



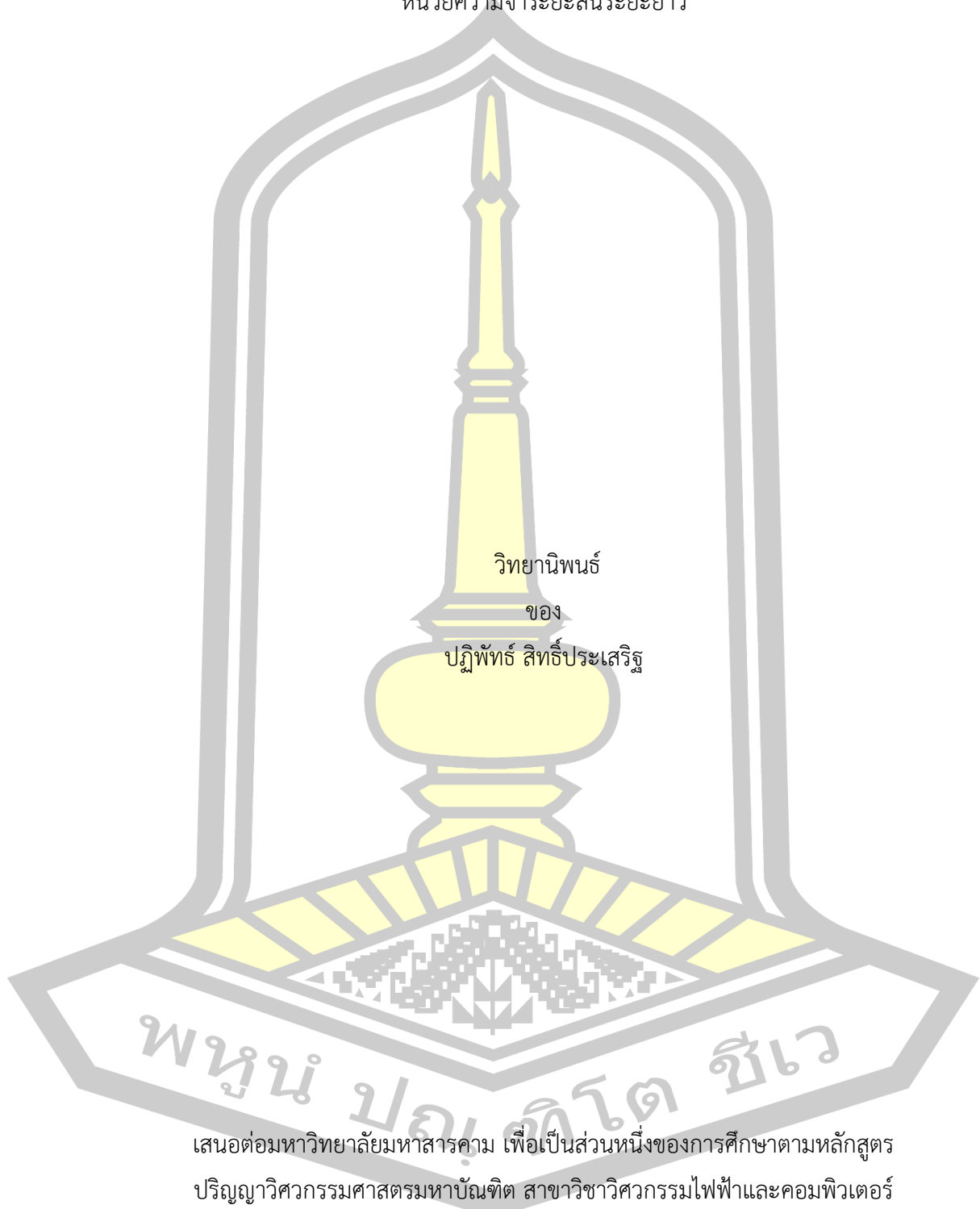
การตรวจจับการลัดวงจรโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันร่วมกับโครงข่ายประสาทเทียมแบบ  
หน่วยความจำระยะสั้นระยะยาว

วิทยานิพนธ์  
ของ  
ปฏิพัทธ์ สิทธิประเสริฐ

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์  
มิถุนายน 2566

ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม

การตรวจจับการลัดโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันร่วมกับโครงข่ายประสาทเทียมแบบ  
หน่วยความจำระยะสั้นระยะยาว



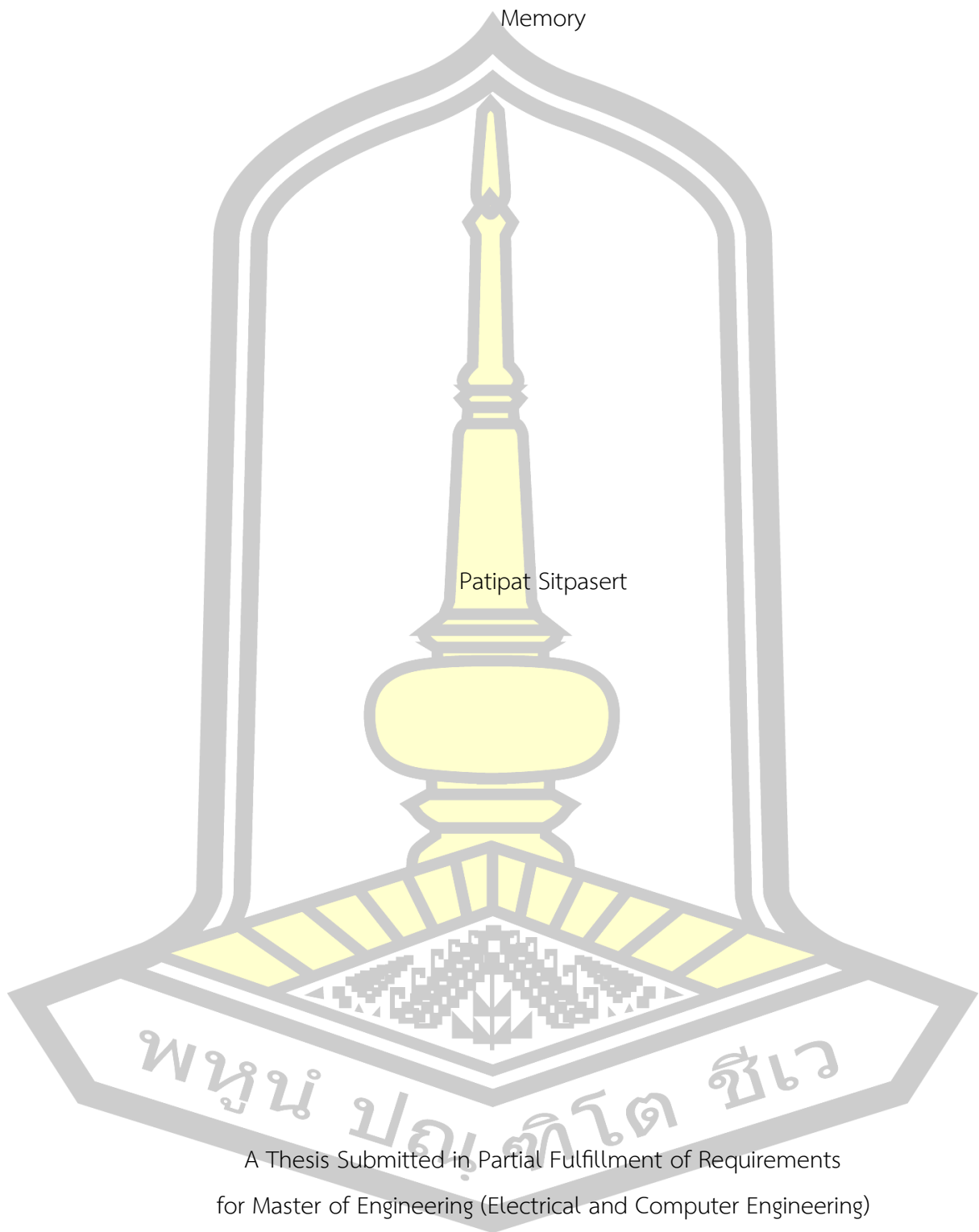
เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

มิถุนายน 2566

ลิขสิทธิเป็นของมหาวิทยาลัยมหาสารคาม

Vision-Based Fall Detection Using Convolution Neural Network and Long Short-Term

Memory



Patipat Sitpasert

A Thesis Submitted in Partial Fulfillment of Requirements  
for Master of Engineering (Electrical and Computer Engineering)

June 2023

Copyright of Maharakham University



คณะกรรมการสอบวิทยานิพนธ์ ได้พิจารณาวิทยานิพนธ์ของนายปฏิพัทธ์ สิทธิประเสริฐ  
แล้วเห็นสมควรรับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ ของมหาวิทยาลัยมหาสารคาม

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(รศ. ดร. อนันต์ เครือทรัพย์ถาวร )

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(รศ. ดร. วรวัฒน์ เสี่ยงมิบูล )

.....กรรมการ

(ผศ. ดร. นิวัตร อังควิเศษฐพันธ์ )

.....กรรมการ

(ผศ. ดร. สุพรรณนิกา วัฒนชะ )

มหาวิทยาลัยอนุมัติให้รับวิทยานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญา วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ ของมหาวิทยาลัย  
มหาสารคาม

.....  
(รศ. ดร. เกียรติศักดิ์ ศรีประทีป )

.....  
(รศ. ดร. กริสน์ ชัยมูล )

คณบดีคณะวิศวกรรมศาสตร์

คณบดีบัณฑิตวิทยาลัย

<b>ชื่อเรื่อง</b>	การตรวจจับการล้มโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันร่วมกับโครงข่ายประสาทเทียมแบบหน่วยความจำระยะสั้นระยะยาว		
<b>ผู้วิจัย</b>	ปฏิพัทธ์ สิทธิประเสริฐ		
<b>อาจารย์ที่ปรึกษา</b>	รองศาสตราจารย์ ดร. วรวัฒน์ เสงี่ยมวิบูล		
<b>ปริญญา</b>	วิศวกรรมศาสตรมหาบัณฑิต	<b>สาขาวิชา</b>	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
<b>มหาวิทยาลัย</b>	มหาวิทยาลัยมหาสารคาม	<b>ปีที่พิมพ์</b>	2566

### บทคัดย่อ

การวิจัยนี้มีจุดมุ่งหมายเพื่อนำเสนอวิธีการใช้การมองเห็นของคอมพิวเตอร์เพื่อตรวจจับและจำแนกการล้มของมนุษย์ในวิดีโอ โดยใช้อัลกอริทึม BlazePose เพื่อตรวจจับและสกัด 33 จุดบนร่างกายมนุษย์ จากนั้นเลือก 4 จุด เพื่อเป็นตัวแทนส่วนบนของร่างกาย แล้ววาดเส้นตรง "r" เพื่อคำนวณมุมของส่วนบนของร่างกาย ความเร็วเชิงเส้นและความเร็วเชิงมุมเพื่อนำไปวิเคราะห์ว่าเกิดการล้มของบุคคลที่ตรวจพบในวิดีโอหรือไม่ ข้อมูลเหล่านี้คล้ายกับข้อมูลที่ได้รับจากเซ็นเซอร์วัดความเร่ง และเซ็นเซอร์วัดมุม จากนั้นได้ใช้ความสามารถของโครงข่ายประสาทเทียมแบบคอนโวลูชันร่วมกับโครงข่ายประสาทเทียมแบบหน่วยความจำระยะสั้นระยะยาวเพื่อสร้างโมเดลสำหรับการตรวจจับการล้ม และใช้อัลกอริทึม DeepSORT ในการติดตามบุคคลในวิดีโอและระบุว่าใครล้ม ในการทดลองได้ทำการทดลองด้วยชุดข้อมูล 3 ชุด และปรากฏว่าโมเดลให้ผลลัพธ์ความถูกต้องสูงถึงร้อยละ 96.66 ค่าสามารถในการตรวจจับเหตุการณ์ล้มร้อยละ 89.95 ค่าความแม่นยำร้อยละ 96.72 และคะแนน F1 ร้อยละ 93.08

คำสำคัญ : การตรวจจับการล้ม, การเรียนรู้ของเครื่อง, อนุกรมเชิงเวลา, โครงข่ายประสาทเทียมแบบหน่วยความจำระยะสั้นระยะยาว, โครงข่ายประสาทเทียมแบบคอนโวลูชัน

พูนัน ปณุกิตโต ชีเว

<b>TITLE</b>	Vision-Based Fall Detection Using Convolution Neural Network and Long Short-Term Memory		
<b>AUTHOR</b>	Patipat Sitpasert		
<b>ADVISORS</b>	Associate Professor Worawat Sa-Ngiamvibool , Ph.D.		
<b>DEGREE</b>	Master of Engineering	<b>MAJOR</b>	Electrical and Computer Engineering
<b>UNIVERSITY</b>	Maharakham University	<b>YEAR</b>	2023

### ABSTRACT

The dissertation presents a method of vision-based model for detecting and classifying human falls in video sequences. We used BlazePose to detect and extract 33 body landmarks of a human body; then, we selected 4 points to represent the upper body. Then, we draw a straight line "r" to calculate the angle of the upper body, linear velocity, and angular velocity to help determine if the person detected has fallen. These data are similar to the data obtained from gyroscope and accelerometer sensors. We then use the capabilities of CNN and LSTM to construct a model for fall detection. In addition, we used DeepSORT to track people in the video and identify who fell. We conducted experiments on three datasets, and our model achieved a high accuracy rate of 96.66%, recall of 89.95%, precision of 96.72% and F1 score of 93.08%.

Keyword : Fall detection, Deep learning, DeepSORT, CNN, LSTM

พหุบัณฑิตวิทยาลัย

## กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี เนื่องจากได้รับความกรุณาและความช่วยเหลือเป็นอย่างดีจาก ศาสตราจารย์ ดร.วรวิวัฒน์ เสงี่ยมวิบูล อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้คำแนะนำและตรวจสอบแก้ไขข้อบกพร่องต่างๆ ด้วยความเอาใจใส่เป็นอย่างดี ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้ ขอกราบขอบพระคุณพ่อ แม่ และครอบครัว ผู้เป็นเบื้องหลัง คอยสนับสนุนและให้กำลังใจ ขอขอบพระคุณบูรพาจารย์ที่ประสิทธิ์ประสาทวิชาอันเป็นประโยชน์ที่ยิ่งใหญ่แก่ผู้วิจัยมาโดยตลอด สุดท้ายนี้ ผู้วิจัยขอขอบพระคุณบุคคลที่ไม่ได้เอ่ยชื่อมา ณ ที่นี้ แต่ได้มีส่วนช่วยเหลือและเฝ้ากำลังใจสนับสนุนในวิทยานิพนธ์เล่มนี้จนสำเร็จด้วยดี

ปฏิพัทธ์ สิทธิประเสริฐ



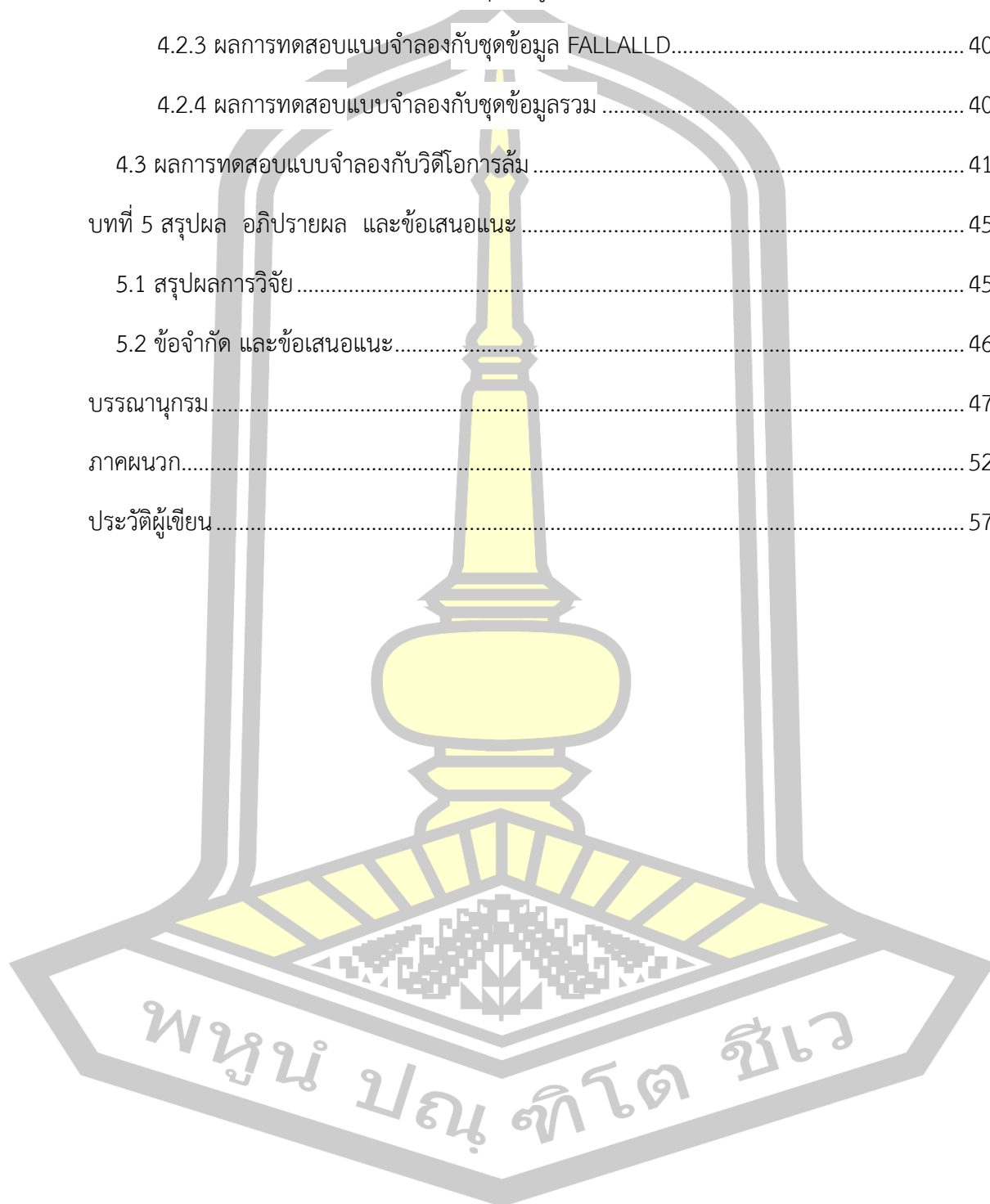
## สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ญ
สารบัญภาพ.....	ฎ
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญของงานวิจัย.....	1
1.2 ความมุ่งหมายของงานวิจัย.....	2
1.3 สมมุติฐานงานวิจัย.....	2
1.4 ขอบเขตงานวิจัย.....	2
1.5 ผลที่คาดว่าจะได้รับ.....	3
บทที่ 2 ปริทัศน์เอกสารข้อมูล และทฤษฎีที่เกี่ยวข้อง.....	4
2.1 พีเจอร์ที่ใช้ในการตรวจจับการล้ม (Fall Detection Features).....	4
2.2 เซ็นเซอร์แบบสวมใส่ (Wearable Sensors).....	6
2.3 เซ็นเซอร์แบบติดตั้งไว้กับที่ (Fixed Location Sensors).....	7
2.4 การตรวจจับโดยใช้กล้อง (RGB Camera Systems).....	8
2.5 แบบผสมผสาน (Combinatory Systems).....	8
2.6 อัลกอริทึม (Algorithms).....	9
2.6.1 ใช้ข้อมูลภาพแบบ RGB.....	9
2.6.2 ใช้ข้อมูลความลึก (Depth).....	11



2.6.3 ใช้ข้อมูลเซ็นเซอร์ความเร่ง (accelerometer).....	12
2.7 ทฤษฎีที่เกี่ยวข้อง .....	13
2.7.1 CNN-LSTM.....	13
2.7.2 DeepSORT.....	15
บทที่ 3 วิธีดำเนินการวิจัย.....	18
3.1 ชุดข้อมูลวิดีโอ (Datasets).....	19
3.1.1 ImVia Fall Detection Dataset [4] .....	19
3.1.2 UR Fall Detection Dataset [26].....	19
3.1.3 FALLALLD [5] .....	20
3.2 การจัดเตรียมข้อมูล (Data Pre-processing).....	22
3.2.1 โครงกระดูก 3 มิติ (3D Skeleton).....	22
3.2.2 การคำนวณมุม (Body Angle) ความเชิงเส้น (Linear Velocity) และ ความเร็วเชิงมุม (Angular Velocity) ของร่างกาย.....	24
3.3 การสอนแบบจำลอง CNN-LSTM .....	27
3.4 ทดสอบแบบจำลอง.....	28
3.4.1 แบบจำลองมาร์คอฟซ่อนเร้น (HMM).....	28
3.4.2 ตัวตรวจจับแบบ YOLOv5.....	29
3.4.3 ตัวติดตามแบบ DeepSORT.....	30
3.5 Evaluation.....	31
บทที่ 4 ผลการวิจัยและการอภิปราย.....	33
4.1 การหาค่าพารามิเตอร์ที่ดีที่สุด.....	33
4.1.1 โครงสร้างของ CNN-LSTM.....	33
4.1.2 จำนวนรอบการเรียนรู้ (Epoch).....	37
4.2 ผลการทดสอบแบบจำลอง.....	38

4.2.1 ผลการทดสอบแบบจำลองกับชุดข้อมูล ImVia Fall Detection Dataset.....	39
4.2.2 ผลการทดสอบแบบจำลองกับชุดข้อมูล UR Fall Detection Dataset .....	39
4.2.3 ผลการทดสอบแบบจำลองกับชุดข้อมูล FALLALLD.....	40
4.2.4 ผลการทดสอบแบบจำลองกับชุดข้อมูลรวม .....	40
4.3 ผลการทดสอบแบบจำลองกับวิดีโอการล้ม .....	41
บทที่ 5 สรุปผล อภิปรายผล และข้อเสนอแนะ .....	45
5.1 สรุปผลการวิจัย .....	45
5.2 ข้อจำกัด และข้อเสนอแนะ.....	46
บรรณานุกรม.....	47
ภาคผนวก.....	52
ประวัติผู้เขียน.....	57



## สารบัญตาราง

	หน้า
ตารางที่ 1 ตัวอย่างการกิจกรรมต่างๆ.....	22
ตารางที่ 2 ผลการทดสอบโครงสร้าง CNN-LSTM แบบต่างๆ.....	36
ตารางที่ 3 ผลการทดสอบแบบจำลอง CNN-LSTM ที่มีการสอนด้วยรอบการเรียนรู้ต่างกัน .....	37
ตารางที่ 4 Confusion Matrix ของแบบจำลองกับชุดข้อมูล ImVia Fall Detection Dataset.....	39
ตารางที่ 5 ผลการวัดประสิทธิภาพของแบบจำลองกับชุดข้อมูล ImVia Fall Detection Dataset ..	39
ตารางที่ 6 Confusion Matrix ของแบบจำลองกับชุดข้อมูล UR Fall Detection Dataset .....	39
ตารางที่ 7 ผลการวัดประสิทธิภาพของแบบจำลองกับชุดข้อมูล UR Fall Detection Dataset .....	39
ตารางที่ 8 Confusion Matrix ของแบบจำลองกับชุดข้อมูล FALLALLD.....	40
ตารางที่ 9 ผลการวัดประสิทธิภาพของแบบจำลองกับชุดข้อมูล FALLALLD.....	40
ตารางที่ 10 Confusion Matrix ของแบบจำลองกับชุดข้อมูลรวม .....	40
ตารางที่ 11 ผลการวัดประสิทธิภาพของแบบจำลองกับชุดข้อมูลรวม.....	40
ตารางที่ 12 สรุปผลการวัดประสิทธิภาพของแบบจำลอง .....	41
ตารางที่ 13 ตัวอย่างข้อมูล $\emptyset$ แบบอนุกรมเวลาที่ใช้ในการสอน และทดสอบแบบจำลองจำนวน 100 ชุดข้อมูล หน้าต่างขนาด 24 เฟรม.....	52

พูน ปณ ทิโต ชีเว

## สารบัญภาพ

	หน้า
ภาพที่ 1 โครงสร้างของระบบตรวจจับการล้มโดยใช้ CNN-LSTM .....	13
ภาพที่ 2 โครงสร้างของ LSTM.....	14
ภาพที่ 3 โครงสร้างของ DeepSORT .....	15
ภาพที่ 4 Deep Appearance Descriptor.....	16
ภาพที่ 5 ขั้นตอนการทำวิจัย.....	19
ภาพที่ 6 ตัวอย่างวิดีโอจากชุดข้อมูล ImVia Fall Detection Dataset [4].....	20
ภาพที่ 7 ตัวอย่างวิดีโอจากชุดข้อมูล UR Fall Detection Dataset [51].....	20
ภาพที่ 8 ตัวอย่างวิดีโอจากชุดข้อมูล FALLALLD [5] .....	21
ภาพที่ 9 ขั้นตอนการจัดเตรียมข้อมูล .....	21
ภาพที่ 10 ตำแหน่ง 33 จุดสำคัญที่ได้จากโมเดล BlazePose GHUM.....	23
ภาพที่ 11 ตัวอย่าง 3D Skeleton ที่ได้จาก BlazePose GNUM.....	23
ภาพที่ 12 พิกัดกล้อง (Camera Coordinate).....	24
ภาพที่ 13 จุดแทนร่างกายท่อนบน(ลำตัว) 4 จุด.....	24
ภาพที่ 14 เส้นตรง $r$ และ มุมแกน $\phi$ , $\theta$ และ $\varphi$ .....	26
ภาพที่ 15 ข้อมูลชนิดอนุกรมเวลาสำหรับ CNN-LSTM.....	26
ภาพที่ 16 โครงสร้างของระบบตรวจจับการล้มโดยใช้ CNN-LSTM .....	27
ภาพที่ 17 โครงสร้างส่วนของ CNN.....	28
ภาพที่ 18 Object Modelling.....	29
ภาพที่ 19 Hidden Markov Model .....	29
ภาพที่ 20 โครงสร้างของ DeepSORT .....	30
ภาพที่ 21 Confusion matrix.....	31

ภาพที่ 22 โครงสร้างรูปแบบที่ 1 มี CNN-LSTM 4 ชุดขนานกัน ..... 33

ภาพที่ 23 โครงสร้างรูปแบบที่ 2 มี CNN-LSTM 3 ชุดขนานกัน ..... 34

ภาพที่ 24 โครงสร้างรูปแบบที่ 3 มี CNN-LSTM 2 ชุดขนานกัน ..... 34

ภาพที่ 25 โครงสร้างรูปแบบที่ 4 มี CNN-LSTM 1 ชุด ..... 35

ภาพที่ 26 โครงสร้างส่วนของ CNN..... 35

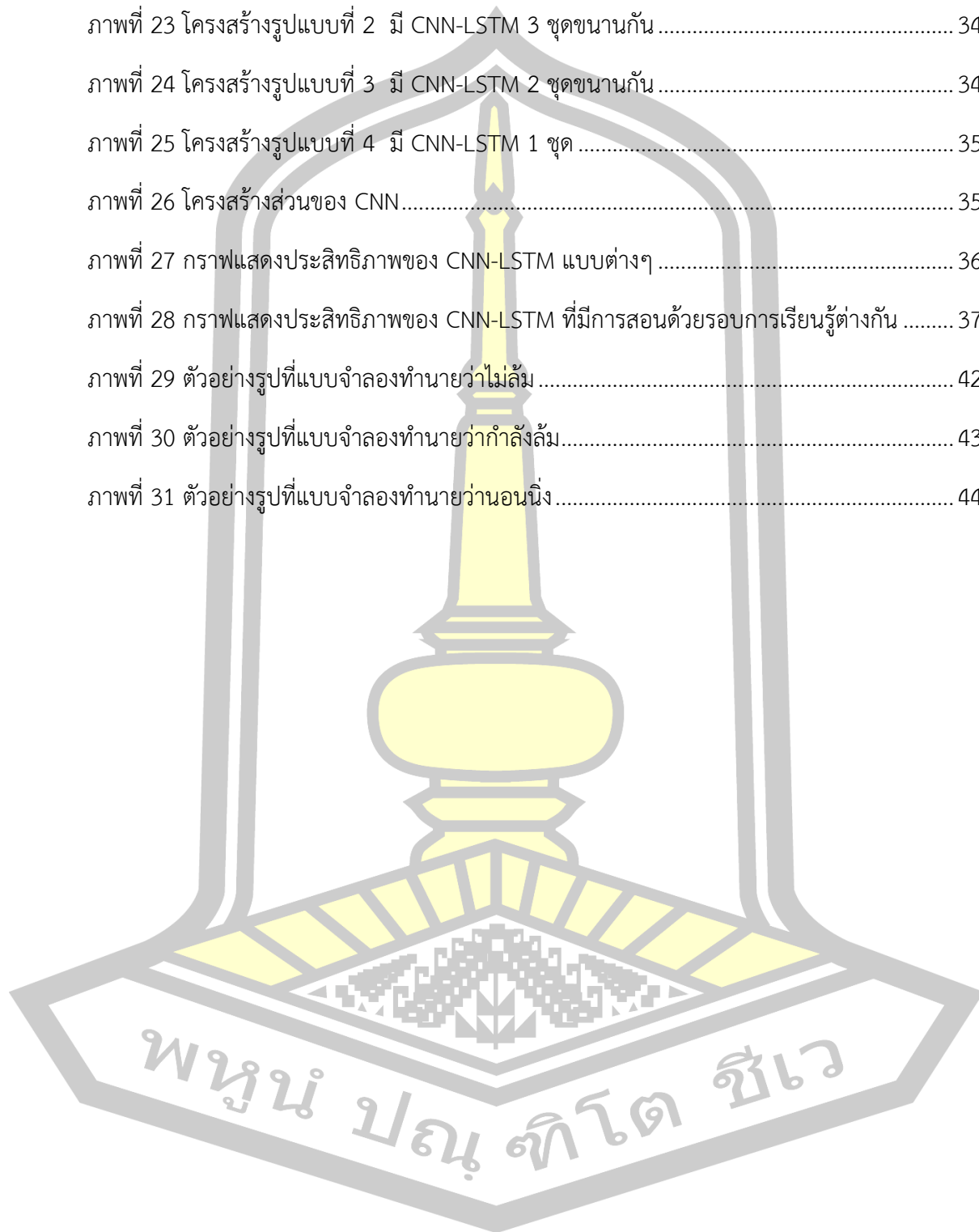
ภาพที่ 27 กราฟแสดงประสิทธิภาพของ CNN-LSTM แบบต่างๆ ..... 36

ภาพที่ 28 กราฟแสดงประสิทธิภาพของ CNN-LSTM ที่มีการสอนด้วยรอบการเรียนรู้ต่างกัน ..... 37

ภาพที่ 29 ตัวอย่างรูปที่แบบจำลองทำนายว่าไม่ล้ม ..... 42

ภาพที่ 30 ตัวอย่างรูปที่แบบจำลองทำนายว่ากำลังล้ม ..... 43

ภาพที่ 31 ตัวอย่างรูปที่แบบจำลองทำนายว่านอนนิ่ง ..... 44



## บทที่ 1

### บทนำ

#### 1.1 ที่มาและความสำคัญของงานวิจัย

ในปัจจุบันสังคมไทยกำลังเข้าสู่สังคมผู้สูงอายุ มีหลายครอบครัวที่ผู้สูงอายุอาศัยอยู่เพียงลำพังหรืออยู่คนเดียวในเวลาที่ลูกหลานออกไปทำงาน ปัญหาการหกล้มในผู้สูงอายุมีอันตรายกว่าคนวัยอื่นหลายเท่าตัว อีกทั้งในผู้สูงอายุยังมีความเสี่ยงต่อการหกล้มได้ง่ายเนื่องจากความเสื่อมของร่างกาย ร่างกายที่ไม่แข็งแรง และการทรงตัวที่ไม่ดีพอ โดยในผู้ที่มีอายุ 65 ปีขึ้นไป มีความเสี่ยงต่อการหกล้มถึงร้อยละ 28–35 และความเสี่ยงจะเพิ่มขึ้นเป็นร้อยละ 32–42 ในผู้ที่มีอายุ 70 ปีขึ้นไป [1] ยิ่งไปกว่านั้นจากสถิติเกี่ยวกับการบาดเจ็บของกองสุขศึกษา กระทรวงสาธารณสุข ยังพบว่ามียัตราการเสียชีวิตจากการหกล้มสูงเป็นอันดับ 2 ในกลุ่มของการบาดเจ็บโดยไม่ได้ตั้งใจอีกด้วย [2]

องค์การอนามัยโลก (World Health Organization: WHO) นิยามการล้มว่าเป็นเหตุการณ์ที่บุคคลเปลี่ยนท่าทางลงมาอยู่บนพื้นดิน พื้น หรือระดับที่ต่ำกว่าโดยไม่ได้ตั้งใจ การล้มอาจเกิดขึ้นจากการหมดสติ ลื่นโดยบังเอิญ สะดุดขณะยืน หรือเดิน เกิดได้จากทั้งปัจจัยภายใน และภายนอก ปัจจัยภายในรวมไปถึงโรคประจำตัว ได้แก่ โรคเกี่ยวกับระบบประสาท โรคหัวใจ หรือ ความผิดปกติอื่นๆ ผลข้างเคียงจากการใช้ยา ขาดการออกกำลังกายอย่างสม่ำเสมอ และเสียการควบคุม โดยเฉพาะในหมู่ผู้สูงอายุซึ่งจะมีความคล่องแคล่ว ความจำ และการมองเห็นที่ลดลง โดยเฉพาะอย่างยิ่งในหมู่ผู้ที่อาศัยอยู่ในสถาบัน เช่น บ้านพักคนชรา หรือสถานพยาบาลโรคเรื้อรังต่างๆ ปัจจัยภายนอกก็จะมีส่วนทำให้เกิดการล้มได้ เช่น บ้านที่มีคนอยู่อาศัยจำนวนมาก ทางเดินเท้าที่ชำรุด เลื่อนกถั่ว เป็นต้น บางทีสภาวะแวดล้อมที่ไม่ปลอดภัยก็อาจทำให้เกิดการล้มได้ เช่น ไม่สมดุล การมองเห็นจำกัด ทำงานบนที่สูง และการทำงานในสภาวะที่อันตรายต่างๆ การล้มหลายประเภทเกิดจากสภาพแวดล้อมในการทำงานเอง เช่น สภาพของทางเดิน การมองเห็นจำกัด และ ไม่มีสมาธิ ความเหนื่อยล้า อื่นๆ และนักกีฬามีแนวโน้มที่จะเกิดการล้มมากกว่าคนทั่วไป

การหกล้มสามารถทำให้อวัยวะต่างๆ เช่น แขน ขา ศีรษะ ฯลฯ เกิดการกระแทก และมีอาการบาดเจ็บตามมา อาจบาดเจ็บเพียงเล็กน้อยไปจนถึงบาดเจ็บรุนแรงและรุนแรงที่สุดคือเสียชีวิต สำหรับคนทั่วไปที่สุขภาพดี และไม่ใช่นักสูงอายุเมื่อเกิดการล้มมักจะไม่ทำให้เกิดการบาดเจ็บร้ายแรง และสามารถรักษาให้หายได้ค่อนข้างเร็ว ในขณะที่ปัญหาการหกล้มเกิดมากกับผู้สูงอายุและมีความอันตรายมากกว่าคนวัยอื่น จากสถิติยังพบว่าผู้สูงอายุเสียชีวิตจากการหกล้มเฉลี่ย 3 คนต่อวัน [3]

แต่ปัญหาดังกล่าวสามารถป้องกันและแก้ไขได้หากเข้าใจถึงความเสี่ยงและการดูแลที่เหมาะสม รวมถึงถ้าเกิดการล้มขึ้นแล้วมีการช่วยเหลือที่ทันท่วงทีจะสามารถช่วยลดอาการบาดเจ็บและอัตราการเสียชีวิตลงได้

## 1.2 ความมุ่งหมายของงานวิจัย

1.2.1 เพื่อศึกษาและวิเคราะห์ท่าทางของคน สำหรับใช้ในการหาคุณลักษณะของข้อมูลเพื่อนำเข้าโครงข่ายการเรียนรู้เชิงลึก

1.2.2 เพื่อศึกษาและพัฒนาวิธีการตรวจจับการล้มด้วยการตรวจจับท่าทางการล้ม ที่มีประสิทธิภาพ

1.2.3 เพื่อเผยแพร่แนวคิดให้ผู้ที่สนใจนำไปพัฒนาในงานวิจัย หรือประยุกต์ใช้ในชีวิตจริง

## 1.3 สมมุติฐานงานวิจัย

1.3.1 ระบบตรวจจับการล้มสามารถบอกได้ว่ามีคนล้มหรือไม่ล้ม

1.3.2 ระบบมีความถูกต้องไม่น้อยกว่าร้อยละ 90

## 1.4 ขอบเขตงานวิจัย

1.4.1 ใช้ข้อมูลจำนวน 3 ชุดข้อมูลดังนี้

1) **ImVia Fall Detection Dataset:** มีทั้งหมด 191 วิดีโอ ซึ่งมีทั้งกิจกรรมปกติ และการล้ม โดยวิดีโอถูกบันทึกจากสถานที่ที่แตกต่างกัน เช่น บ้าน ห้องกาแฟ ออฟฟิศ [4]

2) **UR Fall Detection Dataset:** ชุดข้อมูลนี้มีทั้งหมด 70 วิดีโอ โดยมี 30 วิดีโอ สำหรับการล้ม และ 40 วิดีโอ ที่เป็นกิจกรรมทั่วไป การล้มถูกบันทึกด้วยกล้องไคนะ 2 ตัว แต่กิจกรรมทั่วไปจะบันทึกจากกล้องแค่ 1 ตัว

3) **FALLALLD:** ชุดข้อมูลนี้มี 7 วิดีโอ ครอบคลุมทั้งการล้ม และกิจกรรมทั่วไปในชีวิตประจำวัน ล้มขณะเดิน ล้มไปข้างหน้า ล้มไปข้างหลังจากการลื่น ล้มตอนกำลังนั่ง [5]

1.4.2 เก็บข้อมูล ground truth จาก ชุดข้อมูลเอง

1.4.3 จำแนก 3 สถานะคือการไม่ล้ม กำลังล้ม และ นอนนิ่ง(หลังจากล้ม)

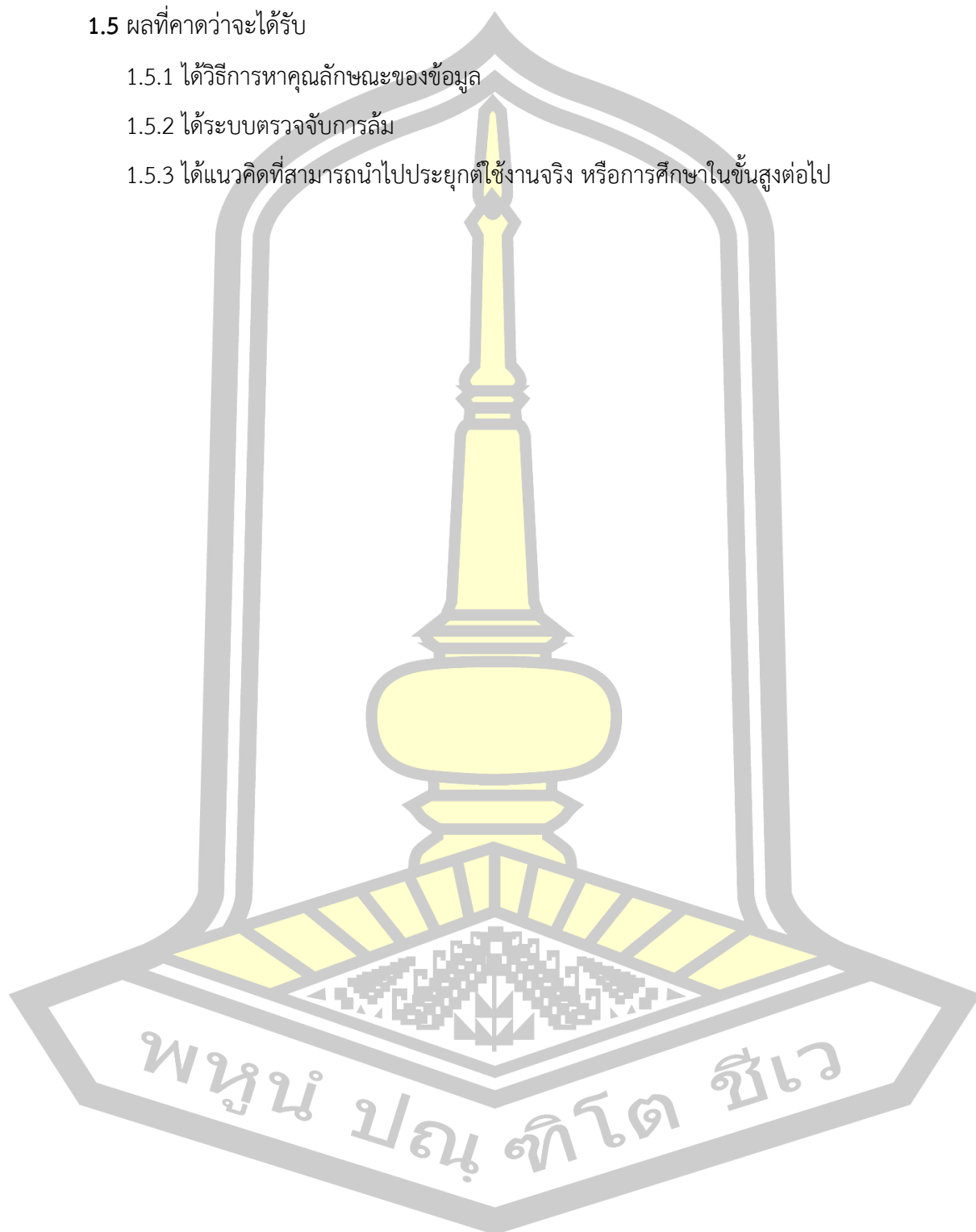
1.4.4 ใช้ Confusion Matrix ในการประเมินประสิทธิภาพโดยรวมของแบบจำลอง

### 1.5 ผลที่คาดว่าจะได้รับ

1.5.1 ได้วิธีการหาคุณลักษณะของข้อมูล

1.5.2 ได้ระบบตรวจจับการล้ม

1.5.3 ได้แนวคิดที่สามารถนำไปประยุกต์ใช้งานจริง หรือการศึกษาในขั้นสูงต่อไป





## ปริทัศน์เอกสารข้อมูล และทฤษฎีที่เกี่ยวข้อง

งานวิจัยเกี่ยวกับระบบตรวจจับการล้มได้มีการศึกษาโดยการใช้เซ็นเซอร์ในหลายๆ รูปแบบ เช่น เซ็นเซอร์ความเร่ง (Accelerometers) เซ็นเซอร์เสียง (Acoustic) เซ็นเซอร์อัลตราโซนิก (Ultrasonic) เซ็นเซอร์อินฟราเรด (Infra-red) เรดาร์ (Radar) กล้องอาร์จีบี (RGB cameras) เซ็นเซอร์ความลึก (Depth) และอื่นๆ ซึ่งวิธีการส่วนใหญ่จะใช้การตรวจจับความเร่งของร่างกายขณะที่เกิดการล้ม หรือตรวจจับการเปลี่ยนแปลงรูปร่างเงาของบุคคล (Silhouette) เช่น จากตำแหน่งตั้งตรงไปยังตำแหน่งนอนหงายหรือคว่ำบนพื้น

ในบทนี้จะกล่าวถึงฟีเจอร์ (Features) หรือลักษณะเฉพาะ ประเภทที่มีบทบาทสำคัญในทางวิสัยทัศน์คอมพิวเตอร์ (Computer Vision) เทคโนโลยีทางด้านฮาร์ดแวร์ (Hardware Technologies) โดยแบ่งออกเป็น 2 กลุ่มหลักๆ คืออุปกรณ์สวมใส่ (Wearable) และแบบเซ็นเซอร์ติดตั้งอยู่กับที่ (Static Sensors) สุดท้ายจะกล่าวถึงเทคนิคทางด้านอัลกอริทึม (Algorithmic Techniques) ต่างๆ ที่ใช้ในการจำแนกเหตุการณ์ล้ม หรือเหตุการณ์อื่นๆ

### 2.1 ฟีเจอร์ที่ใช้ในการตรวจจับการล้ม (Fall Detection Features)

มีการใช้ฟีเจอร์หลายๆอย่างในการตรวจจับการล้ม และงานวิจัยต่างๆ ก็พยายามเลือกใช้ฟีเจอร์ที่ดีที่สุด ฟีเจอร์ดังกล่าวได้แก่ กล่องขอบเขต (Bounding Boxes) ลักษณะรูปร่าง (Shape Descriptors) ความเร่ง (Velocity) ความสูงจากพื้นของคน มุมการล้ม (Fall Angle) และอื่นๆ การใช้กล่องขอบเขต 2 มิติ (2D) จะใช้การวิเคราะห์การเปลี่ยนแปลงการเคลื่อนที่ ส่วน 3 มิติ (3D) จะสนใจความเร็วของการเปลี่ยนแปลงความสูง ความกว้าง และ ความลึกของกล่องขอบเขต หลายๆงานวิจัยใช้อัตราส่วนภาพ (Aspect Ratio) เป็นฟีเจอร์โดยคำนวณจากอัตราส่วนของความกว้างและความสูงของกล่องขอบเขตที่ตรวจจับได้ ถ้าอัตราส่วนภาพน้อยแสดงว่าท่าทางของบุคคลนั้นอยู่ในแนวตั้งในทางตรงข้ามถ้าอัตราส่วนภาพมากแสดงว่าท่าทางอยู่ในแนวนอน

ในหลายๆงานวิจัยจะใช้วงรี (Ellipse) ครอบตัวคนเป็นวัตถุ แล้วใช้ระยะจากจุดศูนย์กลางวงรีไปถึงพื้นในระนาบ 3 มิติเป็นฟีเจอร์อีกที หรือ ใช้มุมการเอียงของแกนหลักของวงรีที่ทำกับพื้นห้องเป็นฟีเจอร์เพื่อตรวจจับการล้ม ถ้ามุมมีค่าน้อยจะตัดสินว่าเกิดการล้ม

บางงานวิจัย [1] [2] ใช้ระยะจากตำแหน่งของศีรษะไปถึงพื้นเป็นพีเจอร์ถ้าศีรษะต่ำกว่าเกณฑ์ (Threshold) จะถูกจำแนกว่าเกิดการล้ม และ บางงานวิจัย [1] ใช้ความเร็วของศีรษะเป็นพีเจอร์ถ้าความเร็วของศีรษะมากกว่าเกณฑ์จะถูกจำแนกว่าเกิดเหตุการณ์ล้มขึ้น

รูปร่างของคนทั้งใน 2 มิติ และ 3 มิติถูกนำมาใช้วิเคราะห์เพื่อตรวจจับการล้ม ในงานวิจัย [3] ใช้การหาตำแหน่งจุดศูนย์กลาง และ มุมของร่างกายคนเทียบกับพื้นระนาบของภาพเงาของคน (Human Silhouette) เพื่อตรวจจับเหตุการณ์ล้ม ในงานวิจัย [6] ใช้กล่องขอบเขตภาพเงาของคน จากนั้นใช้อัตราส่วนภาพ และ เมทริกซ์ความแปรปรวน (Covariance Matrix) เป็นพีเจอร์เพื่อตรวจจับการล้มอีกที

ในหลายๆ งานวิจัยใช้ข้อมูลโครงกระดูก (Skeleton) 2 มิติ หรือ 3 มิติซึ่งได้มาจากกล้องทั่วไป หรือ เซ็นเซอร์ไคเนกซ์ (Kinect Sensor) เป็นพีเจอร์ ใน [7] ข้อมูลโครงกระดูกถูกสกัดและติดตามโดยชุดโปรแกรมของไมโครซอฟต์ไคเนกซ์ (Microsoft Kinect SDK) ซึ่งจะได้ความเร็ว และ ระยะของแต่ละข้อต่อ (Joint) ถึงพื้นพร้อมๆ กัน ถ้าความเร็วของข้อต่อ และระยะถึงพื้นต่ำกว่าเกณฑ์ จะถูกจำแนกว่าเกิดเหตุการณ์ล้มขึ้น ใน [8] ใช้ข้อมูลโครงกระดูก 3 มิติเพื่อคำนวณหามุมระหว่างเส้นต่างๆ โดยใช้จุดศีรษะ ไหล่ กระดูกสันหลัง สะโพก และข้อเข่า เพื่อหาความเอียงของร่างกายเมื่อเกิดการล้มขึ้น

กล่องขอบเขต (Bounding Boxes) เพียงอย่างเดียวไม่ได้ให้ข้อมูลเพียงพอเกี่ยวกับการเคลื่อนไหวของคน และประสิทธิภาพของเทคนิคนี้ขึ้นอยู่กับมุมมองของกล้องด้วย โดยเฉพาะอย่างยิ่งสำหรับวิธีแบบ 2 มิติ อัตราส่วนภาพ (Aspect Ratio) อาจคลาดเคลื่อนได้เนื่องจากตำแหน่งของบุคคล กล้อง และหากมีวัตถุที่บังตัวคน พีเจอร์แบบภาพเงา (Silhouette) ก็มีปัญหากการบดบัง และขึ้นอยู่กับมุมมองของกล้องเหมือนกัน ตำแหน่งของส่วนหัว (Head Location) นั้นดูมีความเสถียรจากการบดบังจากวัตถุมากกว่าเนื่องจากตำแหน่งของส่วนหัวของมันไม่ขึ้นกับส่วนอื่นๆ ของร่างกายที่ตรวจจับได้ แต่การตรวจจับศีรษะอาจเป็นปัญหาได้ในกรณีที่มีการหมุนหรือเอียงศีรษะหรือมุมการมองของกล้อง แต่ตำแหน่งของหัวก็มีปัญหาถ้ามีการหมุน หรือเอียงหัว หรือขึ้นกับมุมมองของกล้องด้วย ข้อมูลโครงกระดูก (Skeleton) จากกล้องไคเนกซ์นั้นไม่เสถียรโดยเฉพาะอย่างยิ่งเมื่อเกิดการล้ม และโดยรูปร่างของโครงกระดูกจะไม่กลับคืนสู่รูปร่างเดิมหลังจากเกิดการล้ม

## 2.2 เซ็นเซอร์แบบสวมใส่ (Wearable Sensors)

วิธีพื้นฐานที่สุดคือส่งสัญญาณแจ้งเตือนโดยการกดปุ่มแจ้งเตือนเมื่อเกิดการล้มเกิดขึ้น ตัวอย่างเช่นในงานวิจัย [9] โดยอุปกรณ์จะติดอยู่กับตัวบุคคลที่มีแนวโน้มที่จะหกล้มและจะกดปุ่มแจ้งเตือน (Alarm) เมื่อเกิดการล้ม อย่างไรก็ตามเทคโนโลยีแบบนี้มีความเปราะบางมากเนื่องจากบุคคลอาจไม่ได้สวมใส่ขณะเกิดการล้ม หรือ อาจไม่สามารถกดปุ่มได้จากการหมดสติ แต่อย่างไรก็ตามเทคโนโลยีนี้มีการขึ้นอย่างแพร่หลายในหลายๆ ประเทศเนื่องจากติดตั้ง และบำรุงรักษา ง่าย อีกทั้งยังราคาไม่แพง

โดยทั่วไปอุปกรณ์แบบสวมใส่จะประมวลผลอัตโนมัติและต่อเนื่องจากการตรวจจับการเคลื่อนไหวโดยใช้เซ็นเซอร์วัดความเร่ง (Accelerometer) และ เซ็นเซอร์วัดความเร็วเชิงมุม (Gyroscope) [9] โดยเซ็นเซอร์ดังกล่าวสามารถตรวจจับการเคลื่อนไหวแบบทันทีทันใดของคนที่สวมใส่ได้ ในลักษณะเดียวกันบางงานวิจัย [10] ใช้ค่าจากเซ็นเซอร์วัดความเร่ง และ เซ็นเซอร์สนามแม่เหล็ก (Magnetic field) ของโทรศัพท์มือถือ เซ็นเซอร์วัดความเร่งและข้อมูลจากกล้องแบบติดกับตัว [11] และเซ็นเซอร์พลังงาน เช่น เครื่องกำเนิดไทรโบอิเล็กทริก (Trieboelectric Generator) [12]

แม้ว่าประสิทธิภาพในการตรวจจับการล้มโดยใช้อุปกรณ์แบบสวมใส่ได้นั้นค่อนข้างสูงเมื่อเกิดเหตุการณ์ที่อันตรายถึงชีวิต แต่เนื่องจากบุคคลนั้นต้องสวมอุปกรณ์อย่างต่อเนื่อง ถ้าคนที่ควรจะใช้เครื่องล้ม (เช่น ผู้สูงอายุเนื่องจากปัญหาความจำ) หรือละเลยความสำคัญของการสวมใส่ทำให้ไม่สามารถตรวจจับการล้มได้ทำให้วิธีการนี้ไร้ประโยชน์ ปัญหาอื่น ๆ ของอุปกรณ์พกพาคือความต้องการในการชาร์จหรือเปลี่ยนแบตเตอรี่เป็นระยะๆ เพื่อใช้งานอย่างต่อเนื่องหรือแม้กระทั่งถอดออกเมื่ออยู่ในห้องอาบน้ำเนื่องจากอุปกรณ์ไม่กันน้ำ โทรศัพท์มือถือมีเทคโนโลยีตรวจจับการตกอยู่แล้ว (เช่น เซ็นเซอร์วัดความเร่ง และ เซ็นเซอร์วัดความเร็วเชิงมุม มีอยู่ในโทรศัพท์ส่วนใหญ่) ในขณะที่สามารถขอความช่วยเหลือได้ไม่ว่าผู้ใช้จะอยู่ในอาคารหรือกลางแจ้ง อย่างไรก็ตาม ตำแหน่งของโทรศัพท์เป็นสิ่งสำคัญ เนื่องจากผู้ใช้อาจใส่โทรศัพท์ไว้ในกระเป๋าเสื้อแจ็กเก็ต กระเป๋าใกล้เอว หรือในกระเป๋าทิ้ง ข้อดีของอุปกรณ์สวมใส่คือเซ็นเซอร์เคลื่อนที่ไปพร้อมกับคน ดังนั้นจึงไม่จำเป็นต้องมีเซ็นเซอร์ในทุกห้อง นอกจากนี้ อุปกรณ์เหล่านี้ไม่ถูกรบกวนด้วยสิ่งกีดขวาง เหมือนกับการใช้กล้องตรวจจับ แต่ก็อาจมีปัญหา ด้านการสื่อสารไร้สายได้ถ้าอยู่ในตำแหน่งที่อับคลื่น

## 2.3 เซ็นเซอร์แบบติดตั้งไว้กับที่ (Fixed Location Sensors)

ระบบเซ็นเซอร์เสียง (Acoustic Sensors) และเซ็นเซอร์ตรวจจับสภาวะแวดล้อม (Ambient Sensors) ใช้ไมโครโฟนหรือเซ็นเซอร์สั่นสะเทือน ตรวจสอบความดังและความสูงของเสียงเพื่อจำแนกการล้ม [13] บางงานวิจัยใช้การตรวจจับการสั่นสะเทือนของพื้น [14] หรือใช้พีเจอาร์ที่ดึงมาจากสัญญาณเรดาร์ [15] งานวิจัย [16] ใช้ระบบไมโครโฟนอาร์เรย์ในการตรวจจับสัญญาณเสียงพบว่าตอนที่เกิดการล้มมีส่วนประกอบของเสียงความถี่สูงประมาณ 1,000 เฮิรตซ์ และมีการใช้เทคนิคการแยกประเภทเสียงเช่น ค่าสัมประสิทธิ์ซีพทริลความถี่เมล (Mel-frequency cepstral coefficients หรือ MFCCs) และอัลกอริทึมเพื่อนบ้านที่ใกล้ที่สุด (Nearest Neighbor) ใช้ในการจำแนกการล้มกับการไม่ล้มออกจากกัน ในงานวิจัย [17] อินฟราเรดเซ็นเซอร์ติดตั้งไว้บนเพดานโดยที่เซ็นเซอร์แต่ละตัวสร้างการตอบสนองแบบไบนารีเพื่อรับรู้ถึงตำแหน่งและการมีอยู่ของบุคคลด้านล่าง สัญญาณจากเซ็นเซอร์จะถูกสร้างเป็นภาพขนาด 5 x 4 พิกเซล โดยจะทำการวิเคราะห์ความแตกต่างของค่าพิกเซลจากเฟรมก่อนหน้าเพื่อตัดสินใจว่าเกิดการล้มหรือไม่

งานวิจัย [18] กล่าวถึงการใช้กระเบื้องอัจฉริยะที่มีเซ็นเซอร์วัดแรง (Force Sensors) และเซ็นเซอร์วัดความเร่งแบบ 3 แกน (3-axis Accelerometers) เซ็นเซอร์วัดแรงช่วยให้สามารถตรวจจับการล้มและรับรู้กิจกรรมอื่นๆ ของคนได้ เช่น การเดิน การยืน การนั่ง การนอน อย่างไรก็ตามท่านอนมีผลต่อความแม่นยำของการตรวจจับการล้มโดยทำให้เกิดผลบวกที่ผิดพลาด (False Positive) ปัญหานี้แก้ไขได้ด้วยการใช้ทั้งเซ็นเซอร์วัดแรงและเซ็นเซอร์วัดความเร่งร่วมกัน งานวิจัยอีกอัน [19] ใช้เซ็นเซอร์แบบไฟเบอร์ที่ไวต่อแรงกด (Pressure-sensitive Fibre Sensors) ฝังไว้ใต้พื้น โดยเน้นการใช้งานที่การตรวจจับการล้มในห้องน้ำ โดยไฟเบอร์เซ็นเซอร์มีราคาถูก ไม่เกะกะ และกันน้ำได้ ทำให้ติดตั้งในห้องน้ำได้ โดยงานวิจัยนี้จะตรวจจับการนอนราบกับพื้นเพราะปกติคนจะไม่นอนบนพื้นห้องน้ำเพื่อออกกำลังกายหรือหลับ

แม้ว่าเซ็นเซอร์ที่ติดตั้งอยู่กับที่อาจมีข้อดีคือไม่ต้องสวมอุปกรณ์ตลอดเวลา และการเฝ้าระวังสามารถทำได้ต่อเนื่องโดยไม่มีปัญหาเรื่องการล้มสวมใส่ และถูกบังโดยวัตถุอื่น แต่ระบบดังกล่าวจำกัดการใช้งานภายในอาคารเท่านั้น นอกจากนี้ อุปกรณ์ที่ใช้ราคาค่อนข้างแพง และความยุ่งยากในการปรับใช้และการบำรุงรักษายังเป็นปัจจัยที่ทำให้วิธีนี้จึงไม่เหมาะกับการใช้งานทั่วไป

## 2.4 การตรวจจับโดยใช้กล้อง (RGB Camera Systems)

ในหลายๆงานวิจัยใช้การวิเคราะห์ภาพจากกล้องหนึ่งตัว [20] หรือหลายตัว [21] แทนการใช้อุปกรณ์ที่สวมใส่กับตัวบุคคล และใช้ใช้อัลกอริทึมการมองเห็นด้วยคอมพิวเตอร์วิเคราะห์การเคลื่อนไหวของคนเพื่อตรวจจับการล้มอีกที กล้องถ่ายภาพความร้อนก็ถูกนำมาใช้เพื่อค้นหาและติดตามเป้าหมายความร้อนและวิเคราะห์การเคลื่อนไหวที่เพื่อตรวจจับพฤติกรรมเฉพาะเมื่อเกิดการล้ม จากนั้นจึงตรวจสอบการไม่ขยับตัวของเป้าหมาย [22] อีกวิธีหนึ่งในการตรวจจับการล้มคือการวิเคราะห์ความเร็วของผู้ล้มตามหลักชีวกลศาสตร์ (Biomechanics) [23] ในงานวิจัย [2] ตรวจจับการล้มโดยใช้การติดตามความเร็วศีรษะใน 3 มิติ วิธีการนี้ตรวจพบการล้มเพียงสองในสามเท่านั้นแต่สามารถแยกความแตกต่างระหว่างการล้มจริงกับเหตุการณ์ที่คล้ายคลึงกัน เช่น การนั่ง แนวทางการใช้การมองเห็นด้วยคอมพิวเตอร์อื่น ๆ มุ่งเน้นไปที่การวิเคราะห์ท่าทางเช่น ใน [24] การวิจัยนี้เน้นไปที่การล้มสามประเภท (ไปด้านหน้า ด้านหลัง และด้านข้าง) แม้ว่าจะสามารถแยกความแตกต่างระหว่างการล้มและการนอน/การนั่งได้ แต่ก็มีข้อจำกัดเนื่องจากภาพที่ใช้สำหรับการวิเคราะห์นั้นถ่ายมาจากมุมมองด้านข้างเท่านั้น

## 2.5 แบบผสมผสาน (Combinatory Systems)

บางงานวิจัยนำเอาฮาร์ดแวร์ และการมองเห็นของคอมพิวเตอร์มาใช้ในการตรวจจับการล้มร่วมกัน ในงานวิจัย [25] นำเสนอแพลตฟอร์มช่วยเหลือในการดำเนินชีวิต (Ambient Assisted Living) สำหรับผู้สูงอายุ หรือ คนพิการ กล่าวถึงอุปกรณ์สวมใส่ได้, สภาพแวดล้อม, การมองเห็นของคอมพิวเตอร์และหลายรูปแบบ งานวิจัย [26] ใช้ข้อมูลจากเซ็นเซอร์วัดความเร่งและวิดีโอความลึก (Depth Video) 2 ตัว งานวิจัย [27] ใช้เซ็นเซอร์เสียง (Acoustic) หรือ เซ็นเซอร์ความเคลื่อนไหว (PIR Sensors) ร่วมกับเทอร์โมไพลส์ (Thermopile) [28] และงานวิจัย [29] ใช้เซ็นเซอร์ความลึกกับ เซ็นเซอร์วัดความเร่งและเซ็นเซอร์เสียงร่วมกัน สุดท้ายนี้ ในงานวิจัย [30] ได้มีการเสนอการผสมผสานระหว่างกล้องและระบบตรวจจับการเต้นของหัวใจ จากการทดลองระบบดังกล่าวให้ผลลัพธ์ที่น่าเชื่อถือมากขึ้น แต่ระบบนี้จะมีความซับซ้อนสูงกว่า และยังต้องติดอุปกรณ์ไว้ตามร่างกายซึ่งอาจจะไม่สะดวกและก่อนให้เกิดการระคายเคืองได้ อย่างไรก็ตามวิธีการมองเห็นของคอมพิวเตอร์อาจยังคงมีความจำเป็นต้องเกี่ยวข้องกับฮาร์ดแวร์หรือแบบผสมผสานดังกล่าว

ในงานวิจัย [31] ใช้สัญญาณจากข้อมูลความดันที่พื้นและภาพอินฟราเรดจะถูกประมวลผล และเมื่อถึงเกณฑ์ที่ตั้งไว้จะถือว่าเกิดการล้ม วิธีการนี้ได้ความถูกต้องร้อยละ 90 ในชุดข้อมูล 120 ตัวอย่าง

งานวิจัย [32] ใช้ตัวกรองแบบเบย์เซียน (Bayesian Filtering) เพื่อหาท่าทางของบุคคล ว่ามีความน่าจะเป็นที่จะล้มหรือลุกขึ้นโดยใช้ข้อมูลภาพที่เกิดจากเซ็นเซอร์ที่พื้นรอบๆ บริเวณนั้น รูปร่าง ขนาด และขนาดของรูปแบบจะถูกรวบรวมสำหรับการแยกประเภทจากชุดของพีเจอร์ที่คำนวณจากชุดข้อมูลทั้งหมด ตัวกรองแบบเบย์เซียน (Bayesian Filtering) ถูกนำมาใช้ในการวิเคราะห์ท่าทางของคนแทนการแบ่งโดยใช้พีเจอร์โดยตรง ระบบมีปัญหาเกี่ยวกับผู้ทดลองที่ล้มแบบคุกเข่าลงเนื่องจากมีรูปแบบที่คล้ายกับคนที่ยืน โดยทดลองจาก 650 เหตุการณ์และผู้ทดลองสิบคนให้ความไว (Sensitivity) และความจำเพาะ (Specificity) ร้อยละ 91

## 2.6 อัลกอริทึม (Algorithms)

เอกสารทางวิชาการแบบทบทวนจำนวนหนึ่งมีการสรุปและพิจารณาเกี่ยวกับระบบตรวจจับการล้มที่ใช้การมองเห็นของเครื่องคอมพิวเตอร์เช่น [33] [34] [35] เอกสารทางวิชาการ [33] พิจารณาเกี่ยวกับวิธีการล่าสุดและจัดหมวดหมู่ในแง่ของการได้มาของข้อมูลภาพ (กล้อง RGB ตัวเดียว กล้อง RGB หลายตัว และเซ็นเซอร์ความลึก) ในขณะที่ [36] [37] กล่าวถึงความแตกต่างของเซ็นเซอร์วัดความเร่ง และ วิธีการใช้เซ็นเซอร์แบบสวมใส่อื่นๆ ปัจจัยอีกอย่างที่ใช้แยกความแตกต่างคือ อัลกอริทึมที่ใช้ว่าเป็นวิธีแบบเฉพาะกิจจากการสังเกต หรือ เป็นวิธีแบบจดจำรูปแบบ (Pattern Recognition) ที่ถูกฝึกสอน (Train) โดยใช้การเรียนรู้ของเครื่อง (Machine Learning) อัลกอริทึมส่วนใหญ่ที่กล่าวถึงในเอกสารทางวิชาการแบบทบทวนต่อไปนี้ใช้นั้นใช้แนวทางการเรียนรู้ด้วยเครื่องเพื่อแยกประเภทว่าเป็นเหตุการณ์ล้มหรือกิจกรรมในชีวิตประจำวันปกติ ข้อมูลต่อไปนี้เป็นการอภิปรายอย่างละเอียดเกี่ยวกับการใช้ตัวตรวจจับแบบ RGB, ความลึก และเซ็นเซอร์วัดความเร่ง

### 2.6.1 ใช้ข้อมูลภาพแบบ RGB

มีการใช้เทคนิคต่างๆ จำนวนมากสำหรับตรวจจับการล้มในงานวิจัยที่ใช้กล้องและเซ็นเซอร์อื่นๆ ในงานวิจัย [38] ใช้วิธี GMM (Gaussian Mixture Model) เพื่อจำแนกกิจกรรมต่างๆ ว่ามีการล้มหรือไม่ล้มโดยพิจารณาจากการเปลี่ยนแปลงรูปร่างของรูปร่างขณะที่เกิดการล้มตามด้วยการไม่ขยับตัว หลังจากเกิดการล้ม การแบ่งส่วนจะดำเนินการเพื่อแยกภาพเงาและจุดภายในภาพเงาจะถูกดึงออกมา



โดยใช้ตัวตรวจจับขอบ **Canny (Canny Edge Detector)** สำหรับการจับคู่รูปร่างมนุษย์สองรูปที่ต่อเนื่องกันโดยใช้บริบทของรูปร่าง ความต่างเฉลี่ยของการจับคู่ (**Mean Matching Cost**) และการวิเคราะห์ **Procrustes** ถูกนำมาใช้สำหรับการวิเคราะห์รูปร่าง ทั้งสองวิธีนี้สามารถวัดปริมาณการเปลี่ยนแปลงรูปร่างได้ การล้มนั้นมีลักษณะเฉพาะโดยจุดสูงสุดบนเส้นโค้งระยะ **Procrustes** หรือเส้นโค้งความต่างเฉลี่ยของการจับคู่ ตามด้วยการไม่มีการเคลื่อนไหวที่ชัดเจนของบุคคลหลังจากการล้มนั้น ตัวแยกประเภทแบบ **GMM** ถูกนำมาใช้เพื่อตรวจจับการล้มนั้น จากการวิเคราะห์ผลลัพธ์ที่ได้จากตัวแยกประเภทแบบ **GMM** การหาความต่างเฉลี่ยของการจับคู่และการวิเคราะห์ **Procrustes** สามารถลดความผิดพลาดลงเหลือร้อยละ **4.60** และ **3.80** ตามลำดับ

ในงานวิจัย [39] จำแนกเหตุการณ์ล้มนั้นโดยใช้กฎ (**Rule-Based**) ที่กำหนดโดยชุดของพีเจอร์จากตัวคนและกล่องขอบเขต เช่น อัตราส่วนภาพ การกระจายความลาดชันในแนวนอนและแนวตั้งของวัตถุในระนาบ **XY** และมุมการล้มนั้น วิธีการลบพื้นหลัง (**Background Subtraction**) แบบปรับได้และกล่องขอบเขตถูกใช้สำหรับการตรวจจับวัตถุที่กำลังเคลื่อนที่ แบบจำลองการตกที่อธิบายจะขึ้นอยู่กับวิธีการวิเคราะห์การแยกคุณลักษณะ การตรวจจับ และการแยกประเภท คุณลักษณะที่แยกออกมารวมถึงการไล่ระดับสีในแนวนอนและแนวตั้ง อัตราส่วนภาพ และมุมจุดศูนย์กลางวัตถุกับแกนแนวนอน ถ้าเมื่อมุมถึงค่าน้อยกว่า **45** องศาจะถือว่ามีอาการล้มนั้นเกิดขึ้น โดยอัลกอริทึมนี้รายงานความถูกต้องร้อยละ **100** โดยความเฉพาะเจาะจง (**Specificity**) และความไว (**Sensitivity**) ถูกประเมินจากชุดข้อมูล **40** วิดีโอ

งานวิจัย [40] ใช้ตัวจำแนกเกาส์เซียน (**Gaussian Classifier**) แบบหลายเฟรมใช้เพื่อหาทิศทางของร่างกายและตำแหน่งของหัวคนในรูปภาพ วิธีการนี้มุ่งเป้าไปที่เหตุการณ์ที่เกี่ยวข้องกับการหกล้มในสถานการณ์ที่บ้านที่ไม่มีใครเห็น โดยนำเสนอการออกแบบและการใช้งานระบบตรวจจับการล้มนั้นแบบเรียลไทม์ การออกแบบนี้เกี่ยวข้องกับการแยกของวัตถุ (**Foreground**) ออกจากพื้นหลังในสตรีมภาพที่ได้จากกล้องสองตัวที่ไม่ปรับเทียบ และตั้งฉากกัน ทิศทางของแกนหลักของร่างกายและอัตราส่วนของความแปรปรวนในทิศทาง **x** และ **y** คำนวณผ่านการวิเคราะห์องค์ประกอบหลัก (**PCA**) โมดูลติดตามส่วนศีรษะใช้สำหรับการตรวจจับของคนและเพิ่มความทนทานของระบบ ตำแหน่งศีรษะถูกประมาณว่าเป็นกลุ่มก้อน (**Blob**) โดยใช้แบบจำลองสีผิวแบบเกาส์เซียน (**Gaussian skin-colour model**) และติดตามโดยการค้นหากลุ่มก้อนสีผิวที่อยู่ใกล้เคียงตำแหน่งศีรษะ การจัดประเภทจะดำเนินการผ่านตัวจำแนกแบบเกาส์เซียนแบบหลายเฟรม ระบบนี้มีความแม่นยำร้อยละ **100** ในวิดีโอ

ที่ไม่มีการบดบัง (Occlude) แต่หลังจากการเพิ่มวิดีโอตัวอย่าง 4 วิดีโอที่มีการบดบังความแม่นยำจะลดลงเหลือแค่ร้อยละ 44

งานวิจัย [41] ใช้กฎเพื่อนบ้านที่ใกล้ที่สุด (Nearest-Neighbour) เพื่อแยกท่าทางโดยใช้อัตราส่วนและความแตกต่างของความสูงและความกว้างของกล่องขอบเขต ใช้ร่วมกับความแตกต่างของเวลาระหว่างเหตุการณ์เพื่อจำแนกการล้ม โดยระบบตรวจจับการล้มนี้ใช้รูปแบบสถิติและฮิสโทแกรมแนวตั้งของภาพเงาเพื่อลดผลกระทบจากการขยับแขนของคน วิธีนี้ใช้การจำแนก k-NN เพื่อจำแนกท่าทางโดยใช้ความแตกต่างและอัตราส่วนความกว้างและความสูงของกล่องสี่เหลี่ยมครอบภาพเงาของร่างกายคน ตัวจำแนกแบบ k-NN และความแตกต่างของเวลาถูกใช้เพื่อตรวจจับเหตุการณ์ล้ม โดยได้ความถูกต้องร้อยละ 84.44 จากตัวอย่าง 15 ตัวอย่าง

งานวิจัย [42] ใช้แบบจำลองแบบซ่อนของมาร์คอฟ (Hidden Markov Model) ซึ่งสามารถตรวจพบการหกล้มได้โดยการวิเคราะห์ท่าทางของบุคคลและการตรวจจับการเปลี่ยนแปลงท่าทางกะทันหัน (เช่น จากการยืนเป็นการนอน) การประมวลผลรวมถึงการจดจำเหตุการณ์และพฤติกรรมที่เป็นอันตราย เช่น การหกล้ม ผ่านการติดตามและการตรวจจับ โดยงานวิจัยนี้ใช้การประมวลผลภาพจากกล้องหลายตัวและมุมมองมีการซ้อนทับกันบางส่วนและเพื่อให้ได้ภาพขณะผ่านจากกล้องหนึ่งไปอีกกล้องหนึ่ง

งานวิจัย [43] ใช้ตรรกศาสตร์คลุมเครือ (Fuzzy Logic) เพื่อตัดสินใจว่าสถานะของแต่ละคนในแต่ละเฟรม (เช่น ตั้งตรง นอน) โดยใช้ว็อกเซล (Voxels) ที่ได้จากภาพเงาของคนถ่ายด้วยกล้องอินฟราเรด งานวิจัยนี้ใช้ระบบกล้องหลายตัวที่ใช้ภาพเงาเพื่อสร้างแบบจำลอง 3 มิติของคน วิธีการตรวจจับการล้มประกอบด้วยสองระดับ ระดับแรกสรุปจำนวนสถานะสำหรับวัตถุในแต่ละภาพ ระดับที่สองเกี่ยวข้องกับบทสรุปทางภาษาศาสตร์ของสถานะของวัตถุที่เรียกว่า Voxel Person งานวิจัยนี้ได้ผลความเฉพาะเจาะจงร้อยละ 93.75 และความไวร้อยละ 100

## 2.6.2 ใช้ข้อมูลความลึก (Depth)

ในงานวิจัย [7] ตรวจจับการล้มโดยใช้การติดตามโครงกระดูก อย่างไรก็ตาม โครงกระดูกทำงานได้อย่างถูกต้องเมื่อวางเซ็นเซอร์ไว้ที่ช่วงและตำแหน่งเฉพาะที่เหมาะสม วิธีการนี้มีจุดมุ่งหมายเพื่อตรวจจับการล้มโดยการดึงข้อมูลโครงกระดูกจากภาพความลึกของกล้อง Kinect ตามอัลกอริทึมตัดสินใจแบบสุ่มอย่างรวดเร็ว (Fast Randomized Decision Forest) อัลกอริทึมนี้สร้างการตรวจจับที่แม่นยำยิ่งขึ้นโดยการหมุนเฟรมอย่างเหมาะสมเพื่อให้ตรงกับกับแนวร่างกายคน



(Human Orientation) วิธีการนี้ได้ความถูกต้องร้อยละ 100 ในชุดข้อมูลขนาดเล็กที่มีตัวอย่างการล้ม 20 ตัวอย่าง

ในงานวิจัย [44] ใช้เรขาคณิตรีมันเนียน (Riemannian Manifolds) ของสถิติความเร็วของการล้ม และการรวมกันของข้อมูล RGB และโครงกระดูกตามลำดับ งานวิจัยทั้งสองใช้ชุดข้อมูลที่เปิดเผยต่อสาธารณะ (public datasets) และผลที่ได้เกือบสมบูรณ์แบบในแง่ของความแม่นยำและอัตราผลบวกที่ผิดพลาด (False Positive) อย่างไรก็ตาม กระบวนการประเมินผลในการศึกษาทั้งหมดข้างต้นไม่ได้พิจารณาถึงลักษณะทางกายภาพของแต่ละบุคคล หรือโดนบดบังจากวัตถุอื่นๆ

### 2.6.3 ใช้ข้อมูลเซ็นเซอร์ความเร่ง (accelerometer)

ในงานวิจัยดังกล่าวจะวางอุปกรณ์ที่มีเซ็นเซอร์ความเร่งไว้บนหรือใกล้เอว ตำแหน่งใกล้กับจุดศูนย์กลางของร่างกาย หรือในตำแหน่งอื่น งานวิจัย [45] ใช้หลายๆ ชุดข้อมูลจากเซ็นเซอร์วัดความเร่งที่บันทึกผ่านโทรศัพท์มือถือซึ่งใส่ไว้ในกระเป๋าเสื้อหรือกระเป๋าคือของผู้ร่วมทดลองการล้ม และสถานการณ์อื่นในชีวิตประจำวัน (ADL) เอกสารทางวิชาการประเภททบทวน [46] กล่าวถึงแนวทางต่างๆ ที่ใช้ สามารถจัดกลุ่มตามเกณฑ์ (ในช่วงก่อนตก การกระแทก หลังการตก ความเร็วขนาดความเร่งและการเปลี่ยนแปลงของสัญญาณ ความเร็วเชิงมุม ความเอียงวิกฤตในระยะก่อนตก) และการเรียนรู้ของเครื่อง (One-class SVM, KFD, k-NN)

งานวิจัย [47] ใช้อัลกอริทึมเพื่อนบ้านที่ใกล้ที่สุด (K-nearest Neighbors) เพื่อตรวจจับการล้มโดยเซ็นเซอร์วัดความเร่งแบบ 3 แกนที่ฝังอยู่ในโทรศัพท์มือถือ ข้อมูลถูกคัดแยกก่อนโดยใช้ซัพพอร์ตเวกเตอร์แมชชีนแบบประเภทเดียว (One-class SVM) หลังจากนั้นถูกจำแนกโดยใช้อัลกอริทึม k-Nearest Neighbor (k-NN) และการวิเคราะห์ Kernel Fisher discriminant (KFD) งานวิจัยนี้ทดสอบกับกิจกรรมประจำวันทั่วไป (เช่น เดิน เดินลงบันไดด้วยความเร็วปกติ) และการล้มประเภทต่างๆ รวมถึงการทำกิจกรรมประจำวันที่มีความเข้มข้นสูง (เช่น วิ่ง กระโดด และยิมนาสติก) โดยผลการทดลองความเฉพาะเจาะจงร้อยละ 97.5 และความไวร้อยละ 84.4

งานวิจัย [48] นำเสนอระบบตรวจจับการล้มที่บ้านแบบเรียลไทม์ซึ่งไม่เพียงแต่สามารถแยกแยะเหตุการณ์การล้มได้ถึง 4 ประเภทที่แตกต่างกัน (ไปข้างหน้า ข้างหลัง ขวา และซ้าย) แต่ยังสามารถพาสะดวก ราคาประหยัด และมีความแม่นยำสูง ระบบประกอบด้วยแถบตรวจจับการล้มแบบเรียลไทม์ เครื่องเซิร์ฟเวอร์ และฟังก์ชันส่งข้อความแบบทันทีของผ่านเครือข่ายจีเอสเอ็ม (GSM) ซึ่ง

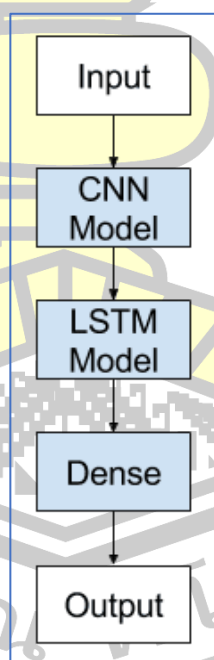
สามารถซื้อความแข็งแรงเดือนเมื่อเกิดการล้ม ส่งข้อความช่วยเหลือฉุกเฉิน โดยทดลองจากอาสาสมัครชายสี่คนแสดงเหตุการณ์ล้ม 120 ครั้งและได้ความแม่นยำที่ร้อยละ 95.83

## 2.7 ทฤษฎีที่เกี่ยวข้อง

### 2.7.1 CNN-LSTM

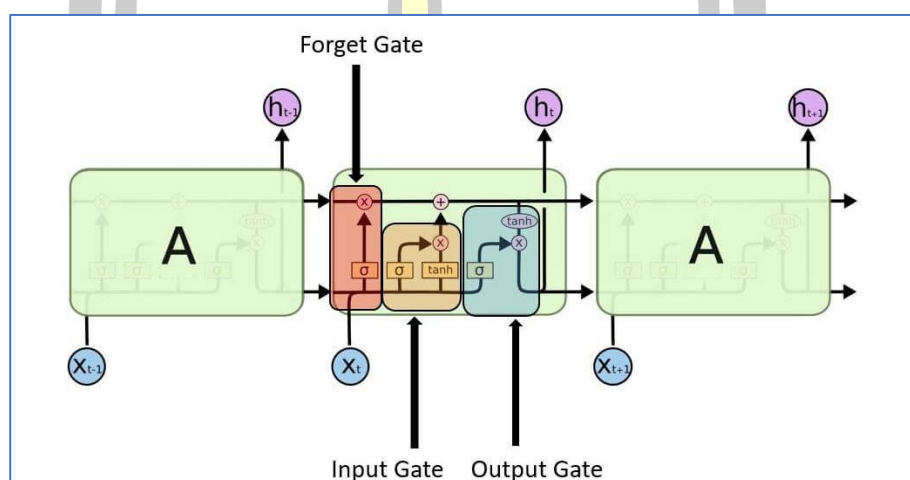
ประกอบด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network หรือ CNN) ในการสกัดฟีเจอร์ และ LSTM (Long Short-Term Memory) ซึ่งเป็นโครงข่ายประสาทเทียมที่เกิดซ้ำ (Recurrent Neural Network หรือ RNN) เพื่อแปลความหมายของฟีเจอร์ตามลำดับเวลา (Time Series) เพื่อรู้จำกิจกรรมต่างๆ ของคนเช่น การล้ม ตามรูปที่ 1

ซึ่ง CNN-LSTMs ได้รับการพัฒนาสำหรับปัญหาการทำนายอนุกรมเวลาด้วยภาพและการประยุกต์ใช้การสร้างคำอธิบายที่เป็นข้อความจากลำดับของภาพ (เช่น วิดีโอ) โดยเฉพาะปัญหาของการรับรู้กิจกรรม (Activity Recognition) คำอธิบายรูปภาพ (Image Description) และคำอธิบายวิดีโอ (Video Description)



ภาพที่ 1 โครงสร้างของระบบตรวจจับการล้มโดยใช้ CNN-LSTM

LSTM ประกอบด้วยโครงข่ายประสาทเทียมสี่ตัวและบล็อกหน่วยความจำจำนวนมากที่เรียกว่าเซลล์ในโครงสร้างแบบลูกโซ่ หน่วย LSTM ทั่วไปประกอบด้วยเซลล์ ประตูทางเข้า (Input Gate) ประตูทางออก (Output Gate) และประตูลืม (Forget Gate) การไหลของข้อมูลเข้าและออกจากเซลล์ถูกควบคุมโดยสามประตู และเซลล์จะจดจำค่าในช่วงเวลาที่กำหนดอัลกอริทึม LSTM ได้รับการดัดแปลงอย่างดีเพื่อจัดหมวดหมู่ วิเคราะห์ และคาดการณ์อนุกรมเวลาของระยะเวลาที่ไม่แน่นอน



ภาพที่ 2 โครงสร้างของ LSTM

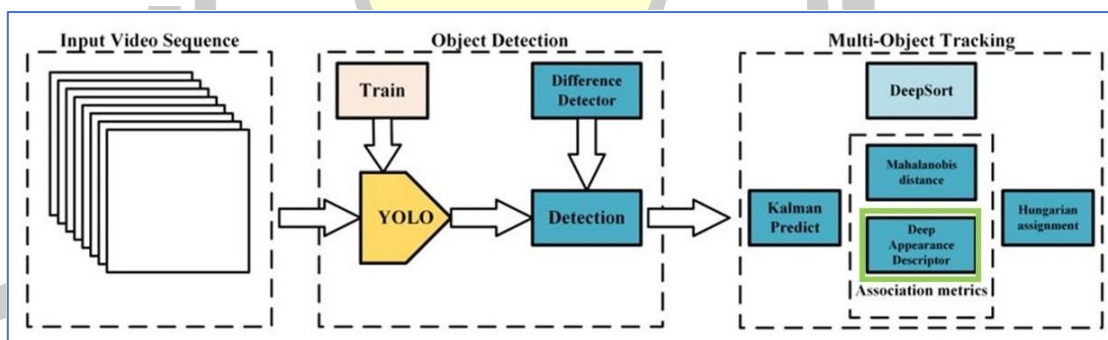
เซลล์ทำหน้าที่เก็บข้อมูลในขณะที่ประตูจัดการหน่วยความจำ มีทางเข้าสามทางได้แก่:

- ประตูเข้า (Input Gate): กำหนดว่าควรใช้ค่าอินพุตใดในการเปลี่ยนหน่วยความจำ ฟังก์ชันซิกมอยด์ (Sigmoid) กำหนดว่าจะยอมให้ค่า 0 หรือ 1 ผ่านหรือไม่ และฟังก์ชัน  $\tanh$  จะกำหนดน้ำหนักให้กับข้อมูลที่ให้ โดยกำหนดความสำคัญในระดับ -1 ถึง 1
- ประตูลืม (Forget Gate): จะทำหน้าที่หาข้อมูลที่ควรลบออกจากบล็อก ถูกกำหนดโดยฟังก์ชันซิกมอยด์ (Sigmoid) สำหรับแต่ละตัวเลขในสถานะเซลล์ ( $c_{t-1}$ ) จะดูที่สถานะก่อนหน้า ( $h_{t-1}$ ) และข้อมูลอินพุต ( $x_t$ ) และสร้างตัวเลขระหว่าง 0 (ลืม) ถึง 1 (เก็บ)
- ประตูทางออก (Output Gate): อินพุตและหน่วยความจำของบล็อกใช้เพื่อกำหนดเอาต์พุต ฟังก์ชันซิกมอยด์จะกำหนดว่าจะยอมให้ค่า 0 หรือ 1 ผ่านหรือไม่ และฟังก์ชัน  $\tanh$  จะกำหนดว่าค่าใดที่อนุญาตให้ส่งผ่าน 0, 1 และฟังก์ชัน  $\tanh$  จะกำหนดน้ำหนักให้กับค่าที่จัดเตรียมไว้ โดยกำหนดความเกี่ยวข้องของค่าต่างๆ ในระดับ -1 ถึง 1 และคูณด้วยเอาต์พุตจากฟังก์ชันซิกมอยด์อีกที

## 2.7.2 DeepSORT

ถึงแม้ว่าปัจจุบันตัวตรวจจับวัตถุที่ถูกพัฒนาด้วย **Deep Learning** ต่างๆ เช่น **YOLO**, **MobileNet**, **Mask R-CNN** และอื่นๆ จะถูกพัฒนา และปรับปรุงประสิทธิภาพสูงขึ้นอย่างต่อเนื่อง อีกทั้งยังสามารถตรวจจับวัตถุได้หลายชนิดมากขึ้นเรื่อยๆ แต่การอาศัยข้อมูลการตรวจจับอย่างเดียวไม่สามารถรับประกันได้ว่าจะสามารถติดตามวัตถุนั้นได้ตลอดเพราะบางทีวัตถุเองอาจมีการบดบังจากวัตถุอื่นอีกที อีกทั้งข้อมูลจากตัวตรวจจับในแต่ละเฟรมก็ขาดข้อมูลความสัมพันธ์ของพีเจอร์รี่ในแต่ละเฟรมที่ติดกัน ทำให้ตัวติดตามจะยังสามารถติดตามวัตถุได้แม้ตัวตรวจจับจะล้มเหลวในบางเฟรม (หรือหลายเฟรมติดต่อกัน) แต่อย่างไรก็ตามการตรวจจับที่ดีมีผลทำให้ได้การติดตามที่ดีตามมา

**DeepSORT** คืออัลกอริทึมติดตามวัตถุซึ่งได้ทำการปรับปรุงมาจากอัลกอริทึม **SORT (Simple Online Realtime Tracking)** โดยเพิ่มตัวบอกลักษณะ (**Appearance Descriptor**) ซึ่งเป็นการเรียนรู้เชิงลึกเข้าไปเพื่อลดการเปลี่ยน (**Switch**) การระบุตัวตน (**ID**) ของวัตถุ ซึ่งทำให้การติดตามมีประสิทธิภาพมากขึ้น นอกจากนี้อัลกอริทึม **SORT** ยังใช้ตัวกรองคาลมานในการทำนายสถานะของวัตถุ และอัลกอริทึมฮังการีเรียนในการกำหนดความเชื่อมโยงของวัตถุในแต่ละเฟรม (**Assignment**)



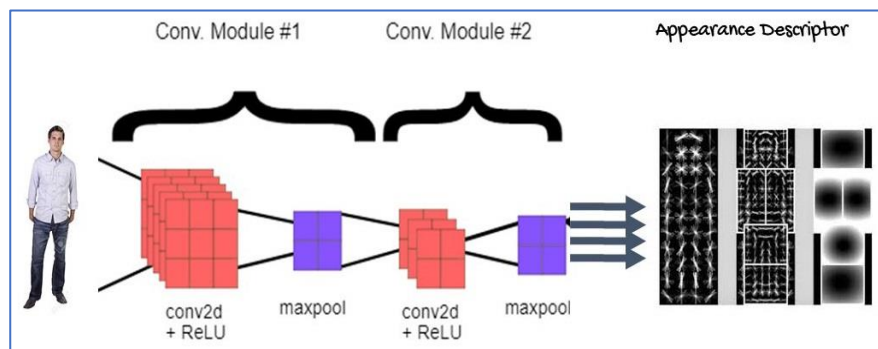
ภาพที่ 3 โครงสร้างของ DeepSORT

โดยโครงสร้าง DeepSORT เป็นดังรูปที่ 3 ประกอบด้วยองค์ประกอบสำคัญ 4 ส่วน ได้แก่

- การตรวจจับวัตถุ (**Detection**): ในขั้นตอนนี้ตัวตรวจจับวัตถุจะตรวจจับวัตถุในเฟรมแล้วส่งต่อไปยังขั้นตอนต่อไป ตัวตรวจจับที่นิยมเช่น **Faster R-CNN**, **YOLO** และอื่นๆ
- การประมาณค่า (**Estimation**): ในขั้นตอนนี้ใช้ตัวกรองคาลมานเพื่อทำนายสถานะวัตถุเฟรมถัดไปจากเฟรมปัจจุบัน โดยใช้แบบจำลองความเร็วคงที่ เมื่อตรวจจับวัตถุเฟรมต่อไป

และถูกกำหนดว่าเป็นวัตถุเดียวกัน จะมีการอัปเดตพารามิเตอร์ของตัวกรองกาลมานเพื่อให้สามารถติดตามวัตถุได้ดีขึ้น

- การเชื่อมโยงข้อมูล (Data association): เมทริกซ์ความสัมพันธ์ระหว่างการทำนายเป้าหมายทั้งหมดที่มีอยู่ และจากการตรวจจับจากเฟรมปัจจุบัน (Association Metrics) ซึ่งระยะห่างคำนวณโดยใช้ระยะทางมหาลาโนบี (Mahalanobis distance) และ ระยะห่างทางลักษณะภายนอก (Appearance distance) โดยใช้ตัวบอกลักษณะ (Appearance Descriptor) ดังรูปที่ 4 หลังจากนั้นอัลกอริทึมยังกาเรียนจะการกำหนดว่าวัตถุจากการตรวจจับตรงกับเป้าหมายจากการทำนายใด ( $IoU > IoU_{min}$ ) เทคนิคนี้แก้ปัญหาการโดนบังจากวัตถุอื่นและช่วยรักษา ID ไว้เมื่อวัตถุกลับมาตรวจเจออีกครั้ง



ภาพที่ 4 Deep Appearance Descriptor

โดย Association Metrics  $D$  หาได้จากสมการ (1)

$$D = \lambda D_k + (1 - \lambda) D_a \quad (1)$$

โดยที่  $D_k$  คือ ระยะทางมหาลาโนบี (Mahalanobis distance)

$D_a$  คือ ระยะทางมหาลาโนบี (Mahalanobis distance) ระยะแบบโคไซน์ (cosine) ระหว่างตัวบอกลักษณะ (Appearance Descriptor)

$\lambda$  คือ คิวค่าถ่วงน้ำหนัก (Weighting factor)

- การสร้างและการลบข้อมูลเฉพาะตัวของแทรีก: เมื่อวัตถุที่ตรวจจับใหม่มีคะแนนต่ำกว่าเกณฑ์ที่กำหนด จะไม่มีการกำหนดให้กับแทรีกที่มีอยู่แล้วแต่จะจัดประเภทเป็นออบเจ็กต์ใหม่ แทรีกจะถูกลบออกเมื่อไม่ได้กำหนดการตรวจจับสำหรับเฟรมจำนวนหนึ่ง (age/frame)

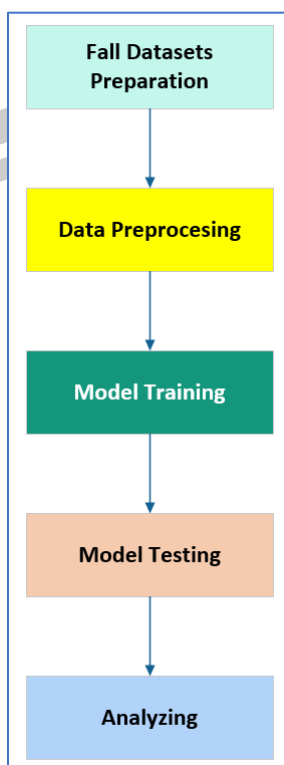


### บทที่ 3

#### วิธีดำเนินการวิจัย

จากการทบทวนเอกสารงานวิจัยเกี่ยวกับการตรวจจับการล้มถึงแม้การตรวจจับการล้มด้วยวิธีใช้เซ็นเซอร์ที่ติดไว้กับตัวผู้ที่เสี่ยงต่อการล้มเช่นผู้สูงอายุ หรือผู้ป่วยที่มีโรคประจำตัว เช่น โรคหัวใจ โรคเกี่ยวกับสมอง จะได้ผลความถูกต้องที่สูง แต่ก็มีจุดอ่อนคือทำให้เกิดการระคายเคือง หรือทำให้เกิดแผลได้ ในผู้สูงอายุที่มีภาวะหลงลืมอาจจะลืมใส่ และยังมีข้อจำกัดเรื่องระยะเวลาการทำงานของอุปกรณ์เนื่องจากแบตเตอรี่ที่ต้องมีการชาร์จสม่ำเสมอทำให้ไม่สะดวกในการใช้งาน ส่วนระบบตรวจจับที่ใช้เซ็นเซอร์ที่ติดตั้งอยู่กับที่ เช่น ตรวจจับเสียง ตรวจจับการสั่นที่พื้น หรือสัญญาณเรดาร์ก็มีความยุ่งยาก ในการติดตั้งและอุปกรณ์มีราคาแพง งานวิจัยนี้ผู้จัดทำจึงเลือกใช้วิธีการมองเห็นของคอมพิวเตอร์ (Computer Vision) โดยใช้กล้อง RGB ที่ติดตั้งอยู่กับที่เพราะในปัจจุบันกล้องวงจรปิด (IP Camera) และเทคโนโลยีเครือข่ายมีความเร็วเพิ่มขึ้นมาก ทำให้การติดตั้งกล้องวงจรปิดหลายๆ ตัวแล้วนำภาพมาประมวลผลที่เดียวผ่านเครือข่ายสามารถทำได้ง่าย และในปัจจุบันการติดกล้องวงจรปิดตามบ้านพักอาศัยได้รับความนิยมเพิ่มขึ้นเรื่อยๆ ดังนั้นถ้าจะนำระบบตรวจจับการล้มโดยใช้กล้องไปประยุกต์ใช้จริงในทางปฏิบัติก็จะสามารถทำได้ง่าย โดยในงานวิจัยนี้จะใช้วิธีตรวจจับการล้มจากการวิเคราะห์ท่ามุมของร่างกายส่วนบน(Upper Body Angle) ความเร็วเชิงเส้น (Linear Velocity) ของจุดที่ลำคอ และความเร็วเชิงมุม (Angular Velocity) ของร่างกายท่อนบนของคนโดยคำนวณมาจาก 4 จุดสำคัญ (Landmark) จากข้อมูลโครงกระดูก (Skeleton) ที่สกัดโดยใช้ไลบรารีการเรียนรู้เชิงลึก BlazePose GHUM แล้วจัดข้อมูลให้อยู่ในรูปแบบอนุกรมเวลา (Time Series) เพื่อทำการสอนแบบจำลอง CNN-LSTM ให้สามารถตรวจจับการล้มได้ ซึ่ง CNN-LSTM เป็นการนำโครงข่ายประสาทเทียมคอนโวลูชัน (CNN) ร่วมกับหน่วยความจำระยะสั้น-ยาว (Long Short – Term Memory หรือ LSTM) ซึ่งผู้วิจัยได้ทำการแบ่งวิธีดำเนินการวิจัยออกเป็น 5 ขั้นตอนตามรูปที่ 5 ดังนี้

1. การจัดเตรียมข้อมูลวิดีโอ (Datasets)
2. เตรียมข้อมูล (Data Pre-Processing) จาก Train Videos
3. การสอนแบบจำลองโดยใช้ข้อมูล Train Datasets (ร้อยละ 70) และวัดผลแบบจำลองที่ได้กับข้อมูล Test Datasets (ร้อยละ 30)
4. ทดสอบแบบจำลองพร้อมติดตามวัตถุกับ Test Videos
5. วิเคราะห์ผลการทดสอบ



ภาพที่ 5 ขั้นตอนการทำวิจัย

### 3.1 ชุดข้อมูลวิดีโอ (Datasets)

งานวิจัยนี้ใช้ชุดข้อมูลสาธารณะ (Dataset) 3 ชุด ดังต่อไปนี้

#### 3.1.1 ImVia Fall Detection Dataset [4]

มีทั้งหมด 191 วิดีโอ ซึ่งมีทั้งกิจกรรมปกติ และการล้ม โดยวิดีโอถูกบันทึกจากสถานที่ที่แตกต่างกัน เช่น บ้าน ห้องกาแฟ ออฟฟิศ และห้องเรียน โดยวิดีโอทั้งหมดจะถูกบันทึกที่อัตราความเร็ววิดีโอที่ 25 เฟรมต่อวินาที และมีความละเอียดที่ 320×240 พิกเซล ดังแสดงในรูปที่ 6

#### 3.1.2 UR Fall Detection Dataset [26]

ชุดข้อมูลนี้มีทั้งหมด 70 วิดีโอ โดยมี 30 วิดีโอ สำหรับการล้ม และ 40 วิดีโอ ที่เป็นกิจกรรมทั่วไป การล้มถูกบันทึกด้วยกล้องไคนะ 2 ตัว แต่กิจกรรมทั่วไปจะบันทึกจากกล้องแค่ 1 ตัว ดังแสดงในรูปที่ 7



### 3.1.3 FALLALLD [5]

ชุดข้อมูลนี้มี 7 วิดีโอที่มีการล้ม และกิจกรรมทั่วไปในชีวิตประจำวัน ดังแสดงในรูปที่ 8



ภาพที่ 6 ตัวอย่างวิดีโอจากชุดข้อมูล ImVia Fall Detection Dataset [4]

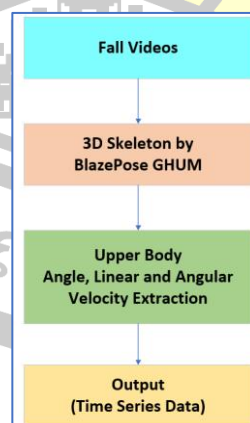


ภาพที่ 7 ตัวอย่างวิดีโอจากชุดข้อมูล UR Fall Detection Dataset [51]



ภาพที่ 8 ตัวอย่างวิดีโอจากชุดข้อมูล FALLALLD [5]

วิดีโอจาก 3 ชุดข้อมูลวิดีโอจะถูกปรับเฟรมเรทให้เท่ากันที่ 25 เฟรม/วินาที ก่อนเพื่อให้ได้ข้อมูลที่ได้เป็นอนุกรมเวลา (Time Series) ที่มีระยะห่างของแต่ละเฟรมเท่ากัน หลังจากนั้นวิดีโอจะถูกแปลงเป็นรูปภาพทั้งหมดขนาดให้ใกล้เคียงกัน คือกว้าง 320 และ สูง 240 พิกเซลโดยประมาณ เพื่อให้ความเร็วเชิงเส้นที่คำนวณได้มีขนาดใกล้เคียงกันในแต่ละวิดีโอ จากนั้นผู้วิจัยจะทำการระบุท่าทางของคนว่าล้มหรือไม่ล้ม (Data Labelling) โดยแบ่งการล้มออกเป็น 3 ระยะตามตารางที่ 1 ได้แก่ ระยะที่ 1 ก่อนล้มได้แก่ การยืน เดิน นั่ง และ อื่นๆ ระยะที่ 2 กำลังล้มเกิดขึ้นจากการหมดสติล้มโดยบังเอิญ สะดุดขณะยืน หรือเดิน จะมีความเร็วจากการเปลี่ยนจากทำยืนเป็นทำนอนเพิ่มขึ้น และ ระยะที่ 3 ระยะนอนนิ่ง หรือ อาจมีการขยับตัวเล็กน้อย จากนั้นผู้วิจัยจะแบ่งวิดีโอออกเป็นร้อยละ 70 สำหรับการสอน และร้อยละ 30 สำหรับทดสอบแบบจำลองต่อไป






ภาพที่ 9 ขั้นตอนการจัดเตรียมข้อมูล

### 3.2 การจัดเตรียมข้อมูล (Data Pre-processing)

การจัดเตรียมข้อมูลสำหรับงานวิจัยจะมี 2 ขั้นตอนหลัก ดังนี้

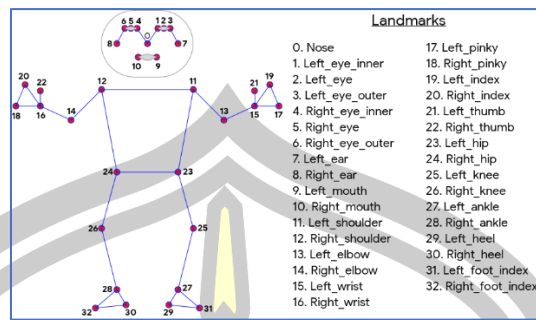
- 1) การสกัดข้อมูลโครงกระดูก 3 มิติโดยใช้ BlazePose GNUM
- 2) การคำนวณหาความเร็วเชิงเส้น (Linear Velocity) และ ความเร็วเชิงมุม (Angular Velocity) เพื่อเป็นข้อมูลแบบอนุกรมเวลาให้แบบจำลอง CNN-LSTM ต่อไป

ตารางที่ 1 ตัวอย่างการกิจกรรมต่างๆ

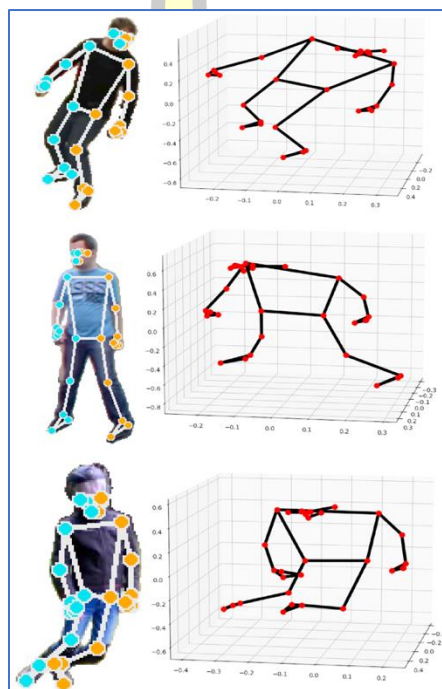
กิจกรรม	ตัวอย่าง	หมายเลขกิจกรรม
ไม่ล้ม (Not Fall)		0
กำลังล้ม (Falling)		1
นอนนิ่ง (Lay Down)		2

#### 3.2.1 โครงกระดูก 3 มิติ (3D Skeleton)

รูปภาพจะถูกตรวจจับคน และสกัดเอาข้อมูลโครงกระดูก 3 มิติ โดยใช้ BlazePose GHUM ของ MediaPipe เป็นระบบโอเพ่นซอร์สจาก Google สำหรับสร้างโซลูชัน ML ข้ามแพลตฟอร์มที่ปรับแต่งได้สำหรับไลพ์สัดและแบบสตรีมมิงโดยเฉพาะ ซึ่งมีจุดเด่นเรื่องความเร็ว ซึ่งข้อมูลที่ได้จะเป็นข้อมูล 3 มิติ (x, y, z) จำนวน 33 ชุดตามตำแหน่งร่างกาย 33 จุดตามรูปที่ 10

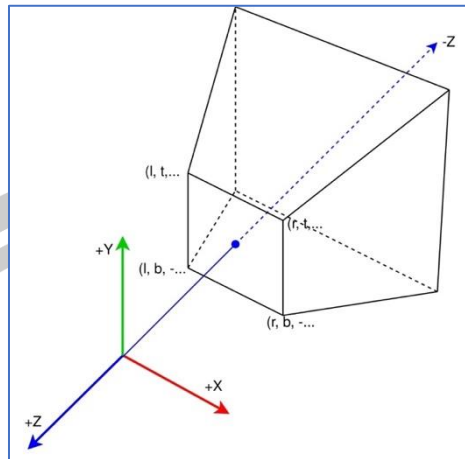


ภาพที่ 10 ตำแหน่ง 33 จุดสำคัญที่ได้จากโมเดล BlazePose GHUM



ภาพที่ 11 ตัวอย่าง 3D Skeleton ที่ได้จาก BlazePose GNUM

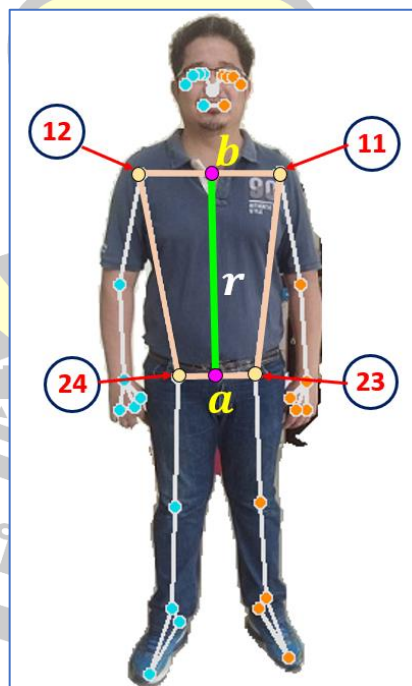
โดยข้อมูล 3 มิติ  $x$ ,  $y$  และ  $z$  ของแต่ละจุดจะใช้พิกัดอ้างอิงแบบพิกัดกล้อง (Camera Coordinate) ตามรูปที่ 12 โดยมีจุด origin อยู่ตรงกลางรูป มีแกน  $+X$  จะชี้ไปทางขวาของรูปภาพ แกน  $+Y$  ชี้ขึ้นด้านบนของรูป และ แกน  $+Z$  ชี้มาด้านหน้า(เข้าหากล้อง) โดยค่า  $x$ ,  $y$  และ  $z$  จะถูกทำให้อยู่ในช่วง  $[-1, +1]$



ภาพที่ 12 พิกัดกล้อง (Camera Coordinate)

3.2.2 การคำนวณมุม (Body Angle) ความเร็วเชิงเส้น (Linear Velocity) และ ความเร็วเชิงมุม (Angular Velocity) ของร่างกาย

จากข้อมูล 3D Skeleton ผู้วิจัยเลือก 4 จุดจาก 33 จุดเพื่อเป็นตัวแทนลำตัวของคน ได้แก่ จุดที่ 12 (ไหล่ขวา) จุดที่ 11 (ไหล่ซ้าย) จุดที่ 24 (สะโพกขวา) และ จุดที่ 23 (สะโพกซ้าย) ดังรูปที่ 13



ภาพที่ 13 จุดแทนร่างกายท่อนบน(ลำตัว) 4 จุด



ผู้วิจัยจึงใช้เส้นตรง  $r$  จากจุด  $a$  (กึ่งกลางระหว่างจุด 23 และ จุด 24) ไปจุด  $b$  (กึ่งกลางระหว่างจุด 11 และ จุด 12) เป็นเส้นตรงแทนร่างกายท่อนบน หรือ ลำตัว โดยตำแหน่งของ  $a$  และ  $b$  หาได้จากสมการ (2) และ (3) ตามลำดับ

$$(x, y, z)_a = \left( \frac{x_{23} + x_{24}}{2}, \frac{y_{23} + y_{24}}{2}, \frac{z_{23} + z_{24}}{2} \right) \quad (2)$$

$$(x, y, z)_b = \left( \frac{x_{11} + x_{12}}{2}, \frac{y_{11} + y_{12}}{2}, \frac{z_{11} + z_{12}}{2} \right) \quad (3)$$

การคำนวณหาค่ามุมของร่างกายท่อนบน (Body Angle)  $\emptyset$   $\theta$  และ  $\varphi$  ที่ทำกับแกน  $x$ ,  $y$  และ  $z$  ดังรูปที่ 14 ค่าความเร็วเชิงเส้น (Linear Velocity) ของจุดกลางหน้าอก (จุด  $b$ ) ของเฟรมปัจจุบันเทียบกับเฟรมก่อนหน้า และ ความเร็วเชิงมุมใน (Angular Velocity) ของมุมของร่างกายท่อนบน (Body Angle) เทียบกับเฟรมก่อนหน้า คำนวณได้จากสมการที่ (4) ถึง (13) ตามลำดับ

$$\vec{r} = (x_b - x_a, y_b - y_a, z_b - z_a) \quad (4)$$

$$\emptyset = \text{atan} \left( \frac{z_r}{x_r} \right) \quad (5)$$

$$\theta = \text{acos} \left( \frac{y_r}{|\vec{r}|} \right) \quad (6)$$

$$\varphi = \text{atan} \left( \frac{x_r}{z_r} \right) \quad (7)$$

$$\dot{v}_{bx} = x_{bk} - x_{bk-1} \quad (8)$$

$$\dot{v}_{by} = y_{bk} - y_{bk-1} \quad (9)$$

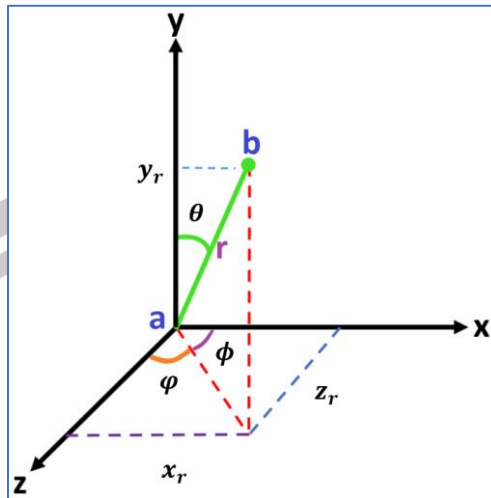
$$\dot{v}_{bz} = z_{bk} - z_{bk-1} \quad (10)$$

$$\dot{\emptyset} = \emptyset_k - \emptyset_{k-1} \quad (11)$$

$$\dot{\theta} = \theta_k - \theta_{k-1} \quad (12)$$

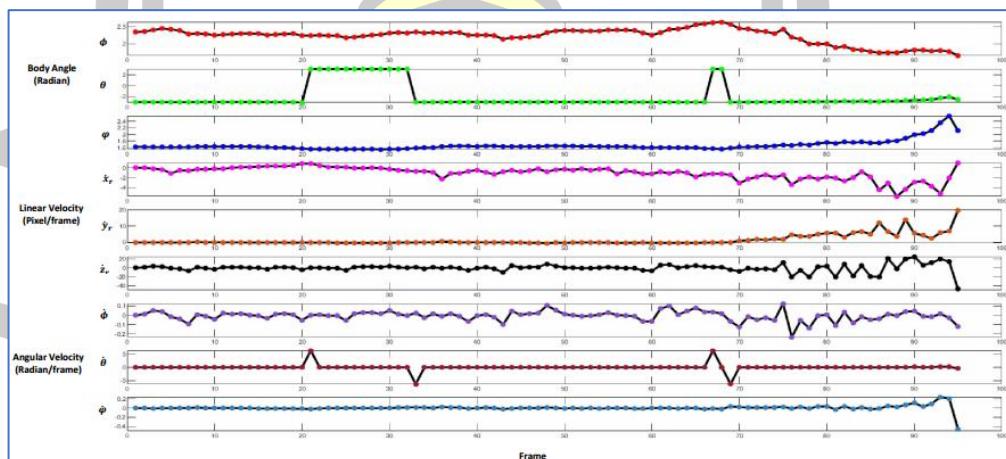
$$\dot{\varphi} = \varphi_k - \varphi_{k-1} \quad (13)$$

พหุปัญญา ทิโต ชีเว



ภาพที่ 14 เส้นตรง  $r$  และ มุมแกน  $\phi$ ,  $\theta$  และ  $\varphi$

ข้อมูลที่ได้จากแต่ละวิดีโอจะถูกเก็บไว้ในรูปแบบอนุกรมเวลา 9 แถวตั้งรูปที่ 15 ประกอบไปด้วยข้อมูลค่ามุมของร่างกายท่อนบนที่ทำกับแกน  $x$ ,  $y$  และ  $z$  ความเร็วเชิงเส้นในแนวแกน  $x$ ,  $y$  และ  $z$  ของจุดที่ลำคอ และความเร็วเชิงมุมรอบแกน  $x$ ,  $y$  และ  $z$  ตามลำดับตามรูปที่ 13 หลังเก็บข้อมูลจนครบทุกเฟรมข้อมูลจะถูกแบ่งออกเป็นหน้าต่าง (window) ขนาด 24 เฟรม (1 วินาที) และเลื่อน (overlap) ไปทุกๆ 12 เฟรม (0.5 วินาที) จากนั้นนำมารวมกับวิดีโออื่นทั้งหมด ทำการสุ่มสลับตำแหน่ง (shuffle) แล้วแบ่งเป็น ร้อยละ 70 สำหรับการสอนแบบจำลอง และร้อยละ 30 สำหรับการทดสอบแบบจำลองตามลำดับ

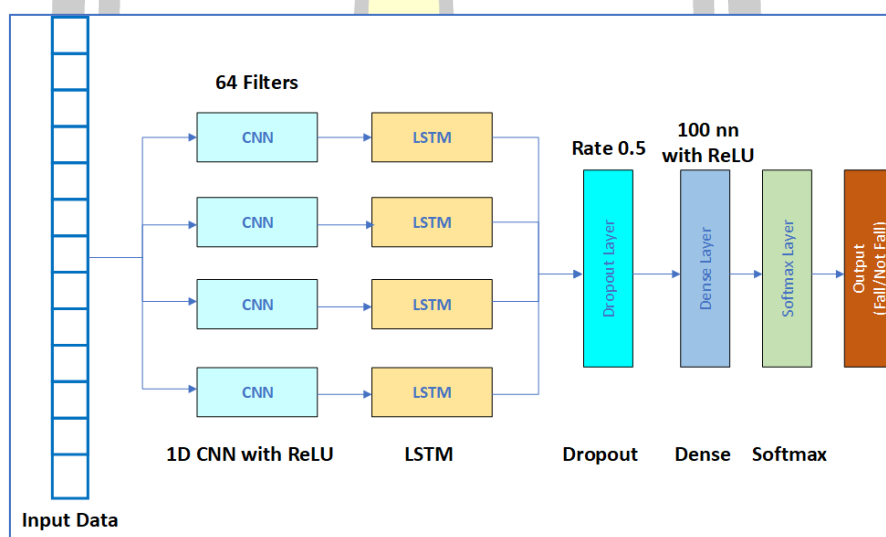


ภาพที่ 15 ข้อมูลชนิดอนุกรมเวลาสำหรับ CNN-LSTM

### 3.3 การสอนแบบจำลอง CNN-LSTM

จากข้อมูลที่ได้จากการเตรียมข้อมูล (Preprocessing) จาก Train Videos ข้อมูลที่ได้จะถูกแบ่งเป็น 2 ส่วน คือ Train Datasets ร้อยละ 70 สำหรับการสอน และ Test Datasets ร้อยละ 30 สำหรับวัดผลแบบจำลอง

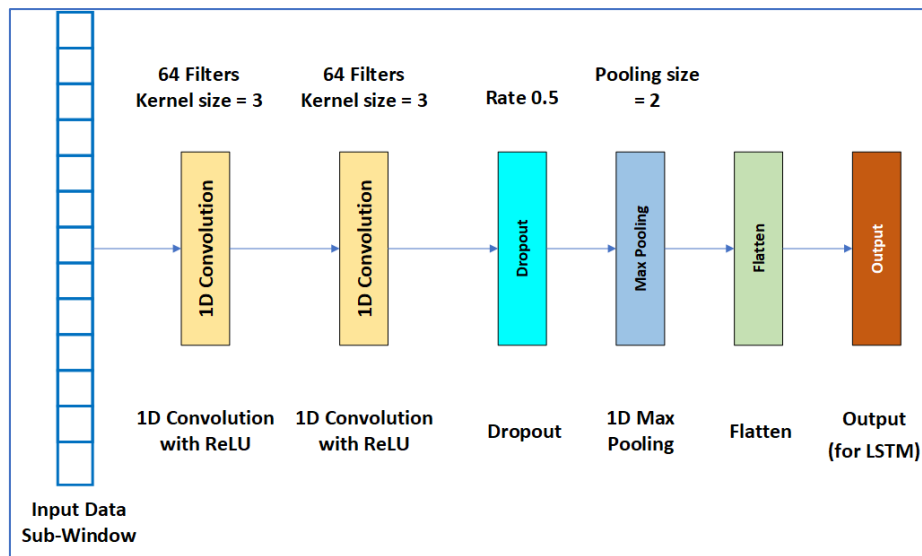
ผู้วิจัยเลือกใช้การเรียนรู้เชิงลึก CNN-LSTM เพื่อวิเคราะห์และจดจำกิจกรรมของคนว่าเกิดการล้มหรือไม่ ประกอบด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network หรือ CNN) ในการสกัดฟีเจอร์ และ LSTM (Long Short-Term Memory) ซึ่งเป็นโครงข่ายประสาทเทียมที่เกิดซ้ำ (Recurrent Neural Network หรือ RNN) เพื่อแปลความหมายของฟีเจอร์ตามลำดับเวลา (Time Series) เพื่อรู้จำกิจกรรมต่างๆ ของคน โดยงานวิจัยนี้สนใจ การล้ม/ไม่ล้มของคนในวิดีโอ มีโครงสร้างตามรูปที่ 16



ภาพที่ 16 โครงสร้างของระบบตรวจจับการล้มโดยใช้ CNN-LSTM

ข้อมูล input 1D 6 แถวยาว 24 เฟรม (1 หน้าต่าง) จะถูกแบ่งออกเป็น 4 ส่วนเท่ากันส่งไปให้กับส่วน CNN ที่มีโครงสร้างตามรูปที่ 17 ซึ่งประกอบด้วยชั้นคอนโวลูชัน 1D ที่มี 64 ตัวกรอง และขนาดเคอร์เนล (Kernel) เป็น 3 จำนวน 2 ชั้น ตามด้วยชั้น Dropout อัตราร้อยละ 0.5 เพื่อลดการ overfitting จากนั้นตามด้วย 1D Max Pooling ขนาดเท่ากับ 2 และ Flatten ตามลำดับ ผลลัพธ์ที่ได้จากส่วน CNN จะถูกส่งไปให้ LSTM เพื่อรู้จำกิจกรรมต่างๆ ของคน(ล้ม/ไม่ล้ม) จากนั้นจะตามด้วย 3 คือ Dropout ร้อยละ 0.5 Dense และ Softmax ตามลำดับ





ภาพที่ 17 โครงสร้างส่วนของ CNN

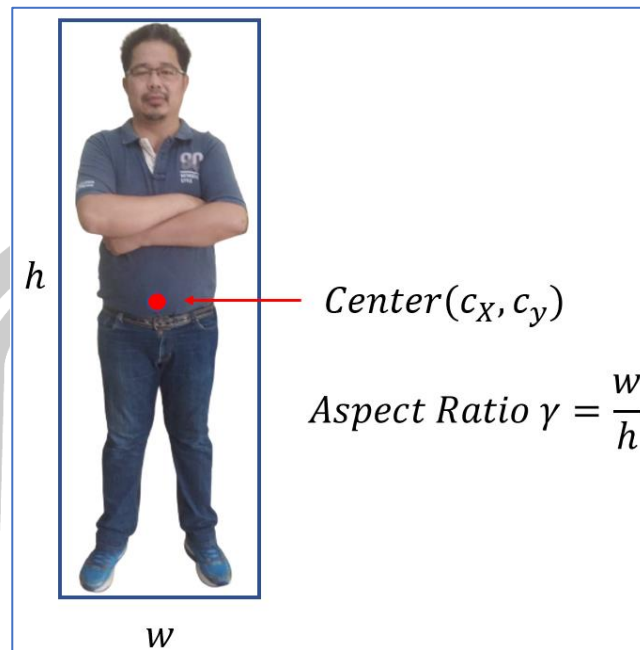
### 3.4 ทดสอบแบบจำลอง

แบบจำลองที่ได้จากการสอนในขั้นตอนก่อนหน้านี้จะถูกทดสอบกับวิดีโอที่เตรียมไว้ร้อยละ 30 สำหรับการทดสอบ และในการวิจัยนี้ผู้วิจัยสนใจที่จะติดตามวัตถุ (Tracking) ไปด้วยเนื่องจากต้องการทราบสถานะแต่ละบุคคลในวิดีโอว่าล้ม หรือไม่ล้ม อีกทั้งข้อมูลที่ป้อนให้แบบจำลอง CNN-LSTM เป็นข้อมูลอนุกรมเวลาแบบต่อเนื่องของแต่ละวัตถุตั้งนั้นเพื่อลดความผิดพลาดที่เกิดจากการสลับข้อมูลกันระหว่างการทำการตรวจจับการล้ม จำเป็นต้องมีการติดตามแต่ละวัตถุอย่างต่อเนื่อง

ในงานวิจัยนี้เลือกใช้ตัวติดตามแบบ DeepSORT และใช้ตัวตรวจจับวัตถุแบบ YOLOv5 โดยสมมุติว่าการเคลื่อนที่ของคนในวิดีโอเป็นไปตามแบบจำลองมาร์คอฟซ่อนเร้น (Hidden Markov model หรือ HMM)

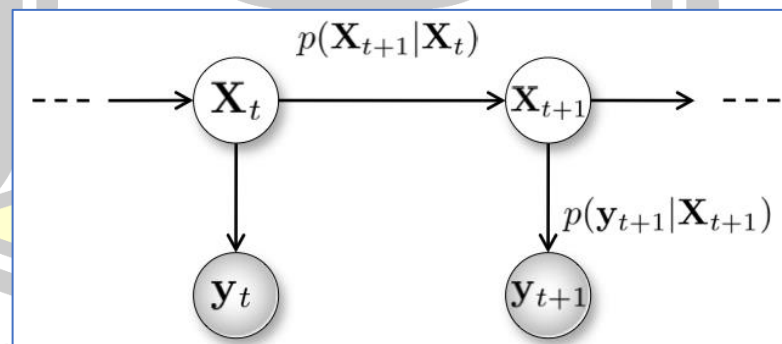
#### 3.4.1 แบบจำลองมาร์คอฟซ่อนเร้น (HMM)

ในงานวิจัยนี้จะใช้แบบจำลองมาร์คอฟซ่อนเร้นเพื่อนิยามการเคลื่อนที่ของคนในวิดีโอ โดยนิยามวัตถุเป็นกรอบสี่เหลี่ยมครอบ (Bounding Box) ที่ได้จากตัวตรวจจับวัตถุ (Detector) ดังรูปที่ 18 กำหนด สถานะ(State)  $x_t = [c_x, c_y, \gamma, h, c_x', c_y', \gamma', h']$  โดยที่  $c_x$  และ  $c_y$  คือจุดศูนย์กลางในแกน  $x$  และ  $y$  ตามลำดับ  $\gamma$  คืออัตราส่วนความกว้างต่อความสูง  $h$  คือความสูง  $c_x', c_y', \gamma', h'$  คือความเร็วของแต่ละตัวแปรตามลำดับ



ภาพที่ 18 Object Modelling

การเปลี่ยนสถานะที่ซ่อนอยู่ (Hidden State)  $\mathbf{x}_t$  จากเฟรมปัจจุบันไปเฟรมถัดไปของวิดีโอ และผลลัพธ์ที่สังเกตพบคือกรอบสี่เหลี่ยมครอบ (Bounding Box)  $\mathbf{y}_t = [c_x, c_y, \gamma, h]$  นั้นสัมพันธ์กันโดยแบบจำลองมาร์คอฟซ่อนเร้น (HMM) ตามรูปที่ 19



ภาพที่ 19 Hidden Markov Model

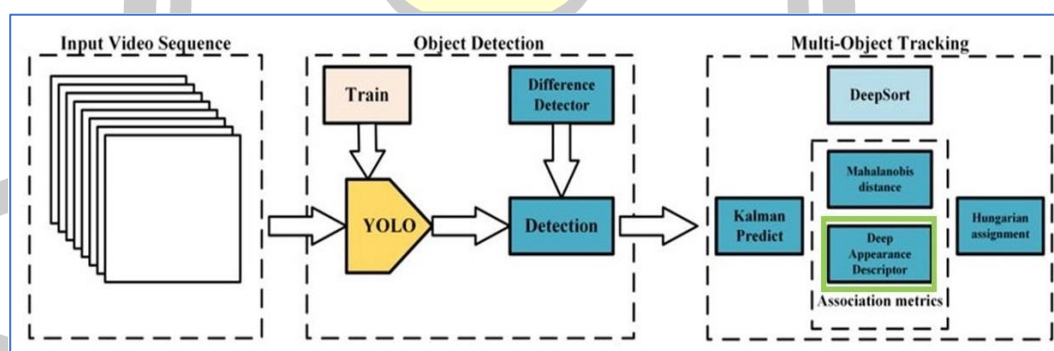
### 3.4.2 ตัวตรวจจับแบบ YOLOv5

ในการตรวจจับวัตถุ (คน) เพื่อให้ได้ผลลัพธ์กรอบสี่เหลี่ยมครอบ (Bounding Box) เพื่อนำไปใช้ในการติดตามวัตถุโดยใช้อัลกอริทึม DeepSORT ต่อไป ผู้วิจัยเลือกใช้ตัวตรวจจับวัตถุแบบ YOLOv5 ที่ถูกสอนโดยชุดข้อมูล COCO ซึ่งสามารถตรวจจับวัตถุได้ 80 ประเภท มีจุดเด่นเรื่อง

ความเร็วและความถูกต้อง โดยจากงานที่ [49] ทดลองกับชุดข้อมูล Tesla P100 ใช้เวลาประมวลผลประมาณ 0.007 วินาทีต่อรูปภาพ หรือ 140 เฟรมต่อวินาที (FPS) ในขณะที่ YOLOv4 ได้แค่ 50 เฟรมต่อวินาที ซึ่งเหมาะที่จะเอามาใช้สำหรับการตรวจจับแบบเรียลไทม์ ในด้านความถูกต้องมีการทดลองใช้ YOLOv5 ทดสอบกับการตรวจจับ และ นับจำนวนเซลล์เม็ดเลือดแดง สามารถให้ค่าเฉลี่ยของความแม่นยำ (mAP) ที่ 0.895 หลังจากเทรนไปแค่ 100 รอบ (Epoch) อีกทั้งขนาดของไฟล์ของ YOLOv5 มีขนาดเล็กแค่ประมาณ 27 เมกะไบต์ แต่ YOLOv4 มีขนาดประมาณ 244 เมกะไบต์ ลดลงเกือบร้อยละ 90

### 3.4.3 ตัวติดตามแบบ DeepSORT

ในงานวิจัยนี้เลือกใช้ DeepSORT เป็นอัลกอริทึมติดตามวัตถุ และกำหนด ID ให้กับแต่ละวัตถุเพื่อให้สามารถแยกคนในวิดีโอได้ในกรณีที่มีหลายคนในวิดีโอเพื่อที่จะระบุสถานะแต่ละคนว่าเกิดการล้ม หรือไม่ล้มได้ โดย DeepSORT คืออัลกอริทึมติดตามวัตถุซึ่งได้ทำการปรับปรุงมาจากอัลกอริทึม SORT (Simple Online Realtime Tracking) โดยเพิ่มตัวบอกลักษณะ (Appearance Descriptor) ซึ่งเป็นการเรียนรู้เชิงลึกเข้าไปเพื่อลดการเปลี่ยน (Switch) การระบุตัวตน (ID) ของวัตถุ ซึ่งทำให้การติดตามมีประสิทธิภาพมากขึ้น นอกจากนี้อัลกอริทึม SORT ยังใช้ตัวกรอกกาลมานในการทำนายสถานะของวัตถุ และอัลกอริทึมฮังการีเรียนในการกำหนดความเชื่อมโยงของวัตถุในแต่ละเฟรม (Assignment) โดยโครงสร้างของ DeepSORT เป็นดังรูปที่ 20



ภาพที่ 20 โครงสร้างของ DeepSORT

### 3.5 Evaluation

ในงานวิจัยนี้ Confusion matrix ถูกนำมาใช้เพื่อวัดประสิทธิภาพของโมเดล โดยความแม่นยำ (Precision) แสดงดังสมการที่ 14 ค่าความไว (Recall) แสดงดังสมการที่ 15 ค่า F1-Score แสดงดังสมการที่ 16 และ ค่าความถูกต้อง (Accuracy) แสดงดังสมการที่ 17

		True Class	
		ล้้ม (Positives)	ไม่ล้้ม (Negatives)
Predicted Class	ล้้ม (Positives)	TP	FP
	ไม่ล้้ม (Negatives)	FN	TN

ภาพที่ 21 Confusion matrix

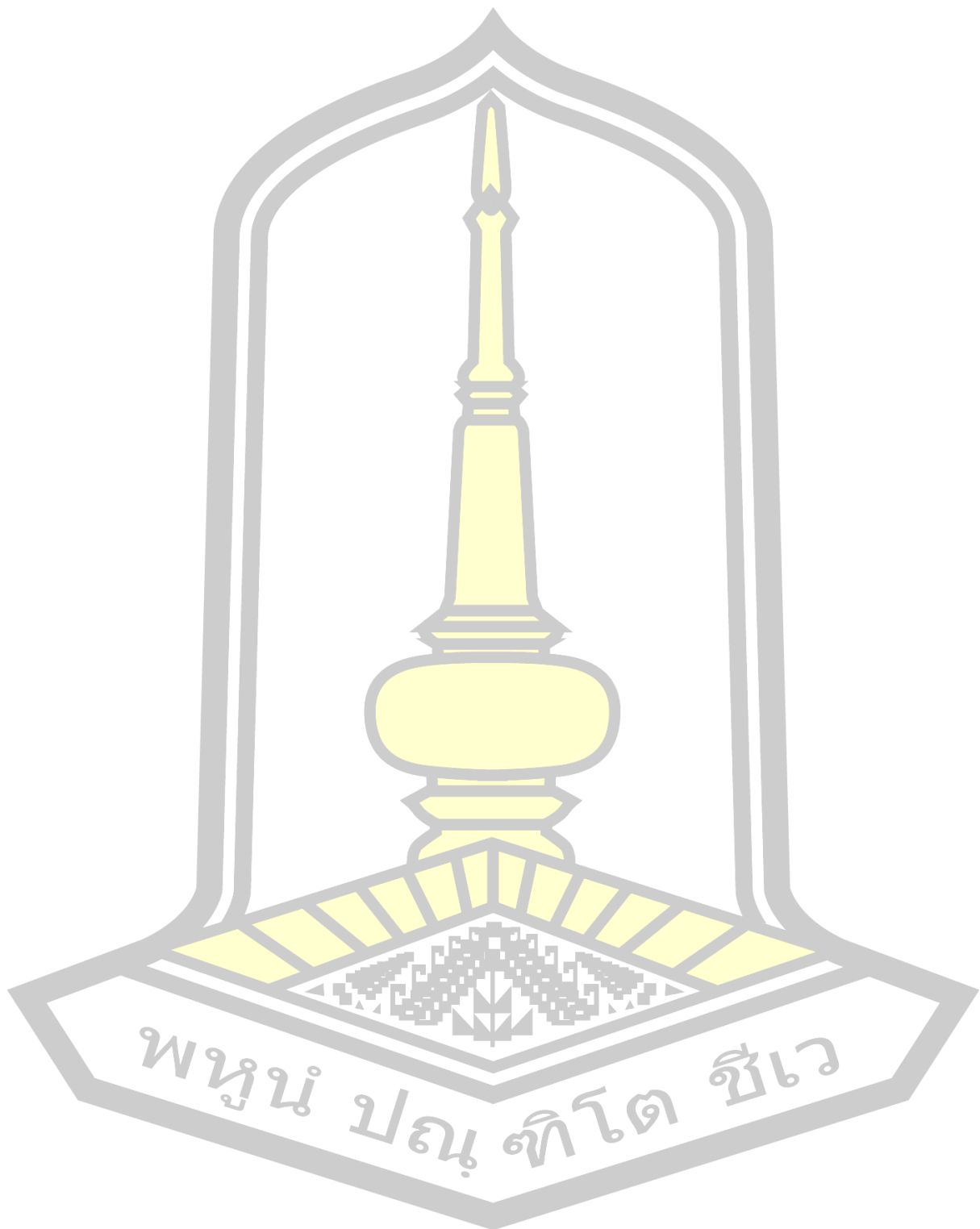
1. True Positive (TP) คือในวิดีโอทดสอบมีคนล้้ม (Truth) และโปรแกรมตรวจจับได้ว่ามีคนล้้ม (Predict)
2. True Negative (TN) คือในวิดีโอทดสอบไม่มีคนล้้ม (Truth) และโปรแกรมตรวจจับได้ว่าไม่มีคนล้้ม (Predict)
3. False Positive (FP) คือในวิดีโอทดสอบไม่มีคนล้้ม (Truth) แต่โปรแกรมตรวจจับได้ว่ามีคนล้้ม (Predict)
4. False Negative (FN) คือในวิดีโอทดสอบมีคนล้้ม (Truth) แต่โปรแกรมตรวจจับได้ว่าไม่มีคนล้้ม (Predict)

$$Precision = \frac{TP}{TP+FP} \quad (14)$$

$$Recall = \frac{TP}{TP+FN} \quad (15)$$

$$F1 - Score = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (16)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (17)$$



## บทที่ 4

### ผลการวิจัยและการอภิปราย

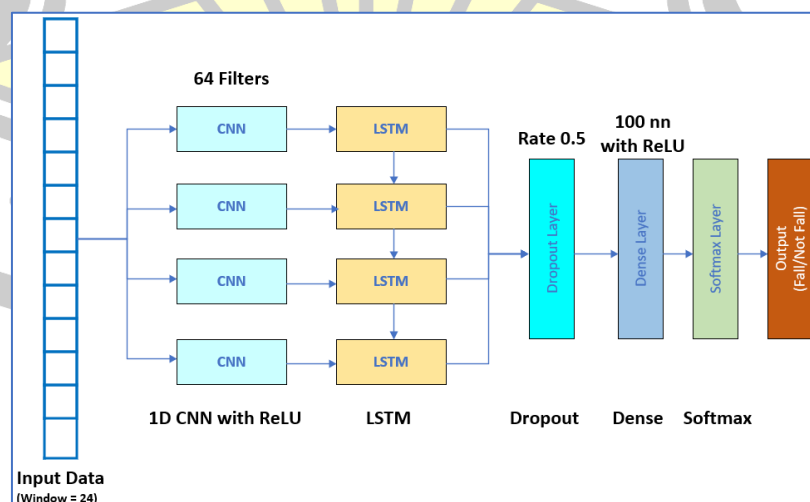
การดำเนินงานระบบตรวจจับการล้มโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันร่วมกับโครงข่ายประสาทเทียมแบบหน่วยความจำระยะสั้น-ระยะยาว (CNN-LSTM) กับข้อมูลมุมของร่างกายส่วนบน ความเร็วเชิงเส้นของจุดที่ลำคอ และ ความเร็วเชิงมุมของมุมของร่างกายส่วนบนซึ่งคำนวณมาจากข้อมูลโครงกระดูกที่สกัดมาจากไลบรารี BlazePose GHUM ที่พัฒนาโดยบริษัท Google โดยข้อมูลจะถูกจัดให้อยู่ในรูปแบบอนุกรมเวลา (Time Series) คล้ายๆกับข้อมูลที่ได้จากเซ็นเซอร์วัดความเร่ง(Accelerometer) และวัดมุม (Gyroscope) ได้ผลดังนี้

#### 4.1 การหาค่าพารามิเตอร์ที่ดีที่สุด

##### 4.1.1 โครงสร้างของ CNN-LSTM

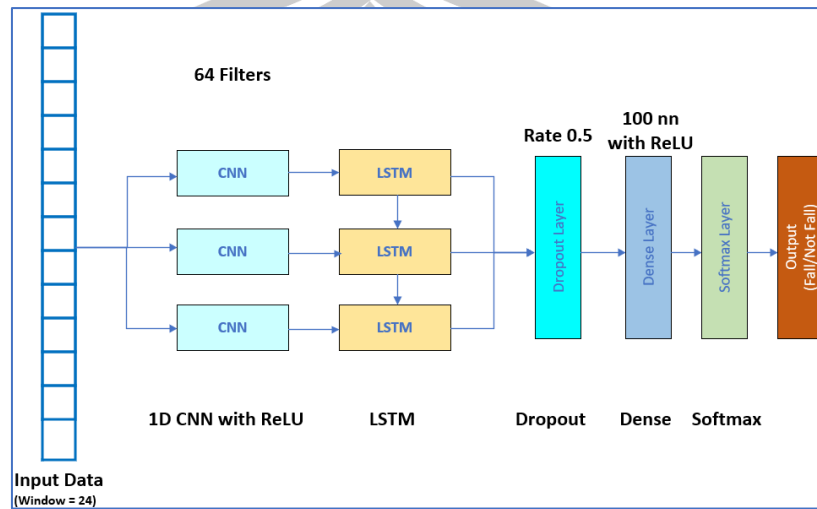
โครงสร้างการเรียนรู้เชิงลึก CNN-LSTM ประกอบด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network หรือ CNN) และ LSTM (Long Short-Term Memory) เนื่องจากข้อมูลนำเข้ามี 9 แถวยาว 24 คอลัมน์ (1 หน้าต่าง) ทำให้สามารถสร้างแบบจำลอง CNN-LSTM ได้ 4 รูปแบบคือ

1. รูปแบบที่ 1 มี CNN-LSTM 4 ชุดขนานกันตามรูปที่ 22 โดยข้อมูลจะถูกแบ่งเป็น 9 แถวยาว 6 คอลัมน์ 4 ชุด



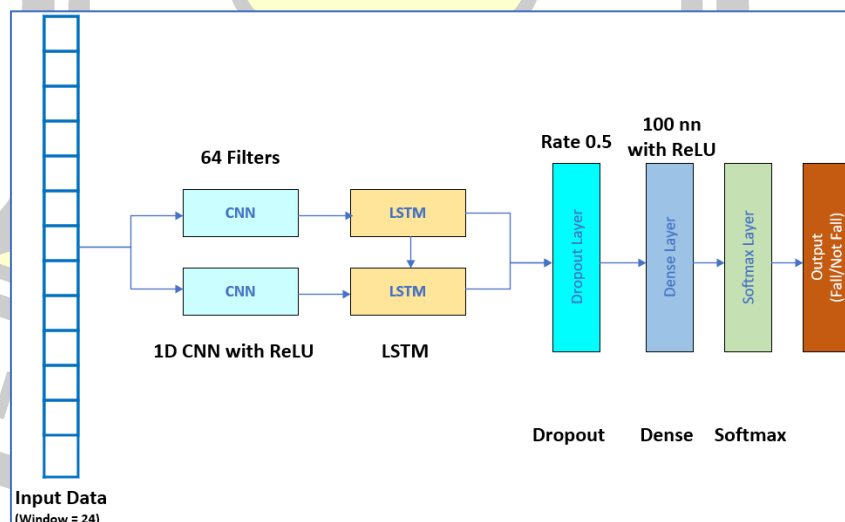
ภาพที่ 22 โครงสร้างรูปแบบที่ 1 มี CNN-LSTM 4 ชุดขนานกัน

2. รูปแบบที่ 2 มี CNN-LSTM 3 ชุดขนานกันตามรูปที่ 23 โดยข้อมูลจะถูกแบ่งเป็น 9 แถวยาว 8 คอลัมน์ 3 ชุด



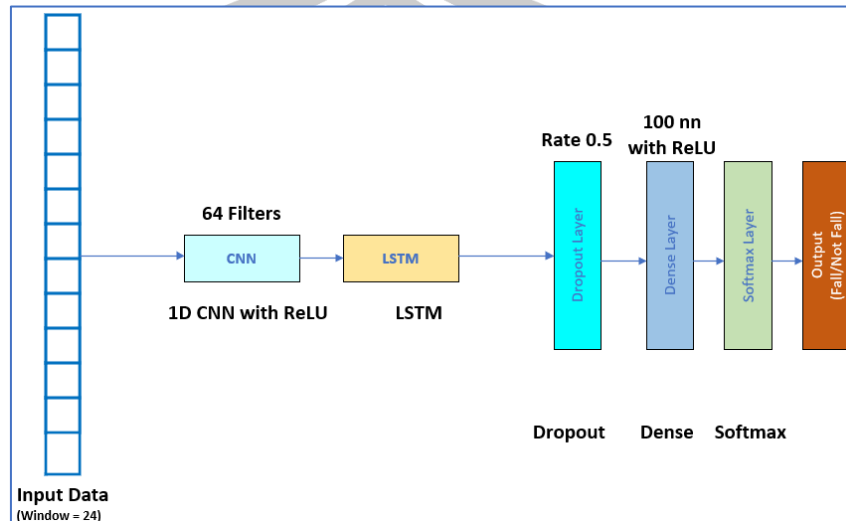
ภาพที่ 23 โครงสร้างรูปแบบที่ 2 มี CNN-LSTM 3 ชุดขนานกัน

3. รูปแบบที่ 3 มี CNN-LSTM 2 ชุดขนานกันตามรูปที่ 24 โดยข้อมูลจะถูกแบ่งเป็น 9 แถวยาว 12 คอลัมน์ 2 ชุด



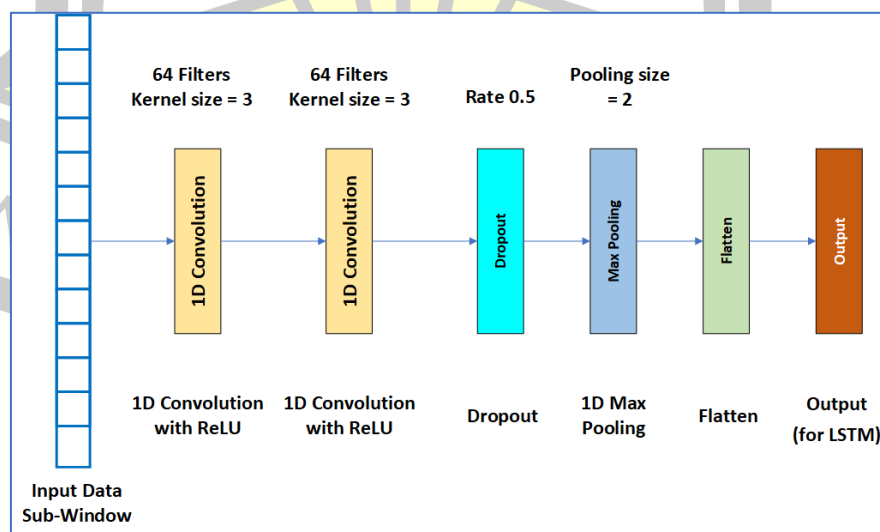
ภาพที่ 24 โครงสร้างรูปแบบที่ 3 มี CNN-LSTM 2 ชุดขนานกัน

4. รูปแบบที่ 4 มี CNN-LSTM 1 ชุดตามรูปที่ 25 โดยข้อมูลจะเป็น 9 แถวยาว 24 คอลัมน์ 1 ชุด



ภาพที่ 25 โครงสร้างรูปแบบที่ 4 มี CNN-LSTM 1 ชุด

ในการทดสอบประสิทธิภาพจะนำข้อมูลสำหรับการสอนและทดสอบแบบจำลองจากหัวข้อที่ 4.1 ทำการสอนและทดสอบแบบจำลอง CNN-LSTM แบบต่างๆ จำนวน 3 รอบเพื่อจะหาค่าเฉลี่ยความถูกต้องที่ได้เพื่อเลือกโครงสร้างที่มีประสิทธิภาพสูงสุดเพื่อใช้ในการทดลองในลำดับถัดไป โดยใช้โครงสร้างและพารามิเตอร์ CNN ตามรูปที่ 26 และใช้จำนวนรอบการเรียนรู้ที่ 50 รอบผลที่ได้ตามตารางที่ 2



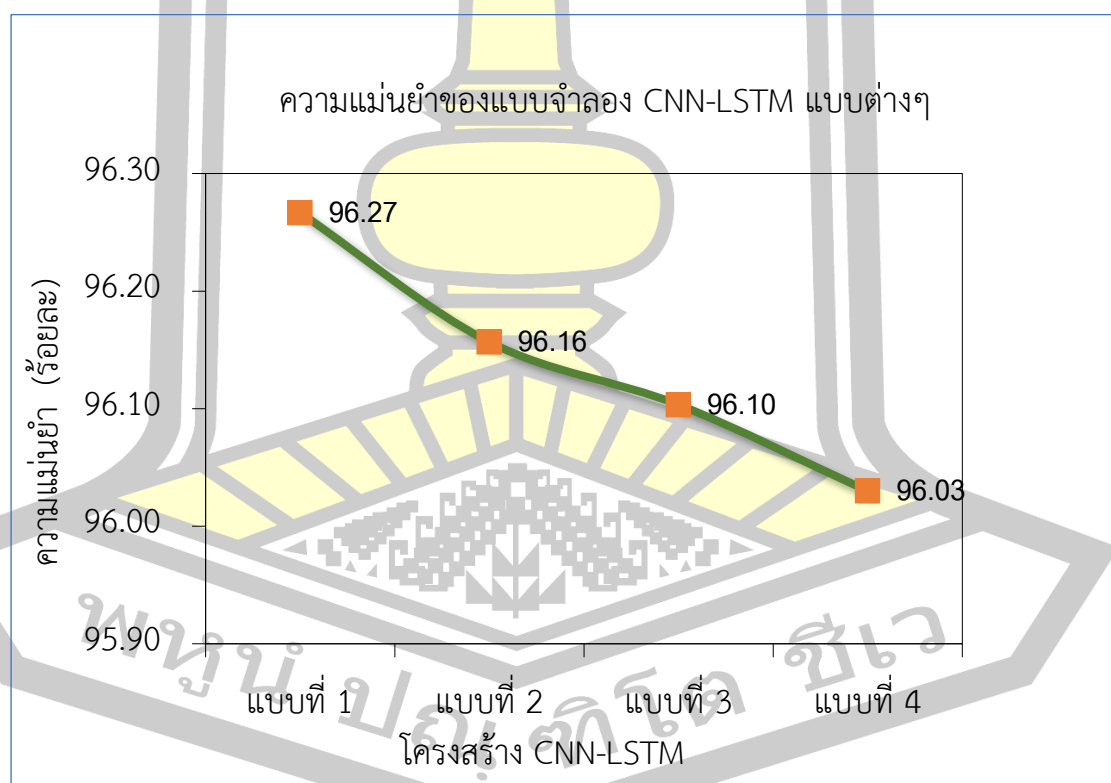
ภาพที่ 26 โครงสร้างส่วนของ CNN



ตารางที่ 2 ผลการทดสอบโครงสร้าง CNN-LSTM แบบต่างๆ

โครงสร้าง	ความยาวข้อมูล	ครั้งที่ 1 (%)	ครั้งที่ 2 (%)	ครั้งที่ 3 (%)	เฉลี่ย (%)
<b>แบบที่ 1</b>	6 คอลัมน์	96.07	96.34	96.39	<b>96.26</b>
แบบที่ 2	8 คอลัมน์	95.85	96.23	96.39	96.15
แบบที่ 3	12 คอลัมน์	95.9	96.12	96.29	96.10
แบบที่ 4	24 คอลัมน์	95.79	96.12	96.18	96.03

จากการทดสอบประสิทธิภาพของโครงสร้าง CNN-LSTM แบบต่างๆจะเห็นว่า โครงสร้างแบบที่ 1 มี CNN-LSTM 4 ระบายย่อยขนานกันโดยข้อมูลนำเข้าจะถูกแบ่งออกเป็น 4 ส่วนยาว 6 คอลัมน์เท่าๆกัน ให้ความถูกต้องเฉลี่ยสูงสุดที่ร้อยละ 96.26 ดังนั้นผู้วิจัยจะเลือกใช้โครงสร้างแบบที่ 1 นี้ในการทดลองในลำดับถัดไป



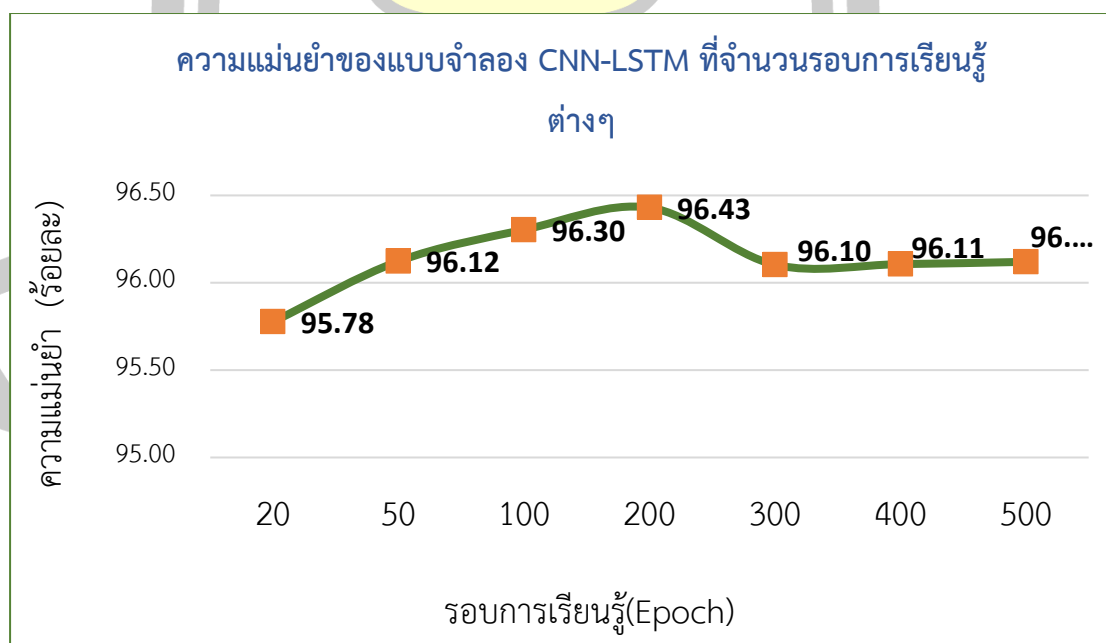
ภาพที่ 27 กราฟแสดงประสิทธิภาพของ CNN-LSTM แบบต่างๆ

#### 4.1.2 จำนวนรอบการเรียนรู้ (Epoch)

ในการหาจำนวนรอบการเรียนรู้ที่เหมาะสมจะทำการสอนและทดสอบแบบจำลอง CNN-LSTM แบบที่ 1 ด้วยจำนวนรอบการเรียนรู้ที่ต่างกันจำนวน 3 รอบเพื่อจะหาค่าเฉลี่ยความถูกต้องที่ได้เพื่อหาจำนวนรอบการเรียนรู้ที่เหมาะสมต่อไป โดยใช้โครงสร้างและพารามิเตอร์ CNN ตามรูปที่ 26 ผลที่ได้ตามตารางที่ 3

ตารางที่ 3 ผลการทดสอบแบบจำลอง CNN-LSTM ที่มีการสอนด้วยรอบการเรียนรู้ต่างกัน

รอบ	ครั้งที่ 1 (%)	ครั้งที่ 2 (%)	ครั้งที่ 3 (%)	เฉลี่ย (%)
20	95.58	95.63	96.12	95.78
50	95.96	96.07	96.34	96.12
100	96.07	96.39	96.45	96.30
<b>200</b>	96.23	96.5	96.56	<b>96.43</b>
300	96.01	96.07	96.23	96.10
400	95.96	96.07	96.29	96.11
500	95.85	96.12	96.39	96.12



ภาพที่ 28 กราฟแสดงประสิทธิภาพของ CNN-LSTM ที่มีการสอนด้วยรอบการเรียนรู้ต่างกัน

จากการทดสอบประสิทธิภาพของแบบจำลอง CNN-LSTM ที่ถูกสอนด้วยจำนวนรอบการเรียนรู้ 20 50 100 200 300 400 และ 500 รอบ (epoch) จำนวน 3 รอบ จะเห็นว่าแบบจำลองที่สอนโดยจำนวนรอบการเรียนรู้มากขึ้นจะให้ความถูกต้องเฉลี่ยสูงขึ้นและสูงที่สุดที่ 200 รอบการเรียนรู้ ให้ความถูกต้องเฉลี่ยสูงสุดที่ร้อยละ 96.43 แต่หลังจาก 200 ถึงแม้จะเพิ่มรอบการเรียนรู้เป็น 300 400 และ 500 ค่าความถูกต้องเฉลี่ยกลับลดลง ดังนั้นผู้วิจัยจะเลือกใช้รอบการเรียนรู้ที่ 200 ในการสอนแบบจำลองเพื่อใช้ในการทดลองในลำดับถัดไป

#### 4.2 ผลการทดสอบแบบจำลอง

หลังจากสอนแบบจำลองด้วยโครงสร้างแบบที่ 1 คือ มี CNN-LSTM จำนวน 4 ชุดขนานกัน ดังรูปที่ 22 ที่รอบการเรียนรู้ 200 รอบจะได้แบบจำลองสำหรับการทดสอบ ในการทดสอบแบบจำลองผู้วิจัยได้ทำการทดลองกับชุดข้อมูล 3 ชุดแยกกัน และ ชุดข้อมูลรวม โดยใช้ตัววัดประสิทธิภาพ 4 ตัวคือ Precision Recall F1 และ Accuracy ตามสมการที่ (18) ถึง (21)

$$Precision = \frac{TP}{TP+FP} \quad (18)$$

$$Recall = \frac{TP}{TP+FN} \quad (19)$$

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (20)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (21)$$

โดยที่

1. True Positive (TP) คือในวิดีโอทดสอบมีคนล้ม (Truth) และโปรแกรมตรวจจับได้ว่ามีคนล้ม (Predict)
2. True Negative (TN) คือในวิดีโอทดสอบไม่มีคนล้ม (Truth) และโปรแกรมตรวจจับได้ว่าไม่มีคนล้ม (Predict)
3. False Positive (FP) คือในวิดีโอทดสอบไม่มีคนล้ม (Truth) แต่โปรแกรมตรวจจับได้ว่ามีคนล้ม (Predict)
4. False Negative (FN) คือในวิดีโอทดสอบมีคนล้ม (Truth) แต่โปรแกรมตรวจจับได้ว่าไม่มีคนล้ม (Predict)

ได้ผลลัพธ์ดังนี้

#### 4.2.1 ผลการทดสอบแบบจำลองกับชุดข้อมูล ImVia Fall Detection Dataset

ตารางที่ 4 Confusion Matrix ของแบบจำลองกับชุดข้อมูล ImVia Fall Detection Dataset

		ผลการทำนาย		
		ไม่ล้ม	กำลังล้ม	นอนนิ่ง
ผลลัพธ์จริง	ไม่ล้ม	1518	1	9
	กำลังล้ม	5	27	0
	นอนนิ่ง	47	0	157

ตารางที่ 5 ผลการวัดประสิทธิภาพของแบบจำลองกับชุดข้อมูล ImVia Fall Detection Dataset

Precision (ร้อยละ)	Recall (ร้อยละ)	F1-Score (ร้อยละ)	Accuracy (ร้อยละ)
95.9	86.89	90.95	96.49

#### 4.2.2 ผลการทดสอบแบบจำลองกับชุดข้อมูล UR Fall Detection Dataset

ตารางที่ 6 Confusion Matrix ของแบบจำลองกับชุดข้อมูล UR Fall Detection Dataset

		ผลการทำนาย		
		ไม่ล้ม	กำลังล้ม	นอนนิ่ง
ผลลัพธ์จริง	ไม่ล้ม	12	0	0
	กำลังล้ม	0	7	0
	นอนนิ่ง	1	0	4

ตารางที่ 7 ผลการวัดประสิทธิภาพของแบบจำลองกับชุดข้อมูล UR Fall Detection Dataset

Precision (ร้อยละ)	Recall (ร้อยละ)	F1-Score (ร้อยละ)	Accuracy (ร้อยละ)
97.43	91.66	93.90	95.65

#### 4.2.3 ผลการทดสอบแบบจำลองกับชุดข้อมูล FALLALLD

ตารางที่ 8 Confusion Matrix ของแบบจำลองกับชุดข้อมูล FALLALLD

		ผลการทำนาย		
		ไม่ล้ม	กำลังล้ม	นอนนิ่ง
ผลลัพธ์จริง	ไม่ล้ม	12	0	0
	กำลังล้ม	1	1	0
	นอนนิ่ง	0	0	32

ตารางที่ 9 ผลการวัดประสิทธิภาพของแบบจำลองกับชุดข้อมูล FALLALLD

Precision (ร้อยละ)	Recall (ร้อยละ)	F1-Score (ร้อยละ)	Accuracy (ร้อยละ)
97.43	83.33	87.55	97.826

#### 4.2.4 ผลการทดสอบแบบจำลองกับชุดข้อมูลรวม

ตารางที่ 10 Confusion Matrix ของแบบจำลองกับชุดข้อมูลรวม

		ผลการทำนาย		
		ไม่ล้ม	กำลังล้ม	นอนนิ่ง
ผลลัพธ์จริง	ไม่ล้ม	1542	1	9
	กำลังล้ม	6	35	0
	นอนนิ่ง	47	0	193

ตารางที่ 11 ผลการวัดประสิทธิภาพของแบบจำลองกับชุดข้อมูลรวม

Precision (ร้อยละ)	Recall (ร้อยละ)	F1-Score (ร้อยละ)	Accuracy (ร้อยละ)
96.48	88.37	92.07	96.56

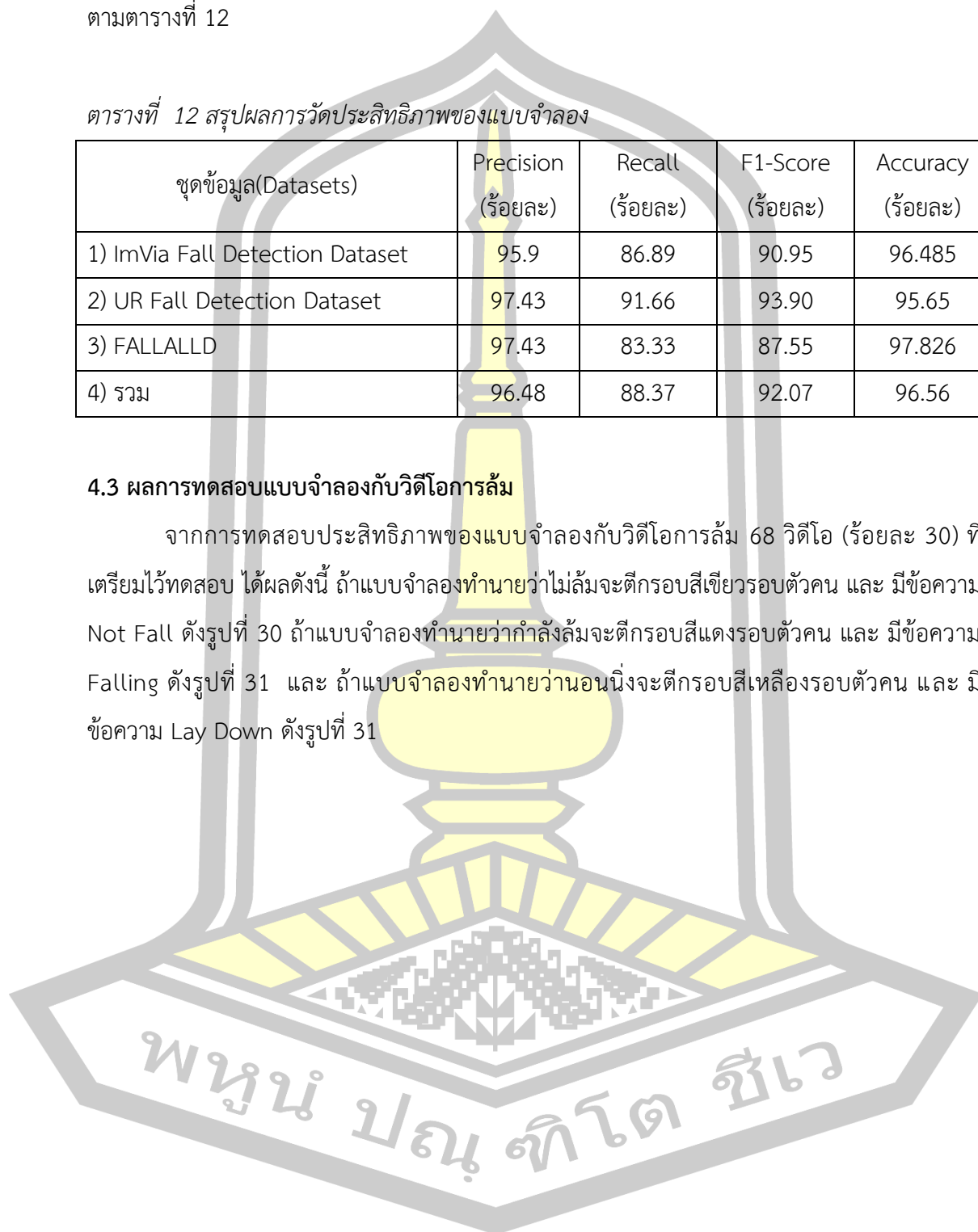
จากการทดสอบประสิทธิภาพของแบบจำลองกับชุดวิดีโอหัวข้อ 4.2.1-4.2.4 สามารถสรุปได้ตามตารางที่ 12

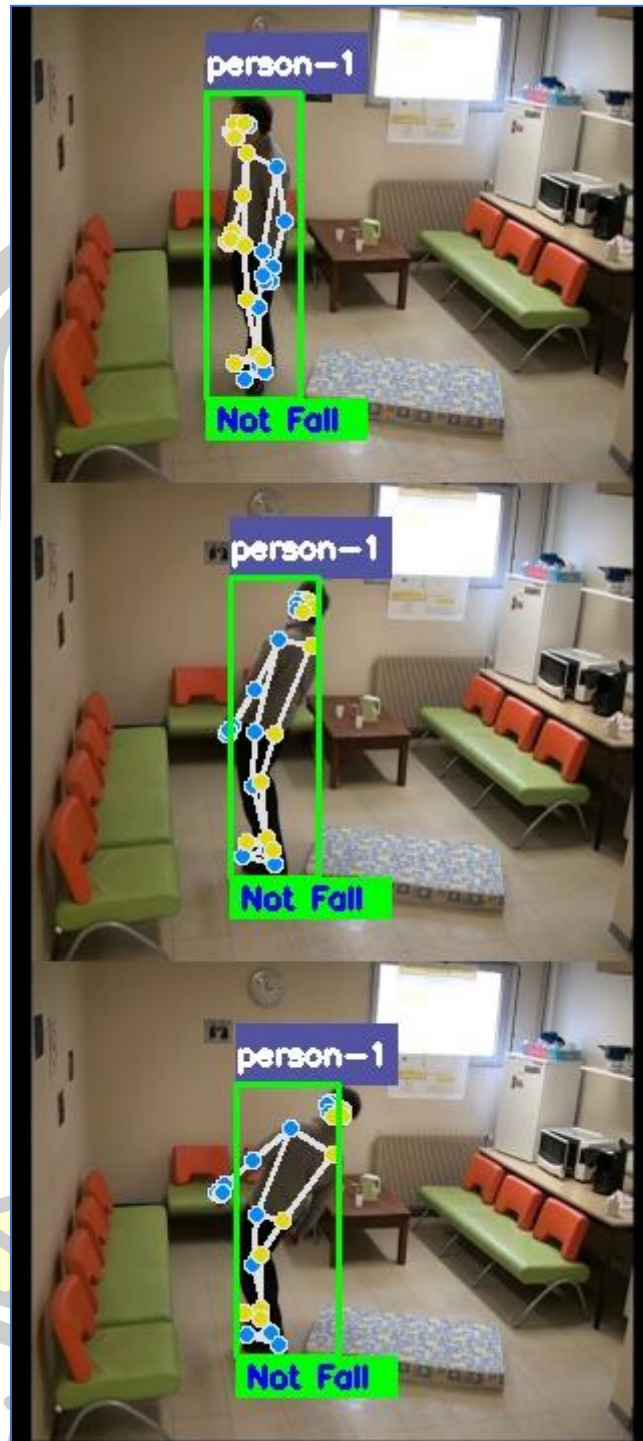
ตารางที่ 12 สรุปผลการวัดประสิทธิภาพของแบบจำลอง

ชุดข้อมูล(Datasets)	Precision (ร้อยละ)	Recall (ร้อยละ)	F1-Score (ร้อยละ)	Accuracy (ร้อยละ)
1) ImVia Fall Detection Dataset	95.9	86.89	90.95	96.485
2) UR Fall Detection Dataset	97.43	91.66	93.90	95.65
3) FALLALLD	97.43	83.33	87.55	97.826
4) รวม	96.48	88.37	92.07	96.56

#### 4.3 ผลการทดสอบแบบจำลองกับวิดีโอการล้ม

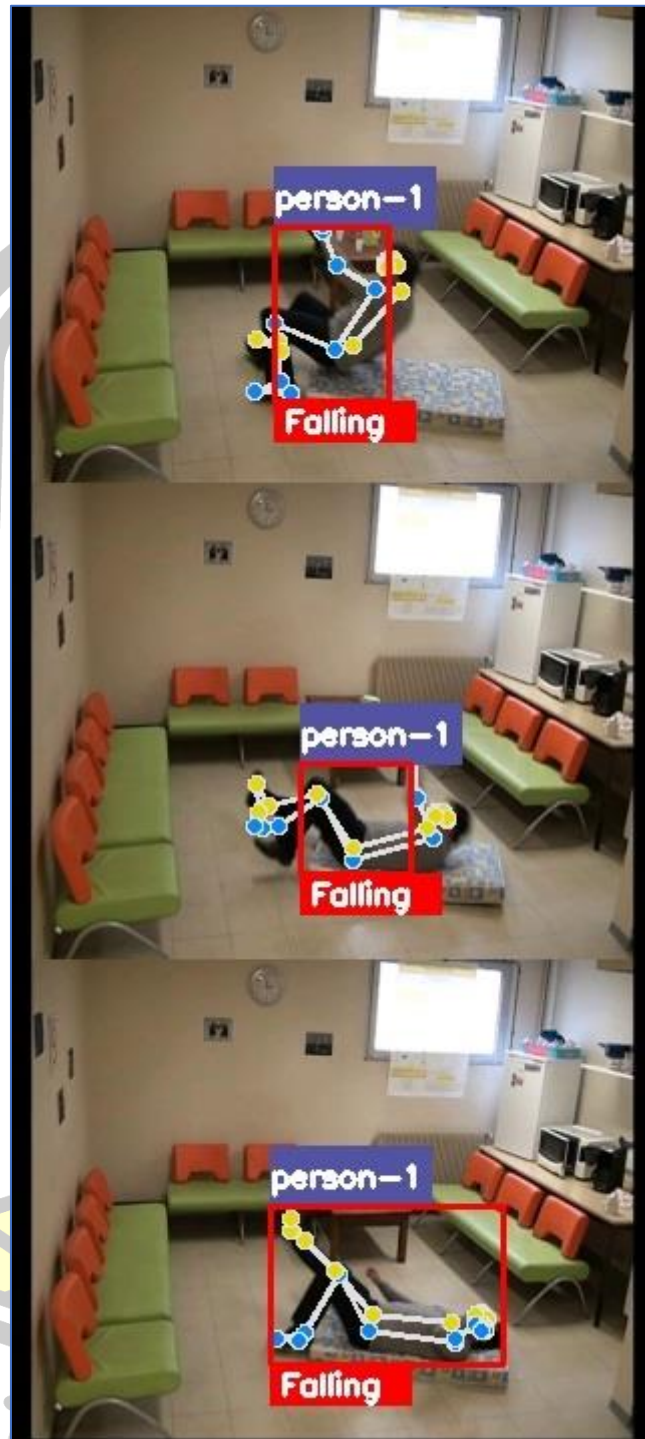
จากการทดสอบประสิทธิภาพของแบบจำลองกับวิดีโอการล้ม 68 วิดีโอ (ร้อยละ 30) ที่เตรียมไว้ทดสอบ ได้ผลดังนี้ ถ้าแบบจำลองทำนายว่าไม่ล้มจะตีกรอบสีเขียวรอบตัวคน และมีข้อความ Not Fall ดังรูปที่ 30 ถ้าแบบจำลองทำนายว่ากำลังล้มจะตีกรอบสีแดงรอบตัวคน และมีข้อความ Falling ดังรูปที่ 31 และ ถ้าแบบจำลองทำนายว่านอนนิ่งจะตีกรอบสีเหลืองรอบตัวคน และมีข้อความ Lay Down ดังรูปที่ 31





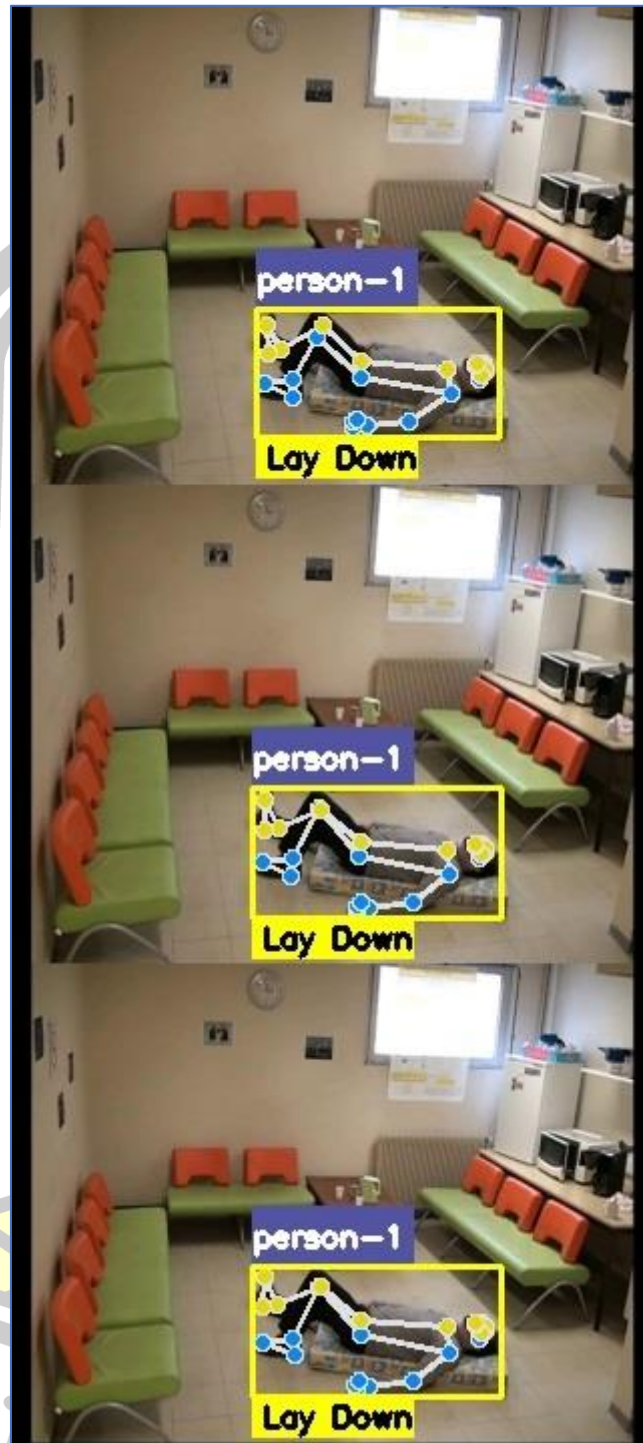
ภาพที่ 29 ตัวอย่างรูปแบบจำลองทำนายว่าไม่ล้ม





ภาพที่ 30 ตัวอย่างรูปที่แบบจำลองทำนายว่ากำลังล้ม





ภาพที่ 31 ตัวอย่างรูปที่แบบจำลองทำนายว่านอนนิ่ง

## บทที่ 5

### สรุปผล อภิปรายผล และข้อเสนอแนะ

งานวิจัยนี้เสนอวิธีการตรวจจับการล้มโดยใช้เทคโนโลยีมองเห็นของเครื่อง เพื่อให้ได้ระบบที่มีความแม่นยำสูงโดยไม่ต้องใช้อุปกรณ์สวมใส่ ซึ่งเป็นสิ่งสำคัญสำหรับผู้สูงอายุ เนื่องจากบางคนอาจทำให้เกิดการระคายเคืองหรือแผล และผู้สูงอายุที่มีภาวะหลังลึมห่างจะสวมใส่อุปกรณ์ได้ นอกจากนี้ อุปกรณ์ที่ต้องชาร์จแบตเตอรี่และบำรุงรักษาสมาเสมอยังมีข้อจำกัดในการใช้งานซึ่งไม่สะดวกสำหรับผู้ใช้งาน

#### 5.1 สรุปผลการวิจัย

โดยงานวิจัยนี้จะใช้วิธีตรวจจับการล้มจากการวิเคราะห์ค่ามุมของร่างกายส่วนบน ค่าความเร็วเชิงเส้นของจุดที่คอ และความเร็วเชิงมุมของร่างกายก่อนบนของคนที่คำนวณมาจาก 4 จุดสำคัญ (Landmark) จุดจากข้อมูลโครงกระดูก (Skeleton) ที่สกัดโดยใช้ไลบรารีการเรียนรู้เชิงลึก BlazePose GHUM ที่พัฒนาโดย Google โดยข้อมูลที่ได้จะถูกจัดให้อยู่ในรูปแบบอนุกรมเวลา (Time Series) 9 แถวคล้ายกับข้อมูลที่ได้จากเซ็นเซอร์ความเร่ง (accelerometer) และ เซ็นเซอร์วัดความเอียง (gyroscope) จากนั้นใช้โดยใช้วิธีการเรียนรู้เชิงลึก CNN-LSTM ซึ่งเป็นการใช้โครงข่ายประสาทเทียมคอนโวลูชัน (CNN) ทำหน้าที่สกัดลักษณะเด่นที่สำคัญ (features) ร่วมกับการใช้โครงข่ายประสาทเทียมหน่วยความจำระยะสั้น-ยาว (Long Short – Term Memory หรือ LSTM) จัดจํารูปแบบการเกิดซ้ำของข้อมูลและจําแนกพฤติกรรมต่างๆ ของคน เพื่อให้สามารถตรวจจับการล้มได้อย่างมีประสิทธิภาพ

วิธีนี้ได้รับความถูกต้องสูง (accuracy) ร้อยละ 96.56 ความแม่นยำในการทำนาย (precision) สูงร้อยละ 96.48 และความแม่นยำในการตรวจหาเหตุการณ์ที่เกี่ยวข้องกับการล้ม (recall) สูงร้อยละ 88.37 อย่างไรก็ตาม แม้ว่าวิธีของเราสามารถลดจำนวนสัญญาณเตือนความผิดปกติที่เกิดจากการเคลื่อนไหวของร่างกายอย่างรวดเร็วที่คล้ายกับการล้มได้ แต่ยังมีข้อจำกัด ข้อมูลโครงสร้างของสเกเลตัน (skeleton) ที่ถูกดึงออกจาก BlazePose มีความผิดพลาดมากกว่าเซนเซอร์ และการล้มเป็นเหตุการณ์ที่เสี่ยงต่อชีวิต ดังนั้น วิธีของเราอาจต้องรวมกับวิธีการตรวจหาการล้มอื่นๆ เพื่อเพิ่มความแม่นยำ เช่น การใช้กล้องหลายตัวเพื่อลดความผิดพลาดที่เกิดจากการบดบังวัตถุหรือมุมมองที่ BlazePose ไม่สามารถตรวจหาจุดสัญญาณทั้ง 33 ได้อย่างแม่นยำ หรืออาจต้องรวมเทคโนโลยีที่ใช้

เซ็นเซอร์และการมองเห็นเข้าด้วยกันเพื่อให้ได้ผลลัพธ์ที่น่าพอใจและเหมาะสมสำหรับการใช้งานจริง วิธีนี้เหมาะสมเป็นระบบสำรองสำหรับการตรวจหาการล้มในผู้สูงอายุที่อาจจะล้มสวมอุปกรณ์ติดตาม หรือมีความไม่สะดวกในการสวมใส่

## 5.2 ข้อจำกัด และข้อเสนอแนะ

ระบบที่พัฒนาขึ้นถึงแม้จะให้ความแม่นยำในการตรวจจับการล้มที่สูงแต่ยังมีข้อจำกัดที่ต้องพิจารณาเพื่อให้นำไปปรับใช้งานให้เหมาะสม และมีประสิทธิภาพสูงสุดดังนี้

5.2.1 ไลบรารีการเรียนรู้เชิงลึก BlazePose สามารถการตรวจจับโครงกระดูกได้ดี และ มีความเร็วสูงมากแต่ก็ยังมีคามผิดพลาดมากกว่าเซ็นเซอร์เพราะเป็นการประมาณตำแหน่งของร่างกาย 33 จุดใน 3 มิติจากรูป 2 มิติ ในบางตำแหน่งและทิศทางของคนในวิดีโอยังมีความผิดพลาดสูง และไม่สามารถตรวจจับตำแหน่งได้ครบทำให้ไม่สามารถนำไปคำนวณหาค่ามุม ความเร็วเชิงเส้น ความเร็วเชิงมุมต่อได้ และการตรวจจับโครงกระดูกของไลบรารี BlazePose จำเป็นต้องตรวจจับใบหน้าให้ได้ก่อน(เฉพาะรูปแรก)ดังนั้นในบางวิดีโอที่ไม่เห็นหน้าไลบรารี BlazePose จะไม่สามารถตรวจจับโครงกระดูกได้ อีกข้อจำกัดของไลบรารี BlazePose เวอร์ชันปัจจุบันคือสามารถตรวจจับ และติดตาม ตำแหน่งของร่างกายได้ที่ละ 1 คนถ้าต้องการให้ตรวจจับได้ที่หลายคนอาจจะต้องมีการใช้ร่วมกับเทคนิคอื่นๆ

5.2.2 ไลบรารีการเรียนรู้เชิงลึก YOLOv5 เป็นโมเดลการตรวจจับวัตถุ (Object Detection) ที่ได้รับความนิยม ความเร็วและความแม่นยำสูงในการตรวจจับวัตถุ โดย YOLOv5 มีการพัฒนาขึ้นจาก YOLOv4 โดยมีการเพิ่มประสิทธิภาพในการตรวจจับวัตถุและลดเวลาในการสร้างโมเดล สามารถตรวจจับวัตถุได้หลายประเภท เช่น รถยนต์ เครื่องบิน คน สัตว์ และอื่นๆ แต่ YOLOv5 ต้องการเครื่องคอมพิวเตอร์ และการ์ดประมวลผลภาพประสิทธิภาพสูงมากในการประมวลผล ดังนั้นในการใช้งานจริงอาจต้องการคอมพิวเตอร์ที่ผลิตมาเพื่อการเรียนรู้เชิงลึกโดยเฉพาะซึ่งมีราคาสูงเช่น Nvidia Jetson AGX ถึงจะทำงานแบบทันทีทันใด (Real Time) ได้ หรือ อาจจะต้องใช้การตรวจจับวัตถุแบบอื่นแทนการใช้ YOLOv5

## บรรณานุกรม

1. Nghiem, A.T., E. Auvinet, and J. Meunier. *Head detection using kinect camera and its application to fall detection*. in *2012 11th international conference on information science, signal processing and their applications (ISSPA)*. 2012. IEEE.
2. Rougier, C., et al. *Monocular 3D head tracking to detect falls of elderly people*. in *2006 international conference of the IEEE engineering in medicine and biology society*. 2006. IEEE.
3. Yang, L., Y. Ren, and W. Zhang, *3D depth image analysis for indoor fall detection of elderly people*. *Digital Communications and Networks*, 2016. **2**(1): p. 24-34.
4. ImViA, *Fall Detection Dataset*. 2020.
5. Saleh, M., M. Abbas, and R.B. Le Jeannes, *FallAID: An open dataset of human falls and activities of daily living for classical and deep learning applications*. *IEEE Sensors Journal*, 2020. **21**(2): p. 1849-1858.
6. Anderson, D., et al. *Recognizing falls from silhouettes*. in *2006 international conference of the IEEE engineering in medicine and biology society*. 2006. IEEE.
7. Bian, Z.-P., L.-P. Chau, and N. Magnenat-Thalmann. *Fall detection based on skeleton extraction*. in *Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry*. 2012.
8. Planinc, R. and M. Kampel, *Introducing the use of depth data for fall detection*. *Personal and ubiquitous computing*, 2013. **17**: p. 1063-1072.
9. Li, Q., et al. *Accurate, fast fall detection using gyroscopes and accelerometer-derived posture information*. in *2009 Sixth International Workshop on Wearable and Implantable Body Sensor Networks*. 2009. IEEE.
10. Dai, J., et al., *Mobile phone-based pervasive fall detection*. *Personal and ubiquitous computing*, 2010. **14**: p. 633-643.
11. Ozcan, K. and S. Velipasalar, *Wearable camera-and accelerometer-based fall detection on portable devices*. *IEEE Embedded Systems Letters*, 2015. **8**(1): p. 6-9.

12. Jeon, S.-B., et al., *Self-powered fall detection system using pressure sensing triboelectric nanogenerators*. *Nano Energy*, 2017. **41**: p. 139-147.
13. Popescu, M., et al. *An acoustic fall detector system that uses sound height information to reduce the false alarm rate*. in *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. 2008. IEEE.
14. Alwan, M., et al. *A smart and passive floor-vibration based fall detector for elderly*. in *2006 2nd International Conference on Information & Communication Technologies*. 2006. IEEE.
15. Liu, L., et al., *An automatic in-home fall detection system using Doppler radar signatures*. *Journal of Ambient Intelligence and Smart Environments*, 2016. **8**(4): p. 453-466.
16. Li, Y., K. Ho, and M. Popescu, *A microphone array system for automatic fall detection*. *IEEE Transactions on Biomedical Engineering*, 2012. **59**(5): p. 1291-1301.
17. Tao, S., M. Kudo, and H. Nonaka, *Privacy-preserved behavior analysis and fall detection by an infrared ceiling sensor network*. *Sensors*, 2012. **12**(12): p. 16920-16936.
18. Daher, M., et al., *Elder tracking and fall detection system using smart tiles*. *IEEE Sensors Journal*, 2016. **17**(2): p. 469-479.
19. Feng, G., et al., *Floor pressure imaging for fall detection with fiber-optic sensors*. *IEEE Pervasive Computing*, 2016. **15**(2): p. 40-47.
20. Zouba, N., et al. *Monitoring activities of daily living (ADLs) of elderly based on 3D key human postures*. in *Cognitive Vision: 4th International Workshop, ICWW 2008, Santorini, Greece, May 12, 2008, Revised Selected Papers*. 2008. Springer.
21. Auvinet, E., et al., *Fall detection with multiple cameras: An occlusion-resistant method based on 3-d silhouette vertical distribution*. *IEEE transactions on information technology in biomedicine*, 2010. **15**(2): p. 290-300.
22. Sixsmith, A. and N. Johnson, *A smart sensor to detect the falls of the elderly*. *IEEE Pervasive computing*, 2004. **3**(2): p. 42-47.
23. Wu, G., *Distinguishing fall activities from normal activities by velocity characteristics*. *Journal of biomechanics*, 2000. **33**(11): p. 1497-1500.

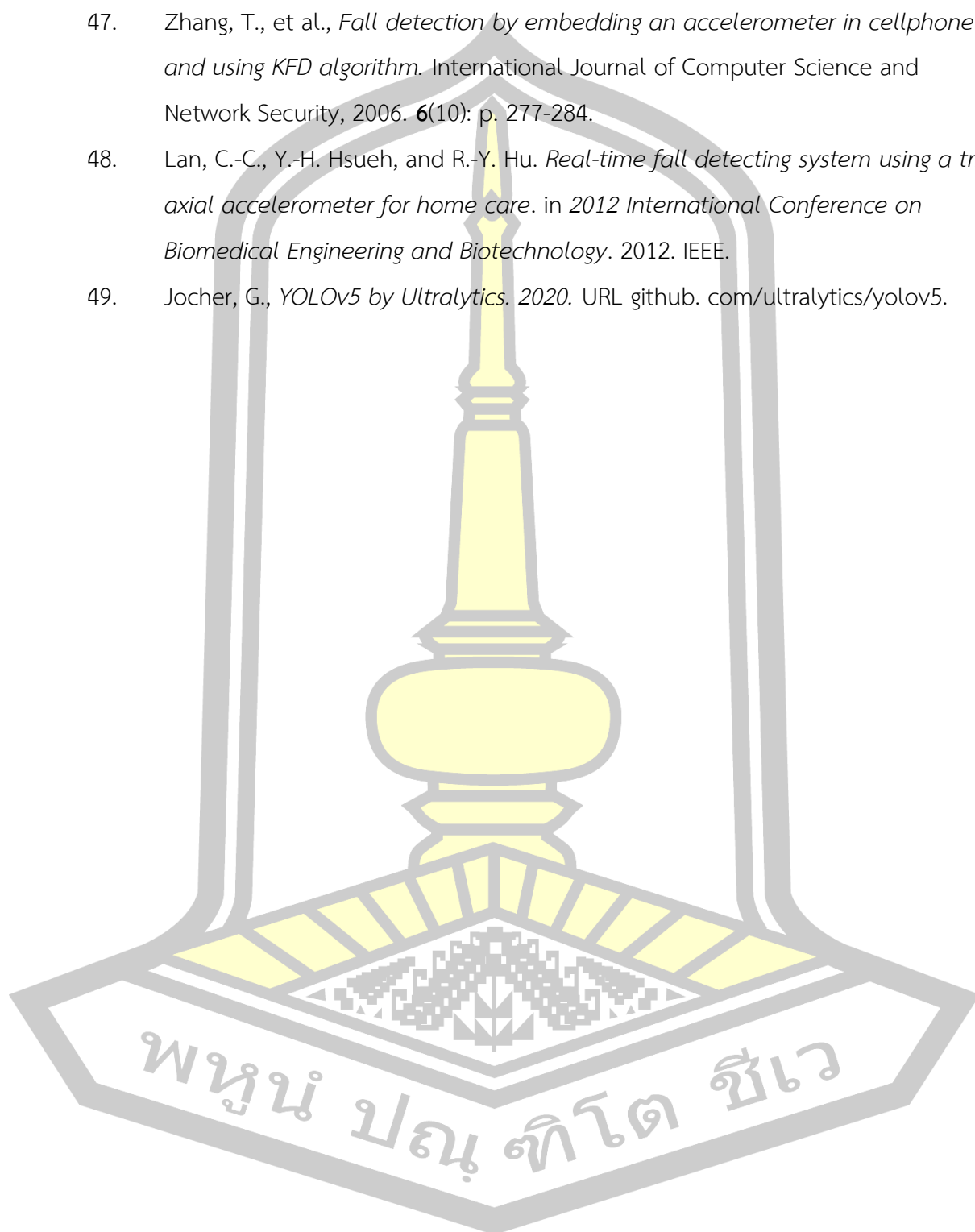


24. Foroughi, H., B.S. Aski, and H. Pourreza. *Intelligent video surveillance for monitoring fall detection of elderly in home environments*. in *2008 11th international conference on computer and information technology*. 2008. IEEE.
25. Spasova, V. and I. Iliev, *A survey on automatic fall detection in the context of ambient assisted living systems*. *International journal of advanced computer research*, 2014. **4**(1): p. 94.
26. Kwolek, B. and M. Kepski, *Human fall detection on embedded platform using depth maps and wireless accelerometer*. *Computer methods and programs in biomedicine*, 2014. **117**(3): p. 489-501.
27. Li, Y., et al. *Improvement of acoustic fall detection using Kinect depth sensing*. in *2013 35th annual international conference of the IEEE Engineering in medicine and biology society (EMBC)*. 2013. IEEE.
28. Brulin, D. and E. Courtial. *Multi-sensors data fusion system for fall detection*. in *Proceedings of the 10th IEEE International Conference on Information Technology and Applications in Biomedicine*. 2010. IEEE.
29. Leone, A., et al. *A multi-sensor approach for people fall detection in home environment*. in *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications-M2SFA2 2008*. 2008.
30. Khawandi, S., P. Chauvet, and B. Daya. *Applying neural network architecture in a multi-sensor monitoring system for the elderly*. in *The Sixth International Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP)*. 2012.
31. Tzeng, H.-W., M.-Y. Chen, and J.-Y. Chen. *Design of fall detection system with floor pressure and infrared image*. in *2010 International Conference on system science and engineering*. 2010. IEEE.
32. Rimminen, H., et al., *Detection of falls among the elderly by a floor sensor using the electric near field*. *IEEE Transactions on Information Technology in Biomedicine*, 2010. **14**(6): p. 1475-1476.
33. Zhang, Z., C. Conly, and V. Athitsos. *A survey on vision-based fall detection*. in *Proceedings of the 8th ACM international conference on PErvasive technologies related to assistive environments*. 2015.

34. Igual, R., C. Medrano, and I. Plaza, *Challenges, issues and trends in fall detection systems*. Biomedical engineering online, 2013. **12**(1): p. 66.
35. Pannurat, N., S. Thiemjarus, and E. Nantajeewarawat, *Automatic fall monitoring: A review*. Sensors, 2014. **14**(7): p. 12900-12936.
36. Delahoz, Y.S. and M.A. Labrador, *Survey on fall detection and fall prevention using wearable and external sensors*. Sensors, 2014. **14**(10): p. 19806-19842.
37. Kulkarni, S. and M. Basu, *A review on wearable tri-axial accelerometer based fall detectors*. J. Biomed. Eng. Technol, 2013. **1**(3): p. 36-39.
38. Rougier, C., et al., *Robust video surveillance for fall detection based on human shape deformation*. IEEE Transactions on circuits and systems for video Technology, 2011. **21**(5): p. 611-622.
39. Vishwakarma, V., C. Mandal, and S. Sural. *Automatic detection of human fall in video*. in *PReMI*. 2007. Citeseer.
40. Hazelhoff, L.B., J. Han, and P.H.d. With. *Video-Based Fall Detection in the Home Using Principal Component Analysis*. in *ACIVS*. 2008.
41. Liu, C.-L., C.-H. Lee, and P.-M. Lin, *A fall detection system using k-nearest neighbor classifier*. Expert systems with applications, 2010. **37**(10): p. 7174-7181.
42. Cucchiara, R., A. Prati, and R. Vezzani, *A multi-camera vision system for fall detection and alarm generation*. Expert Systems, 2007. **24**(5): p. 334-345.
43. Anderson, D., et al., *Linguistic summarization of video for fall detection using voxel person and fuzzy logic*. Computer vision and image understanding, 2009. **113**(1): p. 80-89.
44. Yun, Y. and I.Y.-H. Gu, *Human fall detection in videos via boosting and fusing statistical features of appearance, shape and motion dynamics on Riemannian manifolds with applications to assisted living*. Computer Vision and Image Understanding, 2016. **148**: p. 111-122.
45. Igual, R., C. Medrano, and I. Plaza, *A comparison of public datasets for acceleration-based fall detection*. Medical engineering & physics, 2015. **37**(9): p. 870-878.
46. Schwickert, L., et al., *Fall detection with body-worn sensors: a systematic*



- review. *Zeitschrift für Gerontologie und Geriatrie*, 2013. **46**(8).
47. Zhang, T., et al., *Fall detection by embedding an accelerometer in cellphone and using KFD algorithm*. *International Journal of Computer Science and Network Security*, 2006. **6**(10): p. 277-284.
48. Lan, C.-C., Y.-H. Hsueh, and R.-Y. Hu. *Real-time fall detecting system using a tri-axial accelerometer for home care*. in *2012 International Conference on Biomedical Engineering and Biotechnology*. 2012. IEEE.
49. Jocher, G., *YOLOv5 by Ultralytics*. 2020. URL [github.com/ultralytics/yolov5](https://github.com/ultralytics/yolov5).



ภาคผนวก

ตารางที่ 13 ตัวอย่างข้อมูล Ø แบบอนุกรมเวลาที่ใช้ในการสอน และทดสอบแบบจำลองจำนวน 100 ชุดข้อมูล หน้าต่างขนาด 24 เฟรม

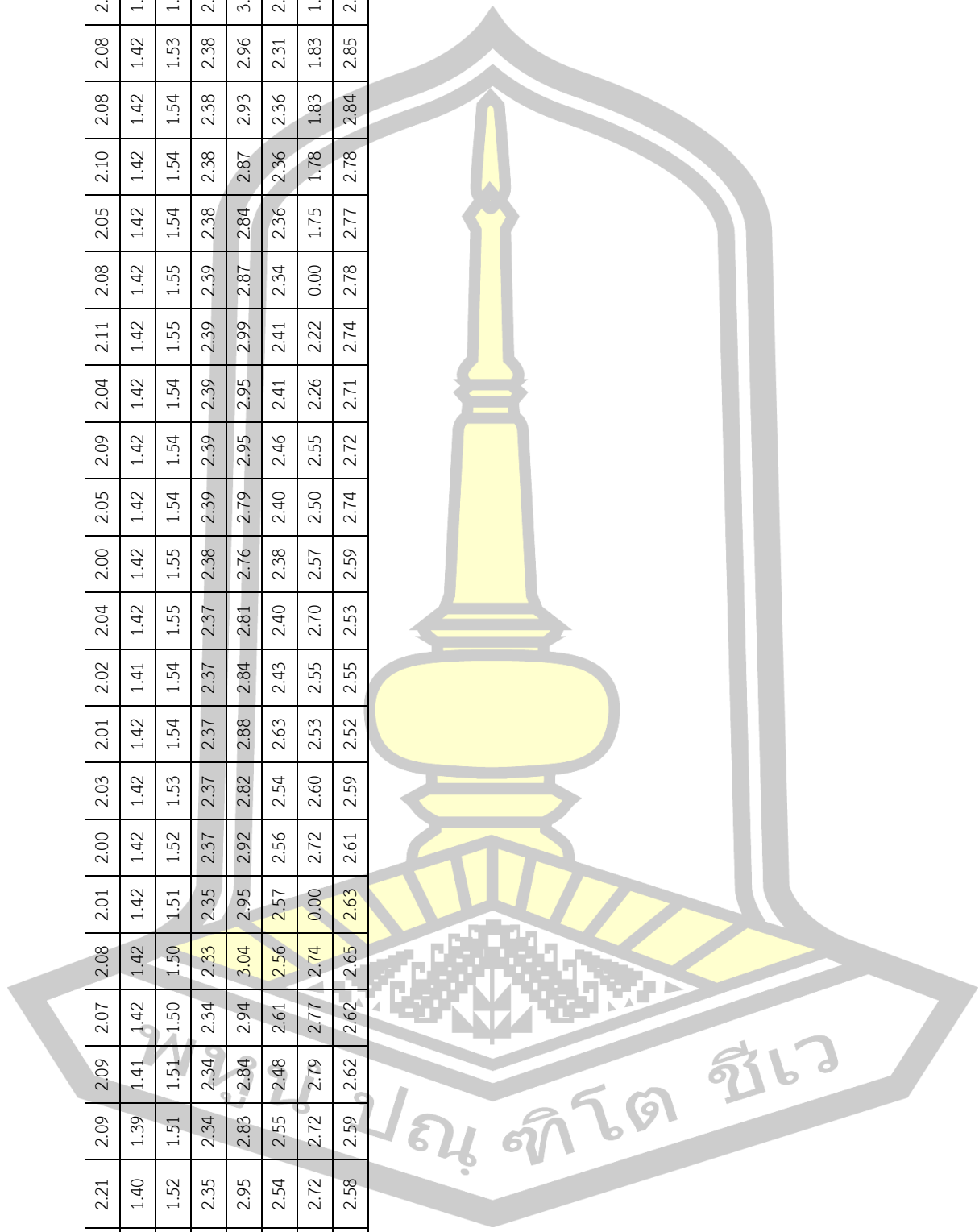
		อนุกรมเวลา(หน้าต่างขนาด 24 เฟรม)																							
ชุดที่		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	2.42	2.40	2.34	2.34	2.32	2.31	2.34	2.27	2.26	2.22	2.18	2.12	2.12	2.13	2.13	2.15	2.15	2.16	2.15	2.19	2.11	2.10	2.12	2.15	
2	2.45	2.45	2.44	2.45	2.45	2.43	2.49	2.53	2.56	2.59	2.63	2.67	2.68	2.67	2.74	2.61	2.54	2.61	2.70	2.72	2.77	2.79	2.85	2.95	
3	1.86	1.85	1.83	1.80	1.77	1.77	1.78	1.81	1.80	1.79	1.79	1.81	1.81	1.81	1.81	1.82	1.84	1.88	1.92	1.93	1.93	1.90	1.90	1.90	
4	2.06	2.08	2.09	2.09	2.05	2.04	2.06	2.06	2.07	2.06	2.05	2.05	2.05	1.98	1.98	1.97	1.98	1.93	1.91	1.87	1.91	1.87	1.88	1.90	
5	2.37	2.37	2.37	2.43	2.45	2.45	2.46	2.49	2.49	2.46	2.43	2.43	2.40	2.42	2.42	2.44	2.48	2.49	2.49	2.50	2.49	2.45	2.45	2.47	
6	2.34	2.47	2.78	2.54	2.48	2.42	2.54	2.59	2.50	2.30	2.60	2.55	2.77	2.75	2.87	2.77	2.65	2.77	2.40	2.48	2.55	2.71	2.60	2.64	
7	1.65	1.69	1.69	1.73	1.75	1.77	1.79	1.84	1.91	1.92	1.95	2.00	2.08	2.17	2.53	2.27	2.68	2.67	2.74	2.70	2.67	2.63	2.69	2.34	
8	2.72	2.69	2.69	2.68	2.64	2.59	2.56	2.48	2.41	2.40	2.42	2.33	2.19	2.24	2.24	2.25	2.24	2.27	2.30	2.30	2.28	2.30	2.31	2.35	
9	1.88	1.84	1.83	1.82	1.79	1.78	1.76	1.75	1.74	1.72	1.71	1.69	1.68	1.68	1.67	1.67	1.66	1.64	1.63	1.64	1.64	1.64	1.61	1.59	
10	2.66	2.64	2.64	2.67	2.68	2.63	2.60	2.59	2.66	2.66	2.64	2.66	2.64	2.60	2.57	2.58	2.62	2.60	2.57	2.58	2.63	2.63	2.67	2.66	
11	2.49	2.60	2.67	2.65	2.68	2.67	2.71	2.71	2.72	2.72	2.74	2.68	2.62	2.59	2.64	2.74	2.78	2.80	2.79	2.79	2.76	2.75	2.76	2.75	
12	2.68	2.70	2.70	2.73	2.66	2.69	2.74	2.78	2.78	2.96	2.84	2.88	2.81	2.83	2.82	2.57	2.62	2.45	2.39	2.38	2.48	2.66	2.63	2.48	
13	2.18	2.34	2.41	2.16	2.02	2.07	2.06	2.09	2.25	2.26	2.39	2.54	2.65	2.56	2.62	2.58	2.59	2.57	2.36	2.03	2.14	2.18	2.11	2.14	
14	2.68	2.68	2.62	2.44	2.39	2.30	2.31	2.36	2.60	2.65	2.77	2.79	2.89	2.92	2.93	2.90	2.99	3.05	2.87	2.89	2.90	2.88	2.88	2.60	
15	2.32	2.34	2.40	2.35	2.38	2.32	2.36	2.43	2.54	2.51	2.53	2.33	2.29	2.20	2.18	2.22	2.29	2.31	2.35	2.39	2.41	2.41	2.44	2.53	
16	2.71	2.74	2.69	2.66	2.67	2.68	2.76	2.85	2.95	3.01	2.88	2.81	2.68	2.66	2.86	2.85	2.94	2.84	2.55	2.51	2.48	2.43	2.47	2.40	
17	2.38	2.28	2.25	2.25	2.27	2.27	2.32	2.30	2.30	2.33	2.40	2.34	2.30	2.30	2.31	2.30	2.26	2.31	2.30	2.32	2.31	2.20	2.21	2.22	

18	2.38	2.27	2.26	0.00	2.15	2.66	2.70	2.66	2.41	2.51	2.50	2.45	2.48	2.32	0.00	2.12	2.60	2.60	2.67	2.59	2.55	2.39	2.30	2.23
19	3.00	3.06	2.89	2.94	2.90	2.84	2.79	2.80	2.75	2.79	2.78	2.90	2.74	2.77	2.66	2.64	2.66	2.63	2.64	2.63	2.73	2.75	2.82	2.77
20	1.20	1.21	1.21	1.19	1.18	1.16	1.18	1.19	1.18	1.18	1.20	1.18	1.19	1.19	1.17	1.19	1.18	1.21	1.22	1.24	1.24	1.24	1.23	1.22
21	2.68	2.68	2.65	2.60	2.58	2.59	2.59	2.58	2.64	2.75	2.76	2.79	2.80	2.81	2.79	2.79	2.83	2.84	2.85	2.81	2.79	2.74	2.72	2.70
22	2.11	2.11	2.12	2.07	2.03	2.03	2.03	2.04	2.08	2.10	2.06	2.05	2.08	2.10	2.10	2.09	2.09	2.09	2.10	2.15	2.07	2.08	2.10	2.10
23	1.55	1.57	1.56	1.56	1.57	1.57	1.58	1.58	1.56	1.56	1.54	1.55	1.57	1.56	1.56	1.55	1.55	1.56	1.57	1.57	1.58	1.59	1.58	1.58
24	2.76	2.77	2.75	2.73	2.71	2.71	2.77	2.88	2.87	2.91	2.87	2.81	2.92	2.91	3.01	3.02	2.96	2.96	2.92	2.95	2.96	2.95	2.93	2.95
25	1.95	1.89	1.88	1.84	1.79	1.78	1.81	1.78	1.73	1.76	1.73	1.70	1.69	1.75	1.65	1.62	1.74	1.63	1.64	1.59	1.63	1.65	1.60	1.56
26	2.81	2.87	2.93	2.91	2.90	2.88	2.81	2.75	2.68	2.67	2.70	2.70	2.69	2.72	2.71	2.69	2.70	2.75	2.76	2.81	2.87	2.89	2.78	2.77
27	3.00	3.01	2.89	2.81	2.61	2.44	2.33	2.34	2.37	2.56	2.62	2.63	2.56	2.51	2.49	2.52	2.49	2.51	2.43	2.34	2.36	2.37	2.38	2.43
28	1.12	1.12	1.13	1.12	1.14	1.13	1.13	1.11	1.12	1.12	1.12	1.12	1.14	1.13	1.13	1.11	1.12	1.15	1.14	1.14	1.15	1.16	1.16	1.16
29	2.05	2.05	2.04	2.00	2.08	2.08	2.02	2.01	2.05	2.05	2.01	2.01	2.01	2.01	2.03	2.03	2.02	2.02	2.00	1.99	2.01	2.10	2.04	2.06
30	2.65	2.59	2.60	2.62	2.62	2.63	2.66	2.64	2.61	2.61	2.62	2.62	2.62	2.61	2.63	2.64	2.62	2.63	2.63	2.67	2.59	2.57	2.56	2.57
31	2.83	2.80	2.77	2.81	2.78	2.78	2.86	2.89	2.92	3.01	3.02	3.04	3.02	3.03	3.00	2.95	2.93	2.87	2.78	2.77	2.76	2.74	2.74	2.74
32	2.49	2.49	2.51	2.48	2.52	2.56	2.56	2.62	2.63	2.66	2.68	2.69	2.67	2.69	2.68	2.67	2.69	2.68	2.69	2.69	2.69	2.68	2.65	2.65
33	2.46	1.96	2.32	1.86	2.00	2.19	1.91	2.10	2.16	1.87	1.86	1.94	1.95	1.92	1.94	1.85	2.13	2.06	2.26	2.06	2.33	2.33	2.22	2.20
34	1.35	1.34	1.34	1.34	1.34	1.34	1.36	1.36	1.36	1.36	1.35	1.35	1.35	1.33	1.34	1.33	1.32	1.32	1.33	1.32	1.31	1.31	1.32	1.32
35	2.27	2.30	2.29	2.29	2.29	2.28	2.29	2.28	2.30	2.31	2.32	2.33	2.32	2.32	2.33	2.34	2.35	2.38	2.37	2.36	2.36	2.34	2.34	2.35
36	2.71	2.71	2.70	2.71	2.71	2.71	2.71	2.71	2.71	2.71	2.70	2.70	2.69	2.69	2.70	2.70	2.70	2.70	2.70	2.69	2.69	2.68	2.67	2.66
37	2.12	2.11	2.13	2.15	2.19	2.24	2.39	2.45	2.42	2.38	2.37	2.32	2.21	2.23	2.26	2.34	2.41	2.38	2.38	2.37	2.37	2.36	2.33	2.24
38	1.35	1.34	1.34	1.34	1.35	1.34	1.36	1.35	1.36	1.36	1.37	1.37	1.36	1.36	1.35	1.34	1.35	1.35	1.35	1.35	1.35	1.35	1.34	1.34
39	2.36	2.25	2.18	2.13	2.17	2.10	2.12	2.12	2.14	2.91	2.71	2.83	2.55	2.55	2.90	2.69	2.64	2.24	2.24	1.91	1.85	1.84	1.85	1.83
40	1.84	1.79	1.77	1.76	1.76	1.72	1.71	1.68	1.68	1.68	1.66	1.65	1.64	1.62	1.61	1.59	1.60	1.60	1.61	1.62	1.63	1.61	1.61	1.61
41	2.86	2.76	2.65	2.63	2.61	2.62	2.61	2.54	2.57	2.57	2.65	2.67	2.68	2.72	2.77	2.78	2.72	2.81	2.91	2.92	2.93	2.92	2.92	2.95
42	1.44	1.43	1.43	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.41	1.42	1.42	1.42	1.42	1.42	1.43	1.43	1.43	1.43

43	2.50	2.41	2.43	2.44	2.50	2.48	2.55	2.57	2.65	2.63	2.57	2.54	2.56	2.58	2.59	2.62	2.58	2.56	2.62	2.59	2.62	2.59	2.62	2.56	2.56	2.56
44	2.92	2.78	2.97	3.09	3.12	3.11	2.88	2.79	2.98	3.12	3.12	3.03	3.04	3.07	3.00	2.86	3.00	3.02	3.02	3.00	3.10	3.10	3.02	3.08	2.84	2.84
45	3.02	2.97	2.99	3.02	3.05	3.06	3.06	2.96	2.96	2.97	2.97	2.98	2.98	2.96	3.00	2.99	3.01	3.01	3.01	3.02	3.01	3.02	3.02	3.03	3.03	3.03
46	2.58	2.52	2.49	2.50	2.56	2.57	2.58	2.48	2.45	2.37	2.36	2.48	2.48	2.50	2.52	2.50	2.48	2.58	2.62	2.58	2.73	2.84	2.84	2.84	2.89	
47	1.99	1.96	1.98	1.99	2.03	2.04	2.04	1.99	1.93	1.87	1.88	1.87	1.82	1.73	1.71	1.73	1.75	1.74	1.70	1.68	1.68	1.71	1.71	1.71	0.00	
48	1.96	1.91	1.89	1.93	1.95	2.02	2.30	2.29	2.80	2.56	2.29	2.16	2.35	2.25	2.31	2.31	2.16	2.12	2.16	2.22	2.14	2.09	2.05	2.03	2.03	
49	1.42	1.43	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.41	1.41	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.41	
50	2.30	2.31	2.30	2.35	2.31	2.25	2.27	2.32	2.36	2.27	2.27	2.27	2.27	2.26	2.35	2.33	2.35	2.35	2.22	2.27	2.30	2.38	2.39	2.39	2.39	
51	2.60	2.60	2.59	2.57	2.58	2.56	2.56	2.55	2.56	2.57	2.60	2.60	2.60	2.59	2.59	2.59	2.60	2.60	2.61	2.62	2.62	2.63	2.62	2.61	2.61	
52	2.40	2.42	2.49	2.57	2.70	2.94	2.97	2.95	2.93	2.93	2.96	2.98	2.98	2.98	3.00	3.00	3.01	3.02	2.99	2.90	2.91	2.90	2.86	2.85	2.85	
53	1.76	1.80	1.85	1.91	1.97	2.03	2.14	2.28	2.29	2.34	2.50	2.36	2.50	2.73	2.66	2.60	2.68	2.62	2.81	2.88	2.92	2.86	2.65	2.60	2.60	
54	2.51	2.50	2.60	2.59	2.62	2.58	2.60	2.62	2.59	2.54	2.51	2.31	2.30	2.30	2.27	2.27	2.23	2.14	2.06	2.09	2.12	2.12	2.12	2.11	2.11	
55	2.28	2.37	2.24	2.18	2.19	2.16	2.25	2.24	2.26	2.21	2.20	2.21	2.21	2.26	2.26	2.25	2.28	2.32	2.30	2.33	2.37	2.41	2.47	2.49	2.49	
56	2.15	2.15	2.11	2.12	2.12	2.15	2.18	2.12	1.99	1.96	1.98	2.06	2.02	2.02	2.06	2.27	2.29	2.26	2.25	2.31	2.37	2.39	2.39	2.42	2.42	
57	2.91	2.92	2.96	2.99	3.00	2.94	2.96	2.97	3.03	3.07	3.11	3.06	3.04	3.04	3.10	3.01	2.94	2.90	2.91	2.88	2.89	2.96	2.94	2.95	2.95	
58	1.33	1.31	1.30	1.28	1.29	1.29	1.34	1.31	1.32	1.26	1.28	1.29	1.30	1.29	1.30	1.29	1.29	1.29	1.31	1.30	1.30	1.27	1.29	1.29	1.29	
59	1.96	1.97	1.96	1.96	1.97	2.00	2.01	2.05	2.08	2.13	2.19	2.24	2.28	2.40	2.44	2.39	2.39	2.41	2.38	2.37	2.36	2.36	2.35	2.37	2.37	
60	2.57	2.43	2.47	2.48	2.40	2.42	2.36	2.27	2.40	2.68	2.47	2.32	2.17	2.05	2.08	2.08	2.03	1.99	1.95	1.94	1.95	1.96	1.96	1.98	1.98	
61	1.66	1.65	1.65	1.64	1.64	1.64	1.66	1.66	1.66	1.66	1.66	1.66	1.66	1.66	1.66	1.65	1.65	1.64	1.64	1.64	1.64	1.64	1.64	1.65	1.65	
62	2.13	2.19	2.17	2.18	2.07	2.11	2.14	2.10	2.01	2.04	2.07	2.04	2.05	2.05	2.07	2.31	2.27	2.31	2.31	2.47	2.51	2.45	2.34	2.36	2.36	
63	3.05	2.56	2.35	2.24	2.19	2.26	2.22	2.13	2.09	2.01	1.87	1.83	1.76	1.69	1.62	1.60	1.63	1.62	1.61	1.60	1.57	1.54	1.52	1.50	1.50	
64	2.31	2.31	2.32	2.31	2.33	2.34	2.36	2.40	2.39	2.39	2.39	2.42	2.41	2.40	2.42	2.40	2.34	2.34	2.34	2.39	2.50	2.47	2.35	2.33	2.33	
65	2.88	2.84	2.87	2.94	2.94	2.94	2.95	2.98	2.97	2.96	2.94	2.97	2.98	2.95	2.90	2.88	2.90	2.88	2.85	2.79	2.82	2.88	2.89	2.78	2.78	
66	3.03	3.06	3.05	3.06	3.03	3.06	3.09	3.04	3.00	2.98	2.93	2.83	2.87	2.92	3.02	3.09	3.11	3.10	2.95	3.00	2.96	2.88	2.79	2.74	2.74	
67	1.24	1.25	1.25	1.25	1.26	1.26	1.25	1.25	1.24	1.24	1.24	1.24	1.24	1.25	1.25	1.25	1.25	1.25	1.25	1.25	1.25	1.26	1.25	1.25	1.25	

68	2.53	2.48	2.46	2.37	2.31	2.27	2.23	2.22	2.18	2.14	2.08	2.05	1.99	1.94	1.87	1.83	1.81	1.79	1.76	1.78	1.81	1.87	1.89	1.97
69	2.08	2.11	2.12	2.08	2.05	2.06	2.05	2.04	2.00	1.98	1.97	1.99	1.96	1.93	1.91	1.91	1.87	1.87	1.87	1.91	1.94	2.02	1.98	2.01
70	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.43	1.42	1.42	1.43	1.43	1.42	1.43	1.43	1.43	1.43
71	2.58	2.53	2.51	2.56	2.59	2.59	2.62	2.63	2.68	2.68	2.87	2.89	2.96	2.98	2.94	2.92	2.83	2.81	2.81	2.81	2.76	2.74	2.73	2.75
72	2.65	2.64	2.65	2.64	2.74	2.83	2.88	2.92	2.95	2.90	2.95	2.85	2.91	2.83	2.83	2.83	2.95	3.02	3.04	2.96	3.07	3.01	3.02	3.03
73	2.45	2.45	2.59	2.79	2.89	2.89	2.83	2.82	2.57	2.25	2.27	2.49	2.41	2.32	2.50	2.87	2.97	2.66	2.67	2.58	2.57	2.53	2.51	2.42
74	2.19	2.20	2.18	2.21	2.19	2.18	2.11	2.09	2.06	2.12	2.09	2.10	2.13	2.11	2.08	2.08	2.07	2.03	2.04	2.03	2.02	2.06	2.02	1.98
75	2.64	2.72	2.72	2.78	2.84	2.98	3.02	2.92	2.72	2.75	3.01	3.00	3.05	3.01	2.91	2.88	2.85	2.88	2.87	2.87	2.89	2.91	2.85	2.81
76	1.97	1.97	2.00	1.98	1.98	1.99	2.00	1.95	1.98	1.98	1.98	2.00	1.99	1.99	1.99	1.97	1.99	2.00	1.94	1.92	1.89	1.91	1.90	1.90
77	1.64	1.64	1.63	1.64	1.64	1.63	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.63	1.67	1.86	1.64	1.64	1.64	1.64	1.64	1.64	1.65	1.64
78	2.99	3.01	3.01	3.01	3.01	3.00	2.96	2.95	2.89	2.96	2.95	2.96	2.95	2.93	2.93	2.94	2.94	2.96	2.99	2.98	3.04	2.95	3.00	3.05
79	1.91	1.95	1.94	1.90	1.91	1.92	1.92	1.90	1.90	1.89	1.89	1.88	1.88	1.90	1.90	1.86	1.86	1.86	1.87	1.88	1.87	1.90	1.90	1.88
80	2.31	2.29	2.36	2.37	2.32	2.28	2.27	2.24	2.25	2.43	2.53	2.54	2.46	2.36	2.35	2.36	2.45	2.47	2.47	2.46	2.41	2.44	2.42	2.39
81	2.41	2.38	2.36	2.33	2.33	2.31	2.28	2.28	2.27	2.27	2.26	2.28	2.26	2.29	2.29	2.25	2.29	2.32	2.34	2.35	2.34	2.29	2.29	2.26
82	2.56	2.55	2.47	2.47	2.50	2.56	2.56	2.59	2.68	2.63	2.66	2.64	2.69	2.69	2.76	2.77	2.75	2.86	2.80	3.00	3.02	2.90	2.80	2.71
83	3.02	3.00	2.98	2.96	2.97	2.95	2.94	2.94	2.95	2.98	3.00	3.00	3.01	3.01	3.00	3.02	3.05	3.05	3.05	3.03	3.05	3.02	3.01	3.07
84	2.42	2.50	2.47	2.52	2.50	2.48	2.42	2.35	2.31	2.29	2.29	2.26	2.32	2.32	2.32	2.39	2.48	2.48	2.40	2.47	2.45	2.44	2.37	2.33
85	1.98	2.05	2.01	2.02	2.01	2.04	2.03	2.06	2.20	2.35	2.13	2.31	2.25	2.37	2.16	2.20	2.23	2.40	2.72	2.83	2.37	3.11	2.89	2.92
86	2.38	2.61	2.75	2.30	2.45	2.13	2.34	2.83	2.61	2.28	2.20	2.23	2.27	2.73	3.11	2.06	2.09	2.05	1.92	2.02	1.96	1.97	2.01	1.95
87	2.70	2.69	2.65	2.67	2.62	2.57	2.35	2.37	2.41	2.43	2.62	2.63	2.59	2.62	2.57	2.43	2.52	2.54	2.67	2.62	2.41	2.39	2.14	2.37
88	2.30	2.27	2.27	2.24	2.26	2.24	2.26	2.32	2.38	2.39	2.40	2.40	2.41	2.42	2.43	2.42	2.42	2.40	2.40	2.39	2.40	2.40	2.40	2.41
89	1.65	1.66	1.65	1.63	1.63	1.63	1.63	1.62	1.63	1.63	1.64	1.64	1.65	1.67	1.67	1.68	1.69	1.69	1.67	1.67	1.68	1.69	1.70	1.68
90	2.18	2.19	2.19	2.20	2.20	2.20	2.22	2.22	2.22	2.21	2.22	2.22	2.22	2.22	2.22	2.21	2.22	2.22	2.22	2.22	2.22	2.22	2.21	2.20
91	2.55	2.53	2.52	2.53	2.56	2.58	2.57	2.56	2.57	2.58	2.57	2.55	2.55	2.60	2.61	2.60	2.60	2.58	2.58	2.57	2.52	2.53	2.53	2.53
92	2.12	2.16	2.19	2.33	2.34	2.48	2.54	2.84	3.02	2.95	3.10	3.07	2.76	2.59	2.52	2.50	2.23	2.56	2.42	2.45	2.37	2.17	2.23	2.28

93	2.21	2.09	2.09	2.07	2.08	2.01	2.00	2.03	2.01	2.02	2.04	2.00	2.05	2.09	2.04	2.11	2.08	2.05	2.10	2.08	2.08	2.09	2.10	2.05
94	1.40	1.39	1.41	1.42	1.42	1.42	1.42	1.42	1.42	1.41	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42	1.42
95	1.52	1.51	1.51	1.50	1.50	1.51	1.52	1.53	1.54	1.54	1.55	1.55	1.54	1.54	1.54	1.55	1.55	1.54	1.54	1.54	1.53	1.53	1.53	1.53
96	2.35	2.34	2.34	2.34	2.33	2.35	2.37	2.37	2.37	2.37	2.37	2.38	2.39	2.39	2.39	2.39	2.39	2.38	2.38	2.38	2.38	2.38	2.38	2.37
97	2.95	2.83	2.84	2.94	3.04	2.95	2.92	2.82	2.88	2.84	2.81	2.76	2.79	2.95	2.95	2.99	2.87	2.84	2.87	2.93	2.96	3.02	3.03	3.10
98	2.54	2.55	2.48	2.61	2.56	2.57	2.56	2.54	2.63	2.43	2.40	2.38	2.40	2.46	2.41	2.41	2.34	2.36	2.36	2.36	2.31	2.31	2.32	2.32
99	2.72	2.72	2.79	2.77	2.74	0.00	2.72	2.60	2.53	2.55	2.70	2.57	2.50	2.55	2.26	2.22	0.00	1.75	1.78	1.83	1.83	1.78	1.73	1.73
100	2.58	2.59	2.62	2.62	2.65	2.63	2.61	2.59	2.52	2.55	2.53	2.59	2.74	2.72	2.71	2.74	2.78	2.77	2.78	2.84	2.85	2.83	2.84	2.84



## ประวัติผู้เขียน

ชื่อ	ปฏิพัทธ์ สิทธิประเสริฐ
วันเกิด	10 มีนาคม 2522
สถานที่เกิด	ยโสธร
สถานที่อยู่ปัจจุบัน	644/3 หมู่14 ต.เชียงเคี่ยน, อ.เมือง จ.สกลนคร 47000
ตำแหน่งหน้าที่การงาน	นักพัฒนาซอฟต์แวร์
สถานที่ทำงานปัจจุบัน	บริษัท พายซอฟต์แวร์ จำกัด
ประวัติการศึกษา	พ.ศ. 2545 วิศวกรรมศาสตรบัณฑิต (วศ.บ.) วิศวกรรมไฟฟ้า มหาวิทยาลัยอุบลราชธานี พ.ศ. 2566 วิศวกรรมศาสตรมหาบัณฑิต (วศ.ม.) วิศวกรรมไฟฟ้าและคอมพิวเตอร์ มหาวิทยาลัยมหาสารคาม

พูนุ่ ปณุ่ ทีโตะ ชีเว