



Goggle : People Video Analytics and Deep Learning Platform

- ชื่อภาษาไทย -

นายปฐมพงศ์ สินธุจาม

นายศุภกร เบญจวิกรัย

นายอุกฤษฎ์ เลิศวรรณาการ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมหุ่นยนต์และระบบอัตโนมัติ

สถาบันวิทยาการหุ่นยนต์ภาครสนา

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ปีการศึกษา 2562



Goggle : People Video Analytics and Deep Learning Platform

- ชื่อภาษาไทย -

นายปฐมพงศ์ สินธุจาม

นายศุภกร เบญจวิกรัย

นายอุกฤษฎ์ เลิศวรรณาการ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมหุ่นยนต์และระบบอัตโนมัติ

สถาบันวิทยาการหุ่นยนต์ภาคสนาม

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ปีการศึกษา 2562

- ชื่อภาษาไทย -

นายปฐมพงศ์ สินธุ์งาม

นายศุภกร เบญจวิกรัย

นายอุกฤษฎ์ เลิศวรรณาการ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมหุ่นยนต์และระบบอัตโนมัติ

สถาบันวิทยาการหุ่นยนต์ภาคสนาม

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ปีการศึกษา 2562

คณะกรรมการสอบวิทยานิพนธ์

ประธานกรรมการสอบวิทยานิพนธ์

(ดร.อาบทิพย์ ชีรวงศ์กิจ)

อาจารย์ที่ปรึกษาวิทยานิพนธ์

(นายธนัชชา ชูพจน์เจริญ)

อาจารย์ที่ปรึกษาวิทยานิพนธ์

(รศ.ดร.ชิต เหล่าวัฒนา)

กรรมการสอบวิทยานิพนธ์

(ดร.ปิติวุฒิ ชีรกิตติกุล)

กรรมการสอบวิทยานิพนธ์

(ดร.สุรชัย วงศ์บุณย์ยง)

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ชื่อวิทยานิพนธ์	- ชื่อภาษาไทย -
หน่วยกิต	6
ผู้เขียน	นายปฐมพงศ์ สินธุจิต นายศุภกร เบญจวิกรัย นายอุตุษฐ์ เลิศวรรณการ
อาจารย์ที่ปรึกษา	ที่ปรึกษาวิทยานิพนธ์หลัก นายธนัชชา ชูพจน์เจริญ ที่ปรึกษาวิทยานิพนธ์ร่วม รศ.ดร.ชิต เหล่าวัฒนา
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมหุ่นยนต์และระบบอัตโนมัติ
คณะ	สถาบันวิทยาการหุ่นยนต์ภาคสนาม
ปีการศึกษา	2562

บทคัดย่อ

งานวิทยานิพนธ์นี้เป็นงานที่เกี่ยวกับการออกแบบและจัดทำแพลตฟอร์มหุ่นยนต์อิเล็กทรอนิกส์ด้วยเครื่องพิมพ์สามมิติ โดยใช้ชื่อว่า หุ่นยนต์อิเล็กทรอนิกส์ UTHAI และจุดประสงค์เพื่อให้นักวิจัยท่านอื่นสามารถนำไปพัฒนาต่อได้ง่าย ภาพรวมของวิทยานิพนธ์นี้จะแบ่งออกเป็นทั้งหมดสามส่วน คือ ส่วนแรกเป็นส่วนของการออกแบบและจัดสร้าง ส่วนสองของหุ่นยนต์อิเล็กทรอนิกส์ ส่วนที่สองเป็นส่วนของการพัฒนาโปรแกรมที่ใช้ในระบบด้วย ROS และ ส่วนสุดท้ายเป็นส่วนที่ออกแบบระบบพื้นฐานสำหรับการพัฒนาหุ่นยนต์อิเล็กทรอนิกส์ รวมไปถึงมีเอกสาร คู่มือที่อยู่ในรูปแบบออนไลน์

คำสำคัญ : แพลตฟอร์มหุ่นยนต์อิเล็กทรอนิกส์ / ระบบพื้นฐานหุ่นยนต์ / ROS / เครื่องพิมพ์สามมิติ

กิตติกรรมประกาศ

ขอขอบพระคุณอาจารย์ ดร.นัชชา ชูพจน์เจริญ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก ที่ได้สละเวลามาให้คำปรึกษา ชี้แนะแนวทาง ให้ความรู้ในด้านต่างๆ ที่จำเป็นต่องานวิจัย รวมถึงการให้การสนับสนุนในเรื่องอุปกรณ์ในการทำวิจัย ตลอดจนช่วยตรวจสอบและแก้ไขวิทยานิพนธ์ให้เป็นไปอย่างสมบูรณ์

ขอขอบพระคุณรองศาสตราจารย์ ดร.ชิต เหล่าวัฒนา อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม ที่ได้ชี้แนะแนวทางให้คำแนะนำ และให้เกียรติเข้าร่วมการสอบวิทยานิพนธ์

ขอขอบพระคุณอาจารย์ ดร.ภิวดา มณีวรรณ และนายวิษณุ จุราวี ที่ได้ให้คำแนะนำในการแก้ไขปัญหาด้านต่างๆ ที่เกิดขึ้นระหว่างการทำวิจัย และได้ให้การสนับสนุนอุปกรณ์สำคัญที่ใช้ในการทำวิจัย

ขอขอบพระคุณอาจารย์ อับพิพัฒ์ ธิรวงศ์กิจ ที่กรุณาให้เกียรติเป็นประธานกรรมการสอบวิทยานิพนธ์ ให้คำแนะนำที่เป็นประโยชน์ต่อการจัดทำวิทยานิพนธ์ให้ดำเนินไปอย่างสมบูรณ์

ขอขอบพระคุณอาจารย์ ดร.ปิติวุฒย์ ธีรกิตติคุล ที่กรุณาให้เกียรติเป็นกรรมการสอบวิทยานิพนธ์ ให้คำแนะนำที่เป็นประโยชน์ต่อการวิจัย และการแก้ไขปรับปรุงงานวิจัย ตลอดจนตรวจแก้วิทยานิพนธ์ให้ดำเนินไปอย่างสมบูรณ์

ขอขอบพระคุณอาจารย์ ดร.สุภาชัย วงศ์บุณย์ยง ที่กรุณาให้เกียรติเป็นกรรมการสอบวิทยานิพนธ์ ให้คำแนะนำที่เป็นประโยชน์ต่อการวิจัย และการแก้ไขปรับปรุงงานวิจัย ตลอดจนตรวจแก้วิทยานิพนธ์ให้ดำเนินไปอย่างสมบูรณ์

ขอขอบพระคุณคณาจารย์ และบุคลากรในสถาบันวิทยาการหุ่นยนต์ภาคนามทุกท่าน ที่ได้ให้คำปรึกษาและช่วยเหลือด้านสถานที่พร้อมทั้งส่งเสริมความต้องการต่างๆ ในระหว่างการทำวิทยานิพนธ์

ขอขอบคุณนักศึกษาปริญญาตรี สถาบันวิทยาการหุ่นยนต์ภาคนามทุกท่าน ที่ได้ให้คำแนะนำ ถ้ามี แล้วเป็นกำลังใจมาโดยตลอด

และสุดท้ายนี้ ขอน้อมรำลึกถึงพระคุณบิดา มารดา และครอบครัว ที่ส่งเสริมให้กำลังใจ และให้การสนับสนุนในเรื่องต่างๆ จนกระทั่งข้าพเจ้าประสบความสำเร็จในการศึกษา

นายปฐมพงศ์ สินธุรงาม
นายศุภกร เบญจวิกรัย
นายอุกฤษฎ์ เลิศวรรณากุล

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	ค
กิตติกรรมประกาศ	๔
สารบัญ	จ
รายการรูปภาพ	ช
รายการตาราง.....	ณ
รายการสัญลักษณ์.....	ภ
ประมวลศัพท์และตัวย่อ.....	ภ
บทที่ 1 บทนำ.....	1
1.1 ทีมและความสำคัญ.....	1
1.2 วัตถุประสงค์.....	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	1
1.4 ขอบเขตการดำเนินงาน.....	2
1.5 ภาพรวมของระบบและขั้นตอนการดำเนินงาน	2
1.6 Gantt chart	3
1.7 Milestones	3
บทที่ 2 ทฤษฎี/การวิจัยที่เกี่ยวข้อง.....	4
2.1 Labeling tools	4
2.2 Dataset.....	6
2.3 Optical flow.....	21
2.4 Two-Stream CNN.....	23
บทที่ 3 ระเบียบวิธีวิจัย	24
3.1 หน้าที่ความรับผิดชอบ.....	24
3.2 เครื่องมือที่ใช้ในงานวิจัย	24
3.3 ภาษาที่ใช้ในการพัฒนาระบบ	25
3.4 Library computer ที่ใช้ในการพัฒนาระบบและแอพพลิเคชัน.....	25
3.5 แผนการดำเนินงาน	25

สารบัญ (ต่อ)

เรื่อง	หน้า
3.6 การออกแบบแอพพลิเคชัน labeling tool	26
3.6.1 แอพพลิเคชัน labeling tool	26
3.7 การออกแบบระบบวิเคราะห์การกระทำของมนุษย์.....	30
3.7.1 กำหนดพิกัดเฟรมให้กับหุ่นยนต์ชีวามโนยด์.....	30
3.7.2 การแปลงข้อมูลให้อยู่ในรูปแบบ URDF	31
3.7.3 โครงสร้างการติดต่อสื่อสารระหว่าง Node ใน ROS	34
เอกสารอ้างอิง	38
ภาคผนวก ก ข้อมูลเบื้องต้นของหุ่นยนต์ชีวามโนยด์ UTHAI	39
ก.1 ค่าคุณสมบัติทางพลศาสตร์	39
ประวัติผู้เขียน	52

รายการรูปภาพ

รูป	หน้า
รูปที่ 2.1 UI ของโปรแกรม DarkLabel	4
รูปที่ 2.2 UI ของโปรแกรม OpenLabeling	5
รูปที่ 2.3 ตัวอย่าง categories ต่างๆของ YouTube-8M.....	6
รูปที่ 2.4 ขั้นตอนกระบวนการลดขนาดของชุดข้อมูลให้สามารถใช้งานได้่ายยิ่งขึ้น.....	7
รูปที่ 2.5 โครงสร้างของโมเดล DBoF	8
รูปที่ 2.6 (ซ้าย) โครงสร้างจาก Beyond Short Snippets: Deep Networks for Video Classification, (ขวา) ส่วนที่สามารถใช้งานกับ YouTube-8M ได้	9
รูปที่ 2.7 ด้านซ้าย แสดงการสุ่มตัวอย่าง (sampling)วิดีโอ เป็นคีย์เฟรม(keyframes) , ด้านขวา แสดงคีย์เฟรม (keyframes) ที่ถูก labels ซึ่งเป็น Multiple label annotation.....	12
รูปที่ 2.8 แสดงขั้นตอนการทำงานของการเก็บข้อมูลทำชุดข้อมูล.....	12
รูปที่ 2.9 แสดง interface สำหรับสร้าง action label.....	14
รูปที่ 2.10 แสดง interface สำหรับสร้าง action label	15
รูปที่ 2.11 ตัวอย่างของวิดีโอ class เดียวกันไม่จำเป็นต้องเป็น agents เดียวกัน.....	17
รูปที่ 2.12 User interface ของโปรแกรมทำ label	18
รูปที่ 2.13 สถิติของชุดข้อมูลของ Moments in timel.....	19
รูปที่ 2.14 เปรียบเทียบข้อมูลระหว่าง Dataset	19
รูปที่ 2.15 ภาพที่ได้จากการทำ CAM และผลลัพธ์ที่ได้จากการทำนายด้วยโมเดล resnet50-ImageNet ..	21
รูปที่ 2.16 ตัวอย่างการเคลื่อนที่ของลูกบอล	21
รูปที่ 2.17 แสดงโครงสร้างการทำงานของ two stream	23
รูปที่ 3.1 ภาพรวมระบบของแอพพลิเคชั่น labeling tool.....	26
รูปที่ 3.2 หน้าต่างแบบ Select ของแอพพลิเคชั่น labeling tool.....	27
รูปที่ 3.3 หลังจากตัดส่วนวิดีโอด้วย คีย์เฟรมจะถูกเก็บไว้ในช่องรายการตามประเภท	28
รูปที่ 3.4 หน้าต่างแบบ Detect ของแอพพลิเคชั่น labeling tool	29
รูปที่ 3.5 ตัวอย่าง link ใน urdf	32
รูปที่ 3.6 การอธิบาย link ใน URDF ไฟล์	32
รูปที่ 3.7 ตัวอย่าง joint ใน urdf	33
รูปที่ 3.8 การอธิบาย Joint ใน URDF ไฟล์	33

รายการรูปภาพ (ต่อ)

รูป	หน้า
รูปที่ 3.9 การติดต่อสื่อสารระหว่าง Node.....	34
รูปที่ ก.1 ภาพแสดงช่วงล่างทั้งตัว	39
รูปที่ ก.2 ภาพแสดงก้านต่อ Right Hip Yaw.....	40
รูปที่ ก.3 ภาพแสดงก้านต่อ Left Hip Yaw	41
รูปที่ ก.4 ภาพแสดงก้านต่อ Right Hip Roll.....	42
รูปที่ ก.5 ภาพแสดงก้านต่อ Left Hip Roll	43
รูปที่ ก.6 ภาพแสดงก้านต่อ Right Hip Pitch	44
รูปที่ ก.7 ภาพแสดงก้านต่อ Left Hip Pitch.....	45
รูปที่ ก.8 ภาพแสดงก้านต่อ Right Knee Pitch.....	46
รูปที่ ก.9 ภาพแสดงก้านต่อ Left Knee Pitch	47
รูปที่ ก.10 ภาพแสดงก้านต่อ Right Ankle Pitch	48
รูปที่ ก.11 ภาพแสดงก้านต่อ Left Ankle Pitch.....	49
รูปที่ ก.12 ภาพแสดงก้านต่อ Right Ankle Roll.....	50
รูปที่ ก.13 ภาพแสดงก้านต่อ Left Ankle Roll	51

รายการตาราง

ตาราง	หน้า
ตารางที่ 2.1 ข้อมูลเชิงสถิติของ YouTube-8M.....	6
ตารางที่ 2.2 ประสิทธิภาพของโมเดลที่สร้างจาก YouTube-8M ด้วยวิธีต่างๆตามหัวข้อที่ 1 และ 2 โดย แล้วที่ 1 คือ frame-level ไม่เดลและแควรที่ 2 คือ video-level ไม่เดล	10
ตารางที่ 2.3 ประสิทธิภาพของโมเดลเมื่อถูก transfer learning ด้วยชุดข้อมูล Sports-1M โดยใช้ video-level presentation	11
ตารางที่ 2.4 ประสิทธิภาพของโมเดลเมื่อถูก transfer learning ด้วยชุดข้อมูล ActivityNet โดยใช้ video-level presentation	11
ตารางที่ 2.5 ผลการทดลองของวิธีต่างๆบน Frame Level.....	15
ตารางที่ 2.6 ผลการทดลองของวิธีต่างๆบน Video Level.....	16
ตารางที่ 2.7 ประสิทธิภาพของโมเดลเมื่อถูก transfer learning ด้วยชุดข้อมูล ActivityNet โดยใช้ video-level presentation	16
ตารางที่ 2.8 Classification accuracy ของ TOP-1 และ TOP-5	20
ตารางที่ 2.9 Data transfer performance ของโมเดล Resnet50 I3D.....	20
ตารางที่ 3.1 Message Geometry Point	34
ตารางที่ 3.2 Message Geometry Quaternion	35
ตารางที่ 3.3 Message Geometry Pose.....	35
ตารางที่ 3.4 Message Geometry Vector3.....	35
ตารางที่ 3.5 Message Geometry Twist	35
ตารางที่ 3.6 Message Navigation Odometry	35
ตารางที่ 3.7 Message Geometry Pose2D	36
ตารางที่ 3.8 Message Navigation Path.....	36
ตารางที่ 3.9 Message Geometry PoseStamped.....	36
ตารางที่ 3.10 Message Trajectory JointTrajectory.....	37
ตารางที่ 3.11 Message Trajectory JointTrajectoryPoint.....	37
ตารางที่ 3.12 Message Sensor JointState	37
ตารางที่ 3.13 Message Geometry Wrench.....	37
ตารางที่ 3.14 Message Sensor Imu	38
ตารางที่ 3.15 Message Sensor MegneticField	38

รายการตาราง (ต่อ)

ตาราง	หน้า
ตารางที่ ก.1 ตารางแสดงค่าพารามิเตอร์ทั้งตัว.....	39
ตารางที่ ก.2 ตารางแสดงค่าพารามิเตอร์ Right Hip Yaw.....	40
ตารางที่ ก.3 ตารางแสดงค่าพารามิเตอร์ Left Hip Yaw	41
ตารางที่ ก.4 ตารางแสดงค่าพารามิเตอร์ Right Hip Roll.....	42
ตารางที่ ก.5 ตารางแสดงค่าพารามิเตอร์ Left Hip Roll	43
ตารางที่ ก.6 ตารางแสดงค่าพารามิเตอร์ Right Hip Pitch	44
ตารางที่ ก.7 ตารางแสดงค่าพารามิเตอร์ Left Hip Pitch.....	45
ตารางที่ ก.8 ตารางแสดงค่าพารามิเตอร์ Right Knee Pitch.....	46
ตารางที่ ก.9 ตารางแสดงค่าพารามิเตอร์ Left Knee Pitch	47
ตารางที่ ก.10 ตารางแสดงค่าพารามิเตอร์ Right Ankle Pitch	48
ตารางที่ ก.11 ตารางแสดงค่าพารามิเตอร์ Left Ankle Pitch.....	49
ตารางที่ ก.12 ตารางแสดงค่าพารามิเตอร์ Right Ankle Roll.....	50
ตารางที่ ก.13 ตารางแสดงค่าพารามิเตอร์ Left Ankle Roll	51

รายการสัญลักษณ์

θ	เชิงตัว
d	distance
kg	Kilogram
m^2	Square Metre

ประมวลศัพท์และตัวย่อ

UTHAI	Universal Template for Humanoid Algorithm Interface
ROS	Robot Operating System
IMU	Inertial Measurement Unit
Dof	Degree of Freedom
CoM	Center of Mass
ZMP	Zero Moment Point
PLA	Polylactic acid
ABS	Acrylonitrile butadiene styrene
KMUTT	King Mongkut's University of Technology Thonburi
Liws	ลูกิวส์ โซลูชันส์ ทรัพย์
θ	เชิงตัว

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

บริษัท เพอเซ็ปท์รา ดำเนินธุรกิจเกี่ยวกับด้าน artificial intelligence service โดยลูกค้านั้นมีความต้องการที่จะให้ทางบริษัทสร้างปัญญาประดิษฐ์(artificial intelligence) เพื่อนำไปใช้งานหรือแก้ปัญหาที่ต่างกันออกไป ทำการสร้างปัญญาประดิษฐ์ (artificial intelligence) เพื่อตอบสนองความต้องการของลูกค้าเหล่านั้นต้องมีข้อมูลที่เหมาะสมกับปัญหานั้นๆ เช่น ร้านขายของแห่งหนึ่งต้องการรู้ว่าในแต่ละวันมีลูกค้าเดินเข้าร้านกี่คน เป็นผู้ชายกี่คน เป็นผู้หญิงกี่คน เป็นต้น ซึ่งการจะได้ข้อมูลที่เหมาะสมกับงานนั้น ต้องใช้มนุษย์ในการสร้างขึ้นมาโดยการเก็บข้อมูลวิดีโอ และสร้าง label สำหรับใช้ในการสร้างโมเดล machine learning ด้วยตัวเอง ถ้าหากมีวิดีโอดูเป็นจำนวนมาก การที่จะใช้มนุษย์ในการสร้าง label นั้นอาจจะต้องใช้มนุษย์เป็นจำนวนมาก หรือ ก่อให้เกิดภาระแก่มนุษย์ อีกทั้งการสร้าง label

นั้นเป็นงานที่ลำบาก และน่าเบื่อ ทางคณะผู้วิจัยจึงมีความต้องการที่จะออกแบบ และพัฒนา video analytics platform ที่มีเครื่องมือในการสร้าง label สำหรับวิดีโอ เพื่อช่วยแบ่งเบาภาระของผู้พัฒนาในการสร้าง label เพื่อนำไปสร้างโมเดล machine learning สำหรับใช้แก้ปัญหาที่ลูกค้าต้องการ โดยโครงการสหกิจนี้เน้นศึกษาการวิเคราะห์และจัดจำการกระทำของมนุษย์จากภาพเคลื่อนไหวเป็นหลัก

1.2 วัตถุประสงค์

- เพื่อออกแบบ และ สร้างระบบที่สามารถตรวจจับมนุษย์ และจัดจำการกระทำพื้นฐานของมนุษย์ภายในสำนักงาน ประกอบด้วย ยืน นั่ง ใช้คอมพิวเตอร์ เล่นโทรศัพท์ เดิน กินข้าว โดยใช้ปัญญาประดิษฐ์มาประมวลผลกับวิดีโอ
- เพื่อพัฒนาเครื่องมือในการทำ video labeling ใน การสร้างข้อมูลที่ใช้สร้างโมเดลจากวิดีโอ ให้สามารถทำได้ง่าย และ มีประสิทธิภาพที่สูงกว่าเครื่องมือตัวอื่นในปัจจุบัน

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- พัฒนาเครื่องมือในการทำ labeling โดยมี artificial intelligence เข้ามาช่วย ที่สามารถสร้าง label ที่สามารถนำไปใช้สร้างโมเดล machine learning ได้
- พัฒนาต้นแบบของ video analytics platform ที่สามารถรับวิดีโอเข้ามาในระบบแล้วสร้างรายงานเกี่ยวกับกิจกรรมของมนุษย์ในวิดีโอด้วย
- สร้างและทดสอบโมเดลสำหรับทำ action recognition อย่างน้อย 2 โมเดล

1.4 ขอบเขตการดำเนินงาน

1. Labeling tool สามารถตัดวิดีโอเฉพาะในช่วงเวลาที่มีมนุษย์อยู่ได้อัตโนมัติ
2. Labeling tool สามารถระบุตำแหน่งได้ว่ามนุษย์แต่ละคนในวิดีโออยู่ตรงส่วนใดของวิดีโอและสามารถระบุการกระทำของมนุษย์ในวิดีโอด้วย ประกอบด้วยการทำได้แก่ ยืน นั่ง ใช้คอมพิวเตอร์ เล่นโทรศัพท์ เดิน กินข้าว
3. Label ผลลัพธ์ที่ได้จาก labeling tool ต้องสามารถนำไปใช้ในการสร้างโมเดลต่อได้
4. พัฒนา Labeling tool ด้วยภาษา Python
5. พัฒนา Labeling tool ที่สามารถให้มนุษย์ทำงานแก้ไขได้ เมื่อระบบอัตโนมัติทำงานผิดพลาด
6. สร้างโมเดลสำหรับการทำ action recognition อย่างน้อย 2 โมเดลที่สามารถระบุการกระทำของมนุษย์ตามที่กำหนดไว้ได้ เพื่อนำไปใช้ใน video analytics platform
7. Video analytics platform ต้องสามารถนำวิดีโอมวลมาเคราะห์ข้อมูลการกระทำและตำแหน่งของมนุษย์แต่ละคนได้ และนำข้อมูลเหล่านั้นไปสร้างรายงานออกมาได้
8. ความละเอียดอย่างต่ำของวิดีโอต้องมากกว่า 640×480 (ยาว x สูง)
9. วิดีโอจะต้องมีเฟรมเรท (fps) อย่างต่ำ 24 fps

1.5 ภาพรวมของระบบและขั้นตอนการดำเนินงาน

งานวิจัยนี้การดำเนินงานวิจัยถูกแบ่งออกเป็นสองส่วน คือ ส่วนที่หนึ่งส่วนเครื่องมือสำหรับการเตรียมชุดข้อมูล (dataset) เป็นส่วนที่ทำเครื่องมือสำหรับช่วยผู้พัฒนาในการสร้างชุดข้อมูล และส่วนที่สองนำชุดข้อมูลไปสร้างโมเดล

ศึกษาค้นคว้าเอกสารและงานวิจัยที่เกี่ยวข้อง

1. ศึกษาเกี่ยวกับการวิเคราะห์ผลวิดีโอ (video analytics)
2. ศึกษาเกี่ยวกับชุดข้อมูลสำหรับการวิเคราะห์ผลวิดีโอ
3. ศึกษาเกี่ยวกับโมเดลใช้ในการวิเคราะห์ผลวิดีโอ
4. ศึกษาเครื่องมือที่ใช้สำหรับช่วยสร้างชุดข้อมูล

ส่วนเครื่องมือสำหรับการเตรียมชุดข้อมูล (dataset)

1. ออกรูปแบบหน้าต่างของแอพพลิเคชัน
2. สร้างระบบของแอพพลิเคชัน
3. ทดสอบการทำงานของแอพพลิเคชัน

ส่วนนำชุดข้อมูลไปสร้างโมเดล

1. สร้างชุดข้อมูลสำหรับสร้างโมเดล
2. สร้างโมเดลสำหรับการทำนายการกระทำของมนุษย์
3. ทดสอบการทำงานของโมเดล

1.6 Gantt chart

แผนการทำางของคณะวิจัยจะแบ่งออกเป็น 3 ช่วง ซึ่งในแต่ละช่วงจะมี task ดังนี้

1. Phase 1 : ศึกษาข้อมูลเกี่ยวกับการทำวิเคราะห์วิดีโอ(video analytics) และเครื่องมือที่ใช้ในการทำโครงการวิจัย
 - (a) task 1 ศึกษา และ พัฒนาใช้เทคนิคที่เกี่ยวข้องกับการทำวิเคราะห์วิดีโอ(video analytics)
 - (b) task2 รวบรวมชุดข้อมูลสำหรับการทำ human action recognition
2. Phase 2 : สร้างและออกแบบแอพพลิเคชันโดยจะประกอบด้วยงานในส่วนหลักๆ คือ
 - (a) task3 โมเดลสำหรับวิเคราะห์ผลการกระทำของมนุษย์
 - (b) task4 หน้าต่าง UI
3. Phase 3 : ปรับปรุงและแก้ไขปัญหา ทำให้แอพพลิเคชันสามารถนำไปใช้งานได้จริง

1.7 Milestones

จะมีการกำหนด milestones ทั้งหมด 3 ครั้ง คือ

1. Milestone 1 วันที่ 12 มิถุนายน 2562
 - (a) ศึกษางานวิจัยที่เกี่ยวข้องกับการทำวิเคราะห์วิดีโอ(video analytics) ต่าง ๆ ได้แก่ Youtube-8M , Moment in time , AVA
2. Milestone 2 วันที่ 14 สิงหาคม 2562
 - (a) สร้างหน้าต่าง UI ครบสมบูรณ์
 - (b) ความคืบหน้าของโมเดลมากกว่า 50 %-symbol
 - (c) ฟังก์ชันการทำงานของแอพพลิเคชันสามารถทำงานได้มากกว่า 50 %-symbol
3. Milestone 3 วันที่ 15 ตุลาคม 2562
 - (a) แอพพลิเคชันสามารถทำงานได้ครบกระบวนการ
 - (b) ได้โมเดลสามารถทำนายการกระทำของมนุษย์ได้

บทที่ 2

ทฤษฎี/การวิจัยที่เกี่ยวข้อง

ในหัวข้อนี้จะกล่าวถึงการวิจัยที่เกี่ยวข้องกับหัวข้อที่ได้รับมา ซึ่งจะเกี่ยวข้องกับการวิเคราะห์วิดีโอ (video analytics) ใน การวิเคราะห์วิดีโอ (video analytics) นั้นสิ่งที่จำเป็นที่จะต้องทำ ได้แก่ การเตรียมชุดข้อมูล (dataset) สำหรับการทำ label และการทำ machine learning โดยโจทย์ที่ได้รับมาคือการสร้างระบบที่สามารถเข้าใจการกระทำของมนุษย์ตามที่ได้ตั้งขอบเขตไว้ ซึ่งเป็นที่มาที่จะกล่าวถึงการวิจัยเกี่ยวข้องที่ให้ความสนใจได้แก่ labeling tools, Youtube-8M, AVA , Moments in time และทฤษฎีที่เกี่ยวข้อง เป็นหัวข้อที่จะกล่าวถึงต่อไปนี้

2.1 Labeling tools

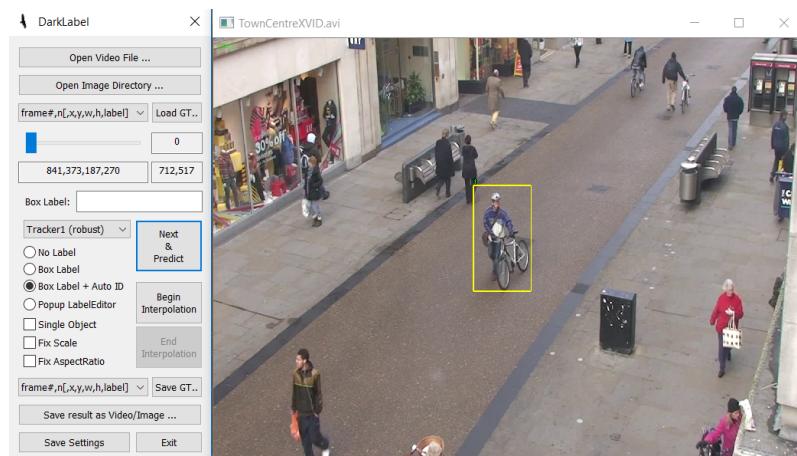
จากการค้นคว้าหาเครื่องมือในการ labeling เพื่อใช้เป็นแนวทางในการทำ Goggle labeling tool พบเครื่องมือที่เป็น open source เปิดให้ทดลองใช้อยู่ 2 เครื่องมือ คือ DarkLabel และ OpenLabeling โดยสรุปข้อสำคัญได้ดังนี้

2.1.1 โปรแกรม DarkLabel

หลังจากใส่วิดีโอเข้าไปในโปรแกรมแล้วมีขั้นตอนการ labeling ดังนี้

- สร้างกรอบสี่เหลี่ยม(boundary box)ครอบบริเวณวัตถุที่สนใจ โดยใช้มนุษย์เป็นคนสร้าง
- กดปุ่ม Next และ Predict อย่างต่อเนื่อง เพื่อ track กรอบสี่เหลี่ยม ในเฟรมถัดๆไป จนกระทั่งการ track เกิดพลาดไป
- ลบกรอบสี่เหลี่ยมที่พลาด และเริ่มทำขั้นตอนที่ 1 ใหม่ อีกครั้งจนครบทุกเฟรมในวิดีโอดังจากที่ได้ทดลองใช้โปรแกรม DarkLabel พบว่า เป็นโปรแกรมที่ค่อนข้างมีการทำงาน

ส่วนใหญ่ที่เป็นแบบทำด้วยมือ ซึ่งทำให้ใช้เวลาในการทำงาน และเสียพลังงานในการทำเป็นอย่างมาก



รูปที่ 2.1: UI ของโปรแกรม DarkLabel

2.1.2 โปรแกรม OpenLabeling

จะมีโปรแกรมการทำงานอยู่ 2 โหมดการทำงาน คือ แบบทำด้วยมือและ แบบอัตโนมัติซึ่งมีการทำงานแยกจากกันอย่างชัดเจน

1. Mode Auto

หลังจากอินพุตวิดีโอเข้าไปในโปรแกรมแล้วมีขั้นตอนการ labeling ดังนี้

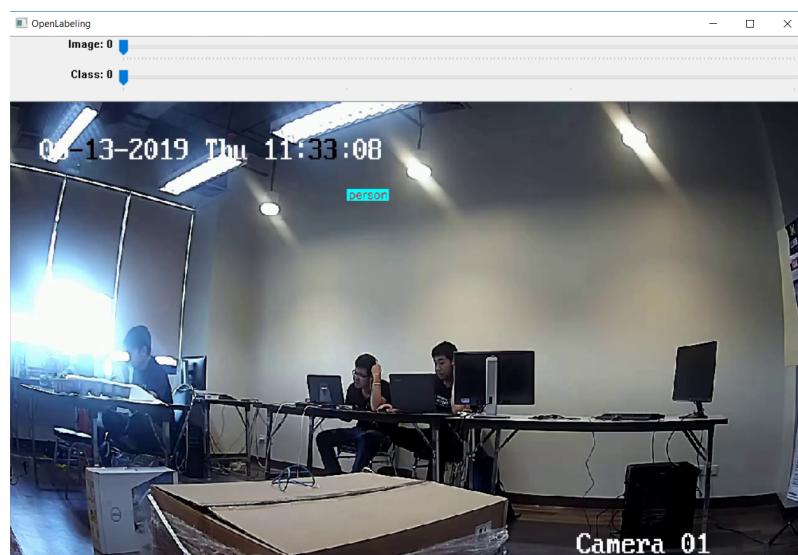
- (a) โปรแกรมจะทำงานอัตโนมัติ โดยใช้โมเดลในการทำนายคีย์เฟรม (predict keyframe) และ track ในภาพที่เหลือ ผลลัพธ์ที่ได้คือ ข้อมูลของชุดข้อมูล

2. Mode Manual

หลังจากอินพุตวิดีโอเข้าไปในโปรแกรมแล้วมีขั้นตอนการ labeling ดังนี้

- (a) สร้างกรอบสี่เหลี่ยม (bounding box) ขึ้นมาโดยใช้มนุษย์เป็นคนสร้าง
- (b) กดปุ่มเพื่อแทร็กกรอบสี่เหลี่ยม (track bounding box) ในเฟรมถัดๆไป จนกระทั่งการแทร็กกรอบสี่เหลี่ยม (track bounding box) เกิดพลาดไป
- (c) ลบกรอบสี่เหลี่ยม (bounding box) ที่พลาด และ เริ่มทำขั้นตอนที่ 1 อีกครั้งจนครบทุกเฟรมในวิดีโอ

หลังจากที่ได้ทดลองใช้โปรแกรม OpenLabeling ทั้ง 2 โหมดการทำงานแล้วพบว่า การทำงานแบบ mode auto การที่เรายังสามารถปรับแก้ไขสิ่งใดในระหว่างกระบวนการ labeling นั้น ทำให้หากเกิดกรณีที่ไม่เดลทำนายกรอบสี่เหลี่ยม (predict bounding) พลาด หรือ เกินมา เราจะไม่สามารถแก้ไขได้ และ การทำงานแบบ mode manual ไม่มีระบบตรวจสอบกรอบสี่เหลี่ยม (detect bounding box) ทำให้ผู้ใช้งานจะต้องสร้างกรอบสี่เหลี่ยม (bounding box) ขึ้นมาเอง

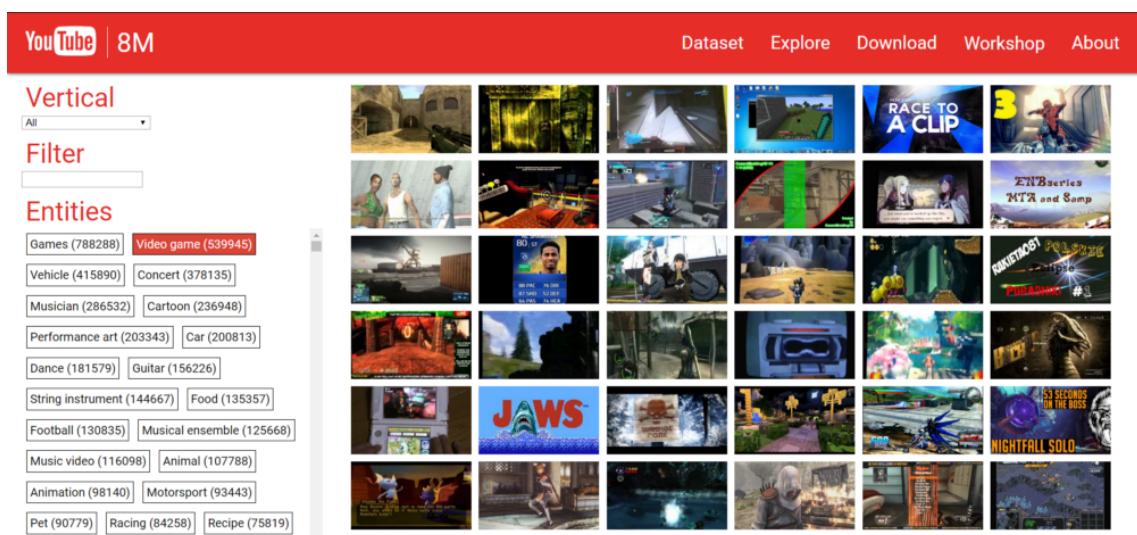


รูปที่ 2.2: UI ของโปรแกรม OpenLabeling

2.2 Dataset

Youtube-8M

YouTube-8M คือชุดข้อมูลวิดีโอที่เป็น multi-label ที่มีจำนวนวิดีโอเยอะที่สุด ซึ่งมีจำนวนมากถึง 8 ล้านวิดีโอ(ในปี 2016) โดยมีจุดมุ่งหมายหลักในการอธิบายรูปแบบของวิดีโอด้วยคำสั้นๆ เช่น ถ้าวิดีโอนั้นเป็นวิดีโอที่มี มนุษย์กำลังปั่นจักรยานบนถนนกับหน้าผา ชุดข้อมูลนี้จะอธิบายวิดีโอนี้ว่า mountain biking ซึ่งทำให้ YouTube-8M แตกต่างจากชุดข้อมูลวิดีโออื่นๆ ส่วนใหญ่ที่จะเน้น action หรือ activity ของมนุษย์ ซึ่งข้อมูลเชิงสถิติจะเป็นดังตารางที่ 1



รูปที่ 2.3: ตัวอย่าง catagories ต่างๆของ YouTube-8M

Number of video	Class of video	Avg. length of each video(s.)	Avg. class of video
8,264,650	4800	229.6	1.8

ตารางที่ 2.1: ข้อมูลเชิงสถิติของ YouTube-8M

1. วิธีการรวบรวมข้อมูล

การเก็บข้อมูลของ YouTube-8M นั้นใช้เครื่องมือที่ชื่อว่า YouTube annotation system ในการเก็บรวบรวมข้อมูลโดยอาศัยผังความรู้(knowledge graph)ของ Google ในการค้นหาและรวบรวมข้อมูลในฐานข้อมูลของ YouTube

1. กฎในการรวบรวมข้อมูลดังนี้

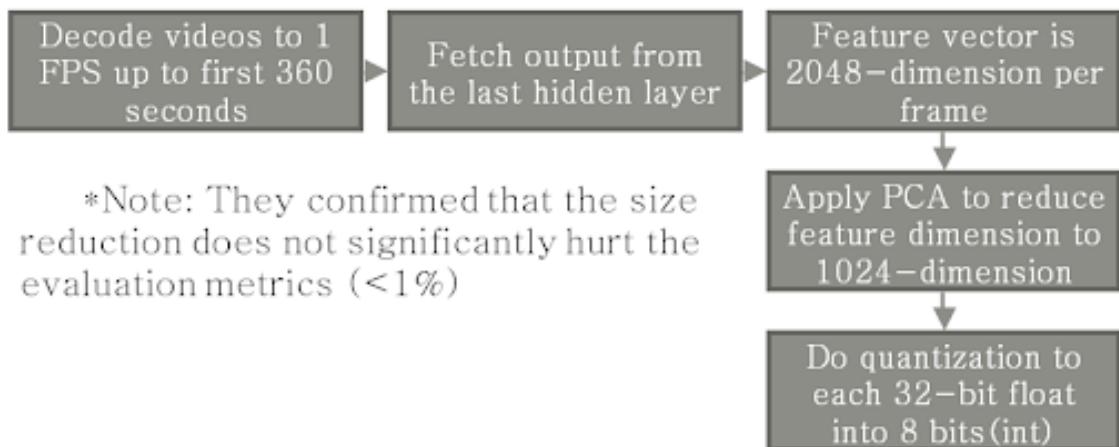
- (a) ทุกๆ หัวข้อต้องเป็นรูปธรรม
- (b) ในแต่ละหัวข้อต้องมีจำนวนวิดีโอมากกว่า 200 วิดีโอ
- (c) ความยาวของวิดีโอต้องอยู่ระหว่าง 120 - 500 วินาที

หลังจากได้กฎในการรวบรวมข้อมูลแล้ว ขั้นตอนต่อไปคือการสร้างคำศัพท์(vocabulary)ที่ใช้ในการค้นหาข้อมูลวิดีโอจากใน YouTube

2. ขั้นตอนในการสร้างคำศัพท์มีดังนี้

- (a) กำหนด whitelist หัวข้อที่เป็นรูปธรรมมา 25 ชนิด เช่น game เป็นต้น
- (b) กำหนด blacklist หัวข้อที่คิดว่าไม่เป็นรูปธรรมไว้ เช่น software เป็นต้น
- (c) รวบรวมหัวข้อที่มีอยู่ใน whitelist อ่าน้อย 1 หัวข้อ และต้องไม่มีอยู่ใน blacklist ซึ่งจะทำให้ได้หัวข้อที่ต้องการมาประมาณ 50,000 หัวข้อ
- (d) จากนั้นใช้ผู้ประเมินจำนวน 3 คน ในการคัดหัวข้อที่คิดว่าเป็นรูปธรรม และสามารถจัดลำหรือเข้าใจได้ง่ายโดยไม่ต้องเขียนชื่อในด้านนั้นๆ ซึ่งผู้ประเมิน ก็จะมีคำถามว่า “ มันยกขนาดไหนถึงจะระบุได้ว่ามีหัวข้อดังกล่าวอยู่ในรูปหรือวิดีโอ โดยใช้เพียงแค่การมองรูปภาพเท่านั้น? ” โดยแบ่งเป็นระดับดังนี้
 - i. บุคคลทั่วไปสามารถเข้าใจได้
 - ii. บุคคลทั่วไปที่ผ่านการอ่านบทความที่เกี่ยวข้องมาแล้วสามารถเข้าใจได้
 - iii. ต้องเสียถูกในด้านใดซักด้านจึงจะเข้าใจได้
 - iv. เป็นไปไม่ได้ ถ้าไม่มีความรู้ที่ไม่ได้เป็นรูปธรรม
 - v. ไม่เป็นรูปธรรม
- (e) หลังจากคำนึงข้างบนและการให้คะแนน จะทำการเก็บไว้เฉพาะหัวข้อที่มีคะแนนเฉลี่ยมากที่สุดอยู่ที่ประมาณ 2.5 คะแนนเท่านั้น
- (f) ทำให้สุดท้ายเหลือเพียงประมาณ 10,000 หัวข้อที่สามารถใช้ได้
- (g) หลังจากได้หัวข้อที่คิดว่าเป็นรูปธรรมแล้วก็นำไปค้นหาและรวบรวมด้วย YouTube annotation system โดยมีขั้นตอนดังนี้
 - i. สุมเลือกวิดีโอมาก 10 ล้านวิดีโอ พร้อมกับหัวข้อของวิดีโอ โดยใช้กฎที่กำหนดไว้ เอาหัวข้อที่มีจำนวนวิดีโอน้อยกว่า 200 วิดีโอออก
 - ii. ทำให้เหลือจำนวนวิดีโอยู่ 8,264,650 วิดีโอ
 - iii. แยกออกเป็น 3 ส่วน Train set, Validate set และ Test set ในอัตราส่วน 70:20:10 ตามลำดับ

เนื่องจากชุดข้อมูลนี้มีขนาดมากกว่า 100 Terabytes และมีความยาวรวมประมาณ 500,000 ชั่วโมง ทำให้การจะใช้คอมพิวเตอร์ทั่วไปเปิดอาจจะใช้เวลานานถึง 50 ปี ทำให้ Google ทำการลดขนาดของข้อมูลลงโดยมีขั้นตอนดังนี้



รูปที่ 2.4: ขั้นตอนกระบวนการกรารลดขนาดของชุดข้อมูลให้สามารถใช้งานได้ง่ายยิ่งขึ้น

2. การทดลองและวิเคราะห์ผล

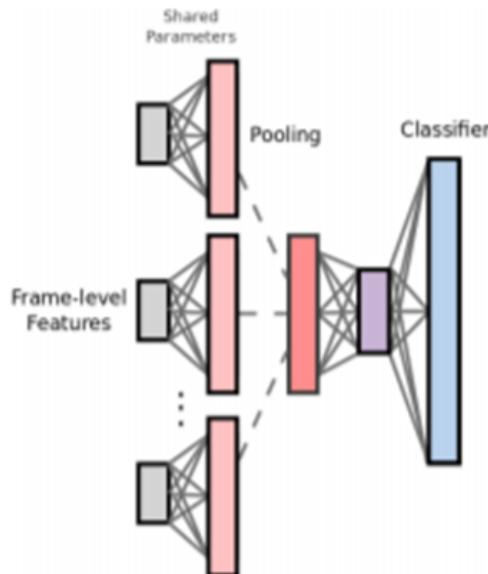
ในบทความ¹ นั้นได้นำเสนอวิธีการในการจัดการข้อมูลซึ่งแบ่งเป็น 2 รูปแบบตามลักษณะของข้อมูลที่ใช้ และอัลกอริทึมหรือเทคนิคที่ใช้ในการสร้างโมเดล ดังนี้

1. คุณลักษณะระดับเฟรม (Frame-level feature)

(a) Frame-Level Models and Average Pooling

อันดับแรกเนื่องจากว่าชุดข้อมูลนี้ไม่มีการระบุหัวข้อในระดับเฟรม จึงใช้วิธีการนำหัวข้อในระดับวิดีโอ มากำหนดให้กับทุกๆเฟรมในวิดีโอแทน จากนั้นสุมเฟรมมา 20 เฟรมในแต่ละวิดีโอ ทำให้มีเฟรมถึง 120 ล้านเฟรม ซึ่งในแต่ละหัวข้อ e ทำให้มี (x_i, y_i^e) 120 ล้านคู่ โดยที่ $x_i \in R^{1024}$ คือคุณลักษณะที่ได้มาจากการ hidden layer สุดท้ายก่อนจะเป็น fully connected และ $y_i^e \in 0, 1$ คือหัวข้อสำหรับหัวข้อ e ของตัวอย่างที่ i^{th} แล้วสร้างโมเดลทั้งหมด 4,800 โมเดลที่เป็นโมเดลแบบ one vs all classifier และเป็นอิสระต่อกันสำหรับแต่ละหัวข้อ และเนื่องจากการประเมินผลนั้นมีพื้นฐานมาจากหัวข้อในระดับวิดีโอ ทำให้ต้องทำการรวมความน่าจะเป็นของแต่ละหัวข้อในระดับเฟรมไปเป็นความน่าจะเป็นในระดับวิดีโอ โดยใช้การเฉลี่ยค่าความน่าจะเป็นทั้งหมดในหัวข้อนั้นๆ และใช้ average pooling เพื่อลดผลจากการตรวจจับความผิดปกติและความโดยเด่นของข้อมูลของแต่ละหัวข้อภายในวิดีโอ

(b) Deep Bag of Frames (DBoF) Pooling



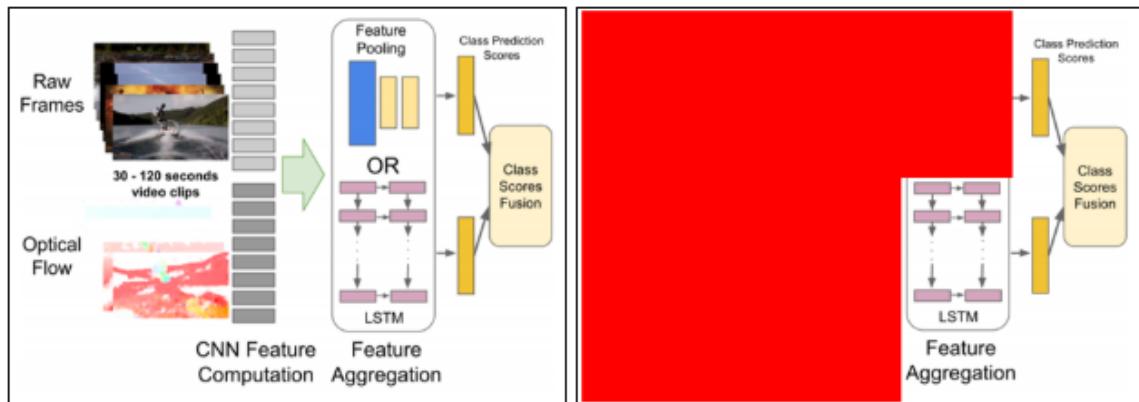
รูปที่ 2.5: โครงสร้างของโมเดล DBoF

หลักการคล้ายๆกับ Deep Bag of Words โดยที่จะสุมเฟรม มา k เฟรม โดยที่แต่ละเฟรมเป็น N dimension input มาผ่าน fully connected ที่มี M units ($M > N$) และใช้ RELU activations และทำ batch normalization ก่อนจะนำรวมด้วย max pooling โดยที่ทั้งโครงข่ายใช้ Stochastic Gradient Descent(SGD)

¹YouTube-8M, <https://arxiv.org/pdf/1609.08675.pdf>

(c) Long short-term memory(LSTM)

ในบทความ² นี้ได้ใช้ LSTM แบบเดียวกับของ Beyond Short Snippets: Deep Networks for Video Classification³ แต่เนื่องจาก YouTube-8M นั้นผ่านการทำ preprocess มาแล้วทำให้ไม่สามารถใช้ raw video frame ได้ จึงทำได้เฉพาะ LSTM และ softmax layer เท่านั้น ตามรูปที่ 2.6



รูปที่ 2.6: (ซ้าย) โครงสร้างจาก Beyond Short Snippets: Deep Networks for Video Classification, (ขวา) ส่วนที่สามารถใช้งานกับ YouTube-8M ได้

2. คุณลักษณะระดับวิดีโอ (Video-level feature)

(a) Video-level representation

ในบทความ⁴ นี้ได้สำรวจวิธีการแยกเวกเตอร์คุณลักษณะระดับวิดีโօความยากง่ายที่จากคุณลักษณะระดับเฟรมซึ่งการทำแบบนี้ทำให้ได้ประโยชน์ 3 ข้อ คือ 1) โมเดลทั่วไปที่ไม่ใช่ neural network สามารถนำไปใช้งานได้ 2) ขนาดข้อมูลเล็กลง 3) เหมาะกับการนำไปสร้างโมเดล domain adaptive มากขึ้น

i. First, Second order and ordinal statistic

จากคุณลักษณะในระดับเฟรม $x_{1:F_v}^v$ โดยที่ x_j^v คือคุณลักษณะระดับเฟรมในเฟรมที่ j ของวิดีโօ v และ F_v คือจำนวนเฟรมทั้งหมดของวิดีโօ v ทำการหาค่าเฉลี่ย μ_v และส่วนเบี่ยงเบนมาตรฐาน σ_v พร้อมทั้งดึง ordinal statistics 5 อันดับแรกของแต่ละ dimension K ออกมา $Top_k(x^v(j)_{1:F_v})$ จะทำให้ได้เวกเตอร์คุณลักษณะ(feature-vector) $\varphi_{1:F_v}^v$ ของวิดีโօเป็นดังนี้

$$\varphi_{1:F_v}^v = \begin{bmatrix} \mu_{1:F_v}^v \\ \sigma_{1:F_v}^v \\ Top_k(x^v(j)_{1:F_v}) \end{bmatrix}$$

ii. Feature normalization

ก่อนที่จะทำการสร้าง one vs all classifiers แต่ละตัวนั้นได้ทำการ normalization เวกเตอร์คุณลักษณะ $\varphi_{1:F_v}^v$ จากนั้นนำค่าเฉลี่ย μ_v ออกแล้วใช้ PCA ในการลด มิติของข้อมูล ซึ่งการทำแบบนี้นั้นทำให้การสร้างโมเดลเป็นไปได้เร็วขึ้น

²YouTube-8M,<https://arxiv.org/pdf/1609.08675.pdf>

³AVA,<https://arxiv.org/pdf/1705.08421.pdf>

⁴YouTube-8M,<https://arxiv.org/pdf/1609.08675.pdf>

โดยการสร้างโมเดลด้วย video-level presentation นั้น บทความ⁵ นี้ได้หยิบมาทดสอบ 3 อัลกอริทึม

(b) Model training algorithm approaches

- i. Logistic Regression
- ii. Hinge Loss
- iii. Mixture of Experts (MoE)

(c) Evaluation metrics

- i. Mean Average Precision (mAP)
- ii. Hit@k
- iii. Precision at equal recall rate (PERR)

3. Results

(a) Baseline on YouTube-8M dataset

Input Features	Modeling Approach	mAP	Hit@1	(PERR)
Frame-level, $(x_{1:F_v}^v)$	Logistic + Average	11.0	50.8	42.2
Frame-level, $(x_{1:F_v}^v)$	Deep Bag of Frames	26.9	62.7	55.1
Frame-level, $(x_{1:F_v}^v)$	LSTM	26.6	64.5	57.3
Video-level, μ	Hinge loss	26.6	64.5	57.3
Video-level, μ	Logistic Regression	26.6	64.5	57.3
Video-level, μ	Mixture-of-2-Expert	26.6	64.5	57.3
Video-level, $\mu; \sigma; Top_5$	Mixture-of-2-Expert	26.6	64.5	57.3

ตารางที่ 2.2: ประสิทธิภาพของโมเดลที่สร้างจาก YouTube-8M ด้วยวิธีต่างๆตามหัวข้อที่ 1 และ 2 โดยแก้วที่ 1 คือ frame-level โมเดลและแก้วที่ 2 คือ video-level โมเดล

จากตารางที่ 2.2 จะเห็นว่าการทำ video-level features จากการหาค่าเฉลี่ยของ frame-level features แล้วสร้างโมเดลด้วย Hinge loss หรือ โมเดล Logistic Regression นั้นสามารถเพิ่มประสิทธิภาพได้ไม่น้อย และจากการทดลองทำให้เห็นว่า LSTM ที่มีความลึก 2 layers นั้นสามารถทำให้ผลลัพธ์เป็น state-of-the-art ในขณะนั้นได้ เนื่องจากในขณะที่ DBoF นั้นไม่ได้สนใจลำดับของเฟรม แต่ LSTM ใช้ state information เพื่อคงลำดับของเฟรมเอาไว้

LSTM นั้นดีที่สุดยกเว้น mAP, เนื่องจาก one-vs-all binary MoE classifier นั้นมีประสิทธิภาพดีกว่า, LSTM สามารถเพิ่มประสิทธิภาพบน Hit@1 และ PERR ได้เนื่องจากความสามารถในการเรียนรู้ความสัมพันธ์ระยะยาวในโดเมนของเวลา

⁵YouTube-8M, <https://arxiv.org/pdf/1609.08675.pdf>

(b) Transfer learning video-level presentation from YouTube-8M to Sports-1M dataset

Approach	mAP	Hit@1	(Hit@1)
Logistic Regression (μ)	58.0	60.1	79.6
Mixture-of-2-Expert (μ)	59.1	61.5	80.4
Mixture-of-2-Expert ($[\mu; \sigma; Top_5]$)	61.3	63.2	82.6
LSTM	66.7	64.9	85.6
+Pretrained on YT-8M	67.6	65.7	86.2
Hierarchical 3D Convolution	-	61.0	80.0
Stacked 3D Convolutions	-	61.0	85.0
LSTM with Optical Flow and Pixels	-	73.0	91.0

ตารางที่ 2.3: ประสิทธิภาพของโมเดลเมื่อถูก transfer learning ด้วยชุดข้อมูล Sports-1M โดยใช้ video-level presentation

จากตารางที่ 2.6 จะเห็นว่าโมเดล LSTM ที่ถูก pretrained จาก YouTube-8M นั้นมีประสิทธิภาพ ที่ดีกว่า ยกเว้น LSTM with Optical Flow and Pixels ที่มีการใช้ข้อมูลการเคลื่อนไหว(optical flow) ในการสร้างโมเดลด้วย

(c) Transfer learning video-level presentation from YouTube-8M to ActivityNet dataset

Approach	mAP	Hit@1	(Hit@1)
Mixture-of-2-Expert (μ)	69.1	68.7	85.4
+Pretrained PCA on YT-8M	74.1	72.5	89.3
Mixture-of-2-Expert ($[\mu; \sigma; Top_5]$)	NO	74.2	72.3
+Pretrained PCA on YT-8M	77.6	74.9	91.6
LSTM	57.9	63.4	81.0
+Pretrained on YT-8M	75.6	74.2	92.4
Ma, Bargal et al.	53.8	-	-
Heilbron et al.	43.0	-	-

ตารางที่ 2.4: ประสิทธิภาพของโมเดลเมื่อถูก transfer learning ด้วยชุดข้อมูล ActivityNet โดยใช้ video-level presentation

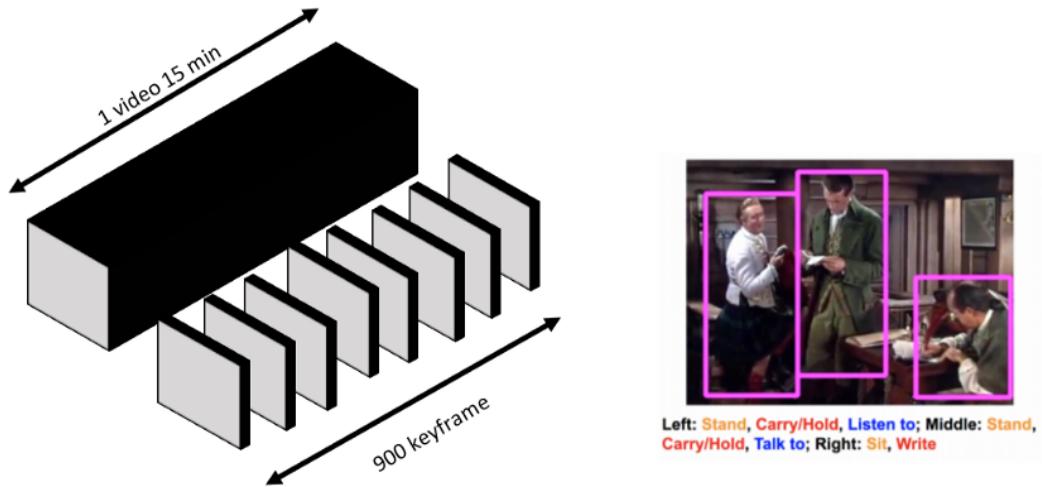
จากตารางที่ 2.7 จะเห็นว่าโมเดลที่ถูก pretrained จาก YouTube-8M นั้นมีประสิทธิภาพที่ดีขึ้น มากเมื่อเทียบกับ benchmark ก่อนหน้า

3. ปัญหาที่พบ

เนื่องจากว่า YouTube-8M นั้นมีจำนวนข้อมูลที่เยอะมาก ทำให้ไม่สามารถตรวจสอบได้ทั้งหมดว่า ground-truth ของแต่ละวิดีโอนั้นมีความถูกต้องมากน้อยขนาดไหน ทำให้อาจเกิดข้อผิดพลาดได้ (ปัจจุบัน ปี 2019 YouTube-8M ได้มีการตรวจสอบข้อมูลอีกรอบ เพื่อเพิ่มประสิทธิภาพของชุดข้อมูลซึ่งทำให้ปัจจุบันจำนวนข้อมูล และจำนวน category นั้นจะลดน้อยลงจากข้อมูลที่ใช้อ้างอิงในบทความ⁶ ข้างต้นที่ได้กล่าวมา)

⁶YouTube-8M, <https://arxiv.org/pdf/1609.08675.pdf>

AVA (Atomic Visual Action)



รูปที่ 2.7: ด้านซ้าย แสดงการสุ่มตัวอย่าง (sampling)วิดีโอ เป็นคีย์เฟรม(keyframes) , ด้านขวา แสดงคีย์เฟรม (keyframes) ที่ถูก labels ซึ่งเป็น Multiple label annotation

AVA คือ ชุดข้อมูลที่รวมวิดีโอที่มีความยาว 15 นาทีและจะถูกแบ่งด้วยความถี่ 1 hz (900 keyframes) จากในหนังโดยการกราฟทำของมนุษย์เป็นศูนย์กลาง เพื่อใช้สำหรับสร้างโมเดลที่เข้าใจกิจกรรมของมนุษย์ในวิดีโอด้วยการคำนวณนิพัทธ์ที่มีความแม่นยำ ซึ่งข้อดีของ AVA คือ ชุดข้อมูลจะมีคำอธิบาย (label) เป็นแบบ multiple label (ในหนึ่งกรอบสี่เหลี่ยม (bounding box) สามารถมีคำอธิบายได้หลายคำอธิบาย) และคำอธิบายของ AVA (label) มีจำนวน 80 class สามารถแบ่งได้เป็น 3 หมวดหมู่ คือ ท่าทาง (Pose) , ปฏิสัมพันธ์กับวัตถุ (Interaction with object) และ ปฏิสัมพันธ์กับบุคคล (Interaction with people) ซึ่งสามารถมีคำอธิบายได้มากถึง 7 คำอธิบาย

1. วิธีการรวบรวมข้อมูล

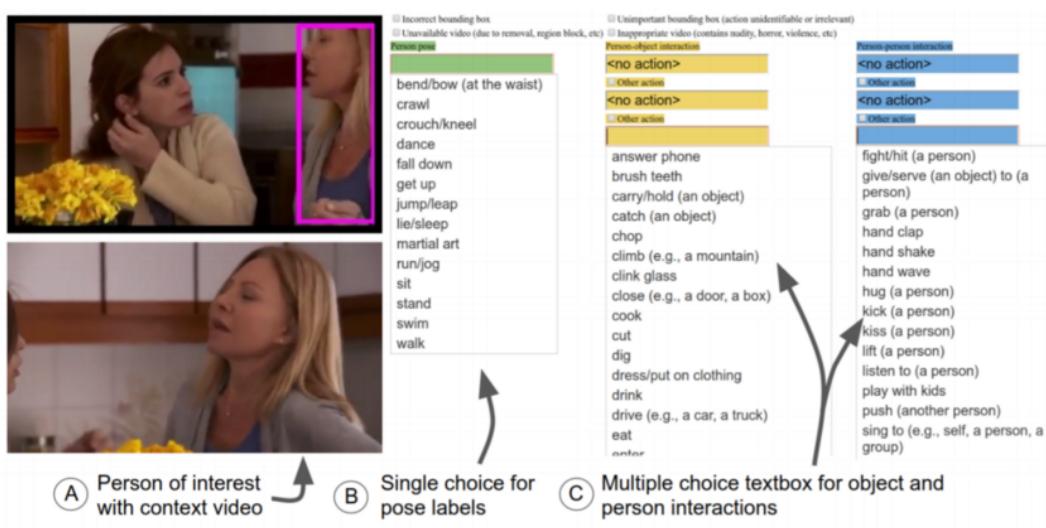


รูปที่ 2.8: แสดงขั้นตอนการทำงานของการเก็บข้อมูลทำชุดข้อมูล

ขั้นตอนการเก็บข้อมูลสำหรับการทำชุดข้อมูลมีขั้นตอนการทำ 5 ขั้น คือ

1. การสร้างคำศัพท์การกระทำ (verb generation) จะมีหลัก 3 ข้อในการรวบรวมคำศัพท์ คือ
 - (a) เก็บรวบรวมคำศัพท์ทั่วไปที่เกิดขึ้นในชีวิตประจำวัน
 - (b) จะต้องมีเอกสารภาษาญี่ปุ่นได้ชัดเจน เช่น การถือของ
 - (c) กำหนดรูปแบบของคำศัพท์ขึ้นมาและใช้ความรู้จากชุดข้อมูลอื่น ในการทำให้ได้ class การกระทำ ของมนุษย์ที่ครอบคลุมของชุดข้อมูล AVA
2. หนังและส่วนที่เลือกมาใช้ (Movie and segment selection) วิดิโอที่ใช้ทำชุดข้อมูล AVA ทั้งหมดจะถูกนำมากจาก youtube โดยเริ่มจากการรวบรวมรายการซึ่งของนักแสดงที่มีชื่อเสียง ซึ่งจะมีความหลากหลายของเชื้อชาติรวมกันอยู่ ซึ่งวิดิโอที่ถูกคัดเลือกจะมีเกณฑ์ดังนี้ คือ
 - (a) วิดิโอต้องอยู่ในหมวด หนัง และ ละครโทรทัศน์
 - (b) จะต้องมีความยาวมากกว่า 30 นาที
 - (c) อัพโหลดเป็นเวลาอย่างน้อย 1 ปี
 - (d) มียอดวิวคนดูมากกว่า 1000 วิว
 - (e) ลงทะเบียนวิดิโอบางประเภท เช่น ข่าว-ดำเนิน , ความลับเอี้ยดต่อ , การ์ตูน , วิดิโอเกม
 - (f) ในการเลือกวิดิโอที่มีข้อจำกัดจะต้องมีวิธีการเลือก คือ
 - i. ไม่ทำการกรองวิดิโอออกด้วย action keywords
 - ii. ไม่ทำให้เป็น uniform label distribution
 - iii. เลือกแค่ส่วนหนึ่งของหนัง คือ ช่วงนาทีที่ 15 - 30 เนื่องจากต้องการที่จะข้ามส่วนต้นของหนัง ซึ่งอาจเป็น ตัวอย่างของหนัง หรือ โฆษณา
3. การตีกรอบบุคคลที่อยู่ภายในภาพ(Person bounding box annotation) ประกอบด้วย 2 ขั้นตอน
 - (a) สร้างกรอบสี่เหลี่ยม (bounding boxes) โดยใช้โมเดล Faster R-CNN สำหรับการตรวจจับมนุษย์
 - (b) นำมนุษย์มาใช้ในการตรวจสอบและแก้ไขกรอบสี่เหลี่ยม (bounding boxes) ที่พลาดไป หรือ ตรวจจับผิด
 - (c) การเชื่อมของบุคคลในช่วงระยะเวลาสั้นๆของเฟรม(Person link annotation) ทำการเชื่อมกรอบสี่เหลี่ยม (bounding boxes) ที่อยู่ในช่วงเวลาเดียวกัน ซึ่งใช้วิธีการ track โดยยึดมนุษย์เป็นศูนย์กลาง ซึ่งจะนำมาคำนวณความใกล้เคียงกันโดยการจับคู่กรอบสี่เหลี่ยม (bounding box) และใช้ person embedding จากนั้นจะใช้ Hungarian algorithm ในการทำตัวเลือกที่ดีที่สุด

2. การสร้างคำอธิบาย (Action annotation)

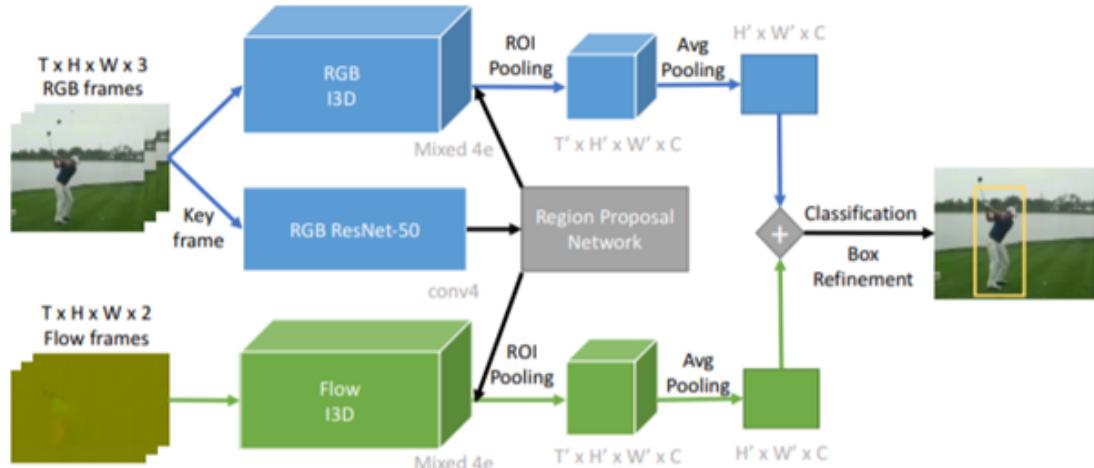


รูปที่ 2.9: แสดง interface สำหรับสร้าง action label

การสร้าง action labels จะถูกสร้างจากเหล่าคนที่เป็น annotators ซึ่งจะใช้ interface ในการสร้าง ซึ่งใน 1 กรอบสี่เหลี่ยม (bounding box) สามารถมี action labels ได้สูงสุดถึง 7 labels นอกจากนั้นสามารถตั้งสถานะบล็อก content ที่ไม่เหมาะสม หรือกรอบสี่เหลี่ยมที่ผิดพลาด (incorrect bounding box) ได้อีกด้วย ในทางปฏิบัติจะสังเกตได้วามันมีโอกาสผิดอย่างหลีกเลี่ยงไม่ได้ เมื่อต้องได้รับคำสั่งให้หา action labels ที่ถูกต้อง จาก 80 class จึงแบ่งขั้นตอนออกเป็น 2 ขั้นตอน คือ

1. Action proposal สอบถามเหล่า annotator เพื่อสร้างข้อเสนอสำหรับ action labels จากนั้นจับกลุ่มเข้าด้วยกัน ซึ่งจะทำให้มีโอกาสถูกต้องมากกว่าเป็นข้อเสนอแยกเดี่ยว
2. Verification annotator จะตรวจสอบข้อเสนอที่ได้จากขั้นตอนแรก ซึ่งในแต่ละวิดีโอดิจิทัลจะใช้มนุษย์ในการตรวจสอบ 3 คน เมื่อ action label ถูก annotator อย่างน้อย 2 คน ตรวจสอบ action label นั้นจะถูกยึดเป็นคำอธิบายหลัก

3. การทดลองและวิเคราะห์ผล



รูปที่ 2.10: แสดง interface สำหรับสร้าง action label

สำหรับโมเดลที่บุกเบิกความ[2]นี้พูดถึงคือ two stream variant ซึ่งจะทำการประมวลผลทั้ง RGB flow และ optical flow และ เป็นโครงสร้างของ Faster RCNN ที่นำ Inception network เข้ามาใช้

1. การทดลองที่ 1 ทดสอบว่าโมเดลได้ให้ประสิทธิภาพการทำงานได้ดีที่สุด

- (a) รายละเอียดการทดลอง : นำชุดข้อมูล JHMDB และ UCF 101 มาเป็นชุดข้อมูลในการทดสอบ ซึ่งการทดลองจะทดสอบด้วย frame level และ video level และมี metrics ในการวัด คือ ใช้ค่า IOU (intersection over union)
- (b) สำหรับ video level จะคำนวณ 3D IOUs ซึ่งเป็นการเปรียบเทียบระหว่าง ground truth tubes และ linked detection tubes (ground truth tube คือ การนำเอกสารอบสี่เหลี่ยม (bounding box) จริงของวัตถุในเฟรมที่ติดต่อกันมาเรียงต่อกันเป็น tube และ linked detection tube คือ การนำเอกสารอบสี่เหลี่ยม (bounding box) ที่ตรวจเจอมาระเบิดต่อ กันเป็น tube) โดยตั้งค่าเกณฑ์ (threshold) ที่ 0.5 และรายงานผลออกเป็น mean average precision

Frame-mAP	JHMDB	UCF101-24
Actionness	39.9	-
Peng w/o MR	56.9	64.8
Peng w/ MR	58.5	65.7
ACT	65.7	69.5
Out approach	73.3	76.3

ตารางที่ 2.5: ผลการทดลองของวิธีต่างๆบน Frame Level

- (c) ผลการทดลองของ frame level ส่วนตารางด้านล่าง คือ video level ซึ่งผลการทดลองได้ผลลัพธ์ คือ วิธี two stream ได้ค่า mAP มากกว่า วิธีการอื่นๆ ทั้ง frame level , video level

2. การทดลองที่ 2 นำโมเดล 2 stream มาทดลองกับชุดข้อมูล AVA ซึ่งได้ผลลัพธ์ดังนี้

- (a) ผลการทดลองของ frame level ส่วนตารางด้านล่าง คือ video level ซึ่งผลการทดลองได้ผลลัพธ์ คือ วิธี two stream ได้ค่า mAP มากกว่า วิธีการอื่นๆ ทั้ง frame level , video level

	Video-mAP	JHMDB	UCF101-24
Peng w/ MR	73.1	35.9	
Singh	72.0	46.3	
ACT	73.7	51.4	
TCNN	76.9	-	
Out approach	78.6	59.9	

ตารางที่ 2.6: ผลการทดลองของวิธีต่างๆบน Video Level

Model	Temp + Mode	JHMDB	UCF101-24	AVA
2D	1 RGB + 5 Flow	52.1	60.1	13.7
3D	5 RGB + 5 Flow	67.9	76.1	13.6
3D	10 RGB + 10 Flow	73.4	78.0	14.6
3D	20 RGB + 20 Flow	76.4	78.3	15.2
3D	40 RGB + 40 Flow	76.7	76.0	15.6
3D	50 RGB + 50 Flow	-	73.2	15.5
3D	20 RGB	73.2	77.0	14.5
3D	20 Flow	67.0	71.3	9.9

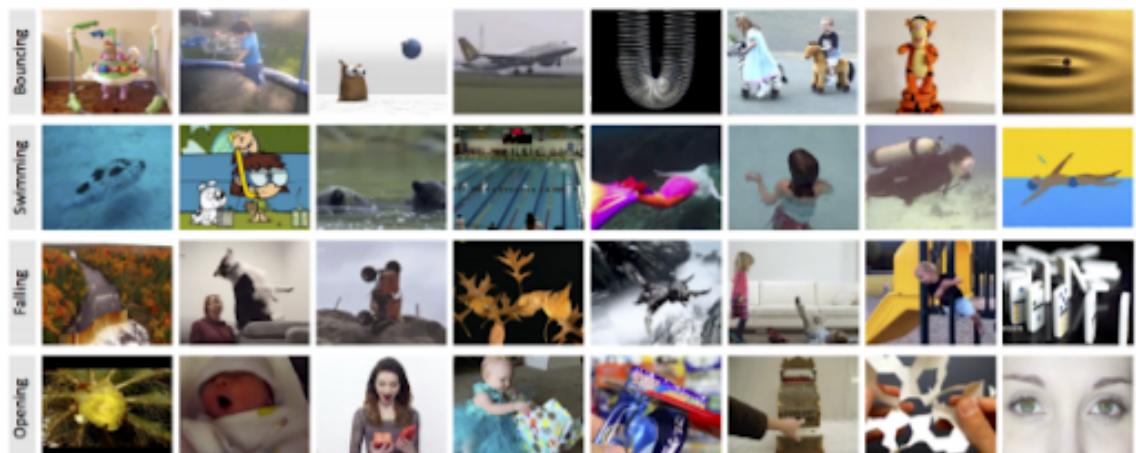
ตารางที่ 2.7: ประสิทธิภาพของโมเดลเมื่อถูก transfer learning ด้วยชุดข้อมูล ActivityNet โดยใช้ video-level presentation

4.สรุปผลการทดลอง

1. สำหรับโมเดล 2 stream ที่เป็น 3D จะได้ประสิทธิภาพมากกว่า 2D
2. สำหรับ AVA 3D โมเดลจะทำงานได้ดีหลังจากผ่านไปมากกว่า 10 เฟรม
3. จะทำให้สังเกตได้ถึงการเพิ่มขึ้นของความยาวของ temporal window
4. การนำ RGB , optical flow มารวมกันจะทำงานได้มีประสิทธิภาพมากกว่าก่าว้าง input
5. JHMDB และ UCF101-24 ผลการทำงานจะ saturate ที่ 20 เฟรม

Moments in time

Moments in time⁷ คือชุดข้อมูลที่ใช้มนุษย์ในการ label ทั้งหมดให้กับวิดีโอสั้นถึง 1 ล้านวิดีโอ และมีจำนวน activity หรือกรรทำต่างกัน 339 class โดยแต่ละวิดีโอมีความยาวอยู่ที่ 3 วินาที เนื่องจากเป็นเวลาเฉลี่ยที่มนุษย์ใช้ในการเข้าใจกับเหตุการณ์ที่เกิดขึ้น (human working memory) รูปแบบของชุดข้อมูลจะมีอยู่ทั้งหมดอยู่ 3 รูปแบบ ได้แก่ ภายนอก (spatial) เสียง (auditory) และการเคลื่อนไหว (temporal) นอกจากนี้ชุดข้อมูลนี้นั้นไม่รวมเพียงแค่การกระทำของมนุษย์เท่านั้น ยังรวมไปถึง สัตว์ สิ่งของ และ ปรากฏการณ์ธรรมชาติ ทำให้ ชุดข้อมูลนี้เป็นการท้าทายรูปแบบใหม่ เพราะด้วยชุดข้อมูลที่มีความซับซ้อนมากขึ้น เช่น การสร้างโมเดลที่สามารถบอกถึงการกระทำ (action) ได้ถึงแม้ว่าสิ่งที่เราสนใจ (มนุษย์ สัตว์ สิ่งของ หรือปรากฏการณ์ธรรมชาติ) จะแตกต่างกัน เป็นต้น



รูปที่ 2.11: ตัวอย่างของวิดีโอ class เดียวกันไม่จำเป็นต้องเป็น agents เดียวกัน

เป้าหมายของชุดข้อมูล Moments in time คือการออกแบบชุดข้อมูลให้มีความหลากหลาย ครอบคลุมความสมดุล และจำนวนข้อมูลที่สูง โดยที่แต่ละ activity หรือการกระทำนั้นจะประกอบไปด้วยวิดีโอมากกว่า 1,000 วิดีโอ และมีการออกแบบมาเพื่อให้สามารถพัฒนาต่อได้ เช่น จำนวน class และชุดข้อมูลภายใน class นั้น ๆ

⁷Moment, <http://moments.csail.mit.edu/TPAMI.2019.2901464.pdf>

1. วิธีการรวมข้อมูล

เริ่มจากการรวมคำ (verb) ที่มีการใช้อยู่ทั่วไปในชีวิตประจำวันมา 4,500 คำจาก VerbNet จากนั้นนำมาแบ่งกลุ่มคำ(verb) ที่มีความหมายใกล้เคียงกันโดยใช้ features จาก Propbank และ FrameNet โดยเก็บข้อมูลเป็นแบบ binary feature vector ซึ่งถ้าคำ (verb) ไหนมีความเกี่ยวข้องกับ feature ก็จะให้ค่าเป็น 1 ถ้าไม่เกี่ยวข้องกันจะให้ค่าเป็น 0 จากนั้นจึงใช้วิธี k-means clustering ในการแบ่งกลุ่ม เมื่อแบ่งกลุ่มแล้วจากนั้นจะเลือกคำ (verb) จากในแต่ละกลุ่มนั้น โดยคำ (verb) ที่เลือกมานั้นจะเป็นที่ใช้บ่อยที่สุดในกลุ่มนั้น และลบคำ (verb) นั้นออกจากกลุ่มทั้งหมด (คำ ๆ หนึ่งสามารถอยู่ได้หลายกลุ่ม) จากนั้นจะทำการบันการนี้ไปเรื่อย ๆ แต่คำ (verb) ที่เลือกมาจะต้องไม่มีความหมายคลุมเครือ ไม่สามารถมองเห็นหรือได้ยินได้ และต้องไม่มีความหมายเหมือนกับคำ (verb) ที่เคยเลือกมาก่อน จนสุดท้ายแล้วได้ออกมาที่ 339 class

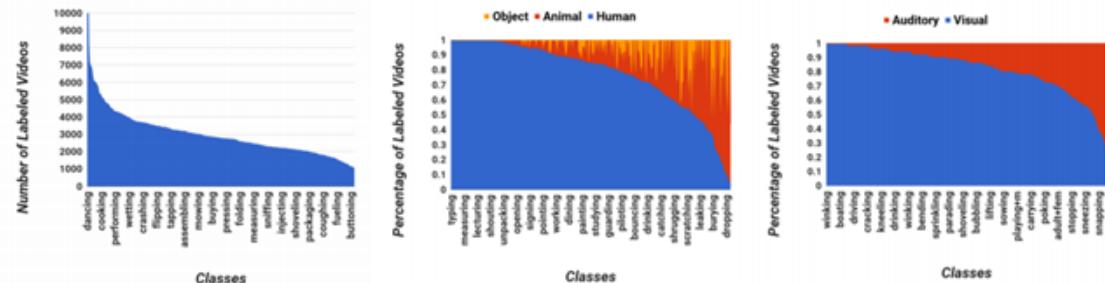
ต่อมาทำการหาชุดข้อมูลวิดีโอด้วยจะตัดออกมาเพียง 3 วินาทีที่เกี่ยวข้องกับคำ (verb) ใน 339 class ที่เลือกมา จากวิดีโอ แหล่งต่างกัน 10 แหล่ง การตัดวิดีโอนั้นจะไม่ใช้พาก Video2Gif (โมเดลที่ระบุตำแหน่งของสิ่งที่น่าสนใจในวิดีโอ) เพราะจะทำให้เกิด bias ขึ้นจะเกิดขึ้นตอนสร้างโมเดลจากนั้นจะทำการส่งข้อมูลของคำ (verb) และวิดีโอที่ตัดไปยัง Amazon Mechanical Turk (AMT หรือตลาดแรงงาน) เพื่อทำการ label โดยพนักงานแต่ละคนของ AMT จะได้ 64 วิดีโอซึ่งเกี่ยวข้องกับคำ (verb) หนึ่ง และอีก 10 วิดีโอที่มีการทำ label อยู่แล้ว โดยวิดีโอที่มีการทำ label ถ้ามีพนักงานของ AMT ตอบเหมือนกันกับที่ทำ label ไว้เกิน 90% ถึงจะนำเข้าไปรวมกับชุดข้อมูลส่วนอีก 64 วิดีโอถ้าเป็นของ training set จะต้องผ่านพนักงานของ AMT อย่างน้อย 3 ครั้ง และต้อง label เหมือนกัน 75% ขึ้นไปถึงจะถือว่าเป็น label ที่ถูกต้อง ถ้าเป็นของ validation และ test set จะต้องผ่านพนักงานของ AMT อย่างน้อย 4 ครั้ง และต้อง label เหมือนกัน 85% ขึ้นไป ที่ไม่ต่างกันทั่วไป 100% เพราะจะทำให้วิดีโอน้ำมากเกินไปที่จะทำให้สามารถจำการกระทำได้



รูปที่ 2.12: User interface ของโปรแกรมทำ label

2. ข้อมูลของ Moments in time

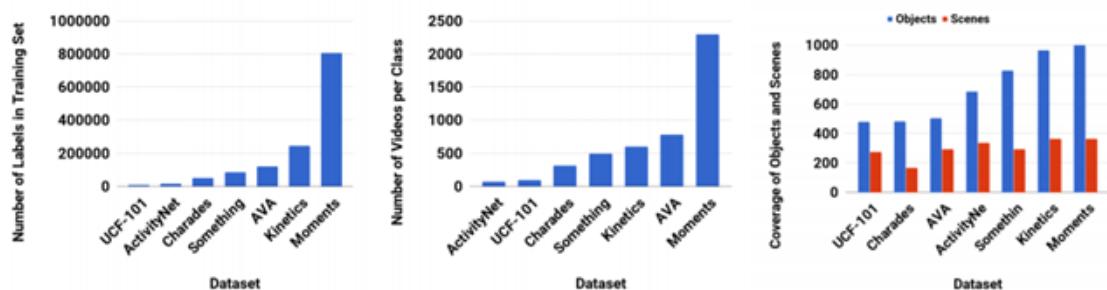
มีวิดีโอมากกว่า 1 ล้านวิดีโอ และมี class ถึง 339 class ที่แตกต่างกัน มีค่าเฉลี่ยวิดีโองแต่ละ class อยู่ที่ 1,757 และค่า median อยู่ที่ 2,775



รูปที่ 2.13: สถิติของชุดข้อมูลของ Moments in time

3. วิธีการทดสอบชุดข้อมูลและผลลัพธ์ที่ได้

โดยการทดสอบแรกจะเป็นการทดสอบเทียบกับชุดข้อมูลอื่นดังภาพด้านล่าง



รูปที่ 2.14: เปรียบเทียบข้อมูลระหว่าง Dataset

จากภาพจะเห็นได้ว่า Moments in time นั้นมีจากหรือสถานที่ที่เหมือน Places = 100% และมีวัตถุเหมือนกับ ImageNet ถึง 99.9 %. ส่วนชุดข้อมูลที่มีความได้เคียงกับ Moments in time มากที่สุดคือชุดข้อมูล Kinetics ที่มีจากหรือสถานที่ที่เหมือน Places = 99.5% และมีวัตถุเหมือน ImageNet ถึง 96.6%

การทดสอบต่อมาจะเป็นการนำ Moments in time มาทดสอบสร้างโมเดลด้วยวิธีต่าง ๆ โดยจะเริ่มจาก การเตรียมข้อมูลข้อมูลดังนี้

1. training set จะมี 802,264 วิดีโอ และมีวิดีโອในแต่ละ class อยู่ที่ 500 ถึง 5,000 วิดีโอ
2. validation set จะมี 33,900 วิดีโอ และมีวิดีโອในแต่ละ class อยู่ที่ 100 วิดีโอ
3. เริ่มการ preprocess จากแยกภาพRGB ออกมานาจิกวิดีโอ และทำการเปลี่ยนขนาดของภาพให้เป็น 340x256 pixel
4. ใช้ TVL1 optical flow algorithm จาก opencv เพื่อลดข้อมูลรบกวนที่จะเกิดขึ้น
5. ทำการแปลงค่าที่อยู่ใน optical flow ให้เป็นเลขจำนวนเต็ม(integer) เพื่อทำให้การคำนวนนั้นเร็วขึ้น
6. ปรับค่า displacement ใน optical flow ให้ค่าสูงสุดเป็น 15 ต่ำสุดเป็น 0 และทำการปรับขนาดให้เป็น ช่วง 0-255
7. เก็บข้อมูลออกมาในรูปแบบของ grayscale image เพื่อลดพื้นที่ ใช้เก็บข้อมูล

8. แก้ปัญหาเรื่องการเคลื่อนไหวของกล้อง(camera motion) โดยการนำค่าเฉลี่ยของ เวกเตอร์(vector) ไปลบกับ displacement
9. สุดท้ายจะเป็นสุ่มตัวภาพอ กมาเพื่อเพิ่มจำนวนข้อมูล

หลังจากการเตรียมข้อมูลเรียบร้อยแล้วจะนำข้อมูลเหล่านั้นมาสร้างโมเดลด้วยวิธีการต่าง ๆ ดังตารางด้านล่าง

Model	Modality	Top-1(%)	Top-5(%)
Chance	-	0.29	1.47
ResNet50-scratch	Spatial	23.65	46.76
ResNet50-Places	Spatial	26.44	50.56
ResNet50-ImageNet	Spatial	27.16	51.68
TSN-Spatial	Spatial	24.11	49.10
BNIception-Flow	Temporal	11.60	27.40
TSN-Flow	Temporal	15.71	34.65
SoundNet	Auditory	7.60	18.00
TSN-2stream	Spatial+Temporal	25.32	50.10
TRN-Multiscale	Spatial+Temporal	28.27	53.87
I3D	Spatial+Temporal	29.51	56.06
Ensemble(SVM)	S+T+A	31.16	57.67

ตารางที่ 2.8: Classification accuracy ของ TOP-1 และ TOP-5

จากภาพจะเห็นได้ว่าผลลัพท์ที่ดีสุดคือการทำ ensemble(SVM) ซึ่งเป็นรวมของโมเดล ResNet50-ImageNet, I3D และ SoundNet จากผลลัพท์จะเห็นค่าที่ได้ออกมาจาก ensemble(SVM) มีค่าใกล้เคียงกับรูปแบบ spatial เพราะประสิทธิภาพเคลื่อนไหว(temporal) และ เสียง(auditory) นั้นมีประสิทธิภาพต่ำ ซึ่งจุดนี้จะทำให้เห็นว่าตัว Moments in time ยังทำให้สามารถพัฒนาต่อไปได้อีก

ต่อมาจะทำการ cross dataset transfer โดยการนำโมเดล ResNet50 I3D pretrained ลงทั้งบน Kinetics และ Moments in time และนำมาเทียบกับชุดข้อมูลอื่น โดยชุดข้อมูลแต่ละชุดจะมีการปรับ frame rate ของวิดีโอให้เป็น 5 fps เมื่อกัน

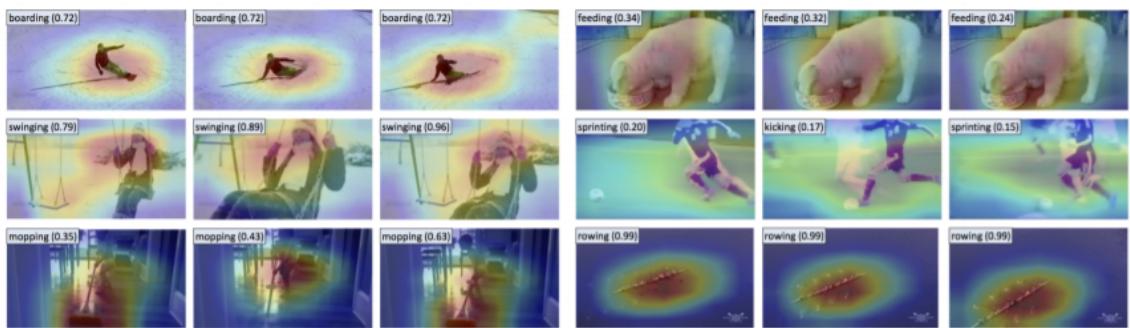
Pretrained	Fine-Tuned		
	UCF	HMDB	Something
Kinetics	Top-1 : 92.6	Top-1 : 62.0	Top-1 : 48.6
	Top-5 : 99.2	Top-5 : 88.2	Top-5 : 77.9
Moments	Top-1 : 91.9	Top-1 : 65.9	Top-1 : 50.0
	Top-5 : 98.6	Top-5 : 89.3	Top-5 : 78.8

ตารางที่ 2.9: Data transfer performance ของโมเดล Resnet50 I3D

จะเห็นได้ว่า Kinetics ให้ผลลัพท์ที่ดีกว่าใน UCF เพราะว่ามีการแชร์ class ด้วยกันอยู่หลายอย่าง ในขณะที่ HMDB นั้นมีการรวม source จากหลายแหล่ง และมีจำนวน class ที่หลากหลายจึงทำให้มีความใกล้เคียงกับตัวข้อมูลของ Moments in time ดังนั้นจึงเทียบผลลัพท์จาก Something ซึ่งจะทำให้เห็นว่า Moments in time มีประสิทธิภาพที่ดีกว่าและวิดีโอมีความยาวมากกว่า 3 วินาทีจะไม่ส่งผลกระทบกับประสิทธิภาพของ Moments in time

4. ปัญหาที่พบ

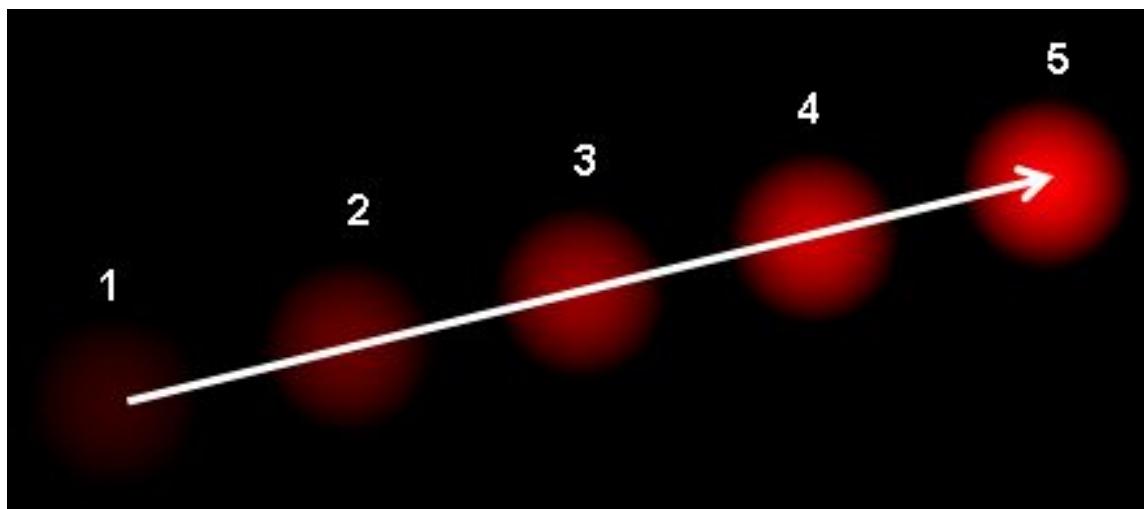
ผลลัพธ์จากการทำนายด้วยโมเดลถ้าผ่านรูปภาพที่มีรายละเอียดเบolare จะทำการ ทำนายโอกาสผิดนั้นค่อนข้างสูง ซึ่งปัญหานี้สามารถทำให้เกิดน้อยลงด้วยการนำวิธี Class Activation Mapping(CAM) จะเป็นการเน้นรูปภาพในส่วนที่มีข้อมูลมากที่สุดและ ทำนายผลออกมา แต่ก็ยังมีจุดที่เป็นปัญหาอยู่ เช่น การกระที่เกิดขึ้นเรื่องมาก (การลื่นล้ม) จะทำให้การทำนาย นั้นมีโอกาสผิดสูงขึ้น



รูปที่ 2.15: ภาพที่ได้จากการทำ CAM และผลลัพธ์ที่ได้จากการทำนายด้วยโมเดล resnet50-ImageNet

2.3 Optical flow

Optical flow⁸ คือรูปแบบของการเคลื่อนที่ของวัตถุในรูปภาพระหว่างภาพซึ่งอาจจากการจากเคลื่อนที่ของวัตถุหรือตัวกล้อง ออกมารูปแบบของ เวกเตอร์ 2 มิติ(vector2D) โดยที่เวกเตอร์(vector) แต่ละตัวจะแสดงถึงทิศทางการเคลื่อนที่ระหว่างภาพดังรูปด้านล่าง



รูปที่ 2.16: ตัวอย่างการเคลื่อนที่ของลูกบอล

จากรูปภาพจะแสดงให้เห็นถึงการเคลื่อนที่ของลูกบอลของภาพที่ต่อเนื่องกัน 5 ภาพโดยที่ลูกศรแสดงถึงทิศทางการเคลื่อนที่ของเวกเตอร์(vector)

⁸Optical flow, shorturl.at/mrtEZ

การทำงานของ optical flow อยู่บนสมมติฐานหลายประการได้แก่

1. ความเข้มของพิกเซล(pixel) ของวัตถุจะไม่เปลี่ยนแปลงระหว่างภาพที่ต่อเนื่องกัน
2. พิกเซล(pixel) ที่อยู่ใกล้กันจะมีการเคลื่อนไหวที่คล้ายกัน

เมื่อพิจารณาพิกเซล(pixel) $I(x,y,t)$ จากภาพแรกจะเคลื่อนไหวเป็นระยะทาง (dx,dy) ไปยังภาพต่อไปหลังจากผ่านไปแล้ว dt เวลา ดังนั้นเมื่อจาก pixel เหล่านี้เหมือนกันและความเข้มไม่มีการเปลี่ยนแปลง จึงทำให้พูดได้ว่า

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

I คือ พิกเซล(pixel) จากภายในภาพ

x คือ ตำแหน่งของพิกเซล(pixel) ในแกน x

dx คือ ระยะทางที่เคลื่อนที่ในแกน x

y คือ ตำแหน่งของพิกเซล(pixel) ในแกน y

dy คือ ระยะทางที่เคลื่อนที่ในแกน y

t คือ เวลา

dt คือ ระยะเวลาที่เปลี่ยนไประหว่างภาพ

จากนั้นใช้การประมาณค่าของ taylor series ทางฝั่งขวาเมื่อและ ลบค่า common term และหารด้วย dt เพื่อให้ได้สมการดังต่อไปนี้

$$f_x u + f_y v + f_t$$

โดยที่

$$f_x = \frac{\delta f}{\delta x}; f_y = \frac{\delta f}{\delta y}$$

$$u = \frac{\delta x}{\delta t}; v = \frac{\delta y}{\delta t}$$

f_x คือ เกรเดียน(gradient) ในแกน x

f_y คือ เกรเดียน(gradient) ในแกน y

f_t คือ เกรเดียน(gradient) ของเวลา

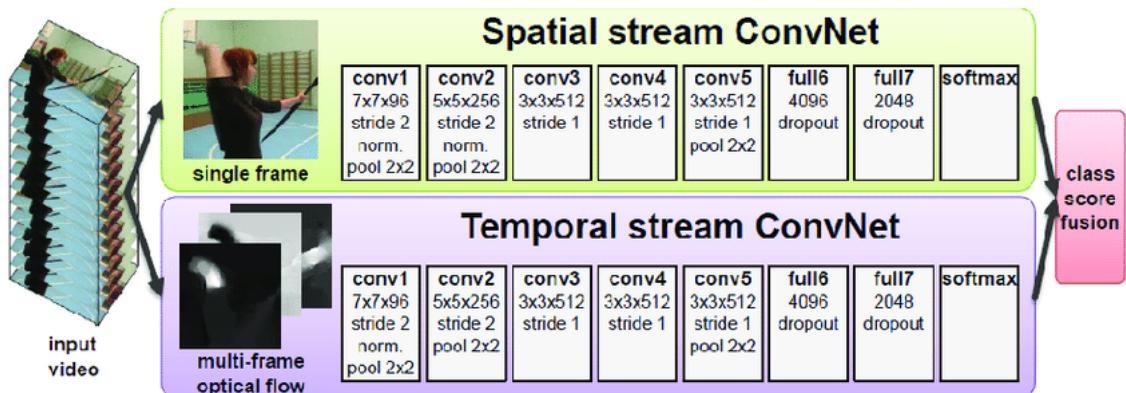
u คือ เวกเตอร์(vector)การเคลื่อนที่ของแกน x

v คือ เวกเตอร์(vector)การเคลื่อนที่ของแกน y

สมการข้างบนนี้จะเรียกว่าสมการ optical flow จากสมการทำให้สามารถหา f_x และ f_y โดยเป็น เกรเดียน(gradient)ของภาพ และ f_t เป็นเกรเดียน(gradient)ของเวลา แต่ u กับ v เป็นตัวแปรที่ไม่ทราบ ทำให้สมการนี้ไม่สามารถแก้ไขโดยมีตัวแปรที่ไม่ทราบถึง 2 ตัว จึงมีการนำวิธีการต่าง ๆ เข้ามาใช้ในการแก้ปัญหานี้ เช่น Lucas-Kanade ⁹

⁹Optical flow,shorturl.at/mrEZ

2.4 Two-Stream CNN



รูปที่ 2.17: แสดงโครงสร้างการทำงานของ two stream

Two-Stream CNN¹⁰ เป็นวิธีการหนึ่งในการทำ video classification โดยจะแบ่งออกเป็นสองกระบวนการทำให้ไปพร้อมกัน คือ กระบวนการแรก คือ นำรูปภาพเดี่ยวๆ มาใช้ซึ่งจะทำให้ได้ข้อมูลจากรูปภาพคือ ฉาบและวัตถุต่างๆ และ กระบวนการที่สอง คือ นำลำดับของรูปภาพมาเพื่อดูการเคลื่อนไหวของวัตถุ และสุดท้ายจะนำข้อมูลที่ได้จากทั้งสองกระบวนการรวมกันโดยใช้การ averaging หรือนำไปผ่าน linear SVM

¹⁰2steamCNN,<https://papers.nips.cc/paper/5353-two-stream-convolutional-networks-for-action-recognition-in-videos.pdf>

บทที่ 3

ระเบียบวิธีวิจัย

ในการทำโครงการวิจัยแอพพลิเคชั่นสำหรับวิเคราะห์วิดีโอ(video analytics) จะมีการทำงานหลากหลายส่วนมาทำงานร่วมกัน ซึ่งทำให้จำเป็นจะต้องมีระเบียบวิจัยสำหรับอธิบายภาพรวม โดยในระเบียบวิจัยนี้จะมีหัวข้อ และระเบียบวิจัยดังนี้

- แผนการดำเนินงาน
- เครื่องมือที่ใช้ในการดำเนินงานวิจัย
- ภาพรวมของแอพพลิเคชั่น
- รายละเอียดของโมเดล

3.1 หน้าที่ความรับผิดชอบ

ปฐมพงศ์ สินธุ์งาม สร้างและทดสอบโมเดลจดจำการกระทำมนุษย์ 3D และออกแบบพร้อมทั้งสร้างระบบ Tracker

ศุภกร เบญจวิกรัย รวบรวมฟังก์ชันต่างๆของแอพพลิเคชั่น และออกแบบพร้อมทั้งสร้างระบบแอพพลิเคชั่นในส่วน Selection และ Detection

อุกฤษฎ์ เลิศวรรณการ สร้างและทดสอบโมเดลจดจำการกระทำมนุษย์ Resnet-50 และออกแบบพร้อมทั้งสร้างระบบ Person ReID

3.2 เครื่องมือที่ใช้ในงานวิจัย

Pycharm community รุ่น 2017.1.2

เป็นโปรแกรมໄวใช้สำหรับเขียนและแก้ไขโค้ดซึ่งข้อดีของโปรแกรมนี้ คือ มี Feature ต่างๆเพื่ออำนวยความสะดวกสะดวกในการเขียนโค้ด เช่น syntax highlighting , Auto-completion ฯลฯ และสามารถนำรันและทดสอบแอพพลิเคชั่นได้

Jupyter รุ่น 2017.1.2

เป็นโปรแกรมໄวใช้สำหรับเขียนและแก้ไขโค้ด แต่เหมาะสมสำหรับรันโมเดล หรือ โค้ดที่รันเป็น section และกัน ซึ่งการรันเป็น section มีข้อดีคือ หากมีการแก้ไขโปรแกรม ไม่จำเป็นต้องรันใหม่ทุกส่วนในโปรแกรม

Qt Creator รุ่น 4.9.2 (Community)

เป็นเครื่องมือสำหรับออกแบบ หน้าต่าง UI ของ pyqt และ เชฟเป็นไฟล์ .ui ซึ่งมีข้อดีคือ เรียกใช้ง่าย มี widget ที่สามารถใช้ได้หลากหลายหมายหนาสำหรับการออกแบบโปรแกรม

3.3 ภาษาที่ใช้ในการพัฒนาระบบ

ใช้ภาษา python ใน การพัฒนาเป็นหลัก เพราะ python เป็นภาษาที่ปัจจุบันมีการใช้กันอย่างแพร่ทำให้มี tools ที่อำนวยความสะดวกให้แก่การพัฒนา และยังเป็นภาษาที่สามารถเข้าใจได้ง่ายและไม่ซับซ้อนจนเกินไป โดยในการทำวิจัยครั้งนี้ได้เลือก python version 3.6.8 มาใช้ในการพัฒนา เนื่องจากเป็น version ที่รองรับการทำงานของ library tensorflow และ cuda version 9

3.4 Library computer ที่ใช้ในการพัฒนาระบบและแอพพลิเคชั่น

1. numpy	version 1.16.4	library ใช้สำหรับการคำนวณและ array
2. pandas	version 0.24.2	library ใช้สำหรับการจัดการเกี่ยวกับ excel
3. opencv	version 4.1.0.25	library ใช้สำหรับการจัดการข้อมูลที่เป็นภาพ
4. pillow	version 6.0.0	library ใช้สำหรับการจัดการข้อมูลที่เป็นภาพ
5. torchsummary	version 1.5.1	library ใช้สำหรับการทำ AI
6. pytorch	version 1.10.0	library ใช้สำหรับการทำ AI
7. torchvision	version 0.3.0	library ใช้สำหรับการทำ AI
8. scikit-learn	version 0.21.2	library ใช้สำหรับการทำ AI
9. scipy	version 1.3.0	library ใช้สำหรับการทำ AI
10. sklearn	version 0.0	library ใช้สำหรับการทำ AI
11. pickleshare	version 0.7.5	library ใช้สำหรับการทำ encoding โมเดล
12. tqdm	version 4.32.1	library ใช้สำหรับนับจำนวนการทำ loop
13. pyqt5	version 5.9.2	library ใช้สำหรับการทำแอพพลิเคชั่น

3.5 แผนการดำเนินงาน

โดยจากที่กล่าวไปตอนต้นในบทนำ การดำเนินงานและการออกแบบการสร้าง labeling tool และระบบวิเคราะห์การกระทำของมนุษย์ในวิดีโอ มีแผนการทำงานซึ่งถูกแบ่งออกเป็นสามส่วนดังนี้ ส่วนแรกคือ ส่วนของการศึกษาความเป็นไปได้ และเทคโนโลยีในปัจจุบันที่เกี่ยวกับการสร้างแอพพลิเคชั่น และการจัดทำกราฟิก ของมนุษย์ด้วยปัญญาประดิษฐ์ เพื่อนำมาประยุกต์ใช้กับงานวิจัยนี้ ส่วนที่สองคือ ส่วนของการออกแบบและสร้างแอพพลิเคชั่นที่ใช้ในการสร้างชุดข้อมูลสำหรับการเทรนโมเดลจากวิดีโอ ส่วนที่สามคือ ส่วนของการออกแบบและสร้างระบบแพลตฟอร์มวิเคราะห์การกระทำของมนุษย์โดยมีข้อกำหนดตามที่กล่าวไว้ในบทนำ

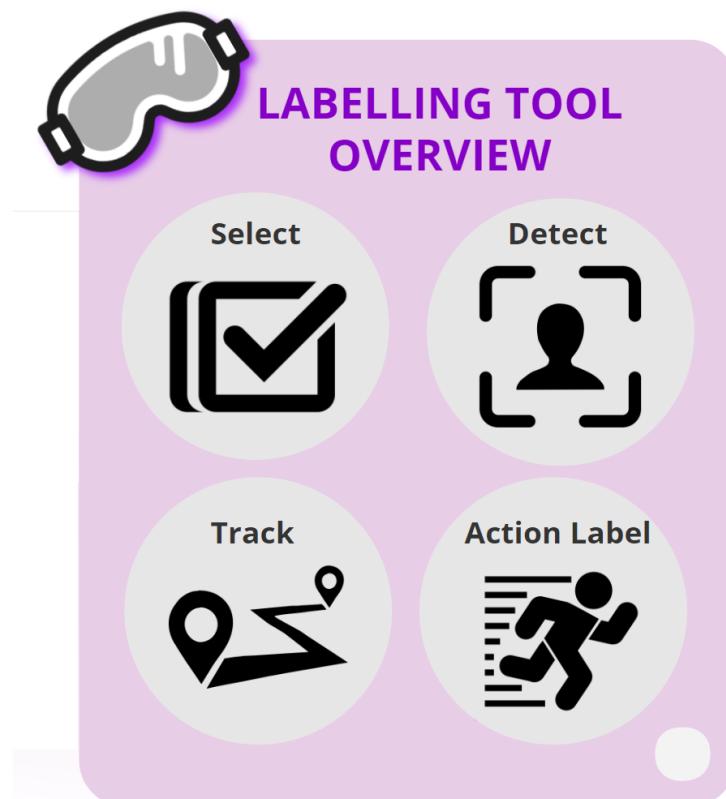
ในการเริ่มทำงานวิจัยนี้นั้นสิ่งจำเป็นที่ต้องทำในอันดับแรกคือการศึกษาสิ่งที่เคยมีอยู่ หรืองานวิจัยอื่นที่ทำเอาไว้แล้ว เพื่อศึกษาและทำความเข้าใจ ข้อดี-ข้อเสีย ของเทคนิคหรือกระบวนการต่างๆ เพื่อนำมาประยุกต์ใช้กับงานวิจัยนี้ ในการศึกษาเกี่ยวกับการออกแบบและการสร้างแอพพลิเคชั่นที่ใช้ในการสร้างชุดข้อมูลสำหรับการเทรนโมเดลจากวิดีโอ ที่มีอยู่แล้วสิ่งที่ต้องให้ความสนใจคือฟังก์ชันการทำงาน การออกแบบและการจัดวางองค์ประกอบต่างๆในหน้าต่าง UI และความสะดวกในการใช้งาน จากนั้นจึงเริ่มศึกษาเกี่ยวกับแพตฟอร์มที่ใช้ในการสร้างแอพพลิเคชั่น ส่วนการศึกษาเกี่ยวกับการสร้างระบบวิเคราะห์การกระทำของมนุษย์นั้น จะมุ่งความสนใจไปที่ชุดข้อมูลสำหรับการวิเคราะห์วิดีโอ โมเดลสำหรับการวิเคราะห์วิดีโอ เทคนิคในการสร้างโมเดล เทคโนโลยีในการทำระบบวิเคราะห์วิดีโอ เพื่อใช้ในการออกแบบและสร้างระบบวิเคราะห์การกระทำของมนุษย์ในวิดีโอด้วยประสิทธิภาพ ในบทนี้ก็จะกล่าวถึงกระบวนการออกแบบและการดำเนินการตามแผนที่วางแผนไว้

3.6 การออกแบบแอพพลิเคชัน labeling tool

การออกแบบ labeling tool นั้น ผู้วิจัยได้เลือกใช้แพลตฟอร์ม PyQt และภาษา Python ในการพัฒนาเนื่องจากแพลตฟอร์ม PyQt นั้นเป็นแพลตฟอร์มที่มีผู้พัฒนาใช้กันอย่างแพร่หลาย จึงทำให้สะดวกในการศึกษาและหาข้อมูลผ่านอินเตอร์เน็ต อีกทั้งยังเป็นแพลตฟอร์มที่สามารถพัฒนาด้วยภาษา Python ได้ และเป็นแพลตฟอร์มที่ใช้งานง่าย สามารถปรับปรุงแก้ไขได้ง่าย เนื่องจากการสร้างแอพพลิเคชันนั้นจำเป็นต้องมีการปรับแก้หน้าต่างอยู่เสมอ

3.6.1 แอพพลิเคชัน labeling tool

ภาพรวมของแอพพลิเคชันที่สร้างขึ้นประกอบด้วยส่วน Select, Detect, Track และ Action label เพื่อช่วยแบ่งเบาภาระของผู้พัฒนาในการสร้าง label สำหรับสร้างโมเดลจากข้อมูลประเกวิดีโอ โดยส่วน Select จะต้องสามารถตัดวิดีโอด้วยส่วนที่ไม่มีมนุษย์อยู่ออกจากวิดีโอด้วย Detect ต้องสามารถหาตำแหน่งของมนุษย์ภายในวิดีโอด้วย Track ต้องสามารถนำตำแหน่งที่ต้องไปของมนุษย์ข้อมูลตำแหน่งของมนุษย์จาก Detect ได้ Action label ต้องสามารถทำงานยกร่างของมนุษย์ได้ในระดับหนึ่ง โดยทุกส่วนการทำงานมนุษย์ต้องสามารถทำงานร่วมกับระบบได้ ดังรูปที่ 3.1

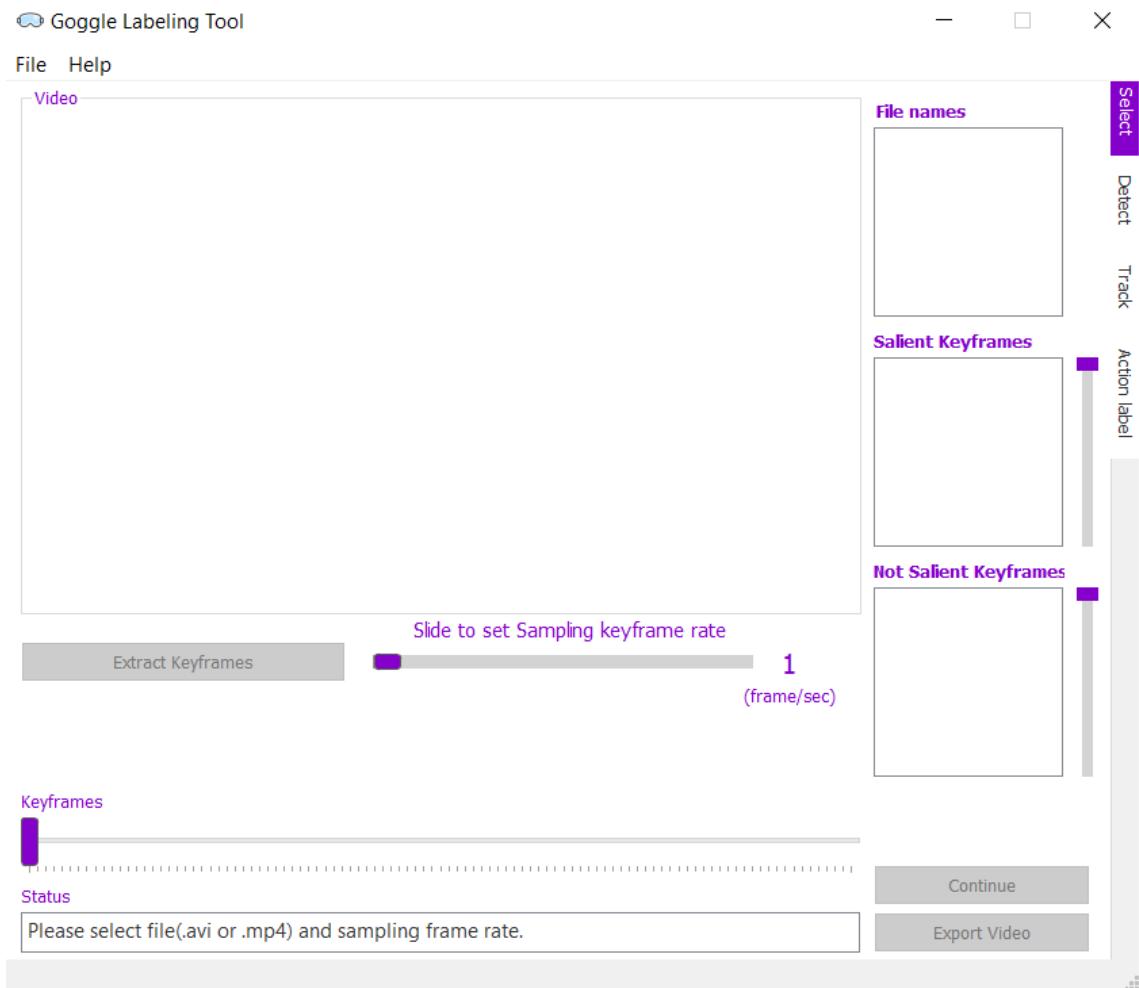


รูปที่ 3.1: ภาพรวมระบบของแอพพลิเคชัน labeling tool

โดยแต่ละส่วนจะมีรายละเอียดดังนี้

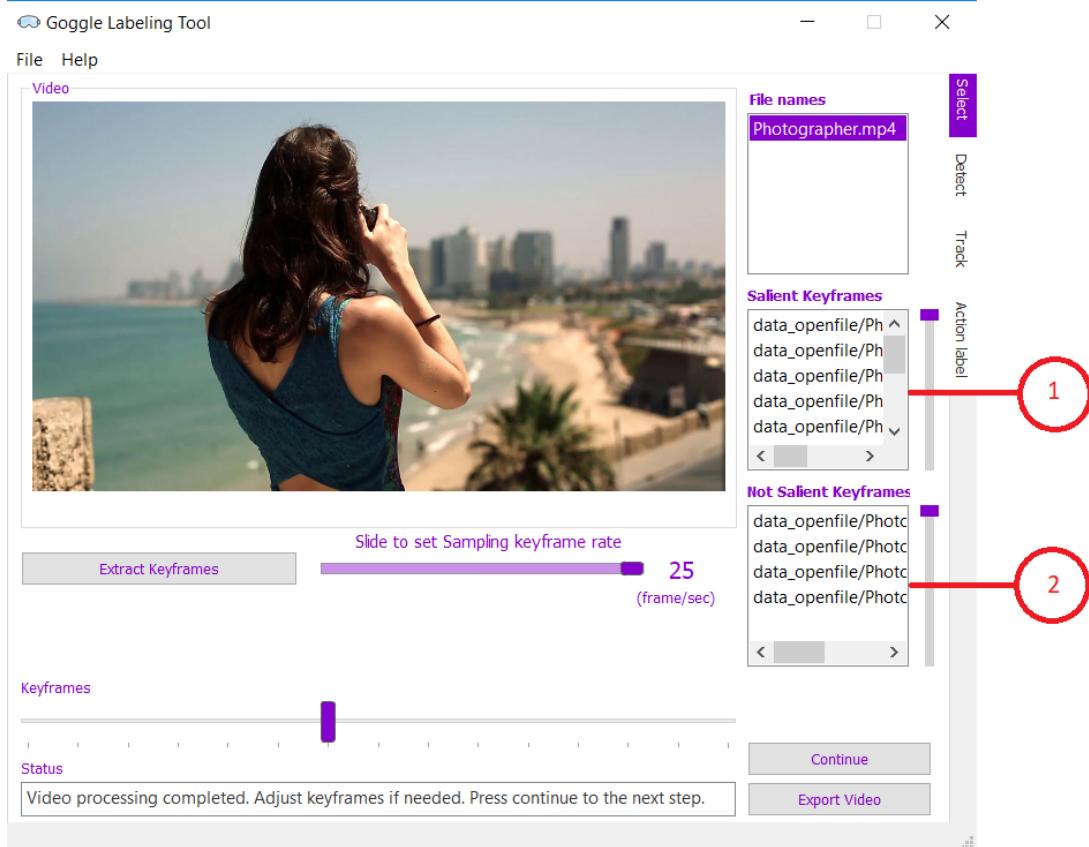
3.6.1.1 Select

ในส่วนของแบบ Select จะมีหน้าต่างเป็นดังรูปที่ 3.2 โดยในส่วนนี้จะมีหน้าที่ในการโหลดวิดีโอที่ต้องการ กำหนดอัตราการหยิบตัวอย่างเฟรมของวิดีโอแล้วเก็บเฟรมเหล่านั้นเป็นคีย์เฟรม(Keyframe) และตัดวิดีโอส่วนที่ไม่มีมนุษย์อยู่ออกໄປ



รูปที่ 3.2: หน้าต่างแบบ Select ของแอพพลิเคชัน labeling tool

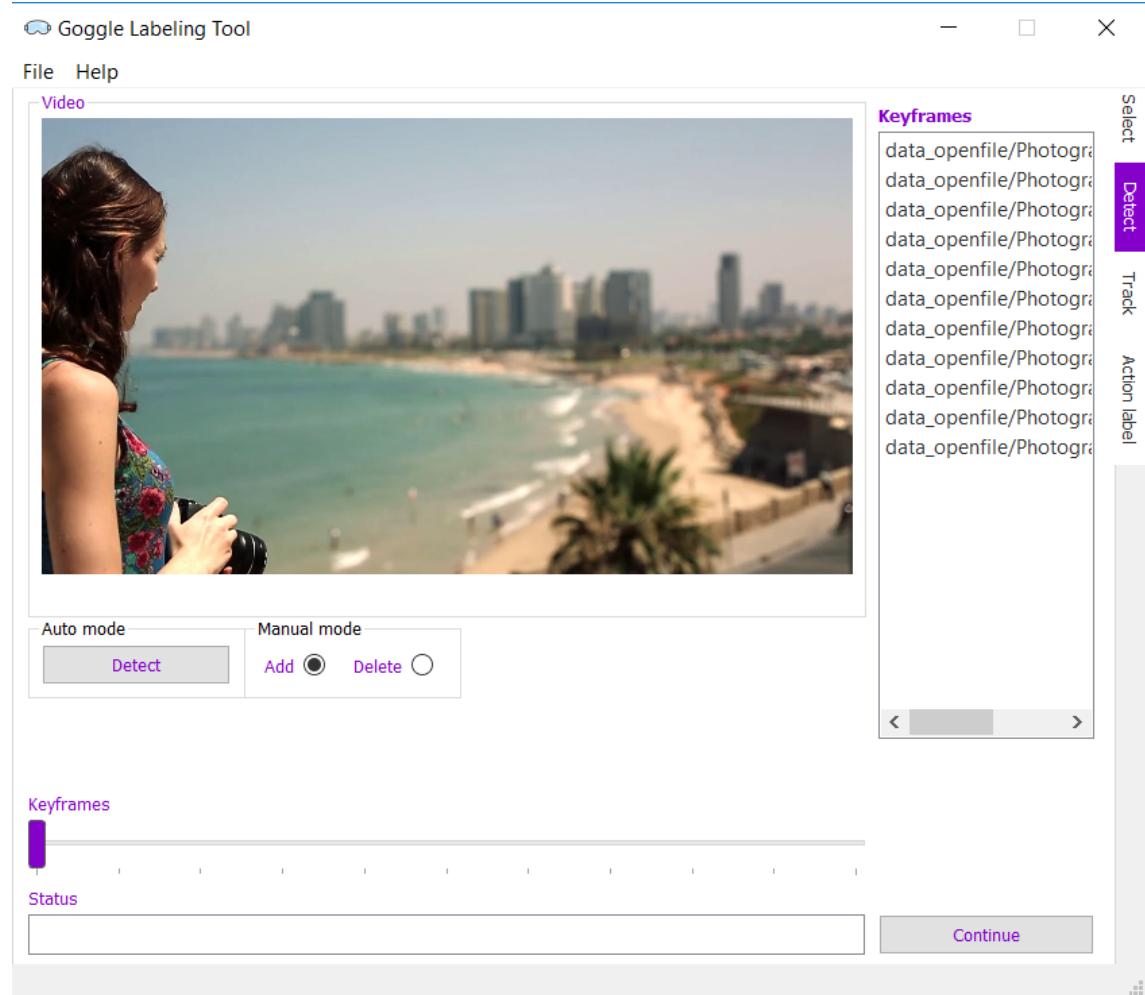
ซึ่งในขั้นตอนการตัดส่วนวิดีโอที่ไม่มีมนุษย์อยู่นั้น ได้ใช้โมเดล YoLo-v3 320 สำหรับตรวจหามนุษย์ ในแต่ละคีย์เฟรม จากนั้นจะแยกคีย์เฟรมที่มีมนุษย์อยู่ และที่ไม่มีมนุษย์อยู่ออกมา แล้วเก็บไว้ในช่องรายการหมายเลข 1 และ 2 ตามลำดับ ดังรูปที่ 3.3



รูปที่ 3.3: หลังจากตัดส่วนวิดีโอลاء คีย์เฟรมจะถูกเก็บไว้ในช่องรายการตามประเภท

3.6.1.2 Detect

ในส่วนของแบบ Detect จะมีหน้าต่างเป็นดังรูปที่ 3.4 โดยในส่วนนี้มีหน้าที่ในการตรวจหาตำแหน่งของมนุษย์ภายในคีย์เฟรม และสร้างกรอบสีเหลืองเพื่อแสดงถึงบริเวณที่มีมนุษย์อยู่จะถูกแบ่งออกเป็นสองแบบ คือแบบอัตโนมัติ(Auto mode) และแบบทำด้วยมือ(Manual mode) ซึ่งในแบบอัตโนมัตินั้นจะใช้โมเดล YoLo-v3 320 ในการตรวจหาตำแหน่งของมนุษย์ในทุกคีย์เฟรม และแบบทำด้วยมือ จะสามารถเพิ่ม หรือลบกรอบสีเหลืองในคีย์เฟรมได้ สามารถใช้ในการแก้ไขความผิดพลาดของแบบอัตโนมัติได้



รูปที่ 3.4: หน้าต่างแบบ Detect ของแอพพลิเคชัน labeling tool

3.7 การออกแบบระบบวิเคราะห์การกระทำของมนุษย์

การออกแบบแบบโปรแกรมด้วย ROS ของหุ่นยนต์ชีวามโนยด์ UTHAI นั้น ผู้วิจัยได้วางการทำงานโดย เริ่มจาก การสร้างแบบจำลองเพื่อใช้ในการจำลองการทำงานของหุ่นยนต์ชีวามโนยด์ UTHAI โดยการจัดสร้างไฟล์ URDF ขึ้น ทดสอบการเคลื่อนไหวของหุ่นยนต์ในการแสดงผลด้วยภาพผ่านโปรแกรม RViz ทดสอบการสั่งการดิจิตอล เชอร์โว ทดสอบการอ่านตำแหน่งจากดิจิตอลเชอร์โว ทดสอบการส่งค่าตำแหน่งจากดิจิตอลเชอร์โวไปประมวลผล หากดศูนย์กลางมวลในโปรแกรม MATLAB และเขียนโปรแกรมอ่านค่าเซนเซอร์ตรวจสอบการสัมผัสพื้น

3.7.1 กำหนดพิกัดเฟรมให้กับหุ่นยนต์ชีวามโนยด์

การกำหนดเฟรมให้กับหุ่นยนต์ชีวามโนยด์ UTHAI นั้น ในวิทยานิพนธ์เล่มนี้ ผู้วิจัยจะใช้หลักตามของ ROS Enhancement Proposals (REPs)¹ ซึ่งการใช้หลักการนี้จะทำให้ การเขียนเป็นระบบระเบียบสามารถหยิบเครื่อง มือต่างๆ ที่สร้างขึ้นมาใช้งานร่วมกันได้ และช่วยทำให้เกิดความเข้าใจเวลาสื่อสาร

base_link เป็นเฟรมที่ติดอยู่กับฐานของหุ่นยนต์ชีวามโนยด์ โดยจะติดตำแหน่งหรือมุมเอียงได้ โดย ส่วนใหญ่แล้วจะติดเฟรม base_link ไว้ที่สังกะส์ของหุ่นยนต์

base_footprint เป็นเฟรมที่แสดงว่าหุ่นยนต์อยู่ตรงไหนเมื่อเทียบกับโลก โดยจะมีระดับอยู่ที่จุดต่ำสุด ของฝ่าเท้า $z = \min(l_sole_z, r_sole_z)$ โดย l_sole_z และ r_sole_z คือความสูงของฝ่าเท้า

l_wrist, r_wrist เป็นเฟรมที่บอกตำแหน่งและมุมเอียงของแขนซ้ายและขวาของหุ่นยนต์ชีวามโนยด์ โดยไม่ต้องคำนึงถึงการติดตั้งอุปกรณ์ใดๆเข้าไปที่ปลายแขนของหุ่นยนต์ชีวามโนยด์

l_gripper, r_gripper เป็นเฟรมที่บอกตำแหน่งและมุมเอียงของที่ปลายแขน (End effector) ถ้ามีอ จับอุปกรณ์อยู่ เฟรมนี้จะใช้ในการอ้างอิงตำแหน่งของอุปกรณ์นั้นๆ แต่ในวิทยานิพนธ์นี้ ไม่ได้ใช้แขนของหุ่นยนต์ ในการหยิบจับเครื่องมือหรือวัตถุ จึงไม่ได้ใช้เฟรมนี้

l_ankle, r_ankle เป็นเฟรมที่บอกตำแหน่งและมุมเอียงของขาซ้ายและขวาโดยไม่ได้คำนึงว่าจุดรับน้ำ หนักของตัวอยู่ที่ไหน

l_sole, r_sole เป็นเฟรมที่บอกตำแหน่งและมุมเอียงของขาซ้ายและขวาที่รองรับน้ำหนักตัวอยู่ โดยจะ บอกการฉายลงในระนาบของ X, Y ที่สัมผัสพื้นและ Z จะอยู่ร่องตืบเดียวกับพื้นสัมผัส

l_toe, r_toe เป็นเฟรมที่บอกตำแหน่งและมุมเอียงของปลายเท้าซ้ายและขวา

gaze เป็นเฟรมที่บอกตำแหน่งและมุมเอียงของหัว โดยการอ้างน้ำหนักของหัว โดยไม่ได้ สนใจเซนเซอร์ว่าจะติดตั้งอย่างไร แต่ในวิทยานิพนธ์นี้ หุ่นยนต์ชีวามโนยด์ UTHAI ไม่มีหัว จึงไม่ได้ใช้เฟรมนี้

torso เป็นเฟรมที่ติดอยู่กับลำตัวซึ่งล่างของหุ่นยนต์ชีวามโนยด์ โดยจะเป็นเฟรมที่ใช้เชื่อม ขา แขน หัว เข้าหากัน

¹<http://www.ros.org/reps/rep-0000.html>

3.7.2 การแปลงข้อมูลให้อยู่ในรูปแบบ URDF

เมื่อออคแบบโครงสร้างของหุ่นยนต์ชีวามาโนยด์ UTHAI ด้วยโปรแกรม Solidworks เสร็จแล้ว ต่อไปเป็นการนำเอาไฟล์ STL ออคมาเพื่อใช้ในการทำระบบจำลองการทำงานของหุ่นยนต์ โดยการใช้งานระบบจำลองเนื่องจากทำให้ผู้วิจัยสามารถที่จะเห็นการทำงานของหุ่นยนต์ชีวามาโนยด์ได้ การสร้างแบบจำลองโดยการใช้เครื่องมือที่มาพร้อมกับ ROS ด้วยโมดูล URDF

3.7.2.1 แพกเกจ ROS สำหรับสร้างแบบจำลอง

ROS ได้ให้เครื่องมือที่ช่วยให้ สามารถที่จะสร้างแบบจำลองของหุ่นยนต์ชีวามาโนยด์สามมิติได้ เครื่องมือใน ROS ที่ชื่อว่า `robot_model` ภายในมีแพกเกจต่างๆที่ใช้สำหรับสร้างแบบจำลองของหุ่นยนต์สามมิติอยู่อย่างครบถ้วน ทำให้เราสามารถทำงานได้สะดวก และรวดเร็วมากขึ้น

`urdf` เป็นหนึ่งในหลายแพกเกจที่อยู่ใน `robot_model`, URDF เป็นไฟล์ XML ที่เอาไว้ใช้บอกลักษณะทางกายภาพของหุ่นยนต์ ซึ่งย่อมาจาก Unified Robot Description Format (URDF) การบอกโครงสร้างของหุ่นยนต์ด้วย URDF จะใช้การบอกเป็นโครงสร้างต้นไม้ของก้านต่อต่างๆในตัวหุ่นยนต์

`joint_state_publisher` เครื่องมือนี้มีประโยชน์มากในการสร้างแบบจำลองหุ่นยนต์ด้วย URDF เนื่องจากสามารถนำตำแหน่งของข้อต่อ มาแสดงเป็น GUI ได้ ทำให้เราสามารถเลื่อนๆหมุนๆไปมาได้ อีกทั้งยังสามารถใช้งานร่วมกับโปรแกรมแสดงผลภาพ RViz ได้

`robot_state_publisher` เป็นเครื่องมือที่ใช้ในการ publish ตำแหน่งของก้านต่อต่างๆในแบบจำลองของหุ่นยนต์ชีวามาโนยด์ออกมายาใน TF อีกทั้งยังให้ความสัมพันธ์ระหว่างเฟรมของหุ่นยนต์ได้ด้วย

`xacro` ย่อมาจาก XML Macros หรือเราสามารถเรียกอีกอย่างว่าเครื่องมือเสริมสำหรับ URDF ซึ่งลักษณะการเขียนเหมือนกับไฟล์ URDF แต่การเขียนนั้นจะสั้นกว่า อ่านง่ายกว่า และสามารถใช้เพื่อทำให้สร้างหุ่นยนต์ที่มีความซับซ้อนง่ายขึ้น สามารถแปลงไฟล์ xacro เป็น urdf ได้ถ้าต้องการ

3.7.2.2 URDF

ในส่วนนี้จะเป็นการอธิบายระบบทางกลของหุ่นยนต์ชีวามาโนยด์เป็นไฟล์ที่ใช้ร่วมกับ ROS เพื่อที่จะสามารถนำไปใช้กับระบบจำลองการทำงานของหุ่นยนต์ในอนาคตได้ ในการอธิบายระบบทางกลนั้นผู้วิจัยได้ใช้ไฟล์ URDF ซึ่งใช้ภาษาการเขียนเป็น XML ในการบอกส่วนประกอบแต่ละส่วนของหุ่นยนต์

Link

ไฟล์ URDF แต่ละชิ้นส่วนของหุ่นยนต์เราจะเรียกว่า link และใน link จะประกอบไปด้วยส่วนย่อยๆ 3 ส่วนคือ `<inertia>` ที่เอาไว้บอกถึงค่าตัวแปรทางฟิสิกส์, `<visual>` ที่เอาไว้แสดงผลให้เราเห็น, `<collision>` ที่เอาไว้ตรวจสอบว่าหุ่นยนต์มีการชนกันกับสิ่งแวดล้อมใหม่ ดังรูปที่ 3.5

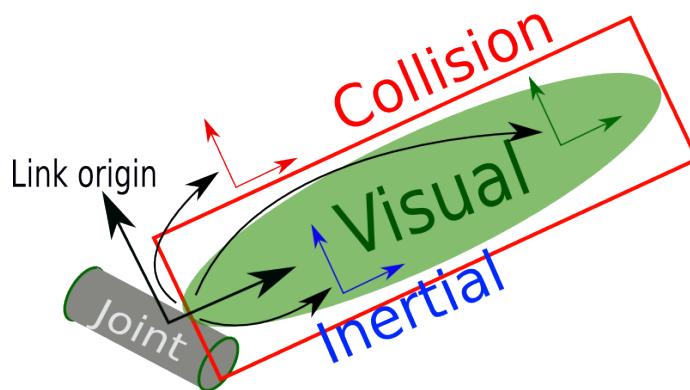
```

<link name="my_link">
  <inertia>
    <origin xyz="0 0 0.5" rpy="0 0 0"/>
    <mass value="1"/>
    <inertia ixx="100" ixy="0" ixz="0" iyy="100" iyz="0" izz="100"/>
  </inertia>
  <visual>
    <origin xyz="0 0 0" rpy="0 0 0"/>
    <geometry>
      <box size="1 1 1" />
    </geometry>
    <material name="Cyan">
      <color rgba="0 1.0 1.0 1.0"/>
    </material>
  </visual>
  <collision>
    <origin xyz="0 0 0" rpy="0 0 0"/>
    <geometry>
      <cylinder radius="1" length="0.5"/>
    </geometry>
  </collision>
</link>

```

รูปที่ 3.5: ตัวอย่าง link ใน urdf

ยังมีอีกหลายตัวที่ใช้ในการอธิบายแต่ละชิ้นส่วนของหุ่นยนต์ แต่ตัวอย่างเป็นเพียงแค่ส่วนหนึ่งเท่านั้น ในความเป็นจริงแล้วเราจะเขียน tags ต่างๆ ก็ตามที่เราต้องการ โดยใน URDF ไฟล์นั้นจะเอาไว้เก็บข้อมูลลักษณะเฉพาะของหุ่นยนต์เอาไว้ และยังสามารถใช้กับซอฟแวร์ตัวอื่นๆ อีกด้วย²



รูปที่ 3.6: การอธิบาย link ใน URDF ไฟล์

²<http://wiki.ros.org/urdf>

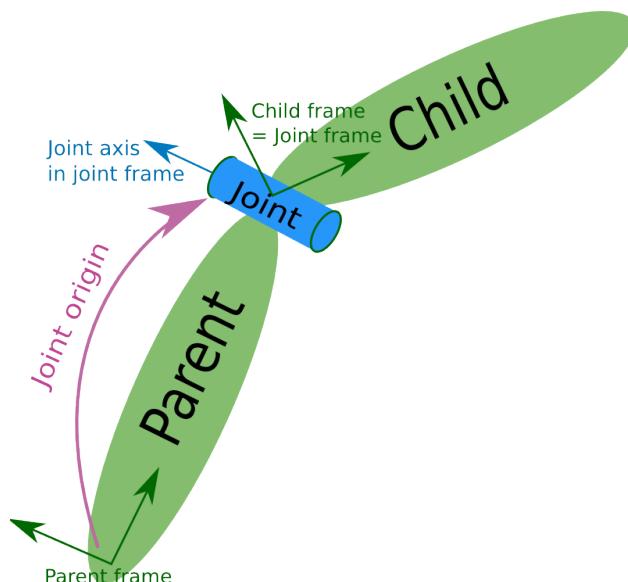
Joint

อีกส่วนที่สำคัญสำหรับการสร้างไฟล์หุ่นยนต์ด้วย URDF ก็คือ Joint tag โดย tag นี้จะอธิบายถึงความสัมพันธ์ระหว่างก้านต่อสองอัน ส่วนนี้ไม่ได้มีเพียงแค่ทำข้อต่อให้เป็นแบบหมุนได้อย่างเดียว ยังมี Fix, Revolution, Linear และ Planar นอกเหนือจากนี้ เราจึงสามารถที่จะเพิ่มองศาสูงสุดต่ำสุดของข้อต่อ รวมไปถึง dynamic properties ต่างๆ ตามที่เห็นดังรูปที่ 3.7

```
<joint name="my_joint" type="floating">
    <origin xyz="0 0 1" rpy="0 0 3.1416"/>
    <parent link="link1"/>
    <child link="link2"/>
    <calibration rising="0.0"/>
    <dynamics damping="0.0" friction="0.0"/>
    <limit effort="30" velocity="1.0" lower="-2.2" upper="0.7"/>
    <safety_controller k_velocity="10" k_position="15"
        soft_lower_limit="-2.0" soft_upper_limit="0.5"/>
</joint>
```

รูปที่ 3.7: ตัวอย่าง joint ใน urdf

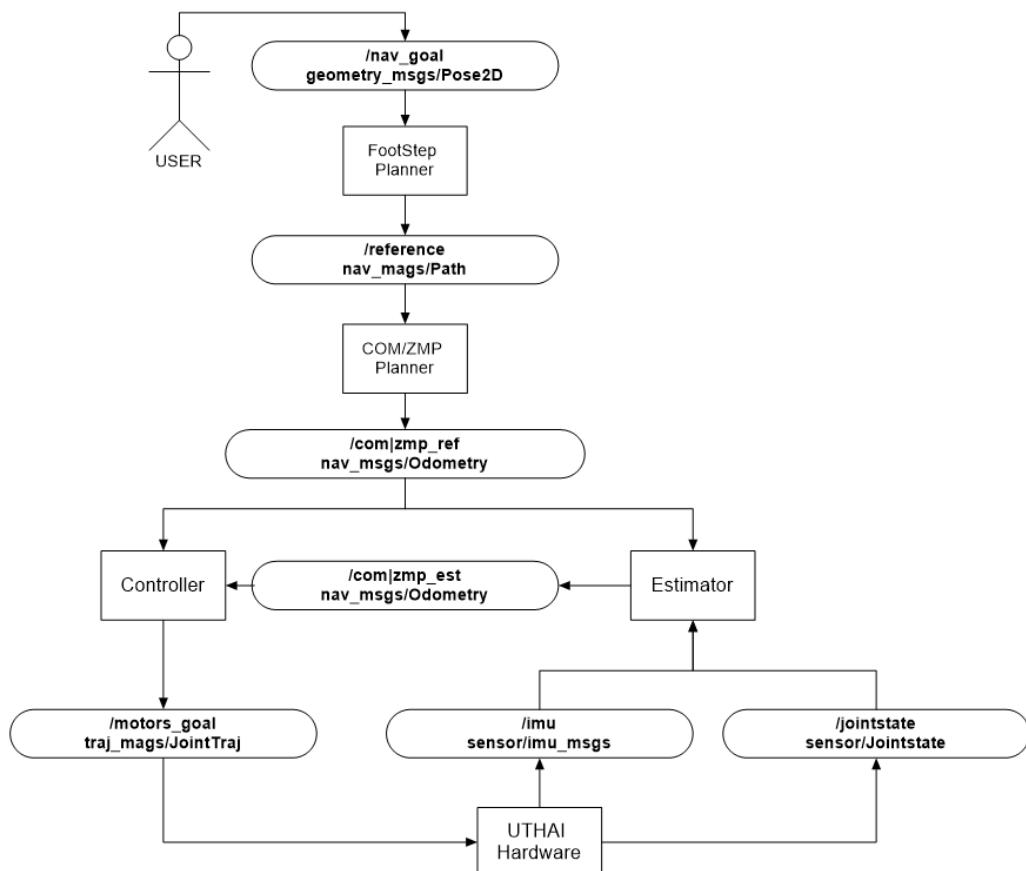
เมื่อเรานำ Joint และ Link มารวมกันเราจะต้องพิจารณาว่ามี wang รูปแบบเป็นไปตามรูปที่ 3.8 โดยจะมีระยะระหว่างแกนของแต่ละข้อต่อ กับ ก้านต่อ ชิ้นส่วนแรกของการสร้างไฟล์ URDF จะมีข้อว่า base_link และ เพرم origin จะเป็นเพرمอ้างอิง เมื่อเราต่อ Joint เข้ากับ Link จะเรียกว่า parent โดยเพرم origin ของข้อต่อจะอยู่จุดเดียวกับเพرم origin ของก้านต่อ ในสถานะเดียวกันก้านต่อที่นำมาต่อจากข้อต่อ เราจะเรียกว่า child และเพرم origin ของก้านต่อ child จะอยู่ที่จุดเดียวกับเพرم origin ของข้อต่อ



รูปที่ 3.8: การอธิบาย Joint ใน URDF ไฟล์

3.7.3 โครงสร้างการติดต่อสื่อสารระหว่าง Node ใน ROS

การติดต่อสื่อสารกันภายใน ROS นั้นจะใช้การส่ง message หากัน ซึ่ง message แต่ละตัวก็จะใช้ในงานที่ต่างกัน ตามระบบที่ต้องการส่ง จากรูปที่ 3.9 เป็นโครงสร้างการส่งข้อมูลหกันของหุ่นยนต์อิริวามานอยด์ ที่ผู้วิจัยได้ออกแบบไว้ โดยเริ่มจากผู้ใช้งานส่งตำแหน่งที่หุ่นยนต์จะต้องเดินไปเป็น Node ที่ทำการคำนวณและสร้างตำแหน่งการวางเท้าของหุ่นยนต์ และหลังจากนั้นจะส่งข้อมูลออกไปเป็น Path เส้นทางไปยัง Node ที่ทำการค้นหาตำแหน่งของ com, zmp ของหุ่นยนต์ เพื่อทำการควบคุมและส่งการหุ่นยนต์ต่อไป



รูปที่ 3.9: การติดต่อสื่อสารระหว่าง Node

การบอกตำแหน่งและมุมเอียง

การบอกตำแหน่งใน 3 มิติ Point คือการบอก x, y, z และการบومุมเอียงจะใช้ Quaternion ในการบอกโดยใช้ตัวแปรสีตัว คือ x,y,z,w หากนำทั้งสองมารวมกันเราจะเรียกว่า Pose

geometry_msgs/Point	
float64	x
float64	y
float64	z

ตารางที่ 3.1: Message Geometry Point

geometry_msgs/Quaternion	
float64	x
float64	y
float64	z
float64	w

ตารางที่ 3.2: Message Geometry Quaternion

geometry_msgs/Pose	
geometry_msgs/Point	position
geometry_msgs/Quaternion	orientation

ตารางที่ 3.3: Message Geometry Pose

การบอกรความเร็วเชิงเส้นและเชิงมุม

การบอกรความเร็วเชิงเส้นใน 3 มิติ คือการบอกรความเร็วตามแนวแกน x, y, z และการบอกรความเร็วเชิงมุม คือการบอกรความเร็วการหมุนรอบแกน x, y, z หากนำทั้งสองมาร่วมกันเราจะเรียกว่า Twist

geometry_msgs/Vector3	
float64	x
float64	y
float64	z

ตารางที่ 3.4: Message Geometry Vector3

geometry_msgs/Twist	
geometry_msgs/Vector3	linear
geometry_msgs/Vector3	angular

ตารางที่ 3.5: Message Geometry Twist

การบอกรตำแหน่งและความเร็ว

หากนำทั้งสองมาร่วมกันจะได้ ตำแหน่ง (Pose) และความเร็ว (Twist) เราจะเรียกว่า Odometry แต่ที่เพิ่มเข้ามาคือ Covariance ซึ่งอาจทำให้เกิดความสับสนได้

nav_msgs/Odometry	
std_msgs/Header	header
geometry_msgs/PoseWithCovariance	pose
geometry_msgs/TwistWithCovariance	twist

ตารางที่ 3.6: Message Navigation Odometry

ตำแหน่งของหุ่นยนต์

การบอกร่องรอยของหุ่นยนต์บนระนาบ 2 มิติ คือการบอก x , y และ θ การบอกนั้นจะบอกว่าตำแหน่งที่หุ่นยนต์อยู่นั้นอยู่ตรงไหนหากเทียบกับแผนที่ รวมไปถึงตำแหน่งของหุ่นยนต์ที่ต้องการจะเดินไปด้วย ซึ่งอ้างอิงมาจากการที่หุ่นยนต์มีความสามารถในการรับและส่งข้อมูลทางบูติค เช่น บอกร่องรอย คำสั่งการเคลื่อนไหว ฯลฯ

geometry_msgs/Pose2D	
float64	x
float64	y
float64	θ

ตารางที่ 3.7: Message Geometry Pose2D

ตำแหน่งการวางแผนทางของหุ่นยนต์

การจะให้หุ่นยนต์นำเท้าไปวางในตำแหน่งที่เราต้องการจากที่ได้จากการคำนวณนั้น จะต้องบอกร่องรอยและบอกรุ่นของจุดที่จะไป จากการสร้างจะได้เป็นรายการของเท้าซ้ายและขวา โดยอิงจาก ตารางที่ 3.3

nav_msgs/Path	
std_msgs/Header	header
geometry_msgs/PoseStamped[]	poses

ตารางที่ 3.8: Message Navigation Path

geometry_msgs/PoseStamped	
std_msgs/Header	header
geometry_msgs/Pose	pose

ตารางที่ 3.9: Message Geometry PoseStamped

ตำแหน่งจุดศูนย์กลางมวลของหุ่นยนต์

ใน Message นี้ใช้อยู่ 2 ที่คือ Message ที่ได้จากการวางแผนของ Node CoM Planner และ Node CoM Estimator โดยทั้งสองจุดใช้ Message เมื่อกันส่งไปยัง Controller เพื่อควบคุมท่าทางต่างๆของหุ่นยนต์ต่อไป Message ที่ใช้คือ Message จากตารางที่ 3.6

nav_msgs/Odometry	
std_msgs/Header	header
geometry_msgs/PoseWithCovariance	pose
geometry_msgs/TwistWithCovariance	twist

Message Navigation Odometry

การควบคุมข้อต่อของหุ่นยนต์

ในการควบคุมข้อต่อแต่ละข้อของหุ่นยนต์ชีวามาโนyd'n'จะใช้ Message trajectory_msgs/JointTrajectory ซึ่งสามารถส่ง ตำแหน่ง ความเร็ว ความเร่ง และ แรงบิด ไปได้ ทำให้หากต้องการเปลี่ยนระบบใหม่สามารถทำได้โดยง่าย

trajectory_msgs/JointTrajectory	
std_msgs/Header	header
string[]	joint_names
trajectory_msgs/JointTrajectoryPoint[]	points

ตารางที่ 3.10: Message Trajectory JointTrajectory

trajectory_msgs/JointTrajectoryPoint	
float64[]	positions
float64[]	velocities
float64[]	accelerations
float64[]	effort
duration	time_from_start

ตารางที่ 3.11: Message Trajectory JointTrajectoryPoint

ค่าเซนเซอร์ข้อต่อของหุ่นยนต์

ที่ข้อต่อของหุ่นยนต์ชีวามาโนyd'm'i' มีเซนเซอร์ที่เอาไว้ใช้ในการอ่านค่าตำแหน่ง ความเร็ว และแรง อยู่ด้วย เราสามารถที่จะใช้ Message sensor_msgs/JointState สำหรับอ่านค่าตำแหน่ง ความเร็ว แรง ของตัวขับเคลื่อนแล้วส่งให้ Estimator Node ได้

sensor_msgs/JointState	
std_msgs/Header	header
float64[]	position
float64[]	velocity
float64[]	effort

ตารางที่ 3.12: Message Sensor JointState

ค่าเซนเซอร์ฝ่าเท้าของหุ่นยนต์

ที่ฝ่าเท้าของหุ่นยนต์ชีวามาโนyd'm'i' มีเซนเซอร์ที่เอาไว้ใช้ในการอ่าน แรงกดที่ฝ่าเท้า ใช้ในการเอามากกว่า เท้าสัมผัสพื้นหรือไม่

geometry_msgs/Wrench	
geometry_msgs/Vector3	force
geometry_msgs/Vector3	torque

ตารางที่ 3.13: Message Geometry Wrench

ค่าเซนเซอร์ IMU ของหุ่นยนต์

เซนเซอร์ IMU เป็นเซนเซอร์ที่เอาไว้ใช้ในการวัด ความเร็วเชิงมุม และ ความเร่งเชิงเส้น หากนำทั้งคู่มารวมกันจะสามารถที่จะแปลงให้วัดมุมอิริยาบถของเซนเซอร์ได้ โดยจะใช้ Message std_msgs/Imu ในการส่งให้ Node Estimator จากตัวหุ่นยนต์

sensor_msgs/Imu	
std_msgs/Header	header
geometry_msgs/Quaternion float64[9]	orientation
geometry_msgs/Vector3 float64[9]	orientation_covariance
geometry_msgs/Vector3 float64[9]	angular_velocity
	angular_velocity_covariance
	linear_acceleration
	linear_acceleration_covariance

ตารางที่ 3.14: Message Sensor Imu

sensor_msgs/MagneticField	
std_msgs/Header	header
geometry_msgs/Vector3 float64[9]	magnetic_field
	magnetic_field_covariance

ตารางที่ 3.15: Message Sensor MagneticField

ภาคผนวก

ภาคผนวก ก

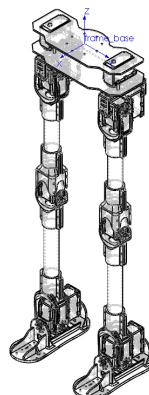
ข้อมูลเบื้องต้นของหุ่นยนต์อิวามานอยด์ UTHAI

ก.1 ค่าคุณสมบัติทางพลศาสตร์

ข้อมูลพลศาสตร์ของหุ่นยนต์อิวามานอยด์ UTHAI ซึ่งจะนำไปใช้ในการทำระบบจำลองด้วยโปรแกรม Gazebo ใน ROS และใช้ในการคำนวณทางคณิตศาสตร์เพื่อทำให้การเดินมีเสถียรภาพ โดยข้อมูลชุดนี้ได้มาจากการคำนวณ Mass Properties ในโปรแกรม SolidWorks และปรับมีค่าใกล้เคียงกับของจริงโดยการเทียบกับเครื่องซึ่งน้ำหนัก

ข้อมูลชุดนี้ประกอบไปด้วย มวล จุดศูนย์กลางมวล และโมเมนต์ความเฉื่อย อีกทั้งข้อมูลยังบอกในมาตรฐาน URDF กับ DH-Parameter ซึ่งทำให้ใช้งานในระบบการคำนวณที่ต่างกันได้

Overall Humanoid

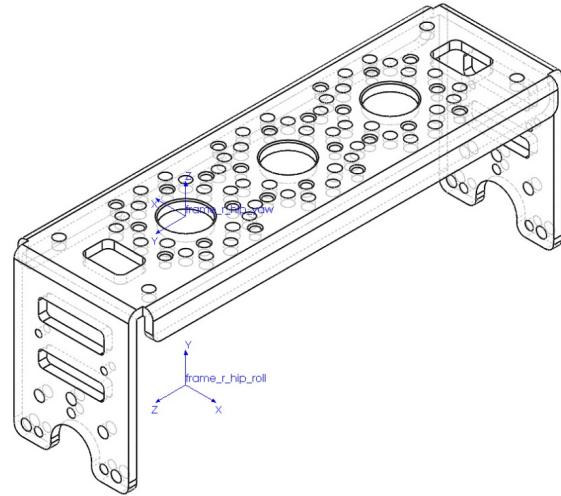


รูปที่ ก.1: ภาพแสดงช่วงล่างทั้งตัว

Link	All Link
Mass (kg)	3.31477475
CoM X (m)	-0.00855772
CoM Y (m)	0.00000000
CoM Z (m)	-0.33375492
Inertia Ixx	0.28641029
Inertia Ixy	-0.00000302
Inertia Ixz	-0.00048106
Inertia Iyy	0.26207601
Inertia Iyz	-0.00061103
Inertia Izz	0.02925799

ตารางที่ ก.1: ตารางแสดงค่าพารามิเตอร์ทั้งตัว

Right Hip Yaw



รูปที่ ก.2: ภาพแสดงก้านต่อ Right Hip Yaw

Link	r_hip_yaw
Mass (kg)	0.09100000
CoM X (m)	0.00000000
CoM Y (m)	0.02864983
CoM Z (m)	-0.02500000
Inertia Ixx	0.00014158
Inertia Ixy	0.00000000
Inertia Ixz	0.00000000
Inertia Iyy	0.00014316
Inertia Iyz	0.00000000
Inertia Izz	0.00002022

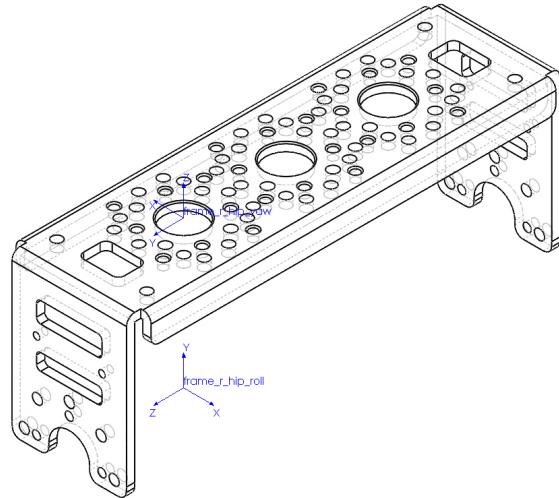
(ก) DH Parameter

Link	r_hip_yaw
Mass (kg)	0.09100000
CoM X (m)	0.00000000
CoM Y (m)	-0.02500000
CoM Z (m)	-0.00735017
Inertia Ixx	0.00014158
Inertia Ixy	0.00000000
Inertia Ixz	0.00000000
Inertia Iyy	0.00002022
Inertia Iyz	0.00000000
Inertia Izz	0.00014316

(ข) URDF

ตารางที่ ก.2: ตารางแสดงค่าพารามิเตอร์ Right Hip Yaw

Left Hip Yaw



รูปที่ ก.3: ภาพแสดงก้านต่อ Left Hip Yaw

Link	l_hip_yaw
Mass (kg)	0.09100000
CoM X (m)	0.00000000
CoM Y (m)	0.02864983
CoM Z (m)	-0.02500000
Inertia Ixx	0.00014158
Inertia Ixy	0.00000000
Inertia Ixz	0.00000000
Inertia Iyy	0.00014316
Inertia Iyz	0.00000000
Inertia Izz	0.00002022

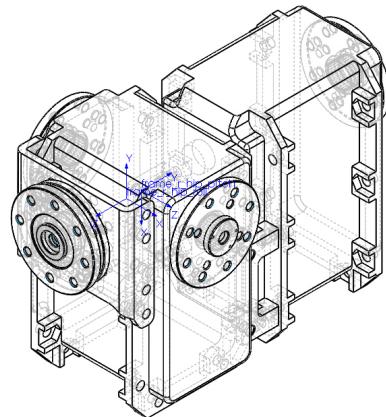
(ก) DH Parameter

Link	l_hip_yaw
Mass (kg)	0.09100000
CoM X (m)	0.00000000
CoM Y (m)	0.02500000
CoM Z (m)	-0.00735017
Inertia Ixx	0.00014158
Inertia Ixy	0.00000000
Inertia Ixz	0.00000000
Inertia Iyy	0.00002022
Inertia Iyz	0.00000000
Inertia Izz	0.00014316

(ข) URDF

ตารางที่ ก.3: ตารางแสดงค่าพารามิเตอร์ Left Hip Yaw

Right Hip Roll



รูปที่ ก.4: ภาพแสดงก้านต่อ Right Hip Roll

Link	r_hip_roll
Mass (kg)	0.34300000
CoM X (m)	0.01526237
CoM Y (m)	0.02152630
CoM Z (m)	0.00000000
Inertia Ixx	0.00026846
Inertia Ixy	0.00000219
Inertia Ixz	-0.00000081
Inertia Iyy	0.00014760
Inertia Iyz	0.00000000
Inertia Izz	0.00032448

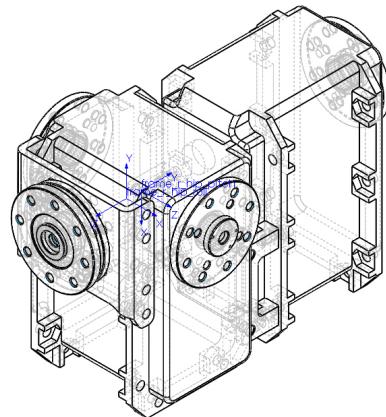
(ก) DH Parameter

Link	r_hip_roll
Mass (kg)	0.34300000
CoM X (m)	0.00000000
CoM Y (m)	-0.01526237
CoM Z (m)	-0.02652630
Inertia Ixx	0.00032448
Inertia Ixy	0.00000081
Inertia Ixz	0.00000000
Inertia Iyy	0.00026846
Inertia Iyz	0.00000219
Inertia Izz	0.00014760

(ข) URDF

ตารางที่ ก.4: ตารางแสดงค่าพารามิเตอร์ Right Hip Roll

Left Hip Roll



รูปที่ ก.5: ภาพแสดงก้านต่อ Left Hip Roll

Link	l_hip_roll
Mass (kg)	0.34300000
CoM X (m)	0.01526237
CoM Y (m)	0.02152630
CoM Z (m)	0.00000000
Inertia Ixx	0.00026846
Inertia Ixy	0.00000219
Inertia Ixz	-0.00000081
Inertia Iyy	0.00014760
Inertia Iyz	0.00000000
Inertia Izz	0.00032448

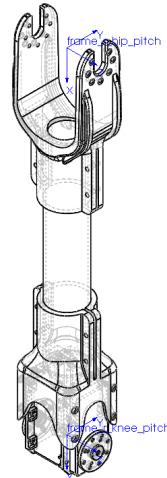
(ก) DH Parameter

Link	l_hip_roll
Mass (kg)	0.34300000
CoM X (m)	0.00000000
CoM Y (m)	-0.01526237
CoM Z (m)	-0.02652630
Inertia Ixx	0.00032448
Inertia Ixy	0.00000081
Inertia Ixz	0.00000000
Inertia Iyy	0.00026846
Inertia Iyz	0.00000219
Inertia Izz	0.00014760

(ข) URDF

ตารางที่ ก.5: ตารางแสดงค่าพารามิเตอร์ Left Hip Roll

Right Hip Pitch



รูปที่ ก.6: ภาพแสดงก้านต่อ Right Hip Pitch

Link	r_hip_pitch
Mass (kg)	0.31800000
CoM X (m)	-0.07862011
CoM Y (m)	0.00000000
CoM Z (m)	0.00000000
Inertia Ixx	0.00011525
Inertia Ixy	0.00000000
Inertia Ixz	0.00000078
Inertia Iyy	0.00254669
Inertia Iyz	0.00000000
Inertia Izz	0.00250848

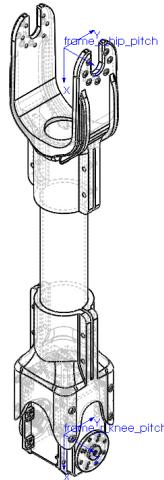
(ก) DH Parameter

Link	r_hip_pitch
Mass (kg)	0.31800000
CoM X (m)	0.22137989
CoM Y (m)	0.00000000
CoM Z (m)	0.00000000
Inertia Ixx	0.00011525
Inertia Ixy	0.00000000
Inertia Ixz	0.00000078
Inertia Iyy	0.00254669
Inertia Iyz	0.00000000
Inertia Izz	0.00250848

(ข) URDF

ตารางที่ ก.6: ตารางแสดงค่าพารามิเตอร์ Right Hip Pitch

Left Hip Pitch



รูปที่ ก.7: ภาพแสดงก้านต่อ Left Hip Pitch

Link	l_hip_pitch
Mass (kg)	0.31800000
CoM X (m)	-0.07862011
CoM Y (m)	0.00000000
CoM Z (m)	0.00000000
Inertia Ixx	0.00011525
Inertia Ixy	0.00000000
Inertia Ixz	0.00000078
Inertia Iyy	0.00254669
Inertia Iyz	0.00000000
Inertia Izz	0.00250848

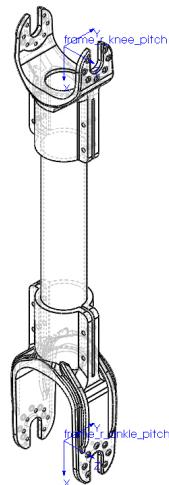
(ก) DH Parameter

Link	l_hip_pitch
Mass (kg)	0.31800000
CoM X (m)	0.22137989
CoM Y (m)	0.00000000
CoM Z (m)	0.00000000
Inertia Ixx	0.00011525
Inertia Ixy	0.00000000
Inertia Ixz	0.00000078
Inertia Iyy	0.00254669
Inertia Iyz	0.00000000
Inertia Izz	0.00250848

(ข) URDF

ตารางที่ ก.7: ตารางแสดงค่าพารามิเตอร์ Left Hip Pitch

Right Knee Pitch



รูปที่ ก.8: ภาพแสดงก้านต่อ Right Knee Pitch

Link	r_knee_pitch
Mass (kg)	0.13800000
CoM X (m)	-0.15211782
CoM Y (m)	0.00000000
CoM Z (m)	0.00000000
Inertia Ixx	0.00011525
Inertia Ixy	0.00000000
Inertia Ixz	0.00000000
Inertia Iyy	0.00127592
Inertia Iyz	0.00000000
Inertia Izz	0.00124960

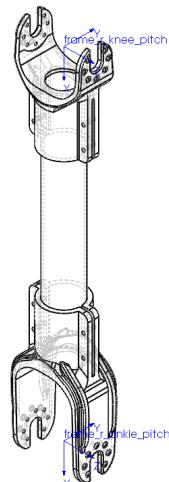
(ก) DH Parameter

Link	r_knee_pitch
Mass (kg)	0.13800000
CoM X (m)	0.16288218
CoM Y (m)	0.00000000
CoM Z (m)	0.00000000
Inertia Ixx	0.00005794
Inertia Ixy	0.00000000
Inertia Ixz	0.00000000
Inertia Iyy	0.00127592
Inertia Iyz	0.00000000
Inertia Izz	0.00124960

(ข) URDF

ตารางที่ ก.8: ตารางแสดงค่าพารามิเตอร์ Right Knee Pitch

Left Knee Pitch



รูปที่ ก.9: ภาพแสดงก้านต่อ Left Knee Pitch

Link	l_knee_pitch
Mass (kg)	0.13800000
CoM X (m)	-0.15211782
CoM Y (m)	0.00000000
CoM Z (m)	0.00000000
Inertia Ixx	0.00011525
Inertia Ixy	0.00000000
Inertia Ixz	0.00000000
Inertia Iyy	0.00127592
Inertia Iyz	0.00000000
Inertia Izz	0.00124960

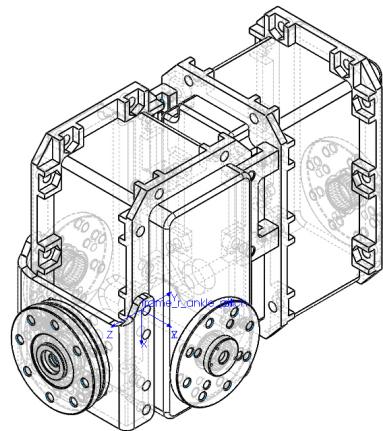
(ก) DH Parameter

Link	l_knee_pitch
Mass (kg)	0.13800000
CoM X (m)	0.16288218
CoM Y (m)	0.00000000
CoM Z (m)	0.00000000
Inertia Ixx	0.00005794
Inertia Ixy	0.00000000
Inertia Ixz	0.00000000
Inertia Iyy	0.00127592
Inertia Iyz	0.00000000
Inertia Izz	0.00124960

(ข) URDF

ตารางที่ ก.9: ตารางแสดงค่าพารามิเตอร์ Left Knee Pitch

Right Ankle Pitch



รูปที่ ก.10: ภาพแสดงก้านต่อ Right Ankle Pitch

Link	r_ankle_pitch
Mass (kg)	0.33138738
CoM X (m)	-0.01526237
CoM Y (m)	0.00000000
CoM Z (m)	-0.02152630
Inertia Ixx	0.00025937
Inertia Ixy	0.00000000
Inertia Ixz	0.00000079
Inertia Iyy	0.00031349
Inertia Iyz	0.00000000
Inertia Izz	0.00014261

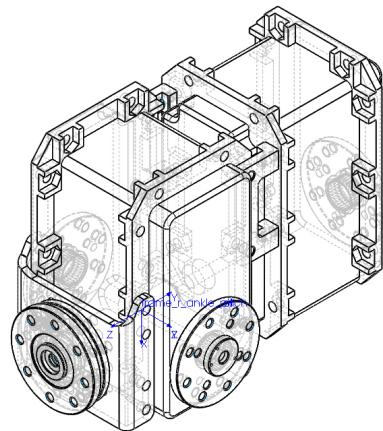
(ก) DH Parameter

Link	r_ankle_pitch
Mass (kg)	0.33138738
CoM X (m)	-0.01526237
CoM Y (m)	0.02152630
CoM Z (m)	0.00000000
Inertia Ixx	0.00025937
Inertia Ixy	0-0.00000212
Inertia Ixz	0.00000079
Inertia Iyy	0.00014261
Inertia Iyz	0.00000000
Inertia Izz	0.00031349

(ข) URDF

ตารางที่ ก.10: ตารางแสดงค่าพารามิเตอร์ Right Ankle Pitch

Left Ankle Pitch



รูปที่ ก.11: ภาพแสดงก้านต่อ Left Ankle Pitch

Link	l_ankle_pitch
Mass (kg)	0.33138738
CoM X (m)	-0.01526237
CoM Y (m)	0.00000000
CoM Z (m)	-0.02152630
Inertia Ixx	0.00025937
Inertia Ixy	0.00000000
Inertia Ixz	0.00000079
Inertia Iyy	0.00031349
Inertia Iyz	0.00000000
Inertia Izz	0.00014261

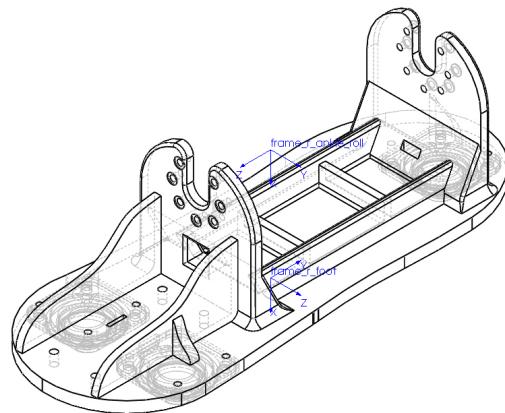
(ก) DH Parameter

Link	l_ankle_pitch
Mass (kg)	0.33138738
CoM X (m)	-0.01526237
CoM Y (m)	0.02152630
CoM Z (m)	0.00000000
Inertia Ixx	0.00025937
Inertia Ixy	0-0.00000212
Inertia Ixz	0.00000079
Inertia Iyy	0.00014261
Inertia Iyz	0.00000000
Inertia Izz	0.00031349

(ข) URDF

ตารางที่ ก.11: ตารางแสดงค่าพารามิเตอร์ Left Ankle Pitch

Right Ankle Roll



รูปที่ ก.12: ภาพแสดงก้านต่อ Right Ankle Roll

Link	r_ankle_roll
Mass (kg)	0.10500000
CoM X (m)	-0.01454118
CoM Y (m)	-0.00034576
CoM Z (m)	-0.00019548
Inertia Ixx	0.00034591
Inertia Ixy	-0.00000857
Inertia Ixz	-0.00000013
Inertia Iyy	0.00004813
Inertia Iyz	-0.00000120
Inertia Izz	0.00032705

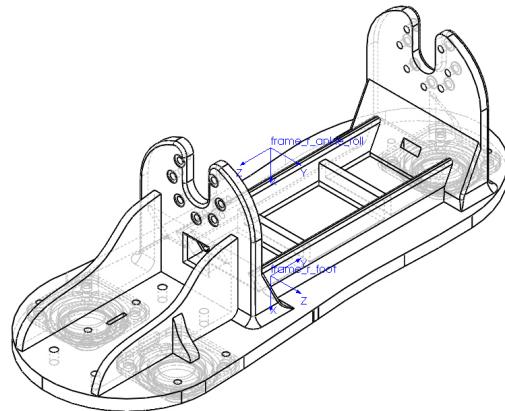
(ก) DH Parameter

Link	r_ankle_roll
Mass (kg)	0.10500000
CoM X (m)	0.03625882
CoM Y (m)	-0.00019548
CoM Z (m)	0.00034576
Inertia Ixx	0.00034591
Inertia Ixy	-0.00000013
Inertia Ixz	0.00000857
Inertia Iyy	0.00032705
Inertia Iyz	0.00000120
Inertia Izz	0.00004813

(ข) URDF

ตารางที่ ก.12: ตารางแสดงค่าพารามิเตอร์ Right Ankle Roll

Left Ankle Roll



รูปที่ ก.13: ภาพแสดงก้านต่อ Left Ankle Roll

Link	l_ankle_roll
Mass (kg)	0.10500000
CoM X (m)	-0.01454118
CoM Y (m)	-0.00034576
CoM Z (m)	-0.00019548
Inertia Ixx	0.00034591
Inertia Ixy	-0.00000857
Inertia Ixz	-0.00000013
Inertia Iyy	0.00004813
Inertia Iyz	-0.00000120
Inertia Izz	0.00032705

(ก) DH Parameter

Link	l_ankle_roll
Mass (kg)	0.10500000
CoM X (m)	0.03625882
CoM Y (m)	-0.00019548
CoM Z (m)	0.00034576
Inertia Ixx	0.00034591
Inertia Ixy	-0.00000013
Inertia Ixz	0.00000857
Inertia Iyy	0.00032705
Inertia Iyz	0.00000120
Inertia Izz	0.00004813

(ข) URDF

ตารางที่ ก.13: ตารางแสดงค่าพารามิเตอร์ Left Ankle Roll

ประวัติผู้เขียน

นายจิรภพ ศรีรัตนอาภรณ์



ชื่อ สกุล	นายจิรภพ ศรีรัตนอาภรณ์
รหัสนักศึกษา	57340500067
วุฒิการศึกษา	วิศวกรรมศาสตรบัณฑิต
วิศวกรรมหุ่นยนต์และระบบอัตโนมัติ	
ชื่อสถาบัน	มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
ปีที่สำเร็จการศึกษา	2560

ประวัติผู้เขียน

นายเจษฎากร ท่าไชยวงศ์



ชื่อ สกุล	นายเจษฎากร ท่าไชยวงศ์
รหัสนักศึกษา	57340500067
วุฒิการศึกษา	วิศวกรรมศาสตรบัณฑิต
	วิศวกรรมหุ่นยนต์และระบบอัตโนมัติ
ชื่อสถาบัน	มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
ปีที่สำเร็จการศึกษา	2560

ประวัติผู้เขียน

นายวุฒิภัทร โชคอนันตทรัพย์



ชื่อ สกุล	นายวุฒิภัทร โชค_anantraphi
รหัสนักศึกษา	57340500067
วุฒิการศึกษา	วิศวกรรมศาสตรบัณฑิต
ชื่อสถาบัน	วิศวกรรมที่นี่ยนต์และระบบอัตโนมัติ
ปีที่สำเร็จการศึกษา	มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
	2560