรับเขียนโปรแกรมที่เหมาะสมสำหรับใช้ในการทดสอบโปรแกรมแต่ละส่วนได้ ซึ่งมีข้อดีคือ หากมีการแก้ไขโปรแกรมเพียงแค่บางส่วน ก็สามารถปรับมาผลเฉพาะส่วนที่ต้องการได้มักจะใช้ในการสร้างโมเดลปัญญาประดิษฐ์

Qt Creator 4.9.2 (community)

เป็นเครื่องมือสำหรับออกแบบหน้าต่างแอพพลิเคชันของ library PyQt ซึ่งมีข้อดีคือ เรียกใช้ง่ายมีวิดเจ็ต (widget) ที่สามารถใช้ได้หลากหลายเหมาะสมสำหรับการออกแบบ

3.5 ภาษาที่ใช้ในการพัฒนาระบบ

ใช้ภาษาไพธอนในการพัฒนาเป็นหลัก เพราะเป็นภาษาที่ปัจจุบันมีการใช้กันอย่างแพร่ มีเครื่องมือและ library ที่อำนวยความสะดวกในการพัฒนาอย่างมาก ทั้งยังเป็นภาษาที่สามารถเข้าใจได้ง่าย โดยในการทำวิจัยครั้งนี้ได้เลือก python 3.6.8 มาใช้ในการพัฒนา เนื่องจากเป็นรุ่นที่รองรับการทำงานของ library Tensorflow 1.12 และ CUDA 9

3.6 Program library ที่ใช้ในการพัฒนาระบบและแอพพลิเคชั่น

Library	Version	Description
numpy	1.16.4	library ใช้สำหรับการคำนวณและ array
pandas	0.24.2	library ใช้สำหรับการจัดการข้อมูลที่อยู่ในรูปแบบของ excel
opencv	4.1.0.25	library ใช้สำหรับการจัดการข้อมูลที่เป็นรูปภาพและวิดีโอ
pillow	6.0.0	library ใช้สำหรับการจัดการข้อมูลที่เป็นรูปภาพ
torchsummary	1.5.1	library ใช้สำหรับการวิเคราะห์โครงสร้างของโมเดล
pytorch	1.10.0	library ใช้สำหรับการสร้างปัญญาประดิษฐ์
torchvision	0.3.0	library ใช้สำหรับการสร้างปัญญาประดิษฐ์
scikit-learn	0.21.2	library ใช้สำหรับการสร้างปัญญาประดิษฐ์
scipy	1.3.0	library ใช้สำหรับการสร้างปัญญาประดิษฐ์
sklearn	0.0	library ใช้สำหรับการสร้างปัญญาประดิษฐ์
pickleshare	0.7.5	library ใช้สำหรับการทำรหัส (encoding) โมเดลปัญญาประดิษฐ์
tqdm	4.32.1	library ใช้สำหรับจัดการการทำงานซ้ำ (loop)
pyqt5	5.9.2	library ใช้สำหรับการทำแอพพลิเคชั่น

3.7 แผนการดำเนินงาน

โดยจากที่กล่าวไปตอนต้นในบทนำการดำเนินงานและการออกแบบการสร้างเครื่องมือกำกับคุณลักษณะด้วยปัญญาประดิษฐ์ และระบบวิเคราะห์การกระทำการของมนุษย์ในวิดีโอ มีแผนการทำงานซึ่งถูกแบ่งออกเป็นสามขั้นตอนดังนี้

- ศึกษาความเป็นไปได้ รวมถึงเทคโนโลยีในปัจจุบันที่เกี่ยวกับการสร้างแอพพลิเคชั่น และการจำแนกการกระทำการของมนุษย์ด้วยปัญญาประดิษฐ์ เพื่อศึกษาและทำความเข้าใจ ข้อดี-ข้อเสีย ของเทคนิคหรือกระบวนการต่างๆ เพื่อนำมาประยุกต์ใช้กับงานวิจัยนี้
- ออกแบบและสร้างแอพพลิเคชั่นที่ใช้ในการสร้างชุดข้อมูลสำหรับการ trenโมเดลจากวิดีโอ
- ออกแบบและสร้างระบบวิเคราะห์การกระทำการของมนุษย์ได้โดยมีข้อกำหนดตามที่กล่าวไว้ในบทนำ

ในการศึกษาเกี่ยวกับการออกแบบและการสร้างแอพพลิเคชันที่ใช้ในการสร้างชุดข้อมูลสำหรับการสร้างโมเดลจากวิดีโอ สิ่งที่ต้องให้ความสนใจคือฟังก์ชันการทำงาน การออกแบบและการจัดวางองค์ประกอบต่างๆ ในหน้าต่างแอพพลิเคชัน และความสะดวกในการใช้งาน จากนั้นจึงเริ่มศึกษาเกี่ยวกับ library ที่ใช้ในการสร้างแอพพลิเคชัน ส่วนการศึกษาเกี่ยวกับการสร้างระบบบุคลิกภาพที่การกระทำมนุษย์ จะมุ่งความสนใจไปที่ชุดข้อมูลสำหรับการบุคลิกภาพวิดีโอ โมเดลสำหรับการบุคลิกภาพวิดีโอ เทคนิคในการสร้างโมเดล เทคโนโลยีในการทำงานบุคลิกภาพ ในบทนี้จะกล่าวถึงกระบวนการออกแบบและการดำเนินการตามแผนที่วางแผนไว้

3.8 การออกแบบหน้าต่างแอพพลิเคชันของเครื่องมือสำหรับคุณลักษณะด้วยปัญญาประดิษฐ์

การออกแบบเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์ ผู้วิจัยได้เลือกใช้ library PyQt และภาษา Python ในการพัฒนา เนื่องจาก PyQt นั้นเป็น library ที่มีผู้พัฒนาใช้กันอย่างแพร่หลาย จึงสะดวกในการศึกษา หาข้อมูลในการสร้างหรือแก้ไข อีกทั้งยังเป็น library ที่สามารถพัฒนาด้วยภาษา Python ได้ และใช้งานง่าย สามารถปรับปรุงแก้ไขได้สะดวก

3.8.1 เครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์

แอพพลิเคชันแบ่งการทำงานออกเป็นสี่ส่วนประกอบด้วยกระบวนการ Select, Detect, Track และ Label เพื่อช่วยแบ่งเบาภาระของผู้พัฒนาในการสร้างชุดข้อมูลสำหรับสร้างโมเดลจากข้อมูลประเภทวิดีโอ โดยกระบวนการ Select จะต้องสามารถตัดวิดีโอด้วยส่วนที่ไม่มีมนุษย์อยู่ออกจากวิดีโอด้วย กระบวนการ Detect จะต้องหาตำแหน่งของมนุษย์ภายในวิดีโอด้วย แล้วใช้กระบวนการ Track นำรายตำแหน่งต่อไปของมนุษย์ข้อมูล ตำแหน่งของมนุษย์ที่ได้จากการกระบวนการ Detect และกระบวนการ Label นั้นต้องสามารถทำการกระทำพื้นฐานของมนุษย์ได้ เช่น ยืน เดิน นั่ง กินข้าว หรือ นอน เป็นต้น โดยทุกส่วนการทำงานมนุษย์ต้องสามารถทำงานร่วมกับปัญญาประดิษฐ์ได้ ดังรูปที่ 3.2

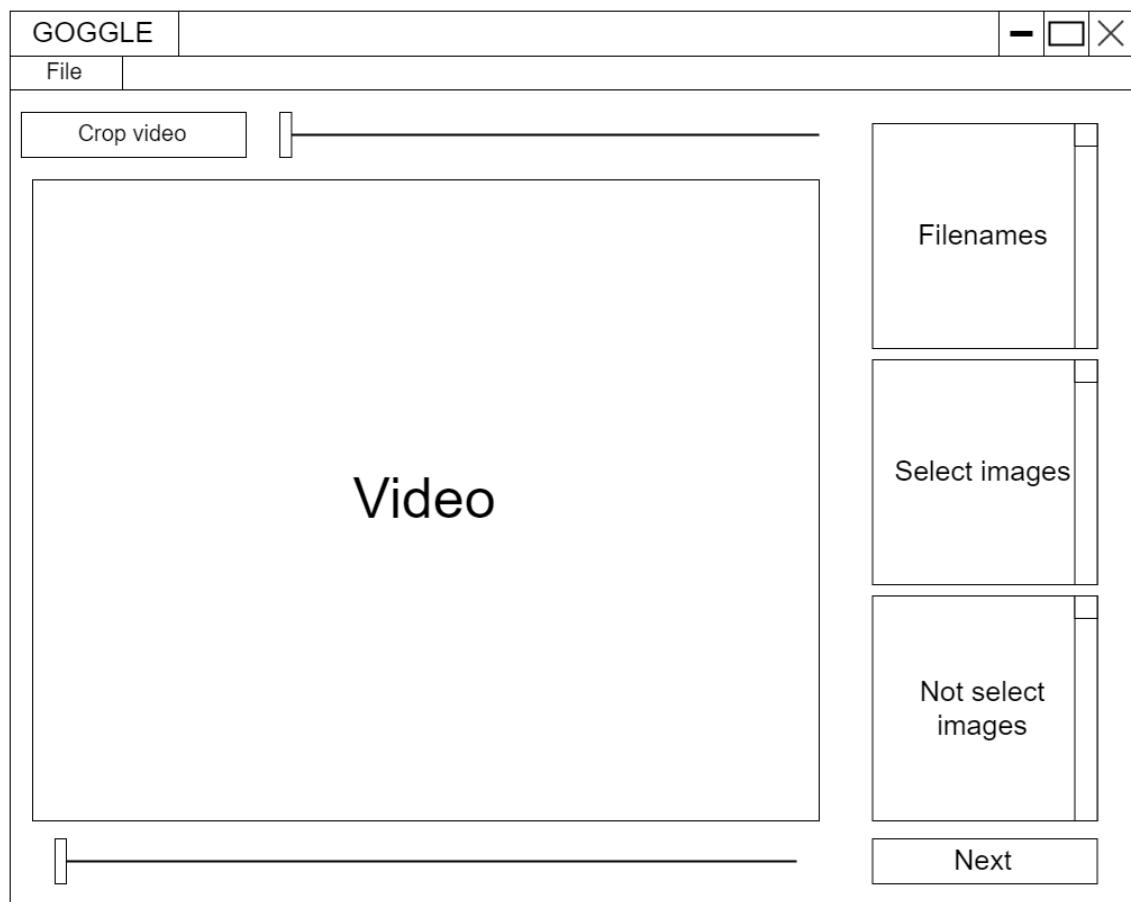


รูปที่ 3.2: กระบวนการหลักของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์

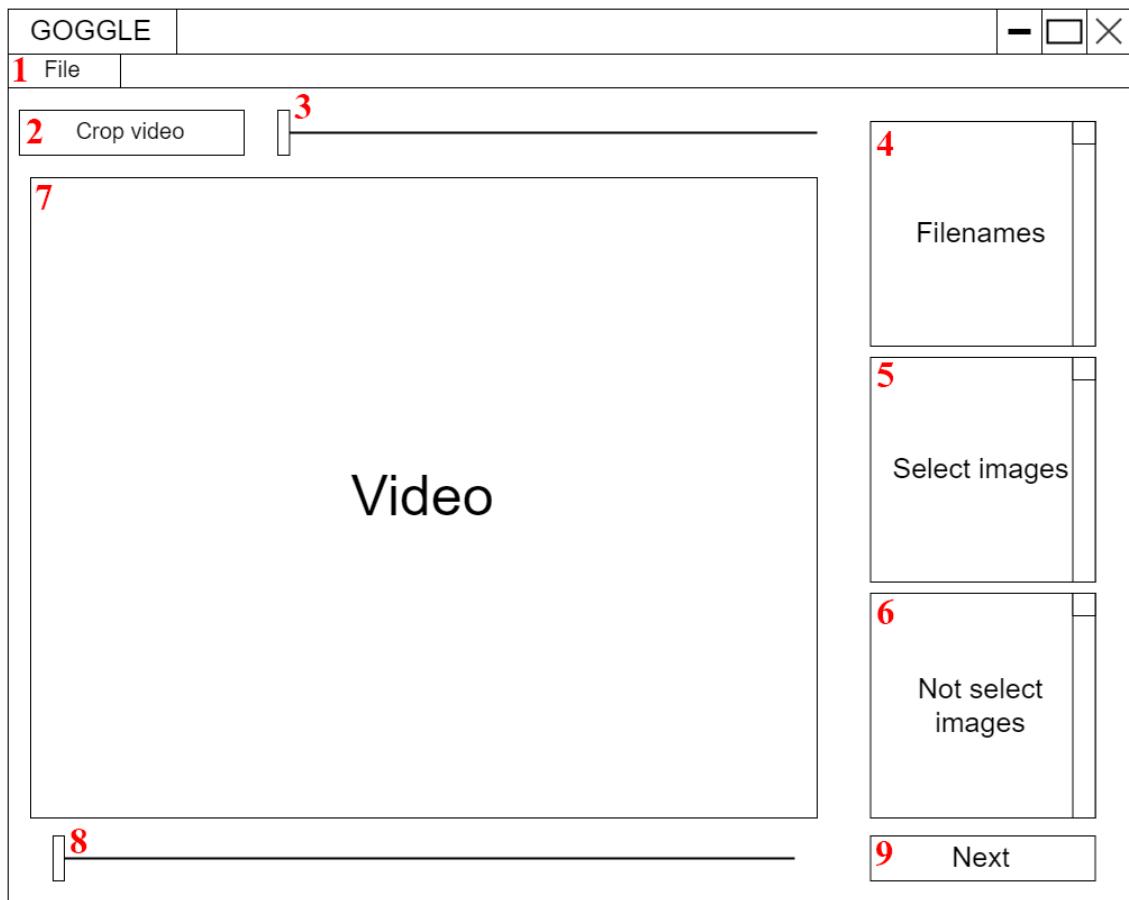
โดยแต่ละกระบวนการจะมีรายละเอียดดังนี้

3.8.1.1 Select

กระบวนการ Select จะต้องสามารถรับวิดีโอเข้ามา แล้วตัดวิดีโອนในช่วงที่ไม่มีมนุษย์อยู่ในเฟรมออกได้ อัตโนมัติด้วยปัญญาประดิษฐ์ แต่เนื่องจากการประมวลผลทุกเฟรมในวิดีโอนั้นจะทำให้เสียเวลามากเกินไป จึงใช้วิธีการเลือกตัวอย่างเฟรมตัวอย่างตราชกที่ (สามารถกำหนดได้) ซึ่งเรียกว่าเฟรมเหล่านี้ว่า คีย์เฟรม (keyframe) จากนั้นใช้ปัญญาประดิษฐ์ประมวลผลคีย์เฟรมที่เหล่านั้น เพื่อลดระยะเวลาในการประมวลผลลง และมนุษย์จะต้องสามารถแก้ไขข้อผิดพลาดของปัญญาประดิษฐ์ได้ เพื่อเพิ่มคุณภาพของชุดข้อมูล จึงออกแบบหน้าต่างได้ดังรูปที่ 3.3



รูปที่ 3.3: หน้าต่าง Select ของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์



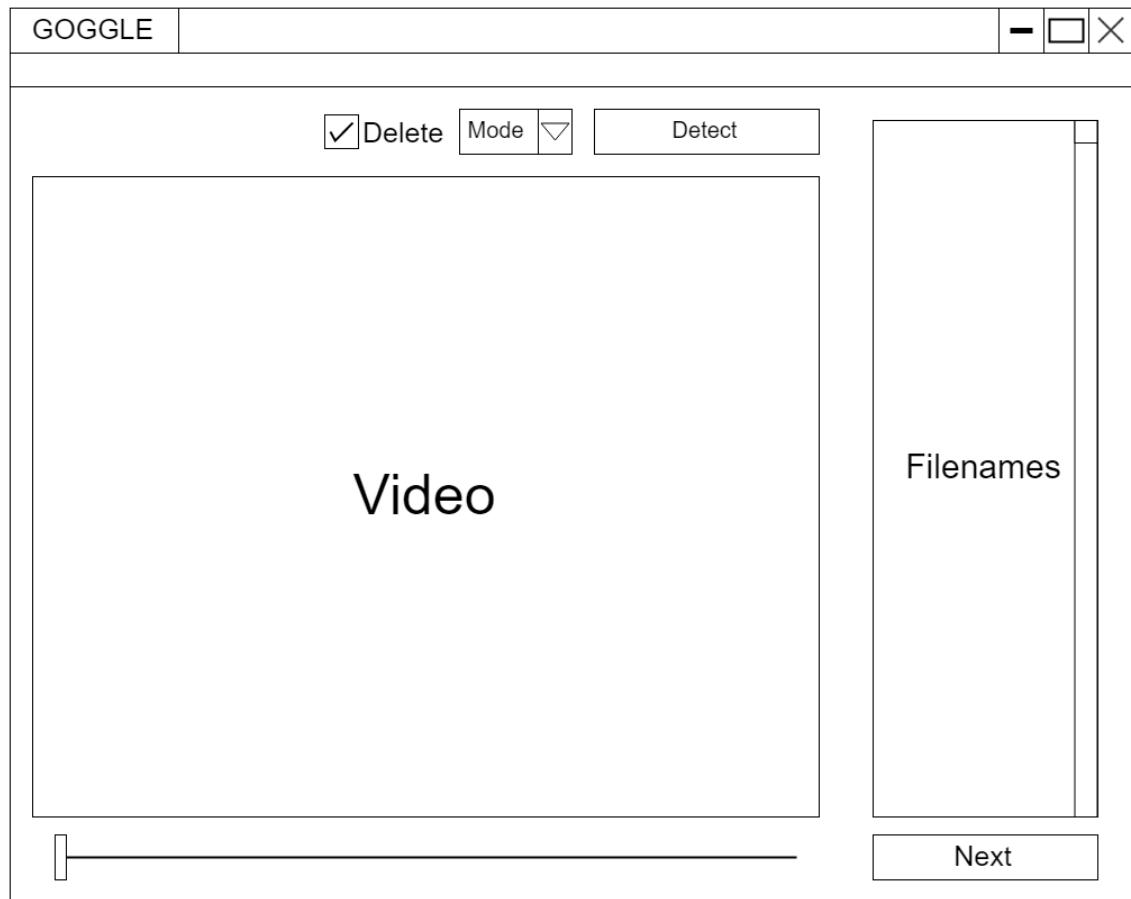
รูปที่ 3.4: ตำแหน่งของแต่ละวิดเจ็ตในหน้าต่าง Select

โดยที่แต่ละวิดเจ็ตตามหมายเลขที่กำหนดตามรูปที่ 3.4 มีรายละเอียดดังนี้

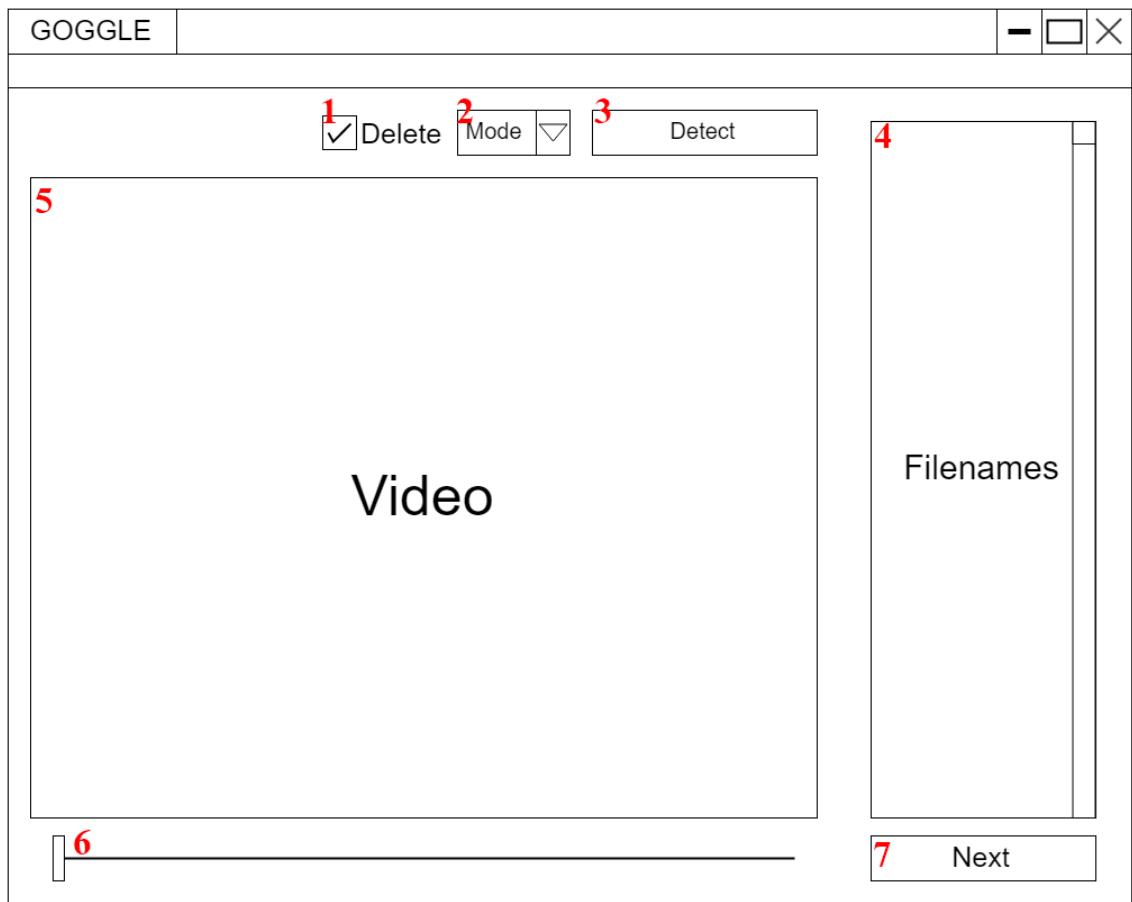
1. หมายเลข 1 คือปุ่มสำหรับเลือกไฟล์วิดีโอที่ต้องการจากในคอมพิวเตอร์เข้ามาในโปรแกรม
2. หมายเลข 2 คือปุ่มสำหรับสั่งให้ระบบทำการสร้างคีย์เฟรมขึ้นมา แล้วใช้ปัญญาประดิษฐ์ประมวลผลเพื่อแยกคีย์เฟรมในเมื่อน้อย และคีย์เฟรมในเมื่อน้อยแบบอัตโนมัติ (Auto mode)
3. หมายเลข 3 คือแถบเลื่อนเพื่อกำหนดความถี่ในการหยิบคีย์เฟรม โดยจะมีช่วงอยู่ที่ 1 เฟรมต่อวินาที จนถึงอัตราเฟรมต่อวินาทีสูงสุดของวิดีโอิที่รับเข้ามา
4. หมายเลข 4 คือกล่องสำหรับแสดงชื่อวิดีโอิที่รับเข้ามาในโปรแกรมเพื่อเลือกเข้ามาใช้ในการประมวลผล
5. หมายเลข 5 คือกล่องสำหรับแสดงว่าคีย์เฟรมได้มีมนุษย์อยู่ในเฟรม โดยที่ผู้ใช้งานสามารถตรวจสอบความถูกต้องและแก้ไขข้อผิดพลาดของปัญญาประดิษฐ์ได้
6. หมายเลข 6 คือกล่องสำหรับแสดงว่าคีย์เฟรมได้มีมนุษย์อยู่ในเฟรม โดยที่ผู้ใช้งานสามารถตรวจสอบความถูกต้องและแก้ไขข้อผิดพลาดของปัญญาประดิษฐ์ได้
7. หมายเลข 7 คือหน้าต่างสำหรับแสดงเฟรมที่เลือกจากหมายเลข 5 หมายเลข 6 หรือหมายเลข 8
8. หมายเลข 8 คือแถบเลื่อนสำหรับเลือนคุณคีย์เฟรมทั้งหมดที่ระบบสร้างขึ้น
9. หมายเลข 9 คือปุ่มสำหรับไปกระบวนการต่อไปหลังจากระบบประมวลผลเสร็จแล้ว

3.8.1.2 Detect

กระบวนการ Detect จะต้องสามารถรับคีย์เฟรมจากการบันทึกการ Select มาประมวลผลด้วยปัญญาประดิษฐ์เพื่อหาตำแหน่งของมนุษย์ที่อยู่ในคีย์เฟรม และสร้างกรอบสีเหลี่ยมครอบบริเวณดังกล่าวได้ในแบบอัตโนมัติ เพื่อแบ่งเบาภาระผู้ใช้ในการที่ต้องสร้างกรอบสีเหลี่ยมครอบตำแหน่งของมนุษย์ด้วยตัวเอง และผู้ใช้ต้องสามารถสร้างหรือลบกรอบสีเหลี่ยมได้ด้วยตัวเองสำหรับแก้ไขความผิดพลาดของปัญญาประดิษฐ์ เพื่อเพิ่มคุณภาพของชุดข้อมูล จึงออกแบบหน้าต่างได้ดังรูปที่ 3.5



รูปที่ 3.5: หน้าต่าง Detect ของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์



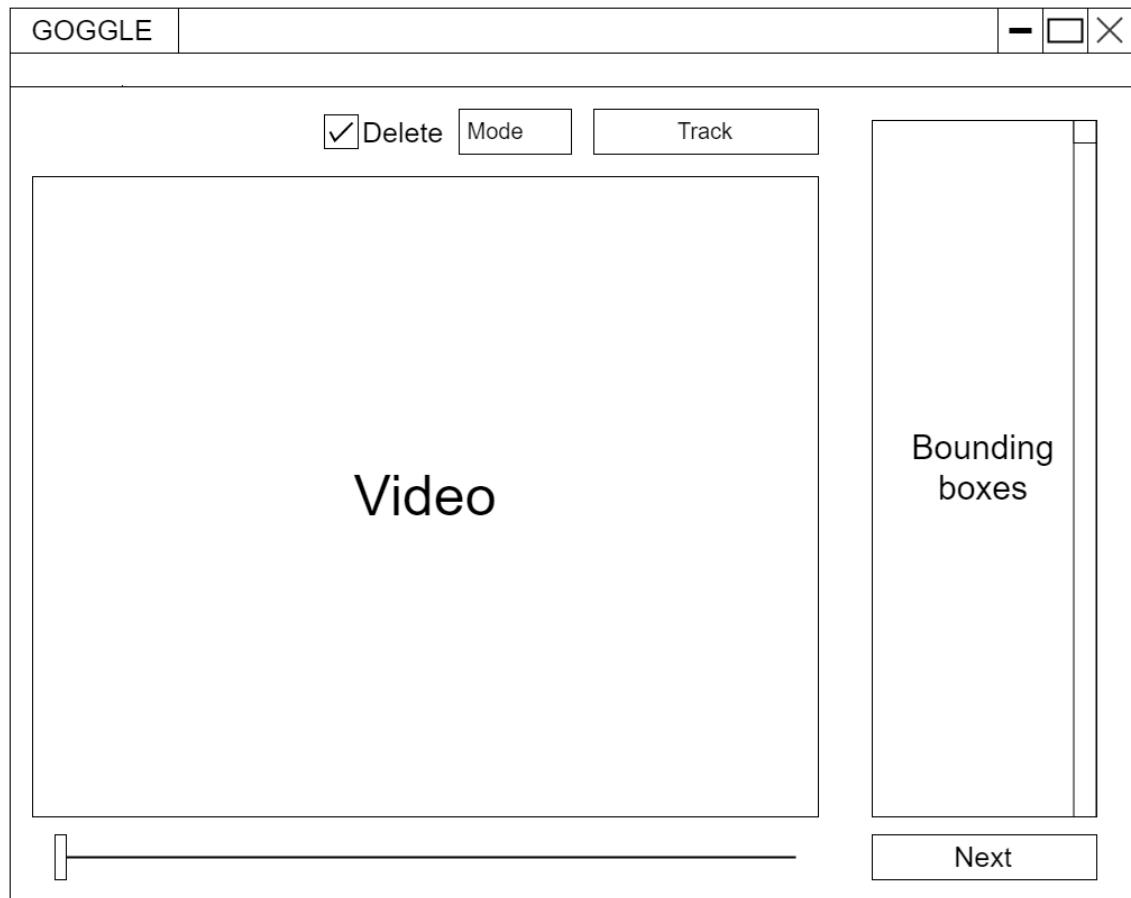
รูปที่ 3.6: ตำแหน่งของแต่ละวิดเจ็ตในหน้าต่าง Detect

โดยที่แต่ละวิดเจ็ตตามหมายเลขที่กำหนดตามรูปที่ 3.6 มีรายละเอียดดังนี้

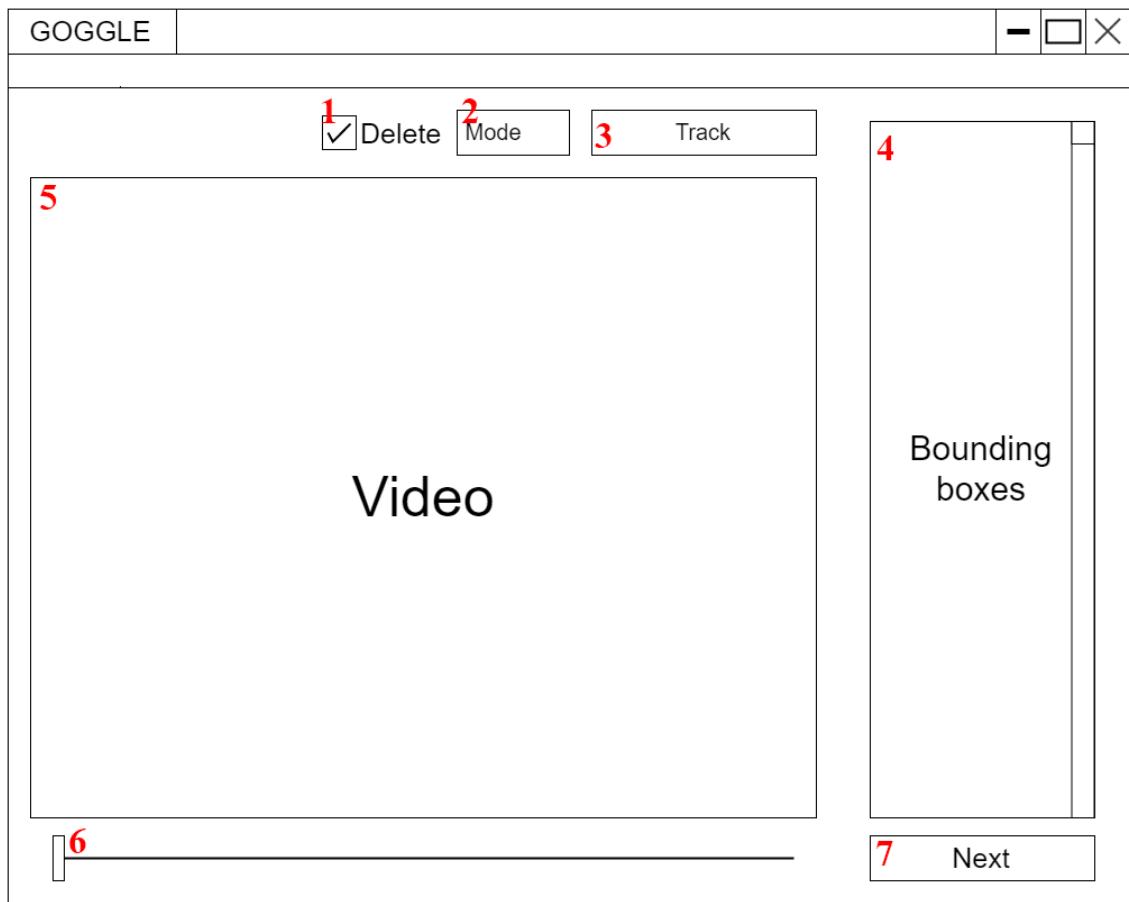
1. หมายเลข 1 คือช่องสำหรับกดเพื่อเปลี่ยนระบบจากสร้างกรอบสี่เหลี่ยมในแบบแก้ไขด้วยตนเอง (Manual mode) เป็นลบรอบสี่เหลี่ยมแทน
2. หมายเลข 2 คือช่องสำหรับเลือกว่าจะใช้ระบบแบบใด ระหว่างแบบอัตโนมัติและแบบแก้ไขด้วยตนเอง
3. หมายเลข 3 คือปุ่มสำหรับสั่งให้ระบบทำการตรวจหาตำแหน่งของมนุษย์ในคิร์เฟรมทั้งหมดแล้วสร้างกรอบสี่เหลี่ยมขึ้นมาครอบบริเวณที่กำหนด
4. หมายเลข 4 คือกล่องสำหรับแสดงคิร์เฟรมทั้งหมด
5. หมายเลข 5 คือหน้าต่างสำหรับแสดงเฟรมที่เลือกจากหมายเลข 4 หรือหมายเลข 6
6. หมายเลข 6 คือแบบเลื่อนสำหรับเลื่อนดูคิร์เฟรมทั้งหมดที่มี เพื่อตรวจสอบความถูกต้องของปัญญาประดิษฐ์
7. หมายเลข 7 คือปุ่มสำหรับไปกระบวนการต่อไปหลังจากระบบประมวลผลเสร็จแล้ว

3.8.1.3 Track

เนื่องจากกระบวนการ Detect นั้นจะทำเฉพาะในคีย์เฟรมทำให้ในเฟรมอื่นๆ นอกเหนือจากนั้นจะไม่มีกรอบสี่เหลี่ยมอยู่ ดังนั้นกระบวนการ Track จึงต้องสามารถทำนายตำแหน่งต่อไปของมนุษย์แล้วสร้างกรอบสี่เหลี่ยมขึ้นมาบนเฟรมระหว่างคีย์เฟรมทั้งหมดได้โดยอัตโนมัติ เพื่อสร้างข้อมูลตำแหน่งของมนุษย์ในเฟรมเหล่านั้น และผู้ใช้ต้องสามารถสร้างหรือลบกรอบสี่เหลี่ยมได้ด้วยตัวเองสำหรับแก้ไขความผิดพลาดของอัลกอริทึม จึงออกแบบหน้าต่างได้ดังรูปที่ 3.7



รูปที่ 3.7: หน้าต่าง Track ของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์



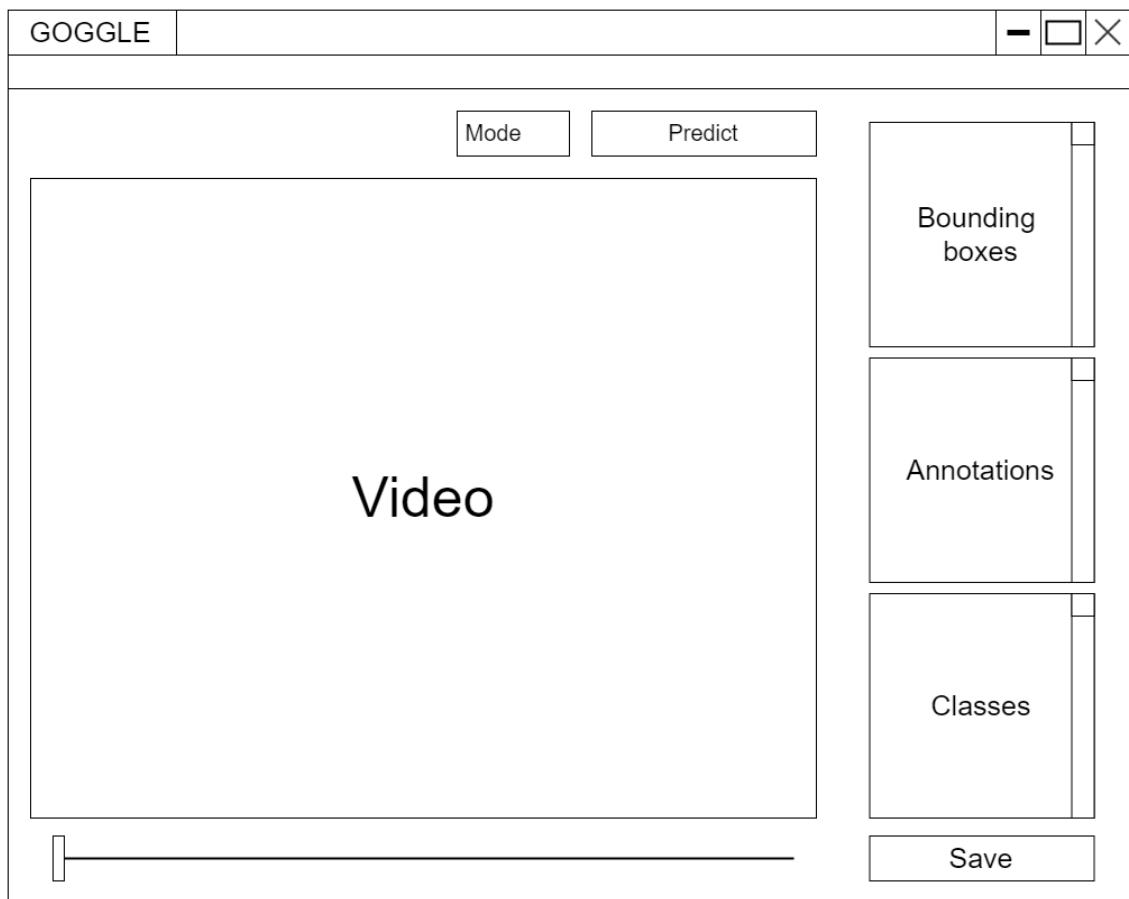
รูปที่ 3.8: ตำแหน่งของแต่ละวิดเจ็ตในหน้าต่าง Track

โดยที่แต่ละวิดเจ็ตตามหมายเลขที่กำหนดตามรูปที่ 3.8 มีรายละเอียดดังนี้

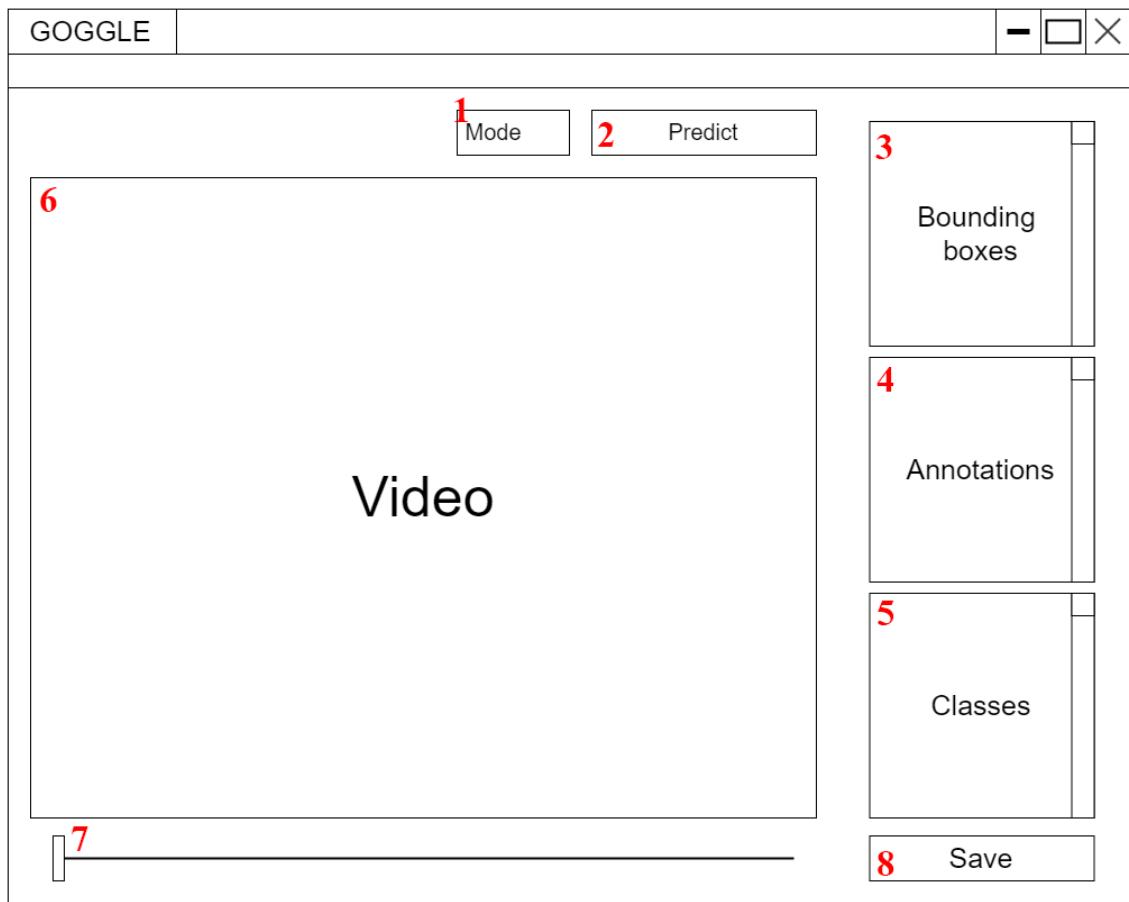
1. หมายเลข 1 คือช่องสำหรับกดเพื่อเปลี่ยนระบบจากสร้างกรอบสี่เหลี่ยมในแบบแก้ไขด้วยตนเองเป็นลบกรอบสี่เหลี่ยมแทน
2. หมายเลข 2 คือช่องสำหรับเลือกว่าจะใช้ระบบแบบใด ระหว่างแบบอัตโนมัติและแบบแก้ไขด้วยตนเอง
3. หมายเลข 3 คือปุ่มสำหรับสั่งให้ระบบทำการตรวจสอบตำแหน่งของมนุษย์ในเฟรมระหว่างคิ้ยวเฟรมทั้งหมดแล้วสร้างกรอบสี่เหลี่ยมขึ้นมาครอบบริเวณที่กำหนด
4. หมายเลข 4 คือกล่องสำหรับแสดงกรอบสี่เหลี่ยมทั้งหมดที่อยู่ในเฟรม
5. หมายเลข 5 คือหน้าต่างสำหรับแสดงเฟรมที่เลือกจากหมายเลข 6
6. หมายเลข 6 คือแถบเลื่อนสำหรับเลื่อนดูเฟรมทั้งหมดที่มี เพื่อตรวจสอบความถูกต้องของอัลกอริทึม
7. หมายเลข 7 คือปุ่มสำหรับไปกระบวนการต่อไปหลังจากระบบประมวลผลเสร็จแล้ว

3.8.1.4 Label

กระบวนการ Label นั้นต้องสามารถทำนายว่าการกระทำของมนุษย์ที่อยู่ในแต่ละเฟรมว่าคืออะไรได้โดยอัตโนมัติด้วยปัญญาประดิษฐ์ และผู้ใช้จะต้องสามารถแก้ไขข้อผิดพลาดของปัญญาประดิษฐ์ได้หากมีการทำนายที่ผิดพลาดเกิดขึ้น หรือถ้าหากผู้ใช้ต้องการเพิ่มการกระทำที่ไม่ได้มีอยู่ในชุดการกระทำพื้นฐานที่มีอยู่แล้วของปัญญาประดิษฐ์ ผู้ใช้ก็สามารถเพิ่มการกระทำนั้นเข้ามาได้ จึงออกแบบหน้าต่างเดดงรูปที่ 3.9



รูปที่ 3.9: หน้าต่าง Label ของเครื่องมือสำหรับกำกับข้อมูลด้วยปัญญาประดิษฐ์



รูปที่ 3.10: ตำแหน่งของแต่ละวิดเจ็ตในหน้าต่าง Label

โดยที่แต่ละวิดเจ็ตตามหมายเลขที่กำหนดตามรูปที่ 3.8 มีรายละเอียดดังนี้

1. หมายเลข 1 คือช่องสำหรับเลือกว่าจะใช้ระบบแบบใด ระหว่างแบบอัตโนมัติและแบบแก้ไขด้วยตนเอง
2. หมายเลข 2 คือปุ่มสำหรับสั่งให้ระบบคำนวณรูปแบบที่มีอยู่ในทุกๆเฟรม
3. หมายเลข 3 คือกล่องสำหรับแสดงกรอบสี่เหลี่ยมทั้งหมดที่มีอยู่ในเฟรมที่เลือก
4. หมายเลข 4 คือกล่องสำหรับแสดงการกระทำของมนุษย์แต่ละคนที่อยู่ในเฟรมที่เลือก โดยจะเรียงลำดับคู่กับกรอบสี่เหลี่ยมที่อยู่ในช่องหมายเลข 3
5. หมายเลข 5 คือกล่องสำหรับแสดงชุดการกระทำที่ปัญญาประดิษฐ์มีอยู่แล้ว ซึ่งในการทำงานแบบแก้ไขด้วยตนเองนั้น จะสามารถค้นหาการกระทำที่มีอยู่แล้วได้ และหากคำที่ใส่เขามานั้นมีอยู่ในชุดการกระทำก็จะเป็นการเพิ่มการกระทำนั้นเข้ามาแทน
6. หมายเลข 6 คือหน้าต่างสำหรับแสดงเฟรมที่เลือกจากหมายเลข 7
7. หมายเลข 7 คือແນບเลื่อนสำหรับเลื่อนดูเฟรมทั้งหมดที่มี เพื่อตรวจสอบความถูกต้องของปัญญาประดิษฐ์
8. หมายเลข 8 คือปุ่มสำหรับสร้างไฟล์ xml ของทุกๆเฟรมสำหรับใช้ในการสร้างโมเดลโดยรายละเอียดข้อมูลภายในไฟล์ xml จะอยู่ในหัวข้อ 3.8.1.5

3.8.1.5 รายละเอียดข้อมูลภายในไฟล์ xml

ไฟล์ xml นั้นเป็นรูปแบบที่นิยมใช้ในการเก็บข้อมูลสำหรับการสร้างโมเดลประเพณีตรวจสอบจับตัวตุ๊ก โดยจะเก็บข้อมูลในรูปแบบของ PASCAL VOC ที่นิยมใช้ในการสร้างโมเดลด้วย library Tensorflow โดยภายในไฟล์จะมีข้อมูลดังรูปที่ 3.11 โดยข้อมูลส่วนสำคัญของรูปแบบนี้นั้นจะถูกใส่หมายเลขอ้างอิงแต่ละหมายเลขนั้นหมาย

```

<annotation>
    <folder>GeneratedData_Train</folder>1
    <filename>000001.png</filename>2
    <path>/my/path/GeneratedData_Train/000001.png</path>3
    <source>
        <database>Unknown</database>
    </source>
    <size> 4
        <width>224</width>
        <height>224</height>
        <depth>3</depth>
    </size>
    <segmented>0</segmented>
    <object>
        <name>21</name> 5
        <pose>Frontal</pose>
        <truncated>0</truncated>
        <difficult>0</difficult>
        <occluded>0</occluded>
        <bndbox> 6
            <xmin>82</xmin>
            <xmax>172</xmax>
            <ymin>88</ymin>
            <ymax>146</ymax>
        </bndbox>
    </object>
</annotation>
```

รูปที่ 3.11: ตัวอย่างข้อมูลภายในไฟล์ xml

ถึง

1. หมายเลขอื่นๆ คือชื่อโฟลเดอร์ที่เก็บไฟล์รูปภาพที่เกี่ยวข้องกับไฟล์ xml นี้อยู่
2. หมายเลขอื่นๆ คือชื่อไฟล์ที่เกี่ยวข้องกับไฟล์ xml นี้
3. หมายเลขอื่นๆ คือเส้นทางในคอมพิวเตอร์ (directory path) ของไฟล์รูปภาพที่เกี่ยวข้องกับไฟล์ xml นี้
4. หมายเลขอื่นๆ คือขนาดและมิติของรูปภาพ ซึ่งจะประกอบด้วยความกว้าง (width) ความยาว (height) และจำนวนช่องสี (depth) โดยที่จำนวนช่องสีที่มีความลึก 3 มักจะหมายถึงภาพสี RGB และจำนวนช่องสีที่มีความลึก 2 จะหมายถึงภาพขาวดำ (gray scale)
5. หมายเลขอื่นๆ คือ label ของวัตถุหรืออย่างอื่น ที่อยู่ในกรอบสีเหลี่ยมที่ถูกกำหนดไว้ในส่วนของหมายเลขอื่นๆ
6. หมายเลขอื่นๆ คือ กรอบสีเหลี่ยมที่ครอบวัตถุที่สนใจ เช่นมนุษย์ เป็นต้น

3.9 การออกแบบการทดสอบการตรวจจับวัตถุ

3.9.1 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์สำหรับการทำการตรวจจับภาพบุคคล สิ่งที่ใช้ในการวัดผล

1. ความเร็วในการทำนายต่อรูปภาพ (มิลลิวินาที)
2. ความแม่นยำ โดยคำนึงถึงอัตราส่วนร่วมของกรอบที่เหลือ หรือ Intersection over Union (IoU)

จุดประสงค์

1. ผู้วิจัยได้ตั้งจุดประสงค์การทดลอง การใช้โมเดลปัญญาประดิษฐ์สำหรับตรวจจับวัตถุ เพื่อวัดผลโมเดล ปัญญาประดิษฐ์ที่ใช้ในปัจจุบัน และหาโมเดลปัญญาประดิษฐ์สำหรับการทำการตรวจจับวัตถุที่มีความเร็วมาก ที่สุดและแม่นยำสูงที่สุดเมื่อทดสอบกับชุดข้อมูลของผู้วิจัย

ตัวแปร

1. โมเดลปัญญาประดิษฐ์ ได้แก่
 - (a) SSD MobileNet v1 ppn
 - (b) YOLO-v3 tiny
 - (c) YOLO-v3 spp
 - (d) YOLO-v3 320
 - (e) Faster RCNN inception v2

ตัวแปรควบคุม

1. ชุดข้อมูล : ชุดข้อมูลสำหรับทดสอบวัดผลที่ผู้วิจัยสร้างขึ้น (สูม 20 เฟรมจากวิดีโอที่ผู้วิจัยใช้สำหรับสร้าง ชุดข้อมูล)

วิธีการทดลอง

1. แบ่งชุดข้อมูลออกเป็นชุดข้อมูลสำหรับทดสอบ และชุดข้อมูลที่มีคำกำกับเพื่อเป็นคำตอบ
 - (a) ชุดข้อมูลสำหรับทดสอบ ประกอบด้วย : ชื่อของวิดีโอ และเฟรม
 - (b) ชุดข้อมูลที่มีคำกำกับเพื่อเป็นคำตอบ ประกอบด้วย : ชื่อของวิดีโอ เฟรม และตำแหน่งของกรอบ สีเหลี่ยม
2. เรียกชื่อและเฟรมของวิดีโอด้วยชุดข้อมูลทดสอบ และนำโมเดลปัญญาประดิษฐ์ทำนายผลลัพธ์ จากนั้น เก็บผลลัพธ์เป็นชุดข้อมูลผลลัพธ์จากการทำนาย
 - (a) ชุดข้อมูลผลลัพธ์จากการทำนาย ประกอบด้วย : ชื่อของวิดีโอ เฟรม และตำแหน่งของกรอบ สีเหลี่ยม
3. ประเมินผลค่าความแม่นยำในการทำงานโดยเทียบระหว่างชุดผลลัพธ์จากการทำนาย และชุดข้อมูลที่มีคำ กำกับเพื่อเป็นคำตอบ โดยกำหนดให้ค่า $\text{IoU} \geq 0.5$ จึงจะนับว่าทำนายได้ถูก
4. เปรียบเทียบผลลัพธ์จากแหล่งที่มา

3.10 การออกแบบการทดสอบการทำงานของระบบนำทางตำแหน่งต่อไปของมนุษย์

3.10.1 ทดสอบประสิทธิภาพการทำงานของระบบนำทางตำแหน่งต่อไปของวัตถุในวิดีโอ สิ่งที่ใช้ในการวัดผล

1. ความเร็วในการนำทางต่อวิดีโอ (วินาที)
2. ความแม่นยำ โดยคำนึงถึงอัตราส่วนร่วมของกรอบที่เหลือ

วัตถุประสงค์

ผู้วิจัยมีความต้องการทดสอบความแม่นยำและความเร็วของการใช้โมเดลปัญญาประดิษฐ์สำหรับตรวจจับวัตถุและสร้างกรอบสีเหลี่ยมทุกๆ N เฟรม แล้วใช้ระบบนำทางตำแหน่งต่อไปของวัตถุในการสร้างกรอบสีเหลี่ยมในเฟรมระหว่างนั้น ว่าด้วยความเร็วที่เพิ่มขึ้นมากนั้นทำให้ความแม่นยำลดลงไปเท่าไหร่

ตัวแปรควบคุม

1. วิดีโอสารสนเทศที่ไม่ติดลิขสิทธิ์ ความยาวประมาณ 10 - 30 วินาที หนึ่งวิดีโอ
2. ใช้โมเดลปัญญาประดิษฐ์สำหรับตรวจจับตำแหน่งวัตถุ ResNet50 ในการสร้างชุดข้อมูลที่มีการกำกับตำแหน่งวัตถุไว้ แล้วใช้มนุษย์ในการตรวจสอบความถูกต้อง เพื่อใช้เป็นค่าตอบของการทำงาน
3. โมเดลปัญญาประดิษฐ์สำหรับตรวจจับตำแหน่งที่ใช้ในการเปรียบเทียบ: YOLO-v3 spp
4. อัลกอริทึมสำหรับระบบนำทางตำแหน่งต่อไปของวัตถุ: DLib
5. อัตราส่วนร่วมของกรอบที่เหลือ: มีส่วนที่ตัดกันมากกว่า 80% ขึ้นไปจึงจะนับว่าผลการทำงานถูกต้อง

วิธีการทดลอง

1. ใช้โมเดลปัญญาประดิษฐ์ YOLO-v3 spp ประมวลผลทุกเฟรมในวิดีโอ และเปรียบเทียบผลลัพธ์กับชุดข้อมูลที่ถูกกำกับตำแหน่งวัตถุไว้แล้ว เพื่อคำนวณหาความแม่นยำ
2. ใช้โมเดลปัญญาประดิษฐ์ YOLO-v3 spp ประมวลผลทุกๆ N เฟรมในวิดีโอ แล้วใช้ระบบนำทางตำแหน่งต่อไปของวัตถุในการสร้างกรอบสีเหลี่ยมในเฟรมระหว่างนั้น และเปรียบเทียบผลลัพธ์กับชุดข้อมูลที่ถูกกำกับตำแหน่งวัตถุไว้แล้ว เพื่อคำนวณหาความแม่นยำ โดยที่ค่า N จะเท่ากับ 10 20 และ 25
3. เปรียบเทียบความเร็วในการประมวลผล และความแม่นยำ

3.11 การออกแบบการทดสอบการระบุตัวตนของมนุษย์

3.11.1 ทดสอบประสิทธิภาพการทำงานของระบบระบุตัวตนของบุคคลภายนอกในภาพสิ่งที่ใช้ในการวัดผล

- ความแม่นยำสำหรับการระบุตัวตนของบุคคลภายนอก

สมมุติฐาน

ผู้จัดได้ตั้งสมมุติฐานว่า ผลลัพธ์ของการทดลองการใช้งานจริงของโมเดลปัญญาประดิษฐ์ ResNet50 ที่สร้างด้วยชุดข้อมูล Market1501 นั้นควรจะมีความแม่นยำในการระบุตัวตนของบุคคลภายนอกมากที่สุดเมื่อเทียบกับโมเดลปัญญาประดิษฐ์ที่สร้างด้วยชุดข้อมูลอื่นๆ เพราะเมื่อเทียบกับโมเดลปัญญาประดิษฐ์ที่ถูกสร้างด้วยชุดข้อมูลอื่นที่มาจากการแหล่งข้อมูลเดียวกัน โมเดลปัญญาประดิษฐ์ ResNet50 ที่สร้างด้วยชุดข้อมูล Market1501 นั้นจะมีความแม่นยำสูงสุด

ตัวแปร

- โมเดลปัญญาประดิษฐ์ ซึ่งได้แก่
 - ResNet50 ของชุดข้อมูล Market1501
 - ResNet50 ของชุดข้อมูล DukeMTMCReID
 - ResNet50 ของชุดข้อมูล CUHK03
 - ResNet50 ของชุดข้อมูล MSMT17

ตัวแปรควบคุม

- ชุดข้อมูล : ชุดข้อมูลที่ทางผู้จัดสร้างขึ้นสำหรับการทดสอบ
- โมเดลปัญญาประดิษฐ์ : YOLO-V3 320 สำหรับการหาตำแหน่งของบุคคล

วิธีการทดลอง

- ดาวน์โหลดโมเดลปัญญาประดิษฐ์ที่ถูกสร้างด้วยชุดข้อมูลต่างๆได้แก่ Market1501, DukeMTMCReID, CUHK03 และ MSMT17
- นำชุดข้อมูลที่ผู้จัดสร้างขึ้นมาผ่านโมเดลปัญญาประดิษฐ์ YOLO-V3 320 เพื่อหาตำแหน่งของบุคคล
- นำโมเดลปัญญาประดิษฐ์แต่ละอันมาทดสอบความแม่นยำสำหรับการระบุตัวตนของบุคคลภายนอกในภาพ ด้วยตำแหน่งของบุคคลที่ได้มาจากขั้นตอนก่อนหน้านี้
- ประเมินผลการทำงานโดยเทียบความแม่นยำสำหรับการระบุตัวตนของบุคคลภายนอกในภาพของแต่ละโมเดล ปัญญาประดิษฐ์ เพื่อหาโมเดลปัญญาประดิษฐ์ที่ได้ผลลัพธ์ดีที่สุด

3.12 การออกแบบการทดสอบการจัดจำการกระทำของมนุษย์

3.12.1 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกเทรน์ผ่าน AVA โดยใช้ชุดข้อมูลของ AVA ในการทดสอบและเทียบผลลัพธ์กับแหล่งอ้างอิง

สิ่งที่ใช้ในการวัดผล

1. ความเร็วในการทำงานต่อรูปภาพ (มิลลิวินาที)
2. ความแม่นยำ (PASCAL mAP)

สมมุติฐาน

ผู้วิจัยได้ตั้งสมมุติฐานว่า ผลลัพธ์ของการทดลองจะมีความแม่นยำเทียบเท่ากับผลลัพธ์จากแหล่งที่มา แต่ความเร็วต่อรูปภาพจะมีความเร็วน้อยกว่าผลลัพธ์จากแหล่งที่มา เนื่องจากแหล่งที่มาของข้อมูลได้ทำการทดสอบโดยใช้กราฟิกการ์ดรุ่น Nvidia GeForce GTX TITAN X ซึ่งเป็นกราฟิกการ์ดที่มีประสิทธิภาพการทำงานดีกว่ากราฟิกการ์ดของผู้วิจัย จึงทำให้สามารถทดสอบด้วยความเร็วที่มากกว่า

ตัวแปรควบคุม

1. ชุดข้อมูล : The validation split of AVA v2.1
2. โมเดลปัญญาประดิษฐ์ : Faster RCNN ResNet101 AVA v2.1

วิธีการทดลอง

1. ดาวน์โหลดชุดข้อมูล The validation split of AVA v2.1
2. แบ่งชุดข้อมูลออกเป็น ชุดข้อมูลสำหรับทดสอบ และชุดข้อมูลที่มีคำกำกับเพื่อเป็นคำตอบ
 - (a) ชุดข้อมูลสำหรับทดสอบ ประกอบด้วย : ชื่อของวิดีโอ
 - (b) ชุดข้อมูลที่มีคำกำกับเพื่อเป็นคำตอบ ประกอบด้วย : ชื่อของวิดีโอ เพرم ตำแหน่งของกรอบสี่เหลี่ยม และรหัสของการกระทำ
3. เรียกชื่อของวิดีโอด้วยชุดข้อมูลทดสอบ และนำโมเดลปัญญาประดิษฐ์ที่เคยฝึกอบรม จากนั้นเก็บผลลัพธ์ เป็นชุดข้อมูลผลลัพธ์จากการทำงาน
 - (a) ชุดข้อมูลผลลัพธ์จากการทำงาน ประกอบด้วย : ชื่อของวิดีโอ เพرم ตำแหน่งของกรอบสี่เหลี่ยม รหัสของการกระทำ และความมั่นใจ
4. ประเมินผลการทำงานโดยเทียบระหว่างชุดผลลัพธ์จากการทำงาน และชุดข้อมูลที่มีคำกำกับเพื่อเป็นคำตอบ
5. เปรียบเทียบผลลัพธ์จากแหล่งที่มา

3.12.2 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกสร้างด้วย AVA และใช้ชุดข้อมูลที่ผู้วิจัยสร้างขึ้นในการทดสอบและเทียบผลลัพธ์กับแหล่งอ้างอิง

สิ่งที่ใช้ในการวัดผล

1. ความในการทำนายเร็วต่อรูปภาพ (มิลลิวินาที)
2. ความแม่นยำ (PASCAL mAP)

สมมุติฐาน

ผู้วิจัยได้ตั้งสมมุติฐานว่าผลลัพธ์ของการทดลองจะมีความแม่นยำต่อลงเมื่อเทียบกับความแม่นยำของการทดลองที่ผ่านมา เนื่องจากชุดข้อมูลที่ผู้วิจัยสร้างขึ้น ได้มีการตัดหมวดหมู่บางอย่างออกไป ทำให้โมเดลปัญญาประดิษฐ์ที่ถูกสร้างด้วย AVA มีหมวดหมู่ของการกระทำไม่ตรงกับชุดข้อมูลที่ผู้วิจัยสร้างขึ้น ซึ่งมีผลทำให้ความแม่นยำลดลง ในส่วนของความเร็วต่อรูปภาพจะมีความเร็วน้อยกว่าผลลัพธ์จากแหล่งที่มา เนื่องจาก แหล่งที่มาของข้อมูลได้ทำการทดสอบโดยใช้กราฟิกการ์ดรุ่น Nvidia GeForce GTX TITAN X card ซึ่งเป็นกราฟิกการ์ดที่มีประสิทธิภาพการทำงานดีกว่า กราฟิกการ์ดของผู้วิจัย จึงทำให้สามารถทดสอบด้วยความเร็วที่มากกว่า

ตัวแปรควบคุม

1. ชุดข้อมูล : ชุดข้อมูลที่ผู้วิจัยสร้างด้วยเครื่องมือกำกับคุณลักษณะ
2. โมเดลปัญญาประดิษฐ์ : Faster RCNN ResNet101 AVA v2.1

วิธีการทดลอง

1. แบ่งชุดข้อมูลออกเป็น ชุดข้อมูลสำหรับทดสอบ และชุดข้อมูลที่มีคำจำกัดเพื่อเป็นคำตอบ
 - (a) ชุดข้อมูลสำหรับทดสอบ ประกอบด้วย : ชื่อของวิดีโอ
 - (b) ชุดข้อมูลที่มีคำจำกัดเพื่อเป็นคำตอบ ประกอบด้วย : ชื่อของวิดีโอ เพرم ตำแหน่งของกรอบสี่เหลี่ยม และรหัสของการกระทำ
2. เรียกชื่อของวิดีโอด้วยชุดข้อมูลทดสอบ และนำโมเดลปัญญาประดิษฐ์ทำนายผลลัพธ์ จากนั้นเก็บผลลัพธ์ เป็นชุดข้อมูลผลลัพธ์จากการทำนาย
 - (a) ชุดข้อมูลผลลัพธ์จากการทำนาย ประกอบด้วย : ชื่อของวิดีโอ เพرم ตำแหน่งของกรอบสี่เหลี่ยม รหัสของการกระทำ และความมั่นใจ
3. ประเมินผลการทำงานโดยเทียบระหว่างชุดผลลัพธ์จากการทำนาย และชุดข้อมูลที่มีคำจำกัดเพื่อเป็นคำตอบ
4. เปรียบเทียบผลลัพธ์กับผลการทดลองที่ผ่านมา

3.12.3 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่ถูกสร้างด้วยชุดข้อมูลที่ผู้วิจัยสร้างขึ้น และใช้ชุดข้อมูลที่ผู้วิจัยสร้างขึ้นในการทดสอบและเทียบผลลัพธ์กับแหล่งอ้างอิง สิ่งที่ใช้ในการวัดผล

1. ความเร็วในการทำนายต่อรูปภาพ (มิลลิวินาที)
2. ความแม่นยำ (PASCAL mAP)

สมมุติฐาน

ผู้วิจัยได้ตั้งสมมุติฐานว่าผลลัพธ์ของการทดลองจะมีความแม่นยำสูงขึ้นเมื่อเทียบกับความแม่นยำของการทดลองที่ผ่านมา เนื่องจากโมเดลปัญญาประดิษฐ์ในการทดลองนี้ เป็นโมเดลปัญญาประดิษฐ์ที่ผู้วิจัยได้สร้างขึ้น ซึ่งจะมีหมวดหมู่ของการกระทำของโมเดลปัญญาประดิษฐ์และชุดข้อมูลทดสอบตรงกัน ในส่วนของความเร็วต่อรูปภาพจะมีความเร็วน้อยกว่าผลลัพธ์จากแหล่งอ้างอิงที่มา เนื่องจากแหล่งอ้างอิงของข้อมูลได้ทำการทดสอบโดยใช้กราฟิกการ์ดรุ่น Nvidia GeForce GTX TITAN X ซึ่งเป็นกราฟิกการ์ดที่มีประสิทธิภาพการทำงานดีกว่ากราฟิกการ์ดของผู้วิจัย จึงทำให้สามารถทดสอบด้วยความเร็วที่มากกว่า

ตัวแปรควบคุม

1. ชุดข้อมูล : ชุดข้อมูลที่ผู้วิจัยสร้างด้วยเครื่องมือกำกับคุณลักษณะ
2. โมเดลปัญญาประดิษฐ์ : โมเดลปัญญาประดิษฐ์ที่ผู้วิจัยสร้างขึ้น

วิธีการทดลอง

1. แบ่งชุดข้อมูลออกเป็น ชุดข้อมูลสำหรับทดสอบ และชุดข้อมูลที่มีคำกำกับเพื่อเป็นคำตอบ
 - (a) ชุดข้อมูลสำหรับทดสอบ ประกอบด้วย : ชื่อของวิดีโอ
 - (b) ชุดข้อมูลที่มีคำกำกับเพื่อเป็นคำตอบ ประกอบด้วย : ชื่อของวิดีโอ เพرم ตำแหน่งของกรอบสี่เหลี่ยม และรหัสของการกระทำ
2. เรียกชื่อของวิดีโอด้วยชุดข้อมูลทดสอบ และนำโมเดลปัญญาประดิษฐ์ทำนายผลลัพธ์ จากนั้นเก็บผลลัพธ์ เป็นชุดข้อมูลผลลัพธ์จากการทำนาย
 - (a) ชุดข้อมูลผลลัพธ์จากการทำนาย ประกอบด้วย : ชื่อของวิดีโอ เพرم ตำแหน่งของกรอบสี่เหลี่ยม รหัสของการกระทำ และความมั่นใจ
3. ประเมินผลการทำงานโดยเทียบระหว่างชุดผลลัพธ์จากการทำนาย และ ชุดข้อมูลที่มีคำกำกับเพื่อเป็นคำตอบ
4. เปรียบเทียบผลลัพธ์กับผลการทดลองที่ผ่านมา

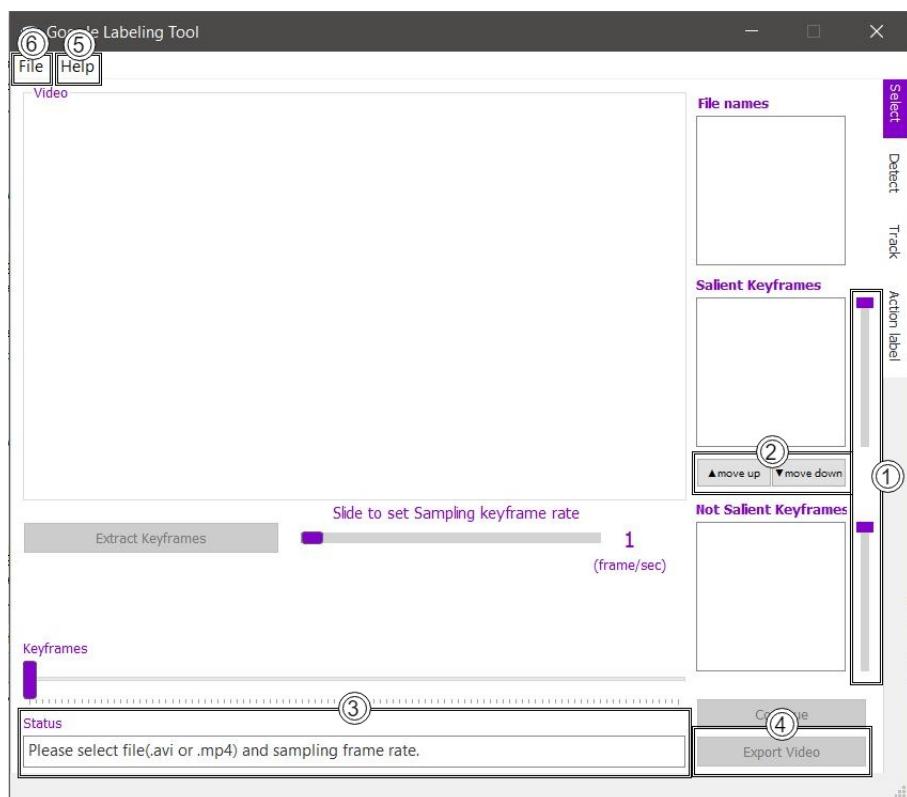
บทที่ 4

ผลการดำเนินงาน

4.1 เครื่องมือกำกับคุณลักษณะ

4.1.1 หน้าต่างแสดงผลของแอพพลิเคชัน

หน้าต่าง Select

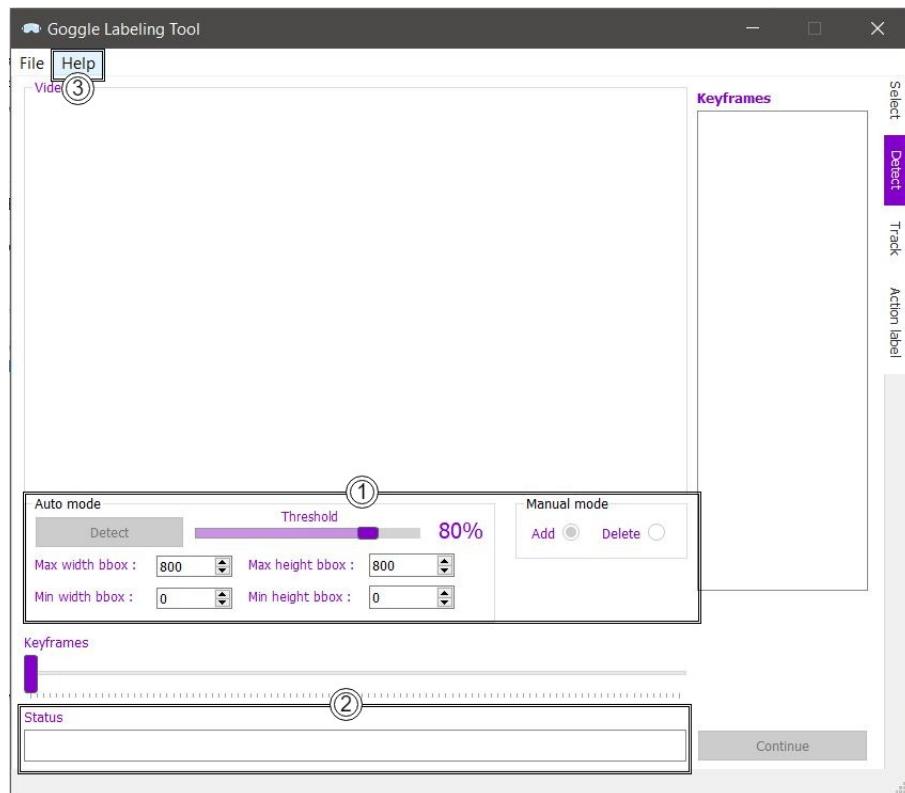


รูปที่ 4.1: รูปหน้าต่างแสดงผลของหน้าต่าง Select

จากรูปที่ 4.1 แสดงหน้าต่าง Select ของแอพพลิเคชัน ซึ่งเมื่อเทียบกับฉบับร่างตามรูปที่ 3.3 จะมีส่วนที่เพิ่มเติมขึ้นมาดังนี้

1. แถบเลื่อนสำหรับเลื่อนคูเพรมที่มีมนุษย์หรือไม่มีมนุษย์ เพื่อเพิ่มความสะดวกในการเลือกคูเพรม
2. ปุ่มสำหรับแก้ไขเพรมที่มีมนุษย์หรือไม่มีมนุษย์
3. แถบแสดงสถานะกระบวนการทำงาน
4. ปุ่มสำหรับนำผลลัพธ์ออกเป็นไฟล์วิดีโอด้วยไฟล์ในช่วงที่มีมนุษย์อยู่
5. แถบสำหรับคำแนะนำช่วยเหลือ
6. ปุ่มสำหรับเปิดไฟล์

หน้าต่าง Detect

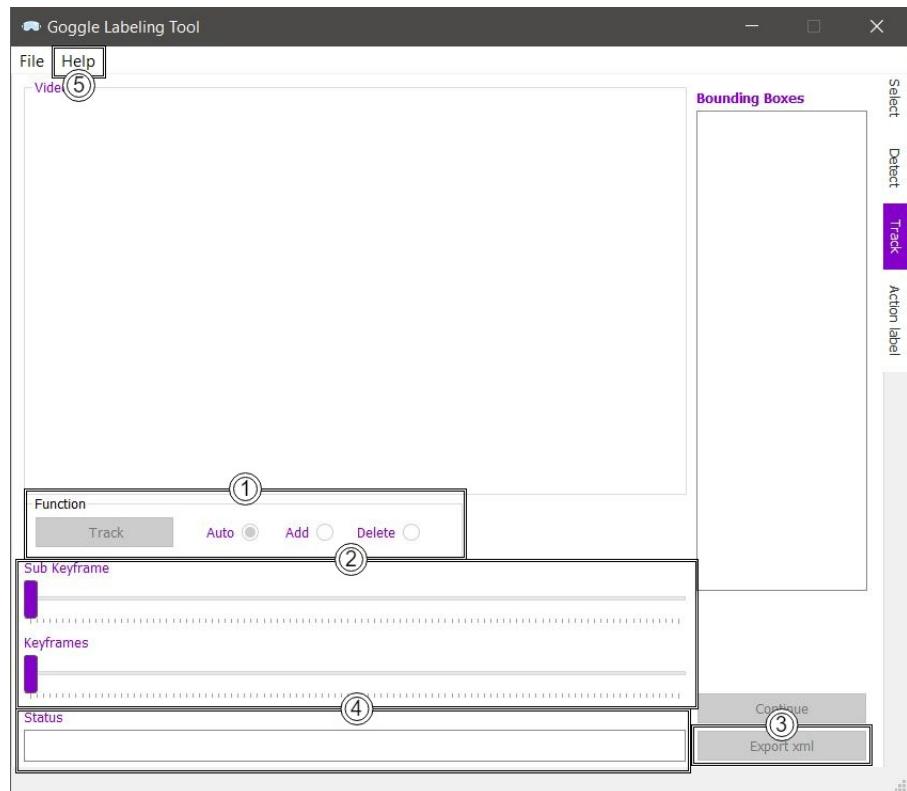


รูปที่ 4.2: รูปหน้าต่างแสดงผลของหน้าต่าง Detect

จากรูปที่ 4.2 แสดงหน้าต่าง Detect ของแอพพลิเคชัน ซึ่งเมื่อเทียบกับรูปที่ (3.5) จะมีส่วนที่เพิ่มเติมขึ้นมาดังนี้

1. ปรับหน้าตาใหม่ด้วยการทำงานแบบอัตโนมัติ และกำหนดเองสามารถใช้งานได้สะดวกขึ้น และเพิ่มความหลากหลายในการปรับแก้ในการทำงานอัตโนมัติ
2. แถบแสดงสถานะกระบวนการทำงาน
3. แถบสำหรับคำแนะนำช่วยเหลือ

หน้าต่าง Track

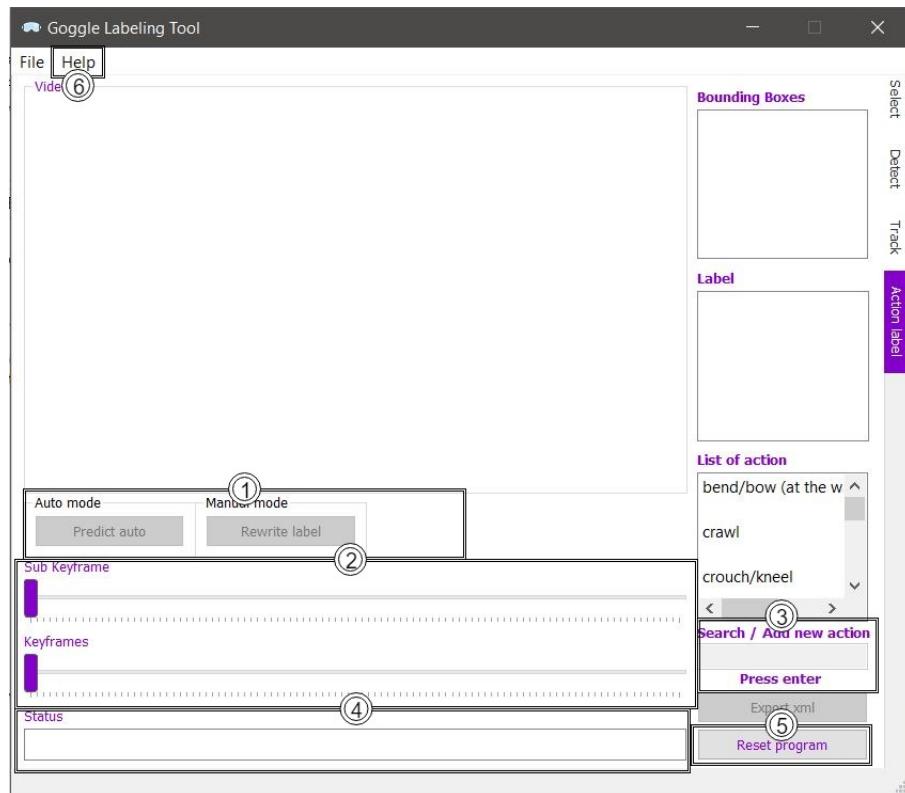


รูปที่ 4.3: รูปหน้าต่างแสดงผลของหน้าต่าง Track

จากรูปที่ 4.3 แสดงหน้าต่าง Track ของแอพพลิเคชัน ซึ่งเมื่อเทียบกับกับฉบับร่างตามรูปที่ (3.7) จะมีส่วนที่เพิ่มเติมขึ้นมาดังนี้

1. ปรับหน้าตาใหม่จากการทำงานแบบอัตโนมัติและกำหนดเองจากฉบับร่างเพื่อให้สามารถใช้งานได้สะดวกขึ้น
2. เพิ่มแถบเลื่อนเป็น 2 แถบเลื่อนทำให้สามารถดูคีย์เฟรมและเฟรมที่อยู่ระหว่างช่วงคีย์เฟรมได้สะดวกขึ้น
3. เพิ่มปุ่มสำหรับนำผลลัพธ์ออกเป็นไฟล์ xml
4. แถบแสดงสถานะกระบวนการการทำงาน
5. แถบสำหรับคำแนะนำช่วยเหลือ

หน้าต่าง Label



รูปที่ 4.4: รูปหน้าต่างแสดงผลของหน้าต่าง Label

จากรูปที่ 4.4 แสดงหน้าต่าง Label ของแอพพลิเคชัน ซึ่งเมื่อเทียบกับกับฉบับร่างตามรูปที่ (3.9) จะมีส่วนที่เพิ่มเติมขึ้นมาดังนี้

1. ปรับหน้าตาใหม่จากการทำงานแบบอัตโนมัติและกำหนดเองจากฉบับร่างเพื่อให้สามารถใช้งานได้สะดวกขึ้น
2. เพิ่มແຕບເລື່ອນປັນ 2 ແຕບເລື່ອນທຳໃຫ້ສາມາຄດຸກີຍີ່ເພິ່ມແຕບຮ່ວມມືນຂອງການກະທຳ
3. ເພີ້ມສຳຫຼັບຄົນຫາຫຼືເພີ້ມໜົວດໝູ່ຂອງການກະທຳ
4. ແຕບແສດງສະຖານະກະບວນການກະທຳ
5. ປຸ່ມສຳຫຼັບເຮີ່ມຕົ້ນການກະທຳໃໝ່
6. ແຕບສຳຫຼັບຄຳແນະນຳໜໍ່ຢ່າງເລື້ອ

4.1.2 ผลลัพธ์การทำงานในแต่ละหน้าต่างของแอปพลิเคชัน

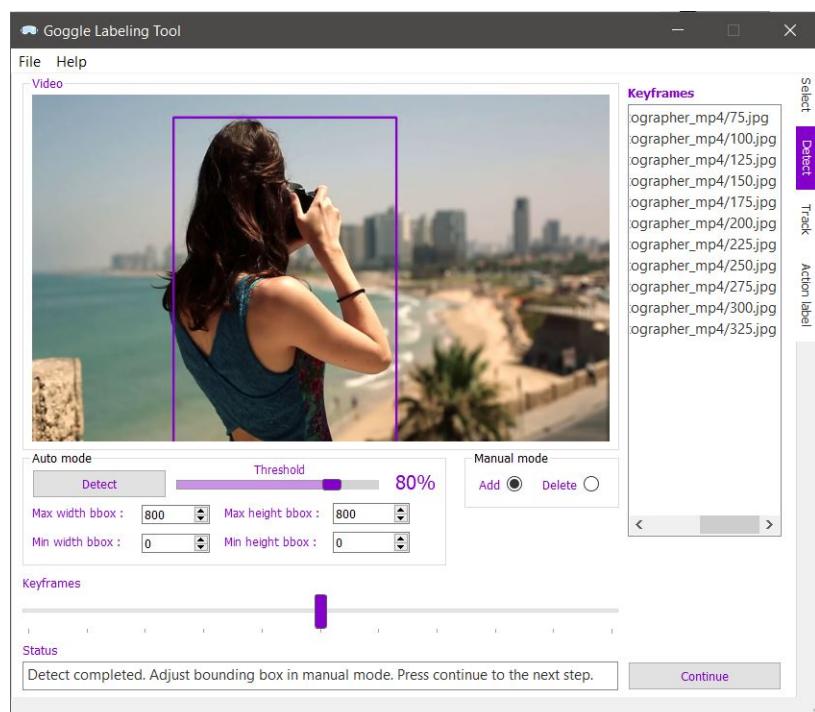
ผลลัพธ์การทำงานของหน้าต่าง Select



รูปที่ 4.5: รูปผลลัพธ์การแยกเฟรมที่มีมนุษย์อยู่ และไม่มีมนุษย์อยู่ภายในเฟรม

ขั้นตอนแรกแอปพลิเคชัน จะสกัดแยกวิดีโอออกเป็นเฟรมทั้งหมด และทำการสั่มคีย์เฟรมอุปกรณ์ตามความถี่ที่ผู้ใช้งานตั้งไว้ จากนั้นแอปพลิเคชันจะนำโมเดล YOLO-v3 320 มาตรวจสอบว่าแต่ละคีย์เฟรมมีเฟรมใดบ้างที่มีมนุษย์อยู่ภายในเฟรม จากนั้นจะทำการแยกเฟรมที่มีมนุษย์อยู่ และไม่มีมนุษย์อยู่ ดังรูปที่ 4.5

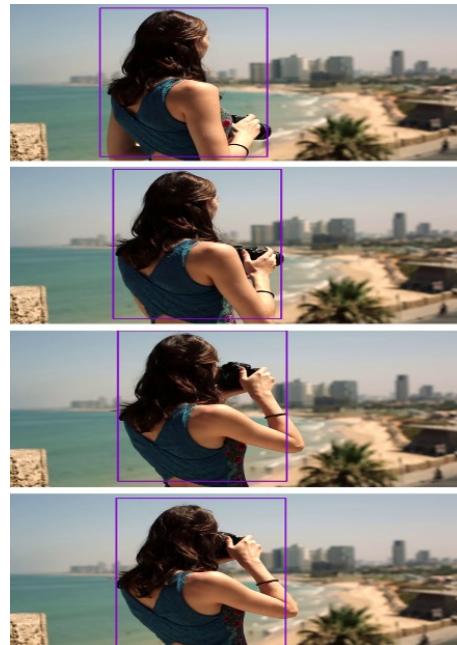
ผลลัพธ์การทำงานของหน้าต่าง Detect



รูปที่ 4.6: รูปคีย์เฟรมที่ถูกตีกรอบสีเหลืองในส่วนที่มีมนุษย์อยู่

แอ��พลิเคชันจะนำคีย์เฟรมที่มีนุชย์ที่ได้จากหน้าต่าง Select นำมาตีกรอบสี่เหลี่ยมในส่วนของเฟรมที่มีมนุชย์อยู่โดยสามารถใช้โหมดการทำงานแบบบอตโนมัติหรือแบบแก้ไขเองก็ได้ ซึ่งผลลัพธ์ที่ได้จะได้คีย์เฟรมที่มีกรอบสี่เหลี่ยม ดังรูปที่ 4.6 จากนั้นจะบันทึกข้อมูลในไฟล์ txt

ผลลัพธ์การทำงานของหน้าต่าง Track



(ก) ตัวอย่างเฟรมที่ถูกตีกรอบสี่เหลี่ยม

```
<?xml version="1.0"?>
- <annotation>
  <folder>D:/Goggle/Goggle_team/out/Photographer_mp4/img</folder>
  <filename>75.jpg.txt</filename>
  <path>D:/Goggle/Goggle_team/out/Photographer_mp4/img/75.jpg</path>
- <source>
  <database>Unknown</database>
</source>
- <size>
  <width>1280</width>
  <height>720</height>
  <depth>3</depth>
</size>
<segmented>0</segmented>
- <object>
  <name>person</name>
  <pose>Unspecified</pose>
  <truncated>0</truncated>
  <difficult>0</difficult>
  - <bndbox>
    <xmin>2</xmin>
    <ymin>35</ymin>
    <xmax>368</xmax>
    <ymax>714</ymax>
  </bndbox>
</object>
</annotation>
```

(ข) ตัวอย่างไฟล์ xml

รูปที่ 4.7: รูปผลลัพธ์การทำงานของหน้าต่าง Track

แอ��พลิเคชันจะนำคีย์เฟรมที่ถูกตีกรอบสี่เหลี่ยมจากหน้าต่าง Detect มาทำนายกรอบสี่เหลี่ยมในเฟรมที่เหลือระหว่างช่วงคีย์เฟรม ซึ่งผลลัพธ์ที่ได้จะได้เฟรมทุกเฟรมที่มีมนุชย์อยู่จะถูกตีกรอบสี่เหลี่ยม ดังรูปที่ 4.7ก จากนั้นสามารถบันทึกข้อมูลออกเป็นไฟล์ xml ได้ดังรูปที่ 4.7ข

ผลลัพธ์การทำงานของหน้าต่าง Label



(ก) ตัวอย่างเฟรมที่ถูกตีกรอบสีเหลืองและคำทำนายการกระทำ

```
<?xml version="1.0"?>
- <annotation>
  <folder>D:/Goggle/Goggle_team/out/Photographer_mp4/Photographer_mp4/img</folder>
  <filename>75.jpg.txt</filename>
  <path>D:/Goggle/Goggle_team/out/Photographer_mp4/Photographer_mp4/img/75.jpg</path>
  - <source>
    <database>Unknown</database>
  </source>
  - <size>
    <width>1280</width>
    <height>720</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  - <object>
    <name>carry/hold (an object)</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    - <bndbox>
      <xmin>2</xmin>
      <ymin>35</ymin>
      <xmax>368</xmax>
      <ymax>714</ymax>
    </bndbox>
  </object>
</annotation>
```

(ข) ตัวอย่างไฟล์ xml

รูปที่ 4.8: รูปผลลัพธ์การทำงานของหน้าต่าง Label

แอพพลิเคชันจะนำกรอบสีเหลืองของทุกเฟรมที่มีมนุษย์อยู่มาทำนายมนุษย์ในกรอบสีเหลืองนั้นกำลังมีการกระทำการอะไรอยู่ โดยสามารถทำงานได้ทั้งหมดอัตโนมัติหรือแบบแก้ไขเอง และสามารถบันทึกข้อมูลออกเป็นไฟล์ xml ได้ดังรูปที่ 4.8ข

4.2 ผลการทดลองการตรวจจับวัตถุ

4.2.1 ข้อมูลรายละเอียดประกอบการทดสอบ

จำนวนเฟรมทั้งหมด: 20 เฟรม

จำนวนมนุษย์ที่อยู่ในเฟรม : 0-5 คน

ความละเอียดรูปภาพ : 1280×720 พิกเซล

ขอบเขตอัตราส่วนร่วมของกรอบที่เหลือที่จะนับว่าการทำนายถูกต้อง: 50% ขึ้นไป

4.2.2 ผลทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์สำหรับการทำการตรวจจับภาพบุคคล

ข้อมูลความแม่นยำของโมเดลปัญญาประดิษฐ์เมื่อทดสอบด้วยชุดข้อมูลของผู้วิจัย

	ความเร็วต่อรูปภาพ (มิลลิวินาที)	ความแม่นยำ (0.5 IOU)
SSD Mobilenet v1 ppm	63.82	37.03
YOLO-v3 320	65.00	64.91
YOLO-v3 tiny	17.21	44.44
YOLO-v3 spp	65.40	70.30
Faster rcnn inception v2	981.21	42.59

ตารางที่ 4.1: ข้อมูลผลการทำงานของโมเดลปัญญาประดิษฐ์สำหรับการทำการตรวจจับภาพบุคคล

จากตารางที่ 4.2 ผู้วิจัยได้ทำการทดสอบความแม่นยำและความเร็วในการประมวลผลของโมเดลปัญญาประดิษฐ์สำหรับการทำการตรวจจับภาพบุคคล พบว่าโมเดลปัญญาประดิษฐ์ที่มีความแม่นยามากที่สุดคือ YOLO-v3 spp และโมเดลปัญญาประดิษฐ์ที่มีความเร็วในการทำนายต่อรูปภาพเร็วที่สุดคือ YOLO-v3 tiny จากผลลัพธ์การทดลองดังกล่าว ทุกโมเดลปัญญาประดิษฐ์ยกเว้น Faster RCNN มีความเร็วในการประมวลผลต่อรูปภาพที่ผู้วิจัยสามารถรับได้ (ไม่เกิน 1 วินาที) ดังนั้นผู้วิจัยจึงเลือกโมเดลปัญญาประดิษฐ์ที่จะใช้จากการความแม่นยามากที่สุด คือ YOLO-v3 spp

4.3 ผลการทดสอบระบบติดตามตำแหน่งของมนุษย์

4.3.1 ข้อมูลรายละเอียดประกอบการทดสอบ

ชื่อวิดีโอ: Photographer beach photography

ความยาววิดีโอ: 15 วินาที

จำนวนเฟรมทั้งหมด: 374 เฟรม

อัตราเฟรมต่อวินาที: 24.9 เฟรมต่อวินาที

ความละเอียดของวิดีโอ: 1920x1080 พิกเซล

ความละเอียดของวิดีโอที่ใช้ในการประมวลผลจริง: 1280x720 พิกเซล

ขอบเขตอัตราส่วนร่วมของกรอบที่เหลืออยู่ที่จะนับว่าการทำนายถูกต้อง: 80% ขึ้นไป

4.3.2 ผลทดสอบประสิทธิภาพ และความเร็วในการประมวลผล

วิธีการทดสอบ	ความแม่นยำ (%)	ความเร็วในการประมวลผล (วินาที)		
ใช้โมเดลปัญญาประดิษฐ์ YOLO-v3 spp ประมวลผลทุกเฟรมในวิดีโอ	95	-	452	-
ใช้โมเดลปัญญาประดิษฐ์ YOLO-v3 spp ประมวลผลทุกๆ N เฟรมในวิดีโอ แล้วใช้ระบบทำนายตำแหน่งต่อไปของวัตถุในเฟรมระหว่างนั้น				
N = 10	85	-10	69	-383
N = 20	80	-15	41	-411
N = 25	75	-20	35	-417

ตารางที่ 4.2: ผลการทดสอบประสิทธิภาพของการตรวจจับกรอบสีเหลี่ยมภายในวิดีโอ

จากตารางที่ 4.2 ผู้วิจัยได้ทำการทดสอบความแม่นยำและความเร็วในการประมวลผลของการใช้โมเดลปัญญาประดิษฐ์ YOLO-v3 spp ประมวลผลทุกเฟรม แม้จะตั้งขอบเขตอัตราส่วนร่วมของกรอบที่เหลืออยู่ที่จะนับว่าการทำนายถูกต้องสูงถึง 80% แต่ความแม่นยำสูงถึง 95% ใช้เวลาในการประมวลผล 452 วินาที เฉลี่ยเฟรมละ 1.2 วินาที ซึ่งถือเป็นความแม่นยำที่สูงมากเมื่อเทียบกับเวลาที่ใช้ในการประมวลผล

ต่อมาเป็นการทดสอบโดยใช้โมเดลปัญญาประดิษฐ์ประมวลผลเฉพาะบางเฟรมทุกๆ 7 วินาที แล้วใช้ระบบทำนายตำแหน่งต่อไปของวัตถุในการสร้างกรอบสีเหลี่ยมในเฟรมระหว่างนั้น เพื่อเพิ่มความเร็วในการประมวลผลโดยระยะที่ใช้ในการทดสอบคือ ทุกๆ 10 เฟรม 20 เฟรม และ 25 เฟรม ซึ่งจากการทดสอบนั้นพบว่ามีการนี้มีความแม่นยำลดลงมาเหลือ 85% น้อยกว่าอยู่เพียง 10% เท่านั้น ถือเป็นความแม่นยำที่สูงเมื่อเทียบกันด้วยระยะเวลาในการประมวลผล ในขณะที่การใช้ระยะประมวลผลเป็น 10 เฟรมนั้นใช้เวลาในการประมวลผลเพียง 69 วินาที น้อยกว่าการใช้โมเดลปัญญาประดิษฐ์ YOLO-v3 spp ประมวลผลทุกเฟรมถึง 383 วินาที ซึ่งเร็วกว่าถึง 6.5 เท่า และความแม่นยำลดลงมาเหลือ 85% น้อยกว่าอยู่เพียง 10% เท่านั้น ถือเป็นความแม่นยำที่สูงเมื่อเทียบกันด้วยระยะเวลาในการประมวลผล ในขณะที่การใช้ระยะประมวลผล 20 เฟรมนั้นจะประมวลผลเร็วกว่าการใช้โมเดลปัญญาประดิษฐ์ YOLO-v3 spp ประมวลผลทุกเฟรมถึง 11 เท่า และมีความแม่นยำต่ำกว่า 15% และเมื่อใช้ระยะประมวลผล 25 เฟรมจะเร็วกว่าประมาณ 13 เท่า และความแม่นยำต่ำลงถึง 20%

4.4 ผลการทดสอบระบบบุคคลตัวตนของมนุษย์

4.4.1 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์สำหรับการระบุตัวตนของบุคคล

ความแม่นยำของโมเดลปัญญาประดิษฐ์จากแหล่งที่มีมาค่าดั้งตารางด้านล่างดังนี้

โมเดลปัญญาประดิษฐ์	rank1/mAP โดยใช้วิธีการทดสอบด้วย Global+DMLI
ResNet50 Market1501	91.0/77.6
ResNet50 DukeMTMCRID	80.7/68.0
ResNet50 CUHK03	60.9/59.7
ResNet50 MSMT17	66.3/40.6

ตารางที่ 4.3: ผลการทดสอบความแม่นยำของโมเดลปัญญาประดิษฐ์

ต่อมานำโมเดลปัญญาประดิษฐ์แต่ละอันมาทดสอบกับตัวอย่างภาพชุดข้อมูลที่ทางคณะผู้วิจัยได้สร้างขึ้น โดยภาพชุดข้อมูลที่นำมาใช้จะผ่านการตรวจหาบุคคลภายในภาพด้วยโมเดลปัญญาประดิษฐ์ YOLO v3 320 และจะเป็นการทดลองในกรณีที่บุคคลในภาพนั้นเป็นบุคคลเดียวกัน



รูปที่ 4.9: ภาพตัวอย่างชุดข้อมูลสำหรับการทดลองครั้งที่ 1

โมเดลปัญญาประดิษฐ์	ค่าสำหรับการระบุบุคคล (Original distance)
ResNet50 Market1501	0.4308
ResNet50 DukeMTMCRID	0.4827
ResNet50 CUHK03	0.4914
ResNet50 MSMT17	0.4668

ตารางที่ 4.4: ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 1



รูปที่ 4.10: ภาพตัวอย่างชุดข้อมูลสำหรับการทดลองครั้งที่ 2

โมเดลปัญญาประดิษฐ์	ค่าสำหรับการระบุบุคคล (Original distance)
ResNet50 Market1501	0.3035
ResNet50 DukeMTMCReID	0.3332
ResNet50 CUHK03	0.3042
ResNet50 MSMT17	0.3684

ตารางที่ 4.5: ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 2



รูปที่ 4.11: ภาพตัวอย่างชุดข้อมูลสำหรับการทดลองครั้งที่ 3

โมเดลปัญญาประดิษฐ์	ค่าสำหรับการระบุบุคคล (Original distance)
ResNet50 Market1501	0.3308
ResNet50 DukeMTMCReID	0.3296
ResNet50 CUHK03	0.3134
ResNet50 MSMT17	0.3968

ตารางที่ 4.6: ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 3

ต่อมาจะเป็นการทดลองในกรณีที่บุคคลในภาพไม่เป็นบุคคลเดียวกัน



รูปที่ 4.12: ภาพตัวอย่างชุดข้อมูลสำหรับการทดลองครั้งที่ 4

โมเดลปัญญาประดิษฐ์	ค่าสำหรับการระบุบุคคล (Original distance)
ResNet50 Market1501	0.7285
ResNet50 DukeMTMCReID	0.6882
ResNet50 CUHK03	0.6727
ResNet50 MSMT17	0.7408

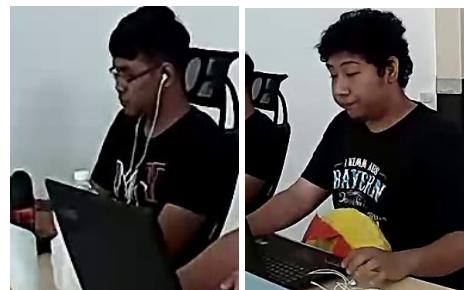
ตารางที่ 4.7: ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 4



รูปที่ 4.13: ภาพตัวอย่างชุดข้อมูลสำหรับการทดลองครั้งที่ 5

โมเดลปัญญาประดิษฐ์	ค่าสำหรับการระบุบุคคล (Original distance)
ResNet50 Market1501	0.6098
ResNet50 DukeMTMCReID	0.6522
ResNet50 CUHK03	0.6 275
ResNet50 MSMT17	0.6155

ตารางที่ 4.8: ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 5



รูปที่ 4.14: ภาพตัวอย่างชุดข้อมูลสำหรับการทดลองครั้งที่ 6

โมเดลปัญญาประดิษฐ์	ค่าสำหรับการระบุบุคคล (Original distance)
ResNet50 Market1501	0.6159
ResNet50 DukeMTMCReID	0.5352
ResNet50 CUHK03	0.5888
ResNet50 MSMT17	0.6119

ตารางที่ 4.9: ผลการทดสอบความแม่นยำสำหรับการระบุบุคคลของโมเดลปัญญาประดิษฐ์ครั้งที่ 6

ค่าความแม่นยำในการระบุบุคคลนั้นค่าอยู่ที่ 0 แสดงบุคคลใน 2 เฟรมนั้นเป็นบุคคลเดียวกัน จากการทดลองครั้งที่ 1 จะเป็นเฟรมที่ไม่ต่อเนื่องกัน การทดลองครั้งที่ 2 และ 3 นั้นจะเป็นเฟรมที่ต่อเนื่องกันมากขึ้นตามลำดับ และการทดลองที่ 4 5 และ 6 นั้นจะนำภาพที่แต่ละบุคคลที่ท่าทางใกล้เคียงกันมาใช้ ซึ่งจะแสดงให้เห็นว่า โมเดลปัญญาประดิษฐ์ที่สามารถให้ผลลัพธ์ที่มีประสิทธิภาพต่อเนื่องมากที่สุดคือ ResNet50 Market1501

4.5 ผลการทดสอบการจดจำการกระทำของมนุษย์

4.5.1 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกเทรนด์ผ่าน AVA เทียบผลลัพธ์กับแหล่งอ้างอิง ได้ผลการทดลองดังตารางต่อไปนี้

	ความเร็วต่อรูปภาพ(มิลลิวินาที)	ความแม่นยำ (PASCAL mAP)
แหล่งอ้างอิง	0.93	11
ผลการทดสอบของผู้วิจัย	5,279	6.8

ตารางที่ 4.10: ผลการทดสอบความแม่นยำของโมเดลปัญญาประดิษฐ์เทียบผลลัพธ์กับแหล่งอ้างอิง

ความเร็วของตอรูปภาพทางผู้วิจัยได้ใช้กราฟฟิกการ์ด GTX 2080 Ti ใน การทดสอบซึ่งจะให้ความเร็วอยู่ที่ 5 วินาที ซึ่งทางแหล่งอ้างอิงนั้นใช้กราฟฟิกการ์ด Nvidia GeForce GTX TITAN X ในส่วนของค่าความแม่นยำที่ไม่เท่ากัน คาดว่าจะเป็นเพราะการประมวลผลของกราฟฟิกการ์ดของรุ่นที่ต่างกันจึงทำให้ค่า mAP ที่ออกมามีเมื่อเท่ากัน

4.5.2 ผลการทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกเทรนด์ผ่าน AVA และ ใช้ชุดข้อมูลที่ผู้วิจัยสร้างขึ้น ในการทดสอบและเทียบผลลัพธ์กับแหล่งอ้างอิง

4.5.3 ทดสอบประสิทธิภาพการทำงานของโมเดลปัญญาประดิษฐ์ที่เคยถูกเทรนด์ผ่านชุดข้อมูลสำหรับการเทรนด์ที่ผู้วิจัยสร้างขึ้น และ ใช้ชุดข้อมูลที่ผู้วิจัยสร้างขึ้น ในการทดสอบและเทียบผลลัพธ์การทดสอบก่อนหน้า

	ความเร็วต่อรูปภาพ(มิลลิวินาที)	ความแม่นยำ (PASCAL mAP)
แหล่งอ้างอิง	X	X
ผลการทดสอบของผู้วิจัย	X	X

ตารางที่ 4.11: ผลการทดสอบความแม่นยำของโมเดลปัญญาประดิษฐ์ เมื่อใช้กับชุดข้อมูลที่ผู้วิจัยสร้างขึ้น

	ความเร็วต่อรูปภาพ(มิลลิวินาที)	ความแม่นยำ (PASCAL mAP)
ผลการทดสอบที่ผ่านมา	X	X
ผลการทดสอบของผู้วิจัย	X	X

ตารางที่ 4.12: ผลการทดสอบความแม่นยำของโมเดลปัญญาประดิษฐ์ที่ผู้วิจัยสร้างขึ้น ใช้กับชุดข้อมูลที่ผู้วิจัยสร้างขึ้น

ເອກສາຣອ້າງອີງ

- [1] Activation functions : Sigmoid, relu, leaky relu and softmax basics for neural networks and deep learning.
- [2] Convolutional neural network (cnn).
- [3] Darklabel1.3-image labeling and annotation tool.
- [4] Going deeper with convolutions.
- [5] Intersection over union (iou) for object detection.
- [6] Object detection (part 1).
- [7] Object detection (part 2).
- [8] Openlabeling: open-source image and video labeler.
- [9] Optical flow.
- [10] Understanding of convolutional neural network (cnn) — deep learning.
- [11] Vgg16 – convolutional network for classification and detection.
- [12] A video dataset of spatio-temporally localized atomic visual actions.
- [13] Ahmad Ali and Sikander Majid Mirza. Object tracking using correlation, kalman filter and fast means shift algorithms. In 2006 International Conference on Emerging Technologies, pages 174–178. IEEE, 2006.
- [14] Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The berkeley framenet project. In Proceedings of the 17th International Conference on Computational Linguistics - Volume 1, COLING '98, pages 86–90, Stroudsburg, PA, USA, 1998. Association for Computational Linguistics.
- [15] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 961–970, 2015.
- [16] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset, 2018.
- [17] Martin Danelljan, Gustav Häger, Fahad Khan, and Michael Felsberg. Accurate scale estimation for robust visual tracking. In British Machine Vision Conference, Nottingham, September 1-5, 2014. BMVA Press, 2014.

- [18] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In Scandinavian conference on Image analysis, pages 363–370. Springer, 2003.
- [19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.
- [20] Michael I Jordan and Robert A Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214, 1994.
- [21] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 1725–1732, 2014.
- [22] Yu Kong and Yun Fu. Human action recognition and prediction: A survey. arXiv preprint arXiv:1806.11230, 2018.
- [23] Luc Van Gool Limin Wang, Yu Qiao Xiaou Tang. Actionness estimation using hybrid fully convolutional networks. <https://arxiv.org/pdf/1604.07279.pdf>, 2016.
- [24] Xiaobin Liu, Shiliang Zhang, Tiejun Huang, and Qi Tian. E ^2 bows: An end-to-end bag-of-words model via deep convolutional neural network. arXiv preprint arXiv:1709.05903, 2017.
- [25] Hao Luo, Wei Jiang, Xuan Zhang, Xing Fan, Jingjing Qian, and Chi Zhang. Alignedreid++: Dynamically matching local information for person re-identification. *Pattern Recognition*, 94:53–61, 2019.
- [26] Shugao Ma, Sarah Adel Bargal, Jianming Zhang, Leonid Sigal, and Stan Sclaroff. Do less and achieve more: Training cnns for action recognition utilizing action images from the web. *Pattern Recognition*, 68:334–345, 2017.
- [27] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Yan Yan, Lisa Brown, Quanfu Fan, Dan Gutfreund, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE transactions on pattern analysis and machine intelligence*, 2019.
- [28] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. arXiv, 2018.
- [29] Karin Kipper Schuler. Verbnet: A Broad-coverage, Comprehensive Verb Lexicon. PhD thesis, Philadelphia, PA, USA, 2005. AAI3179808.
- [30] Ross Shaoqing Ren, Kaiming He and Jian Sun. Faster r-cnn: Towards real-time objectdetection with region proposal networks. arXiv, (0), 2016.
- [31] Du Tran, Lubomir D Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. C3d: generic features for video analysis. CoRR, abs/1412.0767, 2(7):8, 2014.

- [32] Vittorio Ferrari Cordelia Schmid Vicky Kalogeiton, Philippe Weinzaepfel. Action tubelet detector for spatio-temporal action localization. <https://arxiv.org/pdf/1705.01861.pdf>, 2017.
- [33] Cordelia Schmid Xiaojiang Peng. Multi-region two-stream r-cnn for action detection. <https://hal.inria.fr/hal-01349107v1/document>, 2016.
- [34] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4694–4702, 2015.
- [35] Wajdi Zaghouani, Mona Diab, Aous Mansouri, Sameer Pradhan, and Martha Palmer. The revised arabic propbank. In Proceedings of the Fourth Linguistic Annotation Workshop, LAW IV ’10, pages 222–226, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

ภาคผนวก

ภาคผนวก ก

ตัวอย่างชุดข้อมูลที่ผู้จัดสร้างขึ้น

ตัวอย่างชุดข้อมูลสำหรับการทดสอบโมเดลปัญญาประดิษฐ์ในการตรวจจับภาพบุคคล



รูปที่ ก.1: รูปผลลัพธ์การทำงานของหน้าต่าง Track