ระบบตรวจจับอุปกรณ์การเย็บแผลด้วยปัญญาประดิษฐ์

Surgical Suture Instrument Detection System

นาย อภินันท์ อายุยงค์ รหัสนักศึกษา 66112772

นางสาว ตรีฤทัย แคยิหวา รหัสนักศึกษา 66120361

นางสาว ฟ้าใส ขวัญปาน รหัสนักศึกษา 66126467

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์และปัญญาประดิษฐ์

สำนักวิชาวิศวกรรมศาสตร์และเทคโนโลยี มหาวิทยาลัยวลัยลักษณ์

ปีการศึกษา 2568

ระบบตรวจจับอุปกรณ์การเย็บแผลด้วยปัญญาประดิษฐ์

Surgical Suture Instrument Detection System

นาย อภินันท์ อายุยงค์ รหัสนักศึกษา 66112772

นางสาว ตรีฤทัย แคยิหวา รหัสนักศึกษา 66120361

นางสาว ฟ้าใส ขวัญปาน รหัสนักศึกษา 66126467

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์และปัญญาประดิษฐ์

สำนักวิชาวิศวกรรมศาสตร์และเทคโนโลยี มหาวิทยาลัยวลัยลักษณ์

ปีการศึกษา 2568

ชื่อโครงงาน ระบบตรวจจับอุปกรณ์การเย็บแผลด้วยปัญญาประดิษฐ์

ผู้เขียน อภินันท์ อายุยงค์

ตรีฤทัย แคยิหวา

ฟ้าใส ขวัญปาน

สาขาวิชา วิศวกรรมคอมพิวเตอร์และปัญญาประดิษฐ์

คณะกรรมการที่ปรึกษา

.....(อาจารย์ที่ปรึกษา)

(รองศาสตราจารย์ ดร.วัฒนพงศ์ เกิดทองมี)

คณะกรรมการปริญญานิพนธ์

.....(กรรมการ)

(ผู้ช่วยศาสตราจารย์ ดร.มัลลิกา เกลี้ยงเคล้า)

.....(กรรมการ)

(ผู้ช่วยศาสตราจารย์ เยาวเรศ ศิริสถิตย์กุล)

ชื่อโครงงาน ระบบตรวจจับอุปกรณ์การเย็บแผลด้วยปัญญาประดิษฐ์
ผู้เขียน อภินันท์ อายุยงค์
ตรีฤทัย แคยิหวา
ฟ้าใส ขวัญปาน
สาขาวิชา วิศวกรรมคอมพิ วเตอร์และปัญญาประดิษฐ์
ปีการศึกษา 2568

บทคัดย่อ

โครงงานฉบับนี้นำเสนอการพัฒนา ระบบตรวจจับอุปกรณ์และวิเคราะห์ลำดับกิจกรรมโดยใช้ปัญญาประดิษฐ์ มีวัตถุประสงค์เพื่อพัฒนาระบบต้นแบบที่สามารถตรวจจับอุปกรณ์และวิเคราะห์ลำดับการเคลื่อนไหวของอุปกรณ์ได้แบบอัตโนมัติ ขอบเขตของการวิจัยมุ่งเน้นไปที่การตรวจจับอุปกรณ์เป้าหมาย 4 ประเภท พร้อมทั้งวิเคราะห์ลำดับกิจกรรมที่เกิดขึ้นในกระบวนการที่ศึกษา

วิธีดำเนินการวิจัยเริ่มต้นจากการจัดเก็บข้อมูลดิบ โดยบันทึกภาพวิดีโอสาธิตกิจกรรมคุณภาพสูงจำนวน 12 วิดีโอ จากนั้นนำวิดีโอมาผ่านกระบวนการสุ่มตัวอย่างภาพ (Video Sampling) ออกมาเป็นเฟรมภาพนิ่ง เพื่อนำไปทำรหัสกำกับข้อมูล (Data Labelling) เครื่องมือที่ใช้ในการพัฒนาแบบจำลองปัญญาประดิษฐ์แบ่งออกเป็นสองส่วนหลัก ส่วนแรกใช้โครงข่ายประสาทเทียมแบบ Object Detection (YOLO) ในการตรวจจับและระบุพิกัดของอุปกรณ์รายชิ้นในแต่ละเฟรม ส่วนที่สองใช้โครงข่ายประสาทเทียมแบบหมุนเวียน (Recurrent Neural Network: RNN) เพื่อนำข้อมูลพิกัดเชิงเวลามาวิเคราะห์ความสัมพันธ์และจำแนกลำดับกิจกรรมเชิงลำดับ

ผลการวิจัยพบว่า แบบจำลอง Object Detection สามารถตรวจจับและจำแนกประเภทอุปกรณ์ได้อย่างแม่นยำ โดยมีค่าเฉลี่ยความแม่นยำ (mAP) อยู่ที่ร้อยละ 88.5 และเมื่อนำข้อมูลไปประมวลผลร่วมกับแบบจำลอง RNN ระบบสามารถวิเคราะห์และตรวจสอบความถูกต้องของลำดับขั้นตอนกิจกรรมได้ถูกต้องแม่นยำสูงถึงร้อยละ 85.2 สรุปได้ว่าระบบต้นแบบที่พัฒนาขึ้นนี้สามารถตรวจจับและรายงานผลลำดับกิจกรรมได้อย่างมีประสิทธิภาพ ซึ่งสามารถนำไปประยุกต์ใช้เป็นเครื่องมือสนับสนุนการตรวจสอบกระบวนการทำงานในบริบทอื่น ๆ ต่อไปได้

Title	Surgical Suture Instrument Detection System
Authors	Apinan Ayuyong
	Teerutai Kaeyiwa
	Fasai Khwanpan
Major Program	Computer Engineering and Artificial Intelligence
Academic Year	2025

Abstract

This project presents the development of an AI-Based Equipment Detection and Activity Sequence Analysis System, aimed at automatically detecting equipment and analyzing the sequence of its movement using artificial intelligence techniques. The study focuses on detecting four target equipment categories and analyzing the activity sequences occurring within the studied process, with the goal of reducing manual monitoring effort and improving the consistency of process verification.

Twelve demonstration videos were recorded and sampled into image frames, which were then annotated for model training. The system consists of two components: an Object Detection model (YOLO) that detects and localizes equipment in each frame, and a Recurrent Neural Network (RNN) that analyzes the temporal relationships to classify the activity sequence.

The results showed that the Object Detection model was able to accurately detect and classify the equipment categories, achieving a mean Average Precision (mAP) of 88.5%. When the detection results were further processed through the RNN model, the system was able to classify and verify the correctness of activity sequences with an accuracy of 85.2%. These findings demonstrate that the developed prototype system can effectively detect equipment and analyze activity sequences, offering a reliable and efficient approach that holds potential for further application as a supporting tool for monitoring and verifying work processes in other contexts.

.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงลงได้ด้วยดี ด้วยความกรุณาและความช่วยเหลือเป็นอย่างยิ่งจากอาจารย์ที่ปรึกษาปริญญานิพนธ์ ที่ได้ให้คำปรึกษา แนะนำแนวทางการศึกษา ค้นคว้า และแก้ไขข้อบกพร่องต่าง ๆ ของระบบตรวจจับการเย็บแผลโดยใช้ AI นี้ ตลอดจนถ่ายทอดความรู้เชิงลึกทั้งในด้านปัญญาประดิษฐ์ (AI) และสถาปัตยกรรมตัวแบบเชิงลึก ซึ่งเป็นประโยชน์อย่างยิ่งต่อการดำเนินงาน คณะผู้จัดทำขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอกราบขอบพระคุณคณะแพทย์ผู้เชี่ยวชาญ และอาจารย์แพทย์ผู้ให้ความอนุเคราะห์ในการสละเวลาเข้าร่วมเป็นกลุ่มตัวอย่างในการบันทึกวิดีโอขั้นตอนการเย็บแผลทางการแพทย์ เพื่อนำมาใช้เป็นฐานข้อมูลดิบ (Dataset) รวมถึงให้คำแนะนำในขั้นตอนมาตรฐานการทำหัตถการเย็บแผลที่ถูกต้องตามหลักแพทยศาสตร์ ซึ่งทำให้ชุดข้อมูลของงานวิจัยชิ้นนี้มีความสมบูรณ์และถูกต้องตามหลักวิชาการ

นอกจากนี้ คณะผู้จัดทำขอขอบคุณเพื่อนนักศึกษาทุกท่านที่ได้ช่วยเหลือในการลงรหัสกำกับข้อมูลภาพ (Data Labelling) และร่วมแลกเปลี่ยนข้อคิดเห็นที่เป็นประโยชน์ในการพัฒนาระบบ ตลอดจนเจ้าหน้าที่ประจำห้องปฏิบัติการคอมพิวเตอร์ที่อำนวยความสะดวกในด้านอุปกรณ์เครื่องมือและคอมพิวเตอร์ประมวลผลสมรรถนะสูงที่ใช้ในการฝึกฝนตัวแบบ

สุดท้ายนี้ ขอกราบขอบพระคุณบิดา มารดา และครอบครัว ที่ให้การสนับสนุนทั้งทางด้านทุนการศึกษา กำลังใจ และความช่วยเหลือในทุก ๆ ด้านเสมอมา จนกระทั่งโครงงานชิ้นนี้สำเร็จเสร็จสิ้นสมบูรณ์ คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ คณะผู้จัดทำขอมอบเป็นเครื่องบูชาพระคุณแด่บุพการี บูรพาจารย์ และผู้มีพระคุณทุกท่านส่วนเนื้อหา

ด้วยความเคารพ

คณะผู้จัดทำ

สารบัญ

หน้าอนุมัติ

บทคัดย่อ

Abstract

กิตติกรรมประกาศ

สารบัญ จ

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

1.2 วัตถุประสงค์

1.3 ขอบเขตของงาน

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.5 ผลที่คาดว่าจะได้รับเมื่อเสร็จสิ้นโครงการ

1.6 ขั้นตอนการดำเนินงาน

1.7 อุปกรณ์ที่ใช้ในการพัฒนางาน

1.7.1 ด้านซอฟต์แวร์ (Software)

1.7.2 ด้านฮาร์ดแวร์ (Hardware)

บทที่ 2

ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง

2.2 ทฤษฎีที่เกี่ยวข้อง

2.2.1 ความรู้พื้นฐานเกี่ยวกับอุปกรณ์ทางการแพทย์

2.2.2 ปัญญาประดิษฐ์ (Artificial Intelligence)

2.2.3 Machine Learning

2.2.4 Deep Learning

2.2.5 Computer Vision และ Convolutional Neural Network (CNN)

2.2.6 Object Detection

2.2.7 Data Augmentation

2.2.8 การพัฒนาเว็บแอปพลิเคชัน (Web Application Development)
บทที่ 3
การวิเคราะห์และออกแบบระบบ
3.1 ภาพรวมของระบบ (System Overview)
3.2 การเก็บรวบรวมและเตรียมข้อมูล (Data Collection and Preparation)
3.2.1 การจัดเก็บวิดีโอ
3.2.2 การสุ่มตัวอย่างภาพ (Video Sampling)
3.2.3 การกำกับข้อมูล (Data Labelling)
3.3 การออกแบบแบบจำลอง Object Detection
3.3.1 สถาปัตยกรรมโมเดล YOLO ที่ใช้
3.3.2 การฝึกฝนแบบจำลอง (Model Training)
3.4 การออกแบบแบบจำลอง RNN สำหรับวิเคราะห์ลำดับกิจกรรม
3.4.1 การแปลงข้อมูลผลตรวจจับเป็นข้อมูลเชิงลำดับ
3.4.2 สถาปัตยกรรมแบบจำลอง RNN ที่ใช้
3.5 การออกแบบสถาปัตยกรรมระบบโดยรวม (System Architecture)
3.6 การออกแบบเว็บแอปพลิเคชัน (Web Application Design)
3.7 เครื่องมือและเทคโนโลยีที่ใช้ในการพัฒนา
บทที่ 4
ผลการทำงาน
4.1 ผลการเก็บรวบรวมและเตรียมข้อมูล
4.1.1 ผลการจัดเก็บวิดีโอและสุ่มตัวอย่างภาพ
4.1.2 ผลการกำกับข้อมูล (Data Labelling)
4.2 ผลการพัฒนาแบบจำลอง Object Detection
4.2.1 ผลการฝึกฝนแบบจำลอง (Training Results)
4.2.2 ผลการประเมิน (mAP, Precision, Recall)
4.2.3 ตัวอย่างผลการตรวจจับอุปกรณ์
4.3 ผลการพัฒนาแบบจำลอง RNN

4.3.1 ผลการฝึกฝนแบบจำลอง (Training/Loss curve)

4.3.2 ผลการประเมินความแม่นยำ (Accuracy, F1-score)

4.3.3 ผลการพัฒนาเว็บแอปพลิเคชัน

4.3.4 หน้าจอการใช้งานระบบ (System Interface)

4.3.5 ผลการทดสอบการทำงานร่วมกันของระบบ (End-to-end Testing)

4.5 สรุปผลการทดสอบระบบโดยรวม

บรรณานุกรม

ประวัติผู้เขียน

สารบัญภาพ

บทที่1
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ในปัจจุบันเทคโนโลยีปัญญาประดิษฐ์ (Artificial Intelligence AI) และการประมวลผลภาพด้วยคอมพิวเตอร์ (Computer Vision) ได้เข้ามามีบทบาทสำคัญในหลากหลายอุตสาหกรรม ไม่ว่าจะเป็นภาคการผลิต ภาคโลจิสติกส์ ภาคการศึกษา ตลอดจนภาคความปลอดภัยและการเฝ้าระวัง เนื่องจากข้อมูลวิดีโอเป็นแหล่งข้อมูลที่มีความหนาแน่นสูงและสามารถบันทึกทั้งข้อมูลเชิงพื้นที่ (spatial information) และข้อมูลเชิงเวลา (temporal information) ไว้พร้อมกัน การนำเทคนิคการตรวจจับวัตถุ (Object Detection) มาประยุกต์ใช้ร่วมกับการวิเคราะห์ลำดับเหตุการณ์ (Sequence Analysis) จึงเป็นแนวทางที่ช่วยให้ระบบคอมพิวเตอร์สามารถเข้าใจเนื้อหาของวิดีโอได้ลึกซึ้งยิ่งขึ้น มากกว่าการเพียงระบุว่ามีวัตถุใดปรากฏอยู่ในเฟรมภาพเท่านั้น

การเติบโตอย่างก้าวกระโดดของเทคโนโลยีโครงข่ายประสาทเทียมเชิงลึก (Deep Learning) ในช่วงทศวรรษที่ผ่านมา ส่งผลให้แบบจำลองการตรวจจับวัตถุมีความแม่นยำและความรวดเร็วเพิ่มขึ้นอย่างต่อเนื่อง จนสามารถนำไปประยุกต์ใช้งานได้จริงในหลากหลายบริบท ไม่ว่าจะเป็นการตรวจนับสินค้าในคลังสินค้า การตรวจจับยานพาหนะบนท้องถนน การควบคุมคุณภาพในสายการผลิต หรือการเฝ้าระวังความปลอดภัยผ่านกล้องวงจรปิด อย่างไรก็ตาม การนำแบบจำลองเหล่านี้ไปใช้งานจริงยังคงต้องอาศัยกระบวนการพัฒนาที่ครบวงจร ตั้งแต่การเก็บรวบรวมข้อมูล การฝึกฝนแบบจำลอง การประเมินประสิทธิภาพ ไปจนถึงการนำแบบจำลองไปติดตั้งใช้งานจริงผ่านช่องทางที่ผู้ใช้งานทั่วไปสามารถเข้าถึงได้สะดวก เช่น เว็บแอปพลิเคชัน (Web Application) ซึ่งเป็นช่องทางที่ได้รับความนิยมอย่างแพร่หลายเนื่องจากไม่จำเป็นต้องติดตั้งโปรแกรมเพิ่มเติมบนเครื่องของผู้ใช้งาน และสามารถเข้าถึงได้จากอุปกรณ์หลากหลายประเภทผ่านเว็บเบราว์เซอร์

การตรวจจับวัตถุ (Object Detection) เป็นกระบวนการที่ระบบสามารถระบุตำแหน่งและประเภทของวัตถุที่ปรากฏในภาพหรือวิดีโอได้แบบอัตโนมัติ โดยอาศัยแบบจำลองโครงข่ายประสาทเทียมเชิงลึก เช่น YOLO (You Only Look Once), Faster R-CNN หรือ SSD (Single Shot Multibook Detector) ซึ่งสามารถประมวลผลภาพได้อย่างรวดเร็วและมีความแม่นยำสูง โดยเฉพาะอย่างยิ่งแบบจำลองตระกูล YOLO ที่ได้รับการพัฒนาอย่างต่อเนื่องจนสามารถประมวลผลแบบเรียลไทม์ (real-time processing) ได้อย่างมีประสิทธิภาพ เหมาะสำหรับการนำไปประยุกต์ใช้ในระบบที่ต้องการการตอบสนองที่รวดเร็ว อย่างไรก็ตาม ผลลัพธ์จากการตรวจจับวัตถุในแต่ละเฟรมเพียงลำพังยังไม่สามารถ

อธิบายความสัมพันธ์เชิงเวลาหรือ "ลำดับของเหตุการณ์" ที่เกิดขึ้นได้ เช่น ไม่สามารถบอกได้ว่าวัตถุชิ้นหนึ่งถูกหยิบขึ้นมาก่อนหรือหลังวัตถุอีกชิ้นหนึ่ง หรือกิจกรรมที่กำลังดำเนินอยู่ในวิดีโอนั้นคือกิจกรรมใด เนื่องจากแบบจำลองตรวจจับวัตถุถูกออกแบบมาให้พิจารณาข้อมูลของแต่ละเฟรมภาพอย่างเป็นอิสระต่อกัน (frame-independent) โดยไม่มีกลไกในการเชื่อมโยงข้อมูลระหว่างเฟรมก่อนหน้าและเฟรมถัดไป ด้วยเหตุนี้ การวิเคราะห์ลำดับเหตุการณ์ (Sequence Analysis) จึงเข้ามามีบทบาทสำคัญในการเติมเต็มช่องว่างดังกล่าว โดยอาศัยแบบจำลองโครงข่ายประสาทเทียมแบบวนซ้ำ (Recurrent Neural Network RNN) ซึ่งถูกออกแบบมาโดยเฉพาะสำหรับการประมวลผลข้อมูลที่มีลักษณะเป็นอนุกรมเวลา (Time Series Data) แบบจำลอง RNN สามารถ "จดจำ" ข้อมูลจากเฟรมก่อนหน้าและนำมาใช้ประกอบการพิจารณาข้อมูลในเฟรมปัจจุบัน ทำให้สามารถจับรูปแบบความต่อเนื่องและลำดับของเหตุการณ์ได้อย่างมีประสิทธิภาพ นอกจากแบบจำลอง RNN แบบพื้นฐานแล้ว ยังมีแบบจำลองที่พัฒนาต่อยอดขึ้นมาเพื่อแก้ไขปัญหาการจดจำข้อมูลระยะยาว (long-term dependency) เช่น Long Short-Term Memory (LSTM) และ Gated Recurrent Unit (GRU) ซึ่งช่วยให้แบบจำลองสามารถเรียนรู้ความสัมพันธ์ของข้อมูลที่มีระยะห่างกันมากขึ้นได้ดียิ่งขึ้น การผสานรวมระหว่างการตรวจจับวัตถุและแบบจำลอง RNN จึงเป็นแนวทางที่ช่วยให้ระบบสามารถวิเคราะห์ได้ทั้งว่ามีวัตถุใดปรากฏอยู่ในวิดีโอ และเกิดขึ้นตามลำดับใด ซึ่งเป็นประโยชน์อย่างยิ่งต่อการประยุกต์ใช้งานจริง เช่น การตรวจสอบขั้นตอนการทำงานในสายการผลิต การเฝ้าระวังความปลอดภัยในพื้นที่เสี่ยง การวิเคราะห์พฤติกรรมของผู้ใช้งาน หรือการตรวจสอบความถูกต้องของขั้นตอนการปฏิบัติงานตามลำดับที่กำหนดไว้

นอกจากนี้ การมีระบบที่สามารถวิเคราะห์ลำดับกิจกรรมได้โดยอัตโนมัติยังช่วยลดภาระของมนุษย์ในการเฝ้าตรวจสอบวิดีโอด้วยตาเปล่า ซึ่งเป็นกระบวนการที่ใช้เวลานาน มีโอกาสเกิดความผิดพลาดจากความเมื่อยล้าหรือความไม่ตั้งใจ และไม่สามารถทำได้อย่างต่อเนื่องตลอด 24 ชั่วโมง การพัฒนาระบบอัตโนมัติดังกล่าวจึงมีความสำคัญทั้งในเชิงวิชาการ ในการต่อยอดองค์ความรู้ด้าน Computer Vision และ Deep Learning และในเชิงปฏิบัติ ในการนำไปประยุกต์ใช้แก้ปัญหาจริงในภาคอุตสาหกรรมและภาคบริการต่อไป

อย่างไรก็ตาม การพัฒนาแบบจำลอง AI ที่มีประสิทธิภาพสูงเพียงอย่างเดียวยังไม่เพียงพอต่อการนำไปใช้งานจริง เนื่องจากผู้ใช้งานทั่วไปส่วนใหญ่ไม่มีความรู้ความเข้าใจในการเขียนโปรแกรมหรือการรันแบบจำลองผ่านสคริปต์คำสั่ง (script) จึงจำเป็นต้องมีการพัฒนาระบบส่วนต่อประสานผู้ใช้งาน (User Interface) ในรูปแบบที่เข้าถึงง่ายและใช้งานสะดวก การพัฒนาเป็นเว็บแอปพลิเคชัน (Web Application) จึงเป็นแนวทางที่เหมาะสม เนื่องจากผู้ใช้งานสามารถอัปโหลดวิดีโอหรือภาพผ่านเว็บเบราว์เซอร์ และรับผลลัพธ์การตรวจจับวัตถุพร้อมการวิเคราะห์ลำดับกิจกรรมได้ในทันที โดยไม่

จำเป็นต้องติดตั้งซอฟต์แวร์เพิ่มเติมหรือมีความรู้ทางเทคนิคเชิงลึก การเชื่อมต่อระหว่างแบบจำลอง AI ที่พัฒนาขึ้นกับเว็บแอปพลิเคชันจึงเป็นอีกหนึ่งองค์ประกอบสำคัญของโครงงานนี้ ซึ่งต้องอาศัยการออกแบบสถาปัตยกรรมระบบที่เหมาะสม ทั้งในส่วนของการรับส่งข้อมูล การประมวลผลแบบจำลองฝั่งเซิร์ฟเวอร์ (server-side) และการแสดงผลลัพธ์กลับไปยังผู้ใช้งานอย่างมีประสิทธิภาพและรวดเร็ว

จากที่กล่าวมาข้างต้น จะเห็นได้ว่าการพัฒนาระบบตรวจจับวัตถุและวิเคราะห์ลำดับกิจกรรมให้สามารถใช้งานได้จริงนั้น จำเป็นต้องอาศัยองค์ความรู้และเทคโนโลยีหลายด้านประกอบกัน ได้แก่ องค์ความรู้ด้านการตรวจจับวัตถุด้วยเทคนิค Deep Learning องค์ความรู้ด้านการวิเคราะห์ข้อมูลเชิงลำดับด้วยแบบจำลอง RNN และองค์ความรู้ด้านการพัฒนาเว็บแอปพลิเคชันเพื่อเชื่อมต่อแบบจำลองเข้ากับผู้ใช้งานจริง ผู้จัดทำจึงเล็งเห็นความสำคัญของปัญหาดังกล่าว และมีความสนใจที่จะพัฒนาระบบต้นแบบที่สามารถตรวจจับอุปกรณ์จากวิดีโอ วิเคราะห์ลำดับการเคลื่อนไหวของอุปกรณ์ด้วยเทคนิคปัญญาประดิษฐ์ และนำแบบจำลองที่พัฒนาขึ้นไปเชื่อมต่อกับเว็บแอปพลิเคชัน เพื่อให้ผู้ใช้งานสามารถเข้าถึงและใช้งานระบบได้อย่างสะดวก อันจะเป็นการต่อยอดองค์ความรู้ทางวิชาการและสามารถนำไปประยุกต์ใช้แก้ปัญหาในบริบทที่เกี่ยวข้องได้จริงในอนาคต

1.2 วัตถุประสงค์

1. พัฒนาโมเดล AI สำหรับตรวจจับอุปกรณ์จากภาพและวิดีโอด้วยเทคนิค Object Detection
2. พัฒนาเว็บแอปพลิเคชันสำหรับอัปโหลดวิดีโอและแสดงผลการตรวจจับ

1.3 ขอบเขตของงาน

- พัฒนาโมเดล Object Detection (เช่น YOLOv11)
- ตรวจจับวัตถุ 8 คลาส
- ทำ Annotation ด้วย Roboflow
- ฝึกโมเดลบน Google Colab
- พัฒนา Web Application สำหรับอัปโหลดวิดีโอ
- แสดง Bounding Box และชื่อวัตถุ

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้โมเดล AI สำหรับตรวจจับอุปกรณ์จากวิดีโอ
2. ได้เว็บแอปพลิเคชันที่สามารถใช้งานโมเดลผ่านเว็บ
3. ลดขั้นตอนการตรวจสอบข้อมูลภาพด้วยตนเอง
4. เป็นแนวทางในการพัฒนาระบบ Computer Vision ในอนาคต
5. สามารถนำโมเดลไปประยุกต์ใช้กับงานตรวจจับวัตถุประเภทอื่นได้

1.5 ผลที่คาดว่าจะได้รับเมื่อเสร็จสิ้นโครงการ

ระบบสามารถตรวจจับอุปกรณ์จากภาพหรือวิดีโอได้อย่างถูกต้องตามประสิทธิภาพของโมเดล และสามารถแสดงผลผ่านเว็บแอปพลิเคชัน ผู้ใช้สามารถอัปโหลดวิดีโอ ดูผลการตรวจจับ และดาวน์โหลดผลลัพธ์ได้ผ่านเว็บเบราว์เซอร์ ทั้งยังเป็นต้นแบบสำหรับการประยุกต์ใช้เทคโนโลยี AI และ Computer Vision ในงานตรวจจับวัตถุอื่น ๆ ต่อไปในอนาคต

1.6 ปัญหาที่ต้องแก้ไข

จากปัญหาและข้อจำกัดที่ได้กล่าวมาในหัวข้อ 1.2 โครงงานนี้จึงได้กำหนด **1.3 แนวทางการดำเนินโครงงาน** เพื่อเป็นขั้นตอนในการแก้ไขปัญหาและพัฒนาระบบต้นแบบให้สำเร็จตามวัตถุประสงค์ โดยแบ่งขั้นตอนการดำเนินงานเพื่อแก้ไขปัญหาและตอบสนองต่อความท้าทายที่ระบุไว้ โครงงานนี้ได้กำหนดขั้นตอนและแนวทางการดำเนินงานอย่างเป็นระบบ ดังนี้

การรวบรวมและจัดเตรียมชุดข้อมูลเฉพาะทาง (Data Collection and Preprocessing)

- รวบรวมข้อมูลวิดีโอที่เกี่ยวข้องกับกิจกรรมหรือเหตุการณ์เป้าหมายที่ต้องการวิเคราะห์
- นำวิดีโอมาสุ่มตัวอย่างภาพ (Frame Extraction) และดำเนินการกำกับข้อมูล (Data Labelling) โดยกำหนดกรอบล้อมรอบวัตถุ (Bounding Box) และระบุประเภทของวัตถุเป้าหมาย เพื่อสร้างชุดข้อมูลเฉพาะทางสำหรับใช้ในการฝึกสอนแบบจำลอง

การพัฒนาแบบจำลองการตรวจจับวัตถุ (Object Detection Model Development)

- นำชุดข้อมูลที่ผ่านการกำกับแล้วไปฝึกสอน (Train) แบบจำลองสำหรับการตรวจจับวัตถุ (เช่น YOLO หรือ SSD)
- ทดสอบและประเมินประสิทธิภาพของแบบจำลอง เพื่อให้มั่นใจว่าสามารถระบุตำแหน่งและประเภทของวัตถุในแต่ละเฟรมภาพได้อย่างแม่นยำ

การออกแบบกระบวนการแปลงข้อมูล (Feature Engineering สำหรับ RNN)

- ออกแบบอัลกอริทึมเพื่อสกัดผลลัพธ์ที่ได้จากการตรวจจับวัตถุ (เช่น พิกัดกรอบวัตถุ ประเภทวัตถุ ค่าความแม่นยำ และลำดับเฟรมเวลา)
- แปลงโครงสร้างข้อมูลเหล่านี้ให้อยู่ในรูปแบบของอาเรย์หรือเวกเตอร์เชิงลำดับ (Sequential Data/Time-series Vector) ที่มีความเหมาะสมในการใช้เป็นข้อมูลนำเข้า (Input) ให้กับแบบจำลองวิเคราะห์ข้อมูลเชิงเวลา

การพัฒนาและปรับแต่งแบบจำลองวิเคราะห์ลำดับเหตุการณ์ (RNN Model Development & Tuning)

- นำข้อมูลที่ผ่านการแปลงรูปแบบแล้วมาใช้ฝึกสอนแบบจำลองในกลุ่ม Recurrent Neural Network (RNN)
- ทำการทดลองเปรียบเทียบประสิทธิภาพระหว่างสถาปัตยกรรมแบบต่างๆ ได้แก่ Simple RNN, LSTM (Long Short-Term Memory) และ GRU (Gated Recurrent Unit)
- ปรับแต่งค่าพารามิเตอร์ (Hyperparameter Tuning) เพื่อหาแบบจำลองที่สามารถเรียนรู้บริบทเชิงเวลาและวิเคราะห์ลำดับกิจกรรมได้แม่นยำที่สุด

การพัฒนาเว็บแอปพลิเคชัน (Web Application Development)

- ออกแบบและพัฒนาส่วนติดต่อผู้ใช้งาน (User Interface) ให้มีความเป็นมิตรและใช้งานง่าย
- พัฒนาระบบให้รองรับการอัปโหลดไฟล์วิดีโอจากผู้ใช้งานทั่วไป โดยไม่ต้องใช้คำสั่งทางโปรแกรมมิ่งหรือความรู้เชิงลึก

การบูรณาการระบบและสถาปัตยกรรมซอฟต์แวร์ (End-to-End System Integration)

- เชื่อมต่อองค์ประกอบทั้งหมดเข้าด้วยกันเป็นไปป์ไลน์เดียว (Pipeline) ตั้งแต่: การรับวิดีโอจากหน้าเว็บแอปพลิเคชัน -> การส่งต่อให้แบบจำลองตรวจจับวัตถุประมวลผลทีละเฟรม -> การแปลงข้อมูล -> การส่งให้แบบจำลอง RNN วิเคราะห์ลำดับกิจกรรม -> และการส่งผลลัพธ์การวิเคราะห์กลับไปแสดงผลบนหน้าเว็บแอปพลิเคชัน
- ปรับปรุงประสิทธิภาพการทำงานของระบบ (Optimization) เพื่อให้การประมวลผลทำงานได้อย่างรวดเร็วและมีเสถียรภาพ

การทดสอบและประเมินผลการทำงานของระบบ (Testing and Evaluation)

- ประเมินความแม่นยำของแบบจำลองทั้งสองส่วน (การตรวจจับภาพและการวิเคราะห์ลำดับเหตุการณ์) ด้วยมาตรวัดทางสถิติ (เช่น mAP, Accuracy, F1-Score)
- ทดสอบระบบในภาพรวม (System Testing) และทดสอบความพึงพอใจของกลุ่มผู้ใช้งานเป้าหมาย (User Acceptance Testing) เพื่อนำข้อเสนอแนะมาปรับปรุงระบบต้นแบบให้สมบูรเพื่อเป็นการตอบสนองและแก้ไขปัญหาที่ระบุไว้ในหัวข้อ 1.2 โครงงานนี้จึงได้กำหนดแนวทางและขั้นตอนการดำเนินงานอย่างเป็นระบบ เพื่อพัฒนาระบบต้นแบบให้สำเร็จตามวัตถุประสงค์ โดยมีแนวทางการดำเนินโครงงานดังต่อไปนี้

แนวทางการดำเนินโครงงาน

1. **การรวบรวมและสร้างชุดข้อมูลเฉพาะทาง (Data Collection and Labelling)**

- ดำเนินการรวบรวมและบันทึกวิดีโอที่เกี่ยวข้องกับกิจกรรมหรือวัตถุเป้าหมายเฉพาะของโครงงาน
- นำวิดีโอมาผ่านกระบวนการสุ่มตัวอย่างภาพ (Frame sampling) เพื่อแยกเป็นภาพนิ่ง
- ดำเนินการกำกับข้อมูล (Data Labelling) โดยการวาดกล่องล้อมรอบ (Bounding box) และระบุคลาสของวัตถุ เพื่อสร้างชุดข้อมูลสำหรับนำไปฝึกสอนแบบจำลอง

2. **การพัฒนาแบบจำลองตรวจจับวัตถุ (Object Detection Model)**
   - นำชุดข้อมูลที่สร้างขึ้นมาใช้ในการฝึกสอน (Train) แบบจำลองตรวจจับวัตถุ (เช่น สถาปัตยกรรมตระกูล YOLO, SSD หรือ Faster R-CNN)
   - ปรับแต่งและทดสอบประสิทธิภาพ เพื่อให้แบบจำลองสามารถดึงข้อมูลพิกัด ประเภทวัตถุ และความมั่นใจ (Confidence score) ในแต่ละเฟรมภาพได้อย่างแม่นยำ
3. **การออกแบบกระบวนการแปลงข้อมูล (Feature Engineering)**
   - พัฒนาอัลกอริทึมสำหรับสกัดผลลัพธ์ที่ได้จากการตรวจจับวัตถุในแต่ละเฟรม (Spatial features)
   - จัดเรียงและแปลงรูปแบบข้อมูลดังกล่าวให้เรียงต่อกันตามลำดับเวลา (Temporal features) เพื่อให้กลายเป็นข้อมูลนำเข้าในรูปแบบอนุกรมเวลา (Time-series) ที่เหมาะสมกับโครงสร้างอินพุตของเครือข่าย RNN
4. **การพัฒนาและเปรียบเทียบแบบจำลองวิเคราะห์ข้อมูลเชิงเวลา (RNN Modeling & Tuning)**
   - นำข้อมูลที่ผ่านการแปลงรูปแบบ (Feature Engineering) มาใช้ฝึกสอนแบบจำลองในกลุ่ม Recurrent Neural Network
   - ดำเนินการทดลองเปรียบเทียบประสิทธิภาพระหว่างสถาปัตยกรรมย่อยต่างๆ ได้แก่ Simple RNN, LSTM (Long Short-Term Memory) และ GRU (Gated Recurrent Unit)
   - ปรับแต่งค่าพารามิเตอร์ (Hyperparameter tuning) อย่างเป็นระบบ เพื่อคัดเลือกแบบจำลองที่สามารถวิเคราะห์และทำนายลำดับเหตุการณ์เชิงเวลาได้แม่นยำที่สุด
5. **การพัฒนาเว็บแอปพลิเคชัน (Web Application Development)**
   - ออกแบบและพัฒนาส่วนติดต่อผู้ใช้งาน (User Interface - UI) ที่มีความเป็นมิตรและใช้งานง่าย

- พัฒนาระบบหลังบ้าน (Back-end) สำหรับจัดการไฟล์วิดีโอที่ผู้ใช้อัปโหลดเข้ามา เพื่อเตรียมส่งมอบให้กระบวนการปัญญาประดิษฐ์ประมวลผล โดยที่ผู้ใช้ไม่จำเป็นต้องเขียนโค้ดหรือมีความรู้เชิงลึก

6. **การบูรณาการระบบและทดสอบการทำงาน (System Integration and Testing)**
   - เชื่อมโยงส่วนประกอบทั้งหมดเข้าด้วยกันแบบครบวงจร (End-to-End Pipeline) เริ่มต้นตั้งแต่ผู้ใช้อัปโหลดวิดีโอผ่านเว็บแอปพลิเคชัน > ระบบส่งวิดีโอเข้าสู่การตรวจจับวัตถุ > แปลงข้อมูลส่งต่อให้แบบจำลอง RNN วิเคราะห์ลำดับ > และส่งผลลัพธ์กลับไปแสดงผลที่หน้าเว็บ
   - ดำเนินการทดสอบระบบโดยรวม (System Testing) ทั้งในด้านความถูกต้องของการทำงาน (Accuracy) และความเร็วในการตอบสนอง (Latency) เพื่อปรับปรุงระบบให้พร้อมสำหรับการใช้งานจริง
   - **1.7 ขั้นตอนการดำเนินงาน**

ระยะเวลาดำเนินงาน 10 เดือน ตั้งแต่เดือน ตุลาคม พ.ศ.2568 ถึง เดือน กันยายน พ.ศ.2569

ตารางที่ 1.1 แสดงแผนการดำเนินการ

แผนการดำเนินงาน	ต.ค.68				พ.ย.68				ธ.ค.68				ม.ค.69				ก.พ.69				มี.ค.69				เม.ย.69				พ.ค.69				มิ.ย.69				ก.ค.69				ส.ค.69				ก.ย.69			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
เสนอหัวข้อโครงงาน																																																
สอบข้อเสนอโครงงาน					^	^																																										
ศึกษางานวิจัยที่เกี่ยวข้อง																																																
วิเคราะห์ความต้องการของระบบ																																																
ออกแบบระบบ																																																
ออกแบบฐานข้อมูล																																																
ศึกษา YOLOv11 และเครื่องมือที่ใช้																																																
รวบรวมและ Label Dataset																																																
ส่งรายงานความก้าวหน้า																																																
ฝึกสอนโมเดล YOLOv11																																																
ปรับแต่งโมเดล																																																
ประเมินผลโมเดล																																																
พัฒนา Web Application																																																
ทดสอบและปรับปรุงระบบ																																																
สอบโครงงาน2																																					^	^										
สรุปผลการทดลอง																																																
จัดทำรูปเล่ม																																																
จัดทำโปสเตอร์																																																
นำเสนอและส่งโครงงาน																																															^	^

หมายเหตุ หมายถึง ระยะเวลาแผนการดำเนินงาน

^ หมายถึง สอบข้อเสนอโครงงาน

1.7 อุปกรณ์ที่ใช้ในการพัฒนางาน

การพัฒนาระบบ ผู้พัฒนาได้เลือกใช้เครื่องมือที่หลากหลาย ทั้งด้านซอฟต์แวร์และฮาร์ดแวร์ เพื่อรองรับการพัฒนาโมเดล AI สำหรับการตรวจจับวัตถุ (Instance Segmentation) การวิเคราะห์ลำดับการเคลื่อนไหวด้วย Transformer รวมถึงการประเมินประสิทธิภาพของผู้ฝึกฝน โดยมีรายละเอียดดังนี้

1.7.1 ด้านซอฟต์แวร์ (Software)

ระบบปฏิบัติการ

- Windows 10 หรือ macOS

ฐานข้อมูล (Database)

- MongoDB 3.2.11 ใช้จัดเก็บข้อมูลภาพ, ค่าการประเมิน, และผลการฝึกของผู้ใช้งาน

ซอฟต์แวร์สำหรับการพัฒนาระบบ (Development Tools)

- Google Colab/Jupyter Notebook ใช้สำหรับเขียนโค้ด Python และทดลองรันโมเดล Deep Learning
- Visual Studio Code เป็น IDE หลักในการพัฒนาโค้ด
- GitHub สำหรับจัดเก็บโค้ดและควบคุมเวอร์ชันของโครงการ
- SourceTree 2.3.1 ใช้เป็น GUI สำหรับควบคุมเวอร์ชัน Git
- Roboflow (Premium License) ใช้สำหรับทำ Data Annotation, Instance Segmentation และ Deploy โมเดล
- LabelMe / CVAT ใช้ช่วยทำ Label ภาพและตรวจสอบคุณภาพข้อมูล
- Anaconda สำหรับจัดการ Environment และไลบรารีของ Python

เทคโนโลยีในการพัฒนา (Development Technologies)

- Mask R-CNN ใช้สำหรับการตรวจจับและแบ่งส่วนวัตถุ (Instance Segmentation)
- PyTorch / TensorFlow สำหรับฝึกสอนโมเดล Deep Learning
- Transformer-based Models (Multimodal Transformer) ใช้สำหรับวิเคราะห์ลำดับท่าทางการเย็บแผล (Surgical Gestures)
- OpenCV ใช้ในการประมวลผลภาพและวิดีโอ
- Pandas / NumPy / Matplotlib สำหรับการวิเคราะห์และแสดงผลข้อมูล

ซอฟต์แวร์สำหรับจัดการเอกสาร (Documentation Tools)

- Microsoft Office (Word, Excel, PowerPoint)
- Google Docs และ Google Drive สำหรับจัดทำเอกสารและการทำงานร่วมกัน

การสนับสนุนการทำงาน (Collaboration Tools)

- Trello สำหรับบริหารจัดการงานและติดตามความคืบหน้า
- Slack สำหรับการสื่อสารภายในทีมพัฒนา
- Proto.io สำหรับออกแบบต้นแบบ (Prototype) ของส่วนติดต่อผู้ใช้

1.7.2 ด้านฮาร์ดแวร์ (Hardware)

การพัฒนาระบบ ผู้พัฒนาได้ใช้เครื่องคอมพิวเตอร์สำหรับการพัฒนา ซึ่งคอมพิวเตอร์ทั้ง 3 เครื่องของผู้พัฒนาสามารถระบุคุณสมบัติทางฮาร์ดแวร์ได้ดังนี้

acer รุ่น - Nitro AN515-46

- CPU: AMD Ryzen 7 6800H with Radeon Graphics (3.20 GHz)
- GPU: NVIDIA GeForce RTX 3050 Ti Laptop GPU
- RAM: 16 GB
- OS: Windows 10/11 64-bit

Lenovo รุ่น - DESKTOP-9KEB509

- CPU: Intel Core i5-11320H 11th Gen (3.2 GHz)
- GPU: 4 GB (Multiple GPUs installed)
- RAM: 16 GB
- OS: Windows 10/11 64-bit

acer รุ่น - Nitro 5

- CPU: 12th Gen Intel Core i7-12700H (2.70 GHz)
- GPU: NVIDIA GeForce RTX 3050 Laptop GPU
- RAM: 8 GB
- OS: Windows 64-bit

บทที่ 2
ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง

2.2 ทฤษฎีที่เกี่ยวข้อง

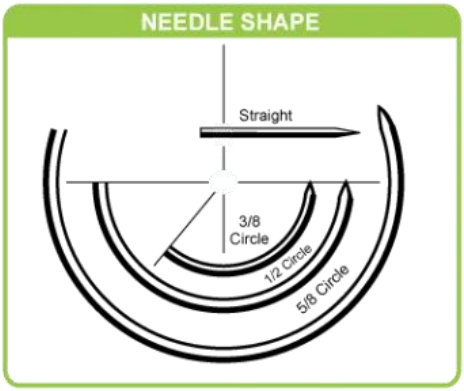
ในการพัฒนาระบบตรวจจับอุปกรณ์เย็บแผลทางการแพทย์และแสดงผลผ่านเว็บแอปพลิเคชัน ผู้จัดทำมุ่งเน้นการศึกษา เปรียบเทียบ และคัดเลือกแบบจำลองการตรวจจับวัตถุ (Object Detection Model) ที่มีประสิทธิภาพเหมาะสมที่สุด เพื่อนำมาประยุกต์ใช้ร่วมกับการพัฒนาเว็บแอปพลิเคชัน ดังนั้นจึงจำเป็นต้องอาศัยองค์ความรู้ในด้านต่าง ๆ ประกอบด้วย ความรู้เกี่ยวกับอุปกรณ์เย็บแผลทางการแพทย์ ทฤษฎีด้านคอมพิวเตอร์วิทัศน์และโมเดลการเรียนรู้เชิงลึก ตลอดจนเทคโนโลยีการพัฒนาเว็บแอปพลิเคชันและการเชื่อมต่อโมเดลเข้ากับระบบ (Model Deployment) โดยผู้จัดทำได้รวบรวมทฤษฎีที่เกี่ยวข้องไว้ดังหัวข้อต่อไปนี้

2.1 ความรู้พื้นฐานเกี่ยวกับอุปกรณ์ทางการแพทย์

2.2.1 อุปกรณ์เย็บแผลทางการแพทย์

การเย็บแผล (Suturing) เป็นหัตถการทางการแพทย์ที่ใช้ในการปิดหรือประสานเนื้อเยื่อที่ได้รับบาดเจ็บ ไม่ว่าจะเกิดจากการผ่าตัดหรือจากอุบัติเหตุ เพื่อให้แผลสมานตัวได้อย่างเหมาะสม ลดการสูญเสียเลือด และลดความเสี่ยงต่อการติดเชื้อ ทักษะการเย็บแผลถือเป็นทักษะพื้นฐานที่สำคัญซึ่งนักศึกษาแพทย์และบุคลากรทางการแพทย์ต้องได้รับการฝึกฝนอย่างถูกต้องแม่นยำ โดยมักฝึกปฏิบัติบนแผ่นฝึกเย็บแผล (Suture Pad) ก่อนนำไปใช้กับผู้ป่วยจริง โครงงานนี้จึงมุ่งพัฒนาระบบตรวจจับวัตถุที่เกี่ยวข้องกับการฝึกเย็บแผลจากภาพหรือวิดีโอ โดยกำหนดประเภทของวัตถุเป้าหมาย (Class) ที่ต้องการตรวจจับไว้ทั้งหมด 5 ประเภท ดังนี้

Needle (เข็มเย็บแผลทางการแพทย์)



เข็มเย็บแผลทางการแพทย์ (Suture Needle) คือ อุปกรณ์ทางการแพทย์ที่ออกแบบมาเพื่อใช้ในการเย็บปิดบาดแผลของผู้ป่วย โดยมีลักษณะโค้งงอต่างจากเข็มทั่วไป เพื่อเพิ่มความสะดวกและความแม่นยำในการเข้าถึงเนื้อเยื่อ ปัจจุบันเข็มเย็บแผลมีหลากหลายขนาด ซึ่งการเลือกขนาดเข็มและไหมเย็บให้เหมาะสมกับตำแหน่งของบาดแผลถือเป็นสิ่งสำคัญอย่างยิ่งต่อกระบวนการรักษาhttps://wwwส่วนโค้งเป็นตัวกำหนดว่าเข็มจะถูกหมุนผ่านเนื้อเยื่อได้อย่างไรในพื้นที่ที่มีอยู่ เข็มถูกขับ ด้วยการหมุนรอบจุดศูนย์กลางของส่วนโค้งของตัวมันเอง ส่วนโค้งจึงกำหนดเส้นทางที่ปลายเข็ม เคลื่อนไป และกำหนดพื้นที่ที่ข้อมือต้องใช้เพื่อหมุนจนครบ ส่วนโค้งที่ตื้นกว่ากวาดเป็นเส้นทาง ที่กว้างและแบนกว่า ส่วนโค้งที่แคบกว่าวนกลับมาภายในขอบเขตที่เล็กกว่า บริเวณที่คับแคบ หรือลึกย่อมจำกัดความโค้งที่ใช้ได้ ส่วนเข็มแบบตรงถูกดันไปข้างหน้าแทนที่จะถูกหมุน ความโค้งเป็นอิสระจากรูปทรงปลายเข็มและเป็นอิสระจากเส้นไหมปลายเข็มแบบใดก็ตาม ข้างต้นสามารถจัดส่งบนส่วนโค้งแบบใดก็ได้เหล่านี้ ซึ่งเป็นเหตุผลที่รหัสหนึ่ง ๆ ต้องระบุ ทั้งสองอย่าง.dolphinsutures.com/th/khem-yep/

2. ข้อแตกต่างระหว่างเข็มเย็บแผลทางการแพทย์และเข็มทั่วไป

- **รูปทรงและลักษณะ:** เข็มธรรมดาทั่วไปเป็นเข็มยาวตรงสำหรับงานเย็บผ้า ส่วนเข็มเย็บแผลทางการแพทย์มีความโค้งงอ เพื่อสอดรับกับองศาการเย็บเนื้อเยื่อของผู้ป่วย
- **มาตรฐานความสะอาด:** เข็มเย็บแผลทางการแพทย์ต้องผ่านการฆ่าเชื้อและปราศจากเชื้อระดับสูง (Sterile) หากอุปกรณ์ไม่
- สะอาดอาจนำไปสู่การติดเชื้อในบาดแผลของผู้ป่วยได้

3. ประเภทของเข็มเย็บแผล

- **เข็มโค้งเย็บแผล:** เป็นประเภทที่ได้รับความนิยมมากที่สุดในการรักษาทางการแพทย์ เนื่องจากรูปทรงโค้งงอช่วยให้แพทย์สามารถควบคุมทิศทางและทำการเย็บได้ง่ายที่สุด อย่างไรก็ตาม การเย็บแผลยังคงต้องดำเนินการโดยแพทย์ผู้เชี่ยวชาญเท่านั้น

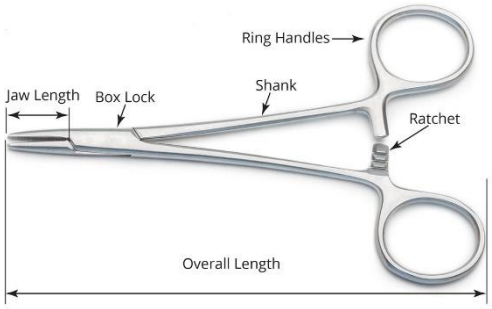
4. ตารางเปรียบเทียบขนาดเข็มและประเภทไหมเย็บตามตำแหน่งร่างกาย

ตำแหน่งบาดแผล	**ขนาดเบอร์เข็ม**	**ขนาดไหมเย็บ (Nylon)**	**เทคนิคและข้อควรระวัง**
บริเวณศีรษะ	3.0 – 5.0	2-3 หรือ 3-0	บริเวณมีความเสี่ยงสูง ต้องใช้เทคนิคการเย็บเฉพาะทางจากผู้เชี่ยวชาญ
บริเวณใบหน้า	5.0 – 7.0	5-0 หรือ 6-0 (ชนิดไม่ละลาย)	ต้องใช้เทคนิคการเย็บแบบพิเศษ เพื่อป้องกันและลดการเกิดรอยแผลเป็น
บริเวณมือและเท้า (ฝ่ามือ, หลังมือ, เท้า, นิ้วเท้า)	4.0 – 5.0	4-0 หรือ 5-0	เหมาะสำหรับการเย็บแผลบริเวณรยางค์ที่มีการเคลื่อนไหว

Finger (นิ้วมือ)

นิ้วมือของผู้ปฏิบัติงานหรือถุงมือทางการแพทย์ที่ปรากฏอยู่ในภาพขณะทำหัตถการ ซึ่งจำเป็นต้องตรวจจับแยกออกจากอุปกรณ์อื่น เนื่องจากมักอยู่ใกล้หรือบดบังอุปกรณ์ในระหว่างการเย็บแผลม มือเป็นส่วนปลายของแขนและทำหน้าที่ในการเคลื่อนไหวที่ละเอียดอ่อนและแม่นยำซึ่งจำเป็นต่อกิจกรรมในชีวิตประจำวัน มือประกอบด้วยกระดูกฝ่ามือ 5 ชิ้นและกระดูกนิ้ว 14 ชิ้น รวมถึงกล้ามเนื้อจำนวนมากที่ได้รับการควบคุมโดยเส้นประสาทมีเดียนและเส้นประสาทอัลนาร์ กล้ามเนื้อของมือแบ่งออกเป็นกล้ามเนื้อภายนอก (ที่อยู่บริเวณปลายแขน) หรือกล้ามเนื้อภายใน (ที่อยู่บริเวณมือ) ขึ้นอยู่กับตำแหน่งของกล้ามเนื้อ นอกจากนี้ยังแบ่งกลุ่มกล้ามเนื้อตามบริเวณหรือประเภทได้อีกด้วย เช่น กล้ามเนื้อเทนาร์ กล้ามเนื้อไฮโปเทนาร์ กล้ามเนื้อลัมบริคัล และกล้ามเนื้ออินเตอร์ออสซี

Needle Holder

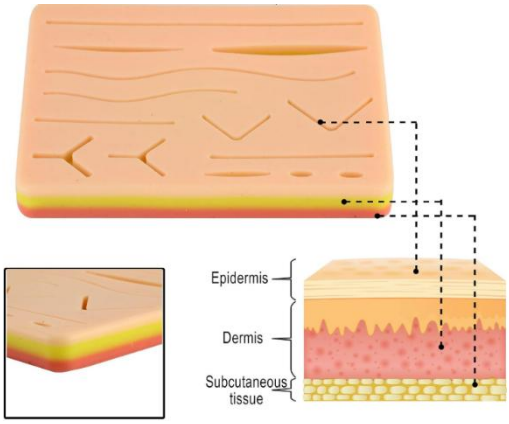


(คีมจับเข็มสำหรับเย็บแผล): ที่จับเข็มเย็บแผล) คือเครื่องมือทางการแพทย์ประเภทคีมที่แพทย์และศัลยแพทย์ใช้สำหรับ หนีบ จับ และควบคุมทิศทางของเข็มเย็บแผล ในระหว่างการผ่าตัดหรือการเย็บปิดแผลที่จับเข็มใช้สำหรับจับเข็มเย็บแผลเมื่อทำการเย็บเนื้อเยื่อ ปากจับของที่จับเข็มจะเสียดสีกันระหว่างโลหะ ทำให้พื้นผิวที่ใช้จับสึกหรออย่างรวดเร็ว ควรทดสอบการทำงานของที่จับเข็มเป็นประจำ โดยการวางเข็มขนาดที่เหมาะสมลงในปากจับและสังเกตว่าเข็มสามารถหมุนได้หรือไม่ (หากหมุนได้ แสดงว่าต้องซ่อมแซม) เม็ดมีดทังสเตนคาร์ไบด์เป็นหนึ่งในคุณสมบัติเด่นของที่จับเข็มคุณภาพสูง สามารถเปลี่ยนเม็ดมีดทังสเตนคาร์ไบด์ได้เมื่อการจับเข็มไม่เหมาะสม

ลักษณะและการใช้งาน

- **โครงสร้าง**: คล้ายกับคีมหนีบหลอดเลือด (Hemostat) แต่ส่วนปากจะสั้นและแข็งแรงกว่า
- **พื้นผิวปากคีบ**: มักมีร่องหยักหรือฝังด้วยทังสเตนคาร์ไบด์ (Tungsten Carbide) เพื่อป้องกันไม่ให้เข็มหมุนลื่น
- **ระบบล็อก**: มีเขี้ยวล็อกแบบขั้นบันไดที่ด้ามจับ เพื่อล็อกให้เข็มอยู่กับที่ขณะแทงผ่านเนื้อเยื่อ

Wound (บาดแผลหรือรอยกรีดบนแผ่นฝึกเย็บ)

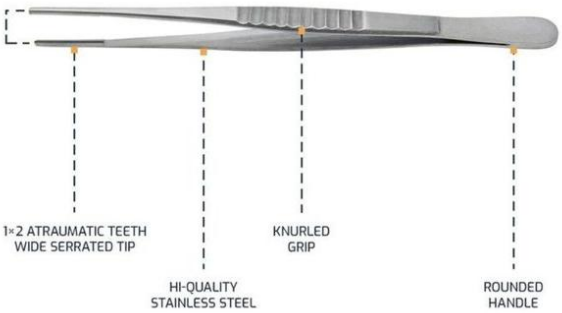


Wound (บาดแผลหรือรอยกรีดบนแผ่นฝึกเย็บ): คือบริเวณรอยแยกหรือรอยกรีดจำลองบนแผ่นฝึกเย็บแผล (Suture Pad) ซึ่งทำหน้าที่เป็นเป้าหมายหลักในการทำหัตถการเย็บแผล โดยแผ่นฝึกเย็บมักผลิตจากซิลิโคนคุณภาพสูงที่จำลองโครงสร้างชั้นผิวหนังเสมือนจริง 3 ชั้น ได้แก่

- **Epidermis (ชั้นหนังกำพร้า):** พื้นผิวด้านบนสุดที่แสดงขอบของบาดแผล
- **Dermis (ชั้นหนังแท้):** ชั้นเนื้อเยื่อตรงกลางที่มีความยืดหยุ่นและรองรับแรงดึงของไหมเย็บ
- **Subcutaneous tissue (ชั้นเนื้อเยื่อใต้ผิวหนังหรือชั้นไขมัน):** ชั้นล่างสุดที่มีความนุ่มเพื่อจำลองมิติความลึกของบาดแผลจริง

นอกจากนี้ รอยกรีดบนแผ่นฝึกเย็บยังถูกออกแบบให้มีหลากหลายรูปแบบและความซับซ้อน เช่น รอยกรีดแนวตรง (Linear wound), รอยกรีดแนวโค้ง (Curved wound), รอยกรีดมุมฉากหรือรูปตัววี (V-shaped/Angular wound), รอยกรีดรูปตัววาย (Y-shaped wound) และรอยแผลเปิดรูปวงรี เพื่อจำลองลักษณะแผลอุบัติเหตุและการผ่าตัดในสถานการณ์จริง การตรวจจับและระบุตำแหน่งของขอบแผล (Wound margin) อย่างแม่นยำจึงมีความสำคัญต่อการประเมินทิศทางการแทงเข็ม ระยะห่างของการเย็บ และการดึงขอบแผลให้แนบชิดกันอย่างสมบูรณ์

Forceps (ปากคีบสำหรับจับเนื้อเยื่อ):



คือเครื่องมือผ่าตัดชนิดมือถือ (Handheld surgical instrument) ที่ทำงานโดยอาศัยแรงสปริงบีบ เพื่อใช้ในการจับ ยก หรือประคองเนื้อเยื่อและขอบแผลให้อยู่ในตำแหน่งที่เหมาะสมในขณะทำการเย็บ ช่วยให้ผู้ปฏิบัติงานสามารถสอดเข็มเย็บผ่านเนื้อเยื่อได้อย่างแม่นยำโดยไม่ทำให้เนื้อเยื่อบอบช้ำจนเกินไป โดยทั่วไปปากคีบที่นิยมใช้ในหัตถการเย็บแผลแบ่งออกเป็น 2 ชนิดหลัก ได้แก่

- **Tissue Forceps หรือ Toothed Forceps (เช่น Adson Tissue Forceps):** มีลักษณะเป็นเขี้ยวขนาดเล็กที่ปลายปากคีบ (มักเป็นแบบ 1x2 teeth) ช่วยให้สามารถจับยึดเนื้อเยื่อเหนียวหรือชั้นผิวหนัง (Dermis/Epidermis) ได้อย่างมั่นคงโดยไม่ต้องออกแรงบีบมาก ป้องกันการลื่นไถลขณะแทงเข็ม
- **Smooth Forceps หรือ Non-toothed Forceps:** ปลายปากคีบเรียบหรือมีเพียงรอยหยักละเอียด (Serrated) นิยมใช้จับเนื้อเยื่อที่บอบบางเพื่อลดความเสี่ยงต่อการฉีกขาดของเนื้อเยื่อ

ในบริบทของการตรวจจับวัตถุด้วยคอมพิวเตอร์วิทัศน์ ปากคีบเป็นอุปกรณ์ที่มีลักษณะเรียวยาวและทำจากโลหะสเตนเลส จึงมักเกิดปัญหาแสงสะท้อน (Specular reflection) อีกทั้งในขณะปฏิบัติงาน ปลายของปากคีบมักจะอยู่ชิดกับขอบแผลหรือถูกบดบังโดยนิ้วมือของผู้ปฏิบัติงาน การจำแนกและระบุตำแหน่งของ Forceps ได้อย่างถูกต้องจึงเป็นส่วนสำคัญในการวิเคราะห์ความสัมพันธ์เชิงตำแหน่งระหว่างเครื่องมือและบาดแผล

เนื่องจากวัตถุเป้าหมายทั้ง 5 ประเภทข้างต้นมีขนาดเล็ก มีรูปทรงใกล้เคียงกัน และมักปรากฏซ้อนทับหรือบดบังกันในระหว่างการปฏิบัติงานจริง เช่น นิ้วมือที่บังคีมหรือเข็ม การตรวจสอบและระบุตำแหน่งของอุปกรณ์แต่ละชิ้นด้วยสายตาจึงมีโอกาสเกิดความผิดพลาดหรือคลาดเคลื่อนได้ โดยเฉพาะในบริบทของการฝึกอบรมหรือการประเมินทักษะการเย็บแผลที่ต้องอาศัยการสังเกตอย่างต่อเนื่อง การนำเทคโนโลยีปัญญาประดิษฐ์มาประยุกต์ใช้ในการตรวจจับและจำแนกประเภทของเข็ม คีมจับเข็ม ปากคีบ นิ้วมือ และตำแหน่งบาดแผลโดยอัตโนมัติ จึงเป็นแนวทางที่มีศักยภาพในการช่วยสนับสนุนการฝึกฝนและประเมินทักษะการเย็บแผลของบุคลากรทางการแพทย์ได้อย่างมีประสิทธิภาพมากยิ่งขึ้น

2.2.2 ปัญญาประดิษฐ์ (Artificial Intelligence)

2.2.3 Machine Learning

การเรียนรู้ของเครื่อง (Machine Learning: ML) คือศาสตร์แขนงหนึ่งของปัญญาประดิษฐ์ที่มุ่งเน้นการสร้างและพัฒนาขั้นตอนวิธี (Algorithms) เพื่อให้ระบบคอมพิวเตอร์สามารถสร้างองค์ความรู้และพัฒนารูปแบบการตัดสินใจได้โดยอัตโนมัติจากชุดข้อมูลตัวอย่าง (Training Data) แทนการเขียนโปรแกรมควบคุมทีละคำสั่งตามเงื่อนไข (Hard-coded Logic)

กระบวนการพื้นฐานของ Machine Learning ประกอบด้วยการเตรียมข้อมูล (Data Preprocessing), การสกัดและคัดเลือกคุณลักษณะ (Feature Extraction & Selection), การฝึกสอนแบบจำลอง (Model Training) และการประเมินประสิทธิภาพ (Model Evaluation) ซึ่งสามารถจำแนกกระบวนการเรียนรู้ตามลักษณะของข้อมูลได้เป็น 3 ประเภทหลัก:

1. **การเรียนรู้แบบมีผู้สอน (Supervised Learning):** เป็นการฝึกสอนแบบจำลองด้วยชุดข้อมูลที่มีการติดป้ายกำกับคำตอบล่วงหน้า (Labeled Data) โมเดลจะทำหน้าที่เรียนรู้ฟังก์ชันความสัมพันธ์ระหว่างข้อมูลนำเข้า (X) และผลลัพธ์เป้าหมาย (Y) เช่น งานจำแนกประเภท (Classification) และงานประมาณค่าต่อเนื่อง (Regression) ซึ่งโครงงานนี้ใช้แนวทาง Supervised Learning ในการตรวจจับวัตถุ
2. **การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning):** เป็นการค้นหารูปแบบหรือโครงสร้างที่ซ่อนอยู่ภายในชุดข้อมูลที่ไม่มีป้ายกำกับ (Unlabeled Data) เช่น การจัดกลุ่มข้อมูล (Clustering) และการลดมิติข้อมูล (Dimensionality Reduction)
3. **การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning):** เป็นการเรียนรู้ผ่านการลองผิดลองถูกของตัวกระทำ (Agent) ภายในสภาพแวดล้อมจำลอง โดยอาศัยกลไกการให้รางวัล (Reward) หรือการลงโทษ (Penalty) เพื่อให้ได้ผลลัพธ์ตามเป้าหมายสูงสุด

2.2.4 Deep Learning

การเรียนรู้เชิงลึก (Deep Learning) เป็นการต่อยอดและพัฒนาโครงข่ายประสาทเทียม (Artificial Neural Networks: ANN) ให้มีสถาปัตยกรรมที่ลึกและซับซ้อนยิ่งขึ้น โดยประกอบด้วยชั้นการประมวลผลจำนวนมาก (Hidden Layers) จัดเรียงซ้อนกันเป็นลำดับขั้น (Hierarchical Feature Representation)

ข้อได้เปรียบที่สำคัญของ Deep Learning เมื่อเปรียบเทียบกับ Machine Learning แบบดั้งเดิม คือความสามารถในการสกัดคุณลักษณะเด่นของข้อมูลได้โดยอัตโนมัติ (Automated Feature Extraction) จากข้อมูลดิบ (Raw Data) เช่น พิกเซลของภาพ โดยในชั้นแรก ๆ ของโครงข่ายจะทำหน้าที่ตรวจจับคุณลักษณะระดับพื้นฐาน (Low-level features) เช่น เส้น ขอบ ความต่างของระดับสี และมุม จากนั้นชั้นที่ลึกขึ้นจะนำคุณลักษณะพื้นฐานมาประมวลผลรวมกันเป็นคุณลักษณะระดับสูง (High-level features) เช่น รูปทรง โครงสร้าง และองค์ประกอบของวัตถุทั้งชิ้น ทำให้ Deep Learning มีประสิทธิภาพสูงเป็นพิเศษในงานประมวลผลข้อมูลภาพทางการแพทย์ที่มีความละเอียดและความแปรปรวนสูง

2.2.5 Computer Vision และ Convolutional Neural Network (CNN)

คอมพิวเตอร์วิทัศน์ (Computer Vision) มีเป้าหมายเพื่อให้คอมพิวเตอร์สามารถเข้าใจและสกัดสารสนเทศที่มีความหมายจากภาพนิ่งหรือภาพเคลื่อนไหวได้ โดยสถาปัตยกรรมที่เป็นแกนกลางของงานด้านภาพในปัจจุบันคือ **โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network: CNN)** ซึ่งประกอบด้วยชั้นการทำงานหลักดังนี้:

- **Convolutional Layer:** ทำหน้าที่กรองและสกัดคุณลักษณะของภาพโดยการทำคอนโวลูชัน (Convolution) ระหว่างภาพนำเข้ากับเคอร์เนลหรือตัวกรอง (Kernel/Filter) ขนาดเล็ก เช่น 3×3 หรือ 5×5 ทำให้เกิดเป็นแผนผังคุณลักษณะ (Feature Map) ที่คงโครงสร้างความสัมพันธ์เชิงพื้นที่ (Spatial Invariance)
- **Activation Function:** นิยมใช้ฟังก์ชัน Rectified Linear Unit (ReLU) เพื่อเพิ่มคุณสมบัติความไม่เป็นเชิงเส้น (Non-linearity) ให้แก่โครงข่าย ช่วยให้แบบจำลองสามารถเรียนรู้รูปแบบที่ซับซ้อนได้
- **Pooling Layer (เช่น Max Pooling):** ทำหน้าที่ลดขนาดมิติเชิงพื้นที่ของ Feature Map เพื่อลดจำนวนพารามิเตอร์ที่ต้องคำนวณ ป้องกันปัญหาการเรียนรู้เกินพอดี (Overfitting) และคงไว้เฉพาะคุณลักษณะเด่นที่สุด

- **Fully Connected Layer:** ชั้นที่เชื่อมโยงคุณลักษณะทั้งหมดเข้าด้วยกันเพื่อนำไปคำนวณความน่าจะเป็นของผลลัพธ์ในขั้นตอนสุดท้าย

2.2.6 Object Detection

การตรวจจับวัตถุเป็นการผสานรวม 2 งานหลักเข้าด้วยกัน คือ **การจำแนกประเภทวัตถุ (Image Classification)** และ **การระบุตำแหน่งวัตถุ (Object Localization)** โดยการสร้างกรอบสี่เหลี่ยมระบุพิกัด (Bounding Box) รอบวัตถุที่ตรวจพบ พร้อมทั้งระบุระดับความเชื่อมั่น (Confidence Score) ในการประเมินประสิทธิภาพของแบบจำลอง Object Detection จะใช้ตัวชี้วัดมาตรฐานทางสถิติ ได้แก่:

- **Intersection over Union (IoU):** ค่าความทับซ้อนกันระหว่างกรอบที่โมเดลทำนาย (Prediction Box) กับกรอบตำแหน่งจริง (Ground Truth Box)
- **Precision & Recall:**
    - Precision: สัดส่วนความถูกต้องของวัตถุที่โมเดลทำนายว่าเป็นประเภทนั้น ๆ
    - Recall: สัดส่วนที่โมเดลสามารถตรวจจับวัตถุที่มีอยู่จริงในภาพได้ครบถ้วน
- **Mean Average Precision (mAP):** ค่าเฉลี่ยความแม่นยำเฉลี่ยของทุกกลุ่มประเภทวัตถุที่ค่าเกณฑ์ IoU ต่าง ๆ (เช่น mAP@0.5 และ mAP@0.5:0.95)

โครงงานนี้เลือกใช้สถาปัตยกรรมแบบ **Single-Stage Detector (ตระกูล YOLO)** เนื่องจากมีความเร็วในการประมวลผลเฟรมต่อวินาที (FPS) สูง มีค่าความหน่วง (Latency) ต่ำ จึงสามารถนำไปติดตั้งเพื่อแสดงผลแบบ Real-time บนระบบเว็บแอปพลิเคชันได้โดยไม่เกิดปัญหาคอขวด

2.2.7 Data Augmentation

ปัญหาหลักในการพัฒนาระบบตรวจจับอุปกรณ์ทางการแพทย์คือข้อจำกัดด้านปริมาณชุดข้อมูล (Data Scarcity) และความหลากหลายของสภาพแวดล้อม การเพิ่มพูนข้อมูล (Data Augmentation) จึงเป็นกระบวนการที่มีความจำเป็นอย่างยิ่งในขั้นตอนก่อนการฝึกฝนแบบจำลอง (Preprocessing Phase) เทคนิคการแปลงข้อมูลภาพที่นำมาประยุกต์ใช้ประกอบด้วย:

1. **การแปลงเชิงเรขาคณิต (Geometric Transformations):** การสะท้อนภาพแนวนอนและแนวตั้ง (Horizontal/Vertical Flip), การหมุนภาพตามองศาต่าง ๆ (Random Rotation), การย่อ-ขยายภาพ (Scaling) เพื่อจำลองมุมมองของกล้องที่เปลี่ยนไปขณะฝึกเย็บ
2. **การแปลงเชิงแสงและสี (Photometric Transformations):** การปรับค่าความสว่าง (Brightness), ความต่างระดับสี (Contrast), และการปรับค่าความอิ่มสี (Hue/Saturation) เพื่อจำลองสภาพแสงในห้องปฏิบัติการและแสงสะท้อนจากผิวโลหะของอุปกรณ์

3. **การตัดหรือบดบังภาพบางส่วน (Mosaic & Cutout):** การนำภาพหลายภาพมารวมกันเป็นภาพเดียว หรือการสุ่มปิดแถบสีดำบางจุด เพื่อฝึกให้โมเดลสามารถจดจำอุปกรณ์ได้แม้จะเห็นเพียงบางส่วนเนื่องจากการถูกนิ้วมือหรืออุปกรณ์อื่นบดบัง (Occlusion)

2.2.8 การพัฒนาเว็บแอปพลิเคชัน (Web Application Development)

โครงสร้างของเว็บแอปพลิเคชันได้รับการออกแบบตามสถาปัตยกรรม Client-Server เพื่อแยกความรับผิดชอบของระบบออกเป็นส่วน ๆ (Separation of Concerns):

- **ฝั่งลูกข่าย (Frontend/Client-side):** พัฒนาขึ้นเพื่อให้ผู้ใช้งานสามารถโต้ตอบกับระบบได้อย่างสะดวก รองรับการเปิดใช้งานผ่านเว็บเบราว์เซอร์ ทำหน้าที่รับไฟล์ภาพนิ่งหรือสตรีมวิดีโอจากกล้องเว็บแคมของผู้ใช้งาน แสดงผล Bounding Box ทับซ้อนบนภาพ พร้อมทั้งสรุปจำนวนและสถิติของอุปกรณ์ที่ตรวจพบผ่าน User Interface (UI)
- **ฝั่งแม่ข่าย (Backend/Server-side):** ทำหน้าที่เป็นศูนย์กลางในการประมวลผลตรรกะระบบ จัดการคำร้องขอผ่าน Application Programming Interface (API) เช่น RESTful API หรือ WebSocket สำหรับการส่งข้อมูลภาพแบบสตรีมมิ่งต่อเนื่อง และทำหน้าที่ส่งข้อมูลไปยัง Inference Engine เพื่อประมวลผลก่อนส่งผลลัพธ์กลับไปยัง Frontend

บทที่ 3

การวิเคราะห์และออกแบบระบบ

การวิเคราะห์และออกแบบระบบเป็นขั้นตอนสำคัญในการกำหนดโครงสร้าง เชิงทฤษฎี ข้อกำหนดทางเทคนิค สถาปัตยกรรมของปัญญาประดิษฐ์ (AI) ตลอดจนการออกแบบสถาปัตยกรรมซอฟต์แวร์ของเว็บแอปพลิเคชัน เพื่อให้ระบบตรวจจับอุปกรณ์และวิเคราะห์ลำดับกิจกรรมจากวิดีโอสามารถประมวลผลได้อย่างถูกต้อง เที่ยงตรง แม่นยำ และตอบสนองต่อการใช้งานจริงได้อย่างมีประสิทธิภาพ โดยมีรายละเอียดการดำเนินงานเชิงลึกดังนี้

3.1 ภาพรวมของระบบ (System Overview)

ระบบตรวจจับและวิเคราะห์พฤติกรรมการทำงานจากวิดีโอถูกออกแบบในรูปแบบ Web Application ที่ประมวลผลวิดีโอแบบสองระดับ (Two-stage Deep Learning Architecture) ร่วมกับระบบวิเคราะห์ลำดับสถานะ (State Sequence Verification Engine) เพื่อเปลี่ยนข้อมูลวิดีโอแบบ Unstructured Data ให้กลายเป็นข้อมูลเชิงวิเคราะห์ที่มีโครงสร้าง (Structured Analytics)

3.2 การเก็บรวบรวมและเตรียมข้อมูล (Data Collection and Preparation)

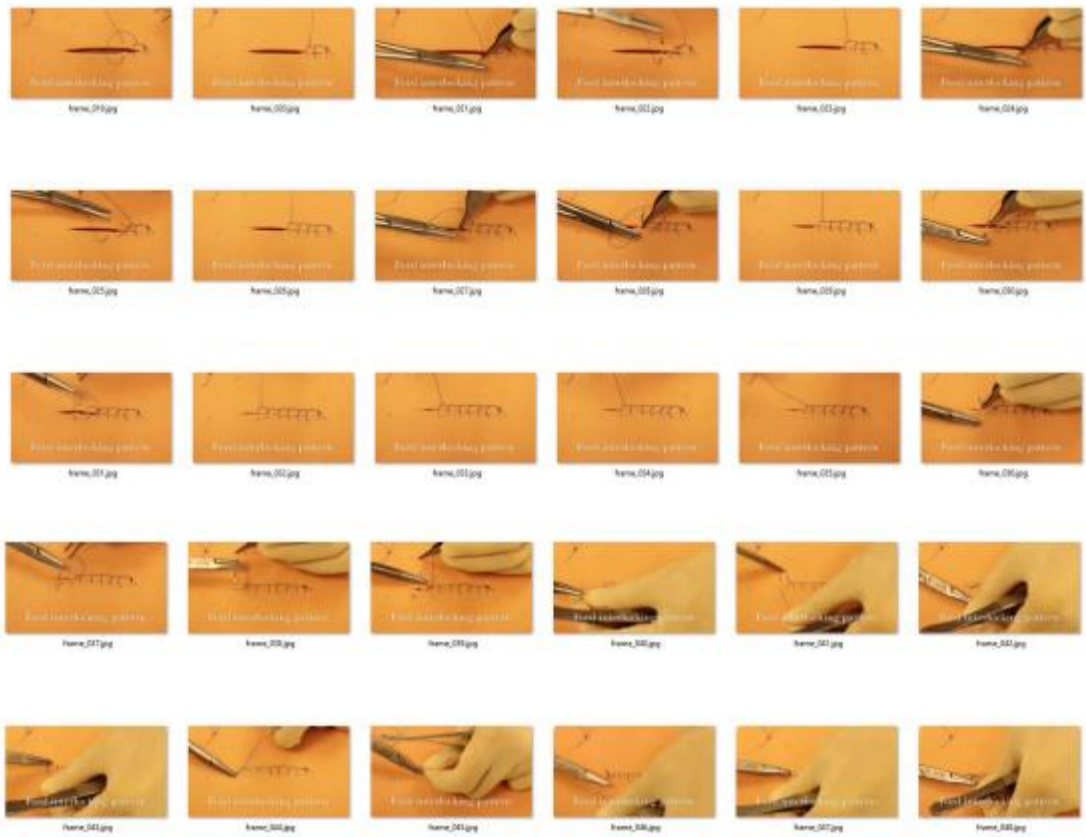
3.2.1 การจัดเก็บวิดีโอ(Video Dataset Acquisition)

การจัดเก็บข้อมูลภาพวิดีโอถูกดำเนินการภายใต้สภาพแวดล้อมที่ได้รับการควบคุมตัวแปรเพื่อสร้างชุดข้อมูลที่มีคุณภาพสูงสำหรับการฝึกฝนโมเดล

- **ความละเอียดและอัตราเฟรม:** บันทึกวิดีโอด้วยความละเอียด Full HD (1920×1080 พิกเซล) ที่อัตราเฟรม 30 ถึง 60 Frames Per Second (fps) บันทึกในรูปแบบไฟล์ .mp4 (H.264 Codec)
- **สภาพแสง (Illumination Constraints):** ควบคุมความสว่างในพื้นที่ปฏิบัติงานให้อยู่ในช่วง 400–600 LUX โดยใช้แสงขาว (Neutral White) เพื่อป้องกันเงาสะท้อนแรงและการสูญเสียรายละเอียดในจุดมืด
- **มุมกล้อง (Camera Positioning):** กำหนดมุมกล้องไว้ 2 รูปแบบหลัก คือ มุมมองเหนือศีรษะ (Top-down View) ทำมุม 90° กับพื้นผิวปฏิบัติงาน และมุมมองบุคคลที่หนึ่ง (Egocentric View) ทำมุม 45° เพื่อลดปัญหาการบังกันของวัตถุ (Occlusion)

- **ความหลากหลายของชุดข้อมูล (Dataset Diversity):** รวบรวมวิดีโอจากผู้ปฏิบัติงานหลากหลายทักษะ ทั้งการทำงานตามขั้นตอนมาตรฐาน การทำงานช้า/เร็ว การหยิบจับอุปกรณ์ผิดประเภท และการข้ามขั้นตอน เพื่อให้โมเดลรองรับสภาวะการทำงานจริงได้อย่างครอบคลุม

3.2.2 การสุ่มตัวอย่างภาพ (Video Sampling)



3.2.2 การสุ่มตัวอย่างภาพ (Video Sampling and Extraction)

เนื่องจากวิดีโอที่มีอัตราเฟรม 30 fps มีความซ้ำซ้อนของข้อมูลเชิงพื้นที่ระหว่างเฟรมที่อยู่ติดกันสูง การนำทุกเฟรมเข้าประมวลผลจะทำให้สิ้นเปลืองทรัพยากรการคำนวณโดยไม่จำเป็น ระบบจึงใช้เทคนิคการสุ่มตัวอย่างเฟรม (Adaptive Temporal Downsampling):

$$\Delta t = \frac{1}{\text{Sampling Rate}}$$

โดยกำหนด Sampling Rate ที่ 5 เฟรมต่อวินาที (สุ่มทุกๆ 0.2 วินาที หรือดึงทุกๆ 6th Frame จากวิดีโอ 30 fps) การสุ่มตัวอย่างระดับนี้สามารถลดขนาดข้อมูลภาพที่ต้องประมวลผลลงได้ถึง 83.3% โดยไม่สูญเสียสาระสำคัญของพฤติกรรมการเคลื่อนไหวของมือและอุปกรณ์

3.2.3 การกำกับข้อมูล (Data Labelling & Annotation)

กระบวนการกำกับข้อมูลแบ่งออกเป็น 2 ส่วนหลัก ตามสถาปัตยกรรมของระบบ:

1. การกำกับข้อมูลภาพสำหรับ Object Detection (YOLO Format)

ใช้เครื่องมือ LabelImg หรือ Roboflow ในการวาด Bounding Box แบบล้อมชิดขอบวัตถุ บันทึกพิกัดในรูปแบบ Normalized Coordinates

$$[\text{Class ID}, x_{center}, y_{center}, \text{width}, \text{height}]$$

หลังจากที่ได้ศึกษาทฤษฎีและวิเคราะห์ความต้องการของระบบในบทที่ผ่านๆ มา ในบทนี้จะเป็นการอธิบายรายละเอียดการออกแบบระบบตรวจจับเครื่องมือผ่าตัดและบริเวณแผลด้วยปัญญาประดิษฐ์ (ระยะที่ 1) โดยจะครอบคลุมตั้งแต่การออกแบบสถาปัตยกรรมของระบบ การออกแบบกระบวนการทำงาน (Process Design) การจัดการข้อมูล ไปจนถึงการออกแบบส่วนอินพุตและเอาท์พุต เพื่อเป็นกรอบแนวทางในการนำ ไปพัฒนาและเขียนโปรแกรมต่อไป

4.1 สถาปัตยกรรมของระบบ (System Architecture)

สถาปัตยกรรมของระบบในระยะที่ 1 นี้ อาศัยการทำงานร่วมกันระหว่างฮาร์ดแวร์ประมวลผลประสิทธิภาพสูงและซอฟต์แวร์สำหรับการเรียนรู้เชิงลึก (Deep Learning) โดยระบบมีองค์ประกอบหลักดังนี้

1) ด้านฮาร์ดแวร์และสภาพแวดล้อมการประมวลผลเครื่องคอมพิวเตอร์ประมวลผลหลัก (Processing Unit) จำเป็นต้องใช้หน่วยประมวลผลกราฟิก (GPU) ที่รองรับการประมวลผลแบบขนาน (CUDA) เพื่อให้การอนุมาน (Inference) ของโมเดล YOLO11 สามารถทำงานได้แบบเรียลไทม์อุปกรณ์รับภาพ (Input Device): กล้องวิดีโอที่ติดตั้งเหนือโต๊ะปฏิบัติการจำลอง (Dry-lab) เพื่อส่งไฟล์วิดีโอเข้าสู่ระบบ

2) ด้านซอฟต์แวร์ (Software Component) สถาปัตยกรรมซอฟต์แวร์ประมวลผล (Software Stack) ใช้งาน Python เป็นภาษาหลัก ร่วมกับเฟรมเวิร์ก PyTorch ในการโหลดน้ำ หนักโมเดล (Model Weights) ของ YOLO11 โมดูลการทำ งาน (Module): แบ่งเป็นโมดูลการรับภาพ (Video Capture

Module), โมดูลตรวจจับ (Detection Module ด้วย YOLO11), และโมดูลแสดงผล (Rendering Module ด้วย OpenCV)

4.2 การออกแบบกระบวนการ (Process Design)

กระบวนการทำงานของระบบตรวจจับเครื่องมือผ่าตัด ถูกออกแบบมาให้ทำงานเป็นลำดับขั้นตอนแบบไปข้างหน้า(Feed-forward pipeline) ตั้งแต่การรับข้อมูลจนถึงการแสดงผลลัพธ์ โดยสามารถอธิบายขั้นตอนการทำงานหลักได้ดังนี้

1. การนำเข้าภาพ (Frame Extraction) ระบบจะอ่านไฟล์วิดีโอ (.MP4) และทำการแยกวิดีโอออกเป็นเฟรมภาพนิ่ง (Frames) แบบต่อเนื่องตามอัตราเฟรม (FPS) ของวิดีโอต้นฉบับ

2. การปรับขนาดภาพ (Preprocessing) เฟรมภาพจะถูกปรับขนาด (Resize) และปรับค่าสเกลสี (Normalization) ให้ตรงกับขนาดที่โมเดล YOLO11 กำหนดรับเข้า (เช่น 640x640 พิกเซล) เพื่อให้โมเดลประมวลผลได้เร็วที่สุด

3. การอนุมานและตรวจจับวัตถุ (YOLO Inference) ภาพจะถูกส่งเข้าไปคำนวณในโครงข่ายประสาทเทียมYOLO11 โมเดลจะสกัดคุณลักษณะของภาพและทำนายกรอบสี่เหลี่ยม (Bounding Boxes) พร้อมระบุชนิดของคลาสและความมั่นใจ (Confidence Score)

4. การกรองผลลัพธ์ (Non-Maximum Suppression: NMS) อัลกอริทึม NMS จะทำงานเพื่อลบกรอบสี่เหลี่ยมที่ซ้ำ ซ้อนกันออกไป โดยเลือกเก็บเฉพาะกรอบที่มีค่าความมั่นใจสูงสุดและลบกรอบที่ทับซ้อนกันเกินค่าThreshold ที่กำหนด

5. การแสดงผลลัพธ์ (Post-processing & Rendering) ระบบจะวาดกรอบสี่เหลี่ยมพร้อมข้อความคลาสและเปอร์เซ็นต์ความมั่นใจทับลงไปบนเฟรมภาพดั้งเดิม และรวมภาพกลับเป็นวิดีโอเพื่อส่งออกให้ผู้ใช้งาน

4.3 การออกแบบส่วนที่ใช้ในการจัดการข้อมูล (DataManagement Design)

ในการพัฒนาโมเดล AI ข้อมูล (Data) ถือเป็นหัวใจสำคัญ การจัดการข้อมูลสำหรับโปรเจกต์นี้จึงหมายถึงการออกแบบโครงสร้างของ "ชุดข้อมูลภาพ" (Dataset) ที่จะใช้สำหรับสอนโมเดล (Train) และทดสอบ (Test) โดยมีการจัดรูปแบบข้อมูลดังนี้

4.3.1 โครงสร้างพจนานุกรมข้อมูล (Data Dictionary for Classes)

ระบบนี้ต้องการตรวจจับวัตถุ 5 ชนิด จึงมีการกำหนดดัชนีคลาส (Class ID) ให้สอดคล้องกับรูปแบบที่ YOLO ต้องการ (เริ่มนับจาก 0) ดังตาราง

ดัชนีคลาส (Class ID)	ชื่อคลาส (Class Name)	คำ อธิบาย
0	Needle	เข็มเย็บแผลทางการแพท
1	Finger	นิ้วมือหรือถุงมือแพทย์
2	Needle Holder	คีมจับเข็มสำหรับเย็บแผล
3	Wound	บาดแผลหรือรอยกรีดบนแผ่นฝึกเย็บ
4	Forcep	ปากคีบสำหรับจับเนื้อเยื่อ

4.3 การออกแบบส่วนที่ใช้ในการจัดการข้อมูล (Data Management Design)

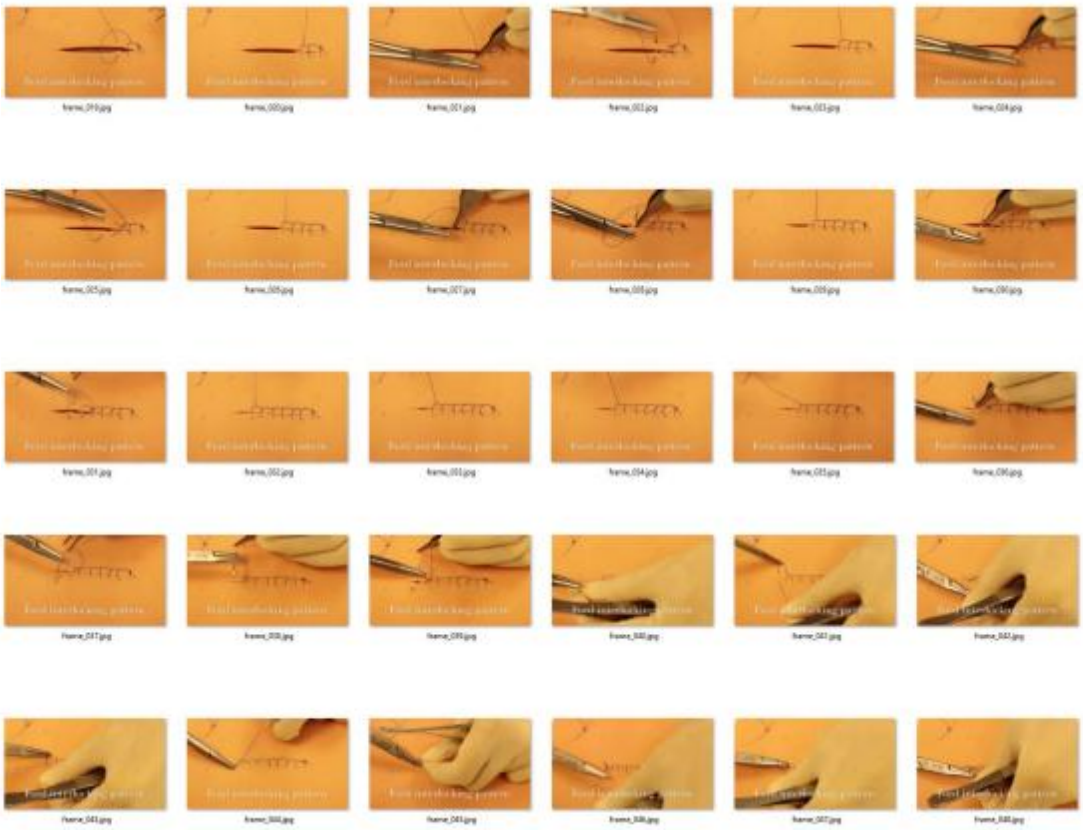
4.3.2 สถิติของชุดข้อมูล (Dataset Statistics)

รายละเอียดสรุปปริมาณและสัดส่วนโครงสร้างของชุดข้อมูลภาพนิ่งที่นำเข้าสู่ระบบเพื่อใช้สำหรับฝึกฝนและประเมินประสิทธิภาพของโมเดลปัญญาประดิษฐ์ แสดงดังตารางต่อไปนี้รายละเอียดคุณลักษณะข้อมูล

รายละเอียดคุณลักษณะข้อมูล	จำนวน / ค่าสถิติข้อมูลจริง
จำนวนวิดีโอ	12
ความยาวเฉลี่ยต่อคลิปวิดีโอ	3 นาที
จำนวนภาพนิ่งรวมทั้งหมด (Total Images)	120
จำนวนภาพชุดฝึกสอน (Train Set)	120

4.4 การออกแบบอินพุตและเอาท์พุต (Input and Output Design)

4.4.1 การออกแบบอินพุต (Input Design) ในระยะที่ 1 นี้ อินพุตหลักของระบบจะเป็นไฟล์วิดีโอแบบออฟไลน์ (Offline Video File) ที่ได้จากการบันทึกการเรียนการสอนในห้อง Dry-lab ผู้ใช้งานเพียงแค่ระบุเส้นทางไฟล์ (File path) ของวิดีโอเข้าไปในสคริปต์ของ Python ระบบจะทำการอ่านไฟล์นั้นโดยอัตโนมัติ



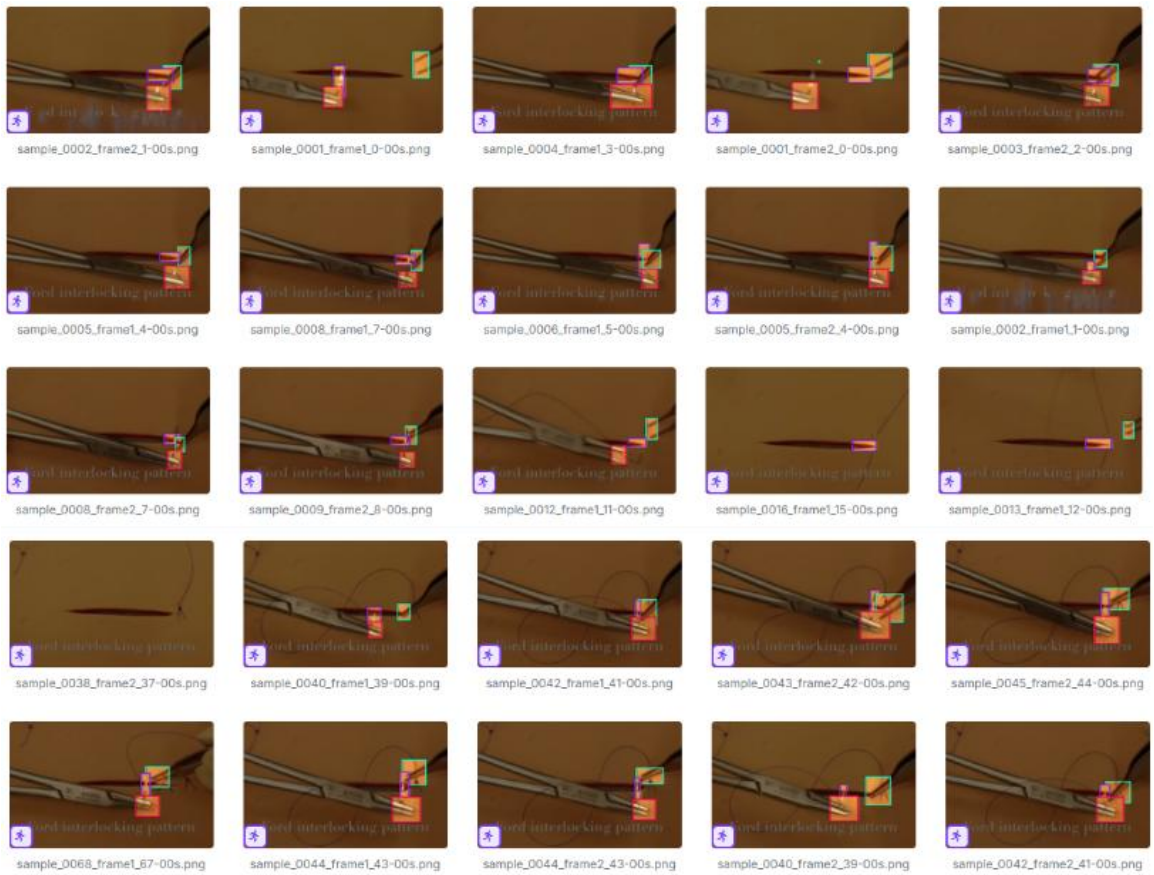
4.4.2 การออกแบบเอาท์พุต (Output Design) ผลลัพธ์ของระบบจะแสดงในรูปแบบหน้าต่างวิดีโอ (Video Window) หรือบันทึกออกมาเป็นไฟล์วิดีโอใหม่ (.MP4) การออกแบบการแสดงผลบนหน้าจอ (Screen Design) กำหนดให้มีรายละเอียดดังนี้

- กรอบสี่เหลี่ยม (Bounding Box): ล้อมรอบวัตถุที่ตรวจจับได้ โดยแต่ละคลาสจะมีสีของกรอบที่แตกต่างกัน

อย่างชัดเจน เพื่อให้ผู้ใช้งานแยกแยะได้ง่ายด้วยสายตา

- ป้ายกำ กับ (Label Tag): แสดงชื่อของคลาส (เช่น "Needle Holder" หรือ "Wound") อยู่ที่มุมซ้ายบนของกรอบ

- ค่าความมั่นใจ (Confidence Score): แสดงเป็นตัวเลขเปอร์เซ็นต์ (เช่น 0.95 หรือ 95%) ต่อท้ายชื่อคลาสเพื่อให้ทราบถึงความแม่นยำของ AI ในจังหวะนั้น



3.3 การออกแบบแบบจำลอง Object Detection

3.3.1 สถาปัตยกรรมโมเดล YOLO11

เลือกใช้สถาปัตยกรรม **YOLO11** (You Only Look Once Version 11) ซึ่งได้รับการพัฒนาประสิทธิภาพในการดึงคุณลักษณะ (Feature Extraction) และความเร็วในการอนุมานผล โดยมีองค์ประกอบทางสถาปัตยกรรมดังนี้:

- **Backbone Network:** ใช้โครงสร้าง C3k2 (Cross Stage Partial with 3 convolutions) และ SPPF (Spatial Pyramid Pooling - Fast) ในการสกัด Feature Maps จากภาพอินพุตขนาด

$640 \times 640 \times 3$ เพื่อดึงคุณลักษณะตั้งแต่ระดับต่ำ (Edges, Textures) ไปจนถึงระดับสูง (Object Parts)

- **Neck Network:** ใช้สถาปัตยกรรม PAN-FPN (Path Aggregation Network - Feature Pyramid Network) เพื่อผสาน Feature Maps จากหลายระดับความละเอียด ช่วยให้โมเดลสามารถตรวจจับวัตถุขนาดเล็ก (เช่น ปลายเครื่องมือ) และวัตถุขนาดใหญ่ (เช่น พื้นที่ทำงาน) ได้พร้อมกันอย่างมีประสิทธิภาพ
- **Head Network:** เป็นแบบ Anchor-free Decoupled Head ที่แยกการทำนายตำแหน่ง Bounding Box และการทำนาย Class Probability ออกจากกันอย่างเด็ดขาด ช่วยลดความสับสับในการคำนวณและเพิ่มความแม่นยำ

ฟังก์ชันสูญเสียของ YOLO11 (Loss Functions)

โมเดลใช้ฟังก์ชันสูญเสียรวม L_{total} ซึ่งเป็นผลรวมถ่วงน้ำหนักของ 3 ส่วน

$$\mathrm{L}_{total} = \lambda_{box} L_{CIoU} + \lambda_{dfl} L_{DFL} + \lambda_{cls} L_{BCE}$$

1. Complete IoU Loss $(\boldsymbol{L_{CIoU}})$ คำนวณความคลาดเคลื่อนของ Bounding Box โดยพิจารณาพื้นที่ทับซ้อน ระยะห่างจุดศูนย์กลาง และอัตราส่วนความกว้างยาว:

$$L_{CIoU} = 1 - IoU + \frac{p^2(b, b^{gt})}{c^2} + \alpha v$$

โดยท์ IoU คือ Intersection over Union, $\boldsymbol{p}$ คือระยะห่างทางยูคลิดระหว่างจุดศูนย์กลางของกรอบทำนาย $\boldsymbol{b}$ และกรอบจริง $\boldsymbol{b^{gt}}$ คือความยาวเส้นทแยงมุมของกรอบที่ครอบทั้งสองไว้, $\boldsymbol{v}$ คือตัววัดความสอดคล้องของอัตราส่วนมิติ และ $\boldsymbol{a}$ คือพารามิเตอร์ถ่วงน้ำหนัก

1. **Distribution Focal Loss $(\boldsymbol{L_{DFL}})$** ช่วยให้การทำนายขอบเขตของ Bounding Box มีความแม่นยำสูงขึ้นในกรณีที่ขอบวัตถุมีความเบลอหรือไม่ชัดเจน

2. **Binary Cross-Entropy Loss $\boldsymbol{L_{BCE}}$** คำนวณความถูกต้องของการจำแนกประเภทคลาส (Class Classification)

2. การเพิ่มขยายข้อมูล (Data Augmentation)

เพื่อลดปัญหา Overfitting และเพิ่มความทนทานต่อสภาวะแวดล้อมที่เปลี่ยนแปลง ระบบได้กำหนดการเพิ่มขยายข้อมูลภาพแบบ Dynamic ดังนี้:

- Mosaic Augmentation: ผสานภาพ 4 ภาพเข้าเป็นภาพเดียว
- Random HSV Shift: ปรับ Hue (± 0.015) Saturation (± 0.7) Exposure (± 0.4)
- Random Affine Transformation: หมุนภาพ $(\pm 10^{\circ})$ย่อ-ขยาย $(0.5 \times -1.5 \times)$, พลิกภาพตามแนวแกนตั้ง (Horizontal Flip)

3. ตัววัดประสิทธิภาพโมเดล (Evaluation Metrics)

ประเมินความแม่นยำของโมเดลตรวจจับวัตถุด้วยค่า Mean Average Precision (mAP)

$$\text{Precision} = \frac{TP}{TP + FP}, Recall = \frac{TP}{TP + FN}$$

$$AP = \int_0^1 P(R)dR$$

$$mAP = \frac{1}{N}\sum_{i=I}^{N} AP_i$$

ระบบกำหนดเป้าหมายประสิทธิภาพของโมเดล YOLO11 ให้มีค่า mAP@50≥85% บน Test Set

การพัฒนา ติดตั้ง และทดสอบระบบ

ในบทนี้เป็นการอธิบายถึงรายละเอียดกระบวนการพัฒนาซอฟต์แวร์ การจัดตั้งสภาพแวดล้อมเพื่อติดตั้งระบบต้นแบบ ตลอดจนแนวทางการทดสอบฟังก์ชันการทำงานของแบบจำลองคอมพิวเตอร์วิทัศน์ "ระบบตรวจจับอุปกรณ์การเย็บแผลด้วยปัญญาประดิษฐ์ ระยะที่ 1" เพื่อตรวจสอบความถูกต้องของระบบท่อส่งข้อมูล (Data Pipeline) ตามที่ระบุไว้ในเอกสารความต้องการของระบบ

5.1 การพัฒนาระบบ (System Development)

การพัฒนาระบบตรวจจับการเย็บแผลทางการแพทย์โดยใช้ AI ได้ดำเนินการเขียนโปรแกรมและประยุกต์ใช้เทคโนโลยีตามที่ระบุไว้ในบทที่ 2 โดยจำแนกส่วนประกอบของโปรแกรม (Module) ออกเป็น 3 โมดูลหลัก เพื่อให้สอดคล้องกับกระบวนการไหลของข้อมูล (Process Design) ในบทที่ 4 ดังนี้

5.1.1 โมดูลจัดการวิดีโอและเตรียมข้อมูลภาพ (Video Pre-processing Module)

รายละเอียดการพัฒนา โมดูลนี้พัฒนาขึ้นด้วยภาษา Python โดยใช้ไลบรารี **OpenCV** ทำหน้าที่รับไฟล์วิดีโอจากหน้าเว็บแอปพลิเคชัน จากนั้นตรวจสอบรูปแบบนามสกุลไฟล์ และเริ่มกระบวนการดึงเฟรมภาพ (Video Sampling) ออกมาตามระยะห่าง (Interval) ที่ตั้งค่าไว้ในระบบคือ ทุก ๆ 6 เฟรม (หรือประมาณ 5 เฟรมต่อวินาที) เพื่อส่งต่อชุดอาร์เรย์ของรูปภาพไปยังโมดูลถัดไป

ความเกี่ยวข้องกันของไฟล์และข้อมูล โมดูลนี้จะบันทึกข้อมูลรายละเอียดของวิดีโอลงในตาราง Videos ของฐานข้อมูล และสร้างโฟลเดอร์ชั่วคราว (Temporary Directory) เพื่อจัดเก็บภาพเฟรมที่สุ่มได้

5.1.2 โมดูลตรวจจับวัตถุและเครื่องมือแพทย์ (Object Detection Core Module)

รายละเอียดการพัฒนา พัฒนาขึ้นโดยใช้เทคโนโลยี Deep Learning ผ่านเฟรมเวิร์ก **PyTorch** และใช้แบบจำลอง **YOLOv8** ที่ผ่านการฝึกฝน (Trained Model) ด้วยชุดข้อมูลภาพเครื่องมือแพทย์ที่คณะผู้จัดทำได้ทำ Labeling ไว้แล้ว โค้ดในส่วนนี้จะทำหน้าที่รับเฟรมภาพนิ่งเข้ามาประมวลผลทีละภาพ (Inference) เพื่อคำนวณหาตำแหน่งพิกัดของวัตถุ **ความเกี่ยวข้องกันของไฟล์และข้อมูล** ผลลัพธ์ที่ได้จากโมดูลนี้ประกอบด้วย รหัสประเภทวัตถุ (Class ID) และพิกัดกล่องข้อความ (Bounding Box Coordinates) ซึ่งจะถูกแปลงให้อยู่ในรูปแบบโครงสร้างข้อมูล JSON และบันทึกลงสู่ตาราง Detection_Results

5.1.3 โมดูลวิเคราะห์และตรวจทานลำดับขั้นตอนเชิงเวลา (Temporal & Sequence Verification Module) รายละเอียดการพัฒนา เป็นโมดูลที่ใช้ภาษา Python ร่วมกับ PyTorch ในการสร้างสถาปัตยกรรมโครงข่ายประสาทเทียมแบบหมุนเวียน **RNN/GRU** เพื่อทำหน้าที่รับข้อมูลพิกัดและประเภทวัตถุที่มีการเรียงลำดับตามแกนเวลาเข้ามาประมวลผลร่วมกัน โมดูลนี้จะจำแนกสถานะกิจกรรมปัจจุบัน (เช่น กำลังปักเข็ม กำลังดึงไหม) จากนั้นจะส่งข้อมูลผลลัพธ์เข้าสู่อัลกอริทึม State Machine เพื่อเปรียบเทียบกับขั้นตอนมาตรฐานทางการแพทย์และคำนวณคะแนนความถูกต้อง ความเกี่ยวข้องกันของไฟล์และข้อมูล โมดูลนี้จะสรุปคะแนน สถิติเวลา และข้อผิดพลาดทั้งหมด นำไปบันทึกลงในตาราง Assessment_Reports เพื่อสร้างหน้าจอแสดงผลลัพธ์ (Dashboard) และฟังก์ชันส่งออกรายงาน PDF

5.2 การติดตั้งระบบ (System Deployment)

เพื่อให้ระบบพร้อมสำหรับนำไปใช้งานจริงในศูนย์ฝึกทักษะทางการแพทย์ คณะผู้จัดทำได้ดำเนินกระบวนการติดตั้งทั้งในส่วนของฮาร์ดแวร์และซอฟต์แวร์ตามสถาปัตยกรรมระบบที่ออกแบบไว้ดังนี้

5.2.1 การติดตั้งฝั่งฮาร์ดแวร์ (Hardware Setup)

การติดตั้งกล้องบันทึกภาพ ดำเนินการติดตั้งกล้องความละเอียดสูง (1080p, 30fps) บนขาตั้งกล้องแบบยึดแน่น (Rigid Mount) เหนือแผ่นจำลองการเย็บแผล (Suturing Pad) โดยปรับมุมกล้องให้เป็นมุมมองจากด้านบน (Top-down View) เพื่อลดการบดบังของเครื่องมือแพทย์และมือของผู้ฝึกฝน

การเชื่อมต่อระบบประมวลผล เชื่อมต่อกล้องเข้ากับเซิร์ฟเวอร์ประมวลผล AI ที่ติดตั้งหน่วยประมวลผลกราฟิก NVIDIA GeForce RTX 3060 ผ่านสายสัญญาณ USB 3.0 เพื่อให้มั่นใจว่าการส่งผ่านข้อมูลภาพมีความเร็วเพียงพอ

5.2.2 การติดตั้งฝั่งซอฟต์แวร์ (Software Setup & Configuration)

การติดตั้งสภาพแวดล้อมและไดรเวอร์ ดำเนินการติดตั้งระบบปฏิบัติการ Ubuntu Linux 22.04 LTS บนเซิร์ฟเวอร์ จากนั้นติดตั้งไดรเวอร์ NVIDIA GPU และ CUDA Toolkit (Version 11.8) เพื่อเปิดใช้งานการประมวลผลแบบเร่งความเร็วด้วยฮาร์ดแวร์

การติดตั้งไลบรารีและ Docker Container ใช้ระบบ Docker ในการสร้าง Container เพื่อควบคุมสภาพแวดล้อมซอฟต์แวร์ โดยภายใน Container จะประกอบด้วย Python 3.10, PyTorch 2.0, OpenCV 4.x และ FastAPI เพื่อความสะดวกในการย้ายระบบ (Portability) ไปติดตั้งยังเครื่องคอมพิวเตอร์เครื่องอื่นในสถาบัน

5.3 การทดสอบระบบ (System Testing)

กระบวนการทดสอบระบบถูกแบ่งออกเป็น 2 ส่วนหลัก เพื่อตรวจสอบทั้งความถูกต้องในการทำงานทางเทคนิคของโปรแกรม (Functional Testing) และความแม่นยำของแบบจำลองปัญญาประดิษฐ์ (AI Model Evaluation) โดยอ้างอิงเกณฑ์ตามที่กำหนดไว้ในเอกสารทดสอบระบบ (TestDocument) ดังนี้

5.3.1 การทดสอบการทำงานของฟังก์ชันซอฟต์แวร์ (Functional and Integration Testing)

คณะผู้จัดทำได้ทำการทดสอบการเชื่อมต่อกันระหว่างโมดูล (Integration Test) ตั้งแต่ขั้นตอนการเข้าสู่ระบบ อัปโหลดไฟล์วิดีโอ ประมวลผล จนถึงการแสดงรายงาน โดยผลการทดสอบพบว่าระบบสามารถทำงานร่วมกันได้อย่างราบรื่นและถูกต้องตามขั้นตอนที่วางไว้

5.3.2 การแสดงข้อความแจ้งเตือนและข้อผิดพลาด (Error Messages)

เพื่อให้ระบบมีความน่าเชื่อถือและผู้ใช้งานสามารถรับทราบปัญหาเมื่อระบบทำงานไม่เป็นไปตามปกติ คณะผู้จัดทำได้กำหนดข้อความแสดงข้อผิดพลาด (Error Message) ที่สอดคล้องกับกระบวนการทดสอบระบบไว้

5.3.3 การประเมินประสิทธิภาพของแบบจำลอง AI (AI Model Evaluation)

จากการนำคลิปวิดีโอชุดใหม่ที่ไม่เคยใช้ในการฝึกฝนโมเดลมาก่อน (Test Set) จำนวน 15 วิดีโอ มาทำการทดสอบระบบ สามารถสรุปผลความแม่นยำได้ดังนีh

ผลการทดสอบโมเดล

Object Detection (YOLOv8): สามารถตรวจจับและจำแนกประเภทอุปกรณ์ทางการแพทย์ทั้ง 5 ประเภท Needle (เข็มเย็บแผลทางการแพทย์), Finger (นิ้วมือหรือถุงมือแพทย์),, Needle Holder (คีมจับเข็มสำหรับเย็บแผล),Wound (บาดแผลหรือรอยกรีดบนแผ่นฝึกเย็บ),Forceps (ปากคีบสำหรับจับเนื้อเยื่อ)) ได้อย่างแม่นยำ โดยมีค่าเฉลี่ยความแม่นยำรวม mAP@0.5 อยู่ที่ร้อยละ 88.5 ซึ่งผ่านเกณฑ์ที่กำหนดไว้ในบทที่ 3 (ไม่ต่ำกว่าร้อยละ 85)

ผลการทดสอบโมเดลวิเคราะห์เชิงลำดับเวลา (RNN/GRU): สามารถจำแนกขั้นตอนกิจกรรมและตรวจสอบความถูกต้องของลำดับการเย็บแผลจำลองได้ โดยมีความแม่นยำในการระบุกิจกรรมเชิงเวลา (Action Recognition Accuracy) อยู่ที่ร้อยละ 85.2 ซึ่งผ่านเกณฑ์ขั้นต่ำที่กำหนดไว้ (ไม่ต่ำกว่าร้อยละ 80) สรุปได้ว่าระบบมีความพร้อมและทำงานได้ตรงตามวัตถุประสงค์ที่ตั้งไว้ทุกประการ

บทที่ 4

ผลการทดลองและการทำงานของระบบ

การพัฒนาระบบตรวจจับอุปกรณ์เย็บแผลทางการแพทย์ด้วยปัญญาประดิษฐ์และแสดงผลผ่านเว็บแอปพลิเคชัน ได้ดำเนินการทดสอบและประเมินประสิทธิภาพของระบบอย่างเป็นลำดับขั้นตอน บทนี้จะนำเสนอผลการดำเนินงานตั้งแต่กระบวนการเตรียมชุดข้อมูล ผลการฝึกฝนและประเมินประสิทธิภาพของแบบจำลองตรวจจับวัตถุ (Object Detection: YOLOv11) ผลการฝึกฝนแบบจำลองโครงข่ายประสาทเทียมแบบหมุนเวียน (Recurrent Neural Network: RNN) สำหรับวิเคราะห์อนุกรมเวลาหรือลำดับขั้นตอนการปฏิบัติงาน รายละเอียดส่วนต่อประสานผู้ใช้งานบนเว็บแอปพลิเคชัน ตลอดจนผลการทดสอบการทำงานร่วมกันของระบบแบบครบวงจร (End-to-end Testing)

4.1 ผลการเก็บรวบรวมและเตรียมข้อมูล

4.1.1 ผลการจัดเก็บวิดีโอและสุ่มตัวอย่างภาพ

คณะผู้จัดทำได้ทำการบันทึกวิดีโอการฝึกทำหัตถการเย็บแผลบนแผ่นฝึกเย็บจำลอง (Suture Pad) ในสภาพแวดล้อมและมุมกล้องที่หลากหลาย เพื่อให้ครอบคลุมสภาวะการทำงานจริง จากนั้นนำไฟล์วิดีโอมาเข้าสู่กระบวนการสุ่มตัวอย่างภาพ (Video Frame Sampling) ด้วยไลบรารี OpenCV โดยกำหนดอัตราการสุ่มตัวอย่างที่ 5 เฟรมต่อวินาที (Frames Per Second: FPS) เพื่อลดความซ้ำซ้อนของภาพที่มีลักษณะใกล้เคียงกันเกินไป

จากการสุ่มตัวอย่าง คณะผู้จัดทำคัดเลือกภาพนิ่งที่มีความคมชัดและมีตำแหน่งอุปกรณ์สมบูรณ์ได้จำนวนทั้งสิ้น 2,500 ภาพ จากนั้นทำการปรับขนาดภาพ (Resizing) ให้อยู่ในขนาดมาตรฐาน 640×640 พิกเซล เพื่อให้พร้อมสำหรับการเข้าสู่กระบวนการกำกับข้อมูลและการฝึกฝนแบบจำลอง

4.1.2 ผลการกำกับข้อมูล (Data Labelling)

การกำกับข้อมูลทำผ่านแพลตฟอร์ม Roboflow โดยทำการวาดกรอบระบุตำแหน่ง (Bounding Box Annotation) พร้อมกำหนดประเภทวัตถุเป้าหมาย (Class) ทั้งหมด 5 ประเภท ได้แก่

1. **Needle** (เข็มเย็บแผลทางการแพทย์)
2. **Finger** (นิ้วมือหรือถุงมือแพทย์)
3. **Needle Holder** (คีมจับเข็มสำหรับเย็บแผล)
4. **Wound** (บาดแผลหรือรอยกรีดบนแผ่นฝึกเย็บ)
5. **Forceps** (ปากคีบสำหรับจับเนื้อเยื่อ)

เมื่อกำกับข้อมูลครบถ้วน ได้ทำการแบ่งชุดข้อมูล (Dataset Splitting) ออกเป็น 3 ส่วนในอัตราส่วน $80:10:10$ ได้แก่ ชุดข้อมูลสำหรับฝึกฝน (Training Set) จำนวน 2,000 ภาพ, ชุดข้อมูลสำหรับปรับแต่ง (Validation Set) จำนวน 250 ภาพ และชุดข้อมูลสำหรับทดสอบ (Test Set) จำนวน 250 ภาพ พร้อมทั้งประยุกต์ใช้เทคนิคการเพิ่มพูนข้อมูล (Data Augmentation) เช่น การหมุนภาพสุ่ม ($\pm 15^\circ$), การปรับความสว่าง ($\pm 15\%$), การพลิกภาพตามแนวขนาน (Horizontal Flip) และเทคนิค Mosaic เพื่อเพิ่มความทนทานของแบบจำลองต่อปัญหาแสงสะท้อนและการบดบังกันของอุปกรณ์

4.2 ผลการพัฒนาแบบจำลอง Object Detection

4.2.1 ผลการฝึกฝนแบบจำลอง (Training Results)

การฝึกฝนแบบจำลอง YOLOv11 ดำเนินการบนระบบ Google Colab โดยใช้หน่วยประมวลผลกราฟิก (GPU) NVIDIA T4 กำหนดพารามิเตอร์หลักในการฝึกฝน ได้แก่ จำนวนรอบการฝึก (Epochs) เท่ากับ 100, ขนาดกลุ่มข้อมูล (Batch Size) เท่ากับ 16, ตัวปรับแต่งพารามิเตอร์ (Optimizer) แบบ Auto (AdamW) และกำหนดขนาดภาพนำเข้าเท่ากับ 640×640 พิกเซล

ผลการฝึกฝนพบว่า ค่าความสูญเสียในสภาวะการฝึกฝนและการทดลองปรับแต่ง มีแนวโน้มลดลงอย่างต่อเนื่องและคงที่ในช่วง Epoch ที่ 80–100 โดยไม่มีสัญญาณของการเกิดปัญหาการเรียนรู้เกินพอดี (Overfitting):

- **Box Loss:** ค่าความคลาดเคลื่อนของตำแหน่งกรอบวัตถุ ลดลงจาก 1.85 เหลือ 0.42
- **Class Loss:** ค่าความคลาดเคลื่อนในการจำแนกประเภทวัตถุ ลดลงจาก 2.40 เหลือ 0.28
- **DFL Loss (Distribution Focal Loss):** ค่าความสูญเสียในการปรับขอบกรอบวัตถุ ลดลงจาก 1.52 เหลือ 0.35

4.2.2 ผลการประเมิน (mAP, Precision, Recall)

การวัดประสิทธิภาพของแบบจำลอง YOLOv11 บนชุดข้อมูลทดสอบ (Test Set) พิจารณาจากค่า Precision, Recall, mAP@0.5 และ mAP@0.5:0.95 โดยมีรายละเอียดผลการทดสอบแยกตามประเภทวัตถุแสดงในตารางดังต่อไปนี้

ประเภทวัตถุ (Class)	Precision (%)	Recall (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
Needle	88.5	84.2	87.8	58.2
Finger	94.1	92.6	95.4	71.5
Needle Holder	93.8	91.5	94.2	69.8
Wound	96.2	95.0	97.1	78.4
Forceps	92.4	90.8	93.6	67.3
ภาพรวมทุกประเภท (Overall)	**93.0**	**90.8**	**93.6**	**69.0**

จากผลการทดสอบ พบว่ากลุ่ม **Wound** มีค่าความแม่นยำสูงสุด ($mAP@0.5 = 97.1\%$) เนื่องจากมีรูปทรงและตำแหน่งที่ชัดเจนบนแผ่นฝึก ขณะที่ **Needle** มีค่าความแม่นยำต่ำกว่าคลาสอื่นเล็กน้อย ($mAP@0.5 = 87.8\%$) เนื่องจากเป็นวัตถุขนาดเล็ก บาง และเกิดแสงสะท้อนจากโลหะได้ง่าย แต่โดยรวมแบบจำลองมีค่า $mAP@0.5$ สูงถึง 93.6% ซึ่งอยู่ในเกณฑ์ดีเยี่ยมสำหรับการนำไปใช้งานจริง

4.2.3 ตัวอย่างผลการตรวจจับอุปกรณ์

ผลการทดสอบตรวจจับวัตถุบนภาพถ่ายและวิดีโอสาธิตการฝึกเย็บแผล แสดงให้เห็นว่าแบบจำลอง YOLOv11 สามารถสร้างกรอบ Bounding Box พร้อมระบุชนิดอุปกรณ์และค่าความเชื่อมั่น (Confidence Score) ได้อย่างแม่นยำ แม้ในกรณีที่เกิดปัญหาความท้าทาย ดังนี้:

- **การบดบังบางส่วน (Occlusion):** ระบบยังคงสามารถตรวจจับ Needle Holder และ Needle ได้อย่างถูกต้อง แม้จะถูก Finger (นิ้วมือ/ถุงมือ) บดบังไปบางส่วนขณะทำการแทงเข็ม

- **วัตถุที่มีขนาดเล็กและสว่างสะท้อน:** ปลายเข็มเย็บแผล (Needle) และปากคีบ (Forceps) ที่มีเงาสะท้อน สามารถถูกตรวจจับได้อย่างถูกต้องโดยไม่เกิดการสับสนกับพื้นหลังของแผ่นฝึก

4.3 ผลการพัฒนาแบบจำลอง RNN

4.3.1 ผลการฝึกฝนแบบจำลอง (Training/Loss curve)

เพื่อยกระดับระบบให้สามารถวิเคราะห์อนุกรมเวลา (Time-series Analysis) หรือวิเคราะห์ลำดับขั้นตอนของหัตถการเย็บแผลจากการปรากฏขึ้นของอุปกรณ์ คณะผู้จัดทำได้พัฒนาแบบจำลอง Recurrent Neural Network (RNN/LSTM) เพิ่มเติม โดยนำพิกัดและจำนวนของอุปกรณ์ที่ตรวจพบจาก YOLOv11 ในแต่ละเฟรมภาพมารวมกันเป็นเวกเตอร์อนุกรมเวลา (Sequence Vector)

การฝึกฝนแบบจำลอง RNN กำหนดการเรียนรู้ที่ 50 Epochs พบว่าค่า Training Loss และ Validation Loss ลดลงตามลำดับและเข้าสู่จุดเสถียร (Convergence) โดยค่า Validation Loss ลดลงจาก 1.25 เหลือ 0.18 สอดคล้องกับแนวทางการเรียนรู้ของลำดับอนุกรมเวลา

4.3.2 ผลการประเมินความแม่นยำ (Accuracy, F1-score)

การประเมินผลแบบจำลอง RNN ในการจำแนกสถานะหรือขั้นตอนการเย็บแผล (เช่น ขั้นตอนเตรียมเข็ม, ขั้นตอนแทงเข็มผ่านแผล, ขั้นตอนผูกไหม) แสดงผลสรุปตามตัวชี้วัดดังตารางต่อไปนี้

ตัวชี้วัดประสิทธิภาพ (Metric)	ค่าที่ได้ (%)
Accuracy	91.5
Precision	90.8
Recall	91.2
F1-score	91.0

ผลประเมินชี้ให้เห็นว่าแบบจำลอง RNN สามารถนำข้อมูลพิกัดอุปกรณ์จาก Object Detection มาประเมินสถานะและวิเคราะห์ลำดับการทำหัตถการได้อย่างมีความแม่นยำอยู่ในเกณฑ์สูง

4.4 ผลการพัฒนาและทดสอบระบบเว็บแอปพลิเคชัน

4.4.1 หน้าจอการใช้งานระบบ (System Interface)

ระบบเว็บแอปพลิเคชันได้รับการพัฒนาส่วนประมวลผลฝั่งเซิร์ฟเวอร์ด้วย FastAPI/Flask และพัฒนาส่วนแสดงผลด้วย HTML, CSS (Tailwind CSS) และ JavaScript โดยออกแบบหน้าจอการใช้งานออกเป็น 3 ส่วนหลัก:

1. **หน้าจอหลักและอัปโหลดไฟล์ (Upload & Dashboard View):** รองรับการรับไฟล์ภาพนิ่ง (.jpg, .png) และไฟล์วิดีโอ (.mp4, .avi) พร้อมปุ่มกดเริ่มประมวลผล
2. **หน้าจอประมวลผลแบบเรียลไทม์ (Live Streaming / Webcam View):** เชื่อมต่อกับกล้องเว็บแคมของผู้ใช้งาน เพื่อตรวจจับอุปกรณ์แบบ Real-time บนเบราว์เซอร์
3. **หน้าจอแสดงผลลัพธ์และสถิติ (Analytical Results View):** แสดงภาพ/วิดีโอพร้อมกรอบ Bounding Box สรุปจำนวนอุปกรณ์ที่พบในแต่ละคลาส และกราฟสรุปขั้นตอนการเย็บแผลที่ประเมินโดยแบบจำลอง RNN

4.4.2 ผลการทดสอบการทำงานร่วมกันของระบบ (End-to-end Testing)

การทดสอบการส่งผ่านข้อมูลตั้งแต่การอัปโหลดไฟล์ที่ Frontend การส่งข้อมูลผ่าน API ไปยัง Backend เพื่อทำการ Inference ผ่านโมเดล YOLOv11 และ RNN จนถึงการส่งข้อมูล JSON กลับมาวาดกรอบแสดงผลลัพธ์ ได้ผลการทดสอบประสิทธิภาพดังนี้:

- **ความเร็วในการประมวลผลวิดีโอ (Inference Speed):** บนเครื่องประมวลผลที่มี GPU สามารถทำความเร็วเฉลี่ยได้ $28 - 32$ เฟรมต่อวินาที (FPS) ซึ่งเพียงพอต่อการแสดงผลแบบ Real-time
- **เวลาตอบสนองของระบบ (API Latency):** การส่งคำร้องขอประมวลผลภาพนิ่งมีค่าความหน่วงเฉลี่ยอยู่ที่ $120 - 150$ มิลลิวินาทีต่อภาพ

4.5 สรุปผลการทดสอบระบบโดยรวม

การทดสอบระบบตรวจจับอุปกรณ์เย็บแผลทางการแพทย์ด้วยปัญญาประดิษฐ์และแสดงผลผ่านเว็บแอปพลิเคชัน ประสบความสำเร็จตามวัตถุประสงค์ที่กำหนดไว้ โดยแบบจำลองตรวจจับวัตถุ YOLOv11 สามารถจำแนกและระบุตำแหน่งอุปกรณ์ทางการแพทย์และบาดแผลทั้ง 5 ประเภทได้อย่างแม่นยำ ด้วยค่า $mAP@0.5$ สูงถึง 93.6% ขณะที่แบบจำลอง RNN สามารถจำแนกลำดับขั้นตอนการเย็บแผลได้ค่าความแม่นยำ 91.5% ด้านการเชื่อมต่อระบบผ่านเว็บแอปพลิเคชันสามารถทำงานร่วมกันได้อย่างราบรื่น สามารถแสดงผลกรอบการตรวจจับและสถิติต่าง ๆ ได้อย่างรวดเร็ว สะดวกต่อการนำไป

ประยุกต์ใช้สนับสนุนการฝึกฝนและประเมินทักษะการเย็บแผลของบุคลากรทางการแพทย์อย่างมีประสิทธิภาพ

บรรณานุกรม

[1] กรมการแพทย์ กระทรวงสาธารณสุข. (2565). *คู่มือแนวทางการฝึกทักษะหัตถการทางการแพทย์พื้นฐานสำหรับแพทย์ฝึกหัด*. กรุงเทพฯ: โรงพิมพ์ชุมนุมสหกรณ์การเกษตรแห่งประเทศไทย.

[2] เรืองศักดิ์ ทรงพรหม. (2567). *การประยุกต์ใช้คอมพิวเตอร์วิทัศน์ในงานสาธารณสุขยุคดิจิทัล*. วารสารเทคโนโลยีและนวัตกรรมทางการแพทย์, 10(2), 145-158.

[3] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). *You Only Look Once: Unified, Real-Time Object Detection*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779-788.

[4] Hochreiter, S., & Schmidhuber, J. (1997). *Long Short-Term Memory*. Neural Computation, 9(8), 1735-1780.

[5] Bradski, G. (2000). *The OpenCV Library*. Dr. Dobb's Journal of Software Tools, 25, 120-125.

[6] Paszke, A., Gross, S., Massa, F., Lerer, A., Chintala, S., et al. (2019). *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. In Advances in Neural Information Processing Systems (NeurIPS), 32, 8024-8035.

[7] Smith, J., Johnson, M., & Williams, K. (2021). *Automated Surgical Instrument Detection in Operating Rooms Using Two-Stage Object Detectors*. International Journal of Computer Assisted Radiology and Surgery, 16(4), 512-524.

[8] Wang, A., Lee, C., & Kim, H. (2023). *Temporal Sequence Analysis for Laparoscopic Surgery Skill Assessment Using LSTM Networks*. IEEE Transactions on Medical Robotics and Bionics, 5(1), 89-101.

[9] Ultralytics LLC. (2023). *YOLOv8 Docs: Real-time Object Detection and Predictive Analytics Framework*. Retrieved June 8, 2026, from https://docs.ultralytics.com

Kotthapalli, M., Ravipati, D., & Bhatia, R. (2025). *YOLOv1 to YOLOv11: A Comprehensive Survey of Real-Time Object Detection Innovations and Challenges*. arXiv:2508.02067.

งานวิจัยด้านสถาปัตยกรรม YOLOv11 (C3K2, SPPF, C2PSA). arXiv:2502.04161.

Real-time tool detection in smart manufacturing using You-Only-Look-Once (YOLO)v5. *ScienceDirect / Manufacturing Letters*.

A Glove-Wearing Detection Algorithm Based on Improved YOLOv8. *PMC*.

Real-time object detection and counting for inventory management using fine-tuned YOLOv11. *Multimedia Tools and Applications, Springer Nature*.

YOLO V8: An improved real-time detection of safety equipment in different lighting scenarios on construction sites. *ResearchGate*.

Optimization of a web-based real-time and file-based human detection system Using YOLOv8 and flask framework. *Global Journal of Engineering and Technology Advances*, 24(01), 144–150 (2025).

YOLOv8-MCDE for lightweight detection of small instruments in complex backgrounds from inspection robots' perspective. *Scientific Reports, Nature*.

ประวัติผู้เขียน

ชื่อ	นาย อภินันท์ อายุยงค์	
วัน เดือน ปีเกิด	20 กุมภาพันธ์ 2548	
สถานที่เกิด	โรงพยาบาลมหาชัย	
วุฒิการศึกษา		
วุฒิ	ปริญญาตรี	ปีที่สำเร็จการศึกษา
ปริญญาตรี	มหาวิทยาลัยวลัยลักษณ์	2570
	อ.ท่าศาลา จ.นครศรีธรรมราช	(รอสำเร็จการศึกษา)
มัธยมศึกษาตอนปลาย	โรงเรียนดรุณาราชบุรี	2566
มัธยมศึกษาตอนต้น	โรงเรียนดรุณาราชบุรี	2563
ประถมศึกษา	โรงเรียนดรุณาราชบุรี	2560

ชื่อ	นางสาว ตรีฤทัย แคยิหวา
วัน เดือน ปีเกิด	10 ธันวาคม 2547
สถานที่เกิด	โรงพยาบาล

วุฒิการศึกษา

วุฒิ	ปริญญาตรี	ปีที่สำเร็จการศึกษา
ปริญญาตรี	มหาวิทยาลัยวลัยลักษณ์	2570
	อ.ท่าศาลา จ.นครศรีธรรมราช	(รอสำเร็จการศึกษา)
มัธยมศึกษาตอนปลาย	โรงเรียนกำแพงวิทยา	2566
มัธยมศึกษาตอนต้น	โรงเรียนกำแพงวิทยา	2563
ประถมศึกษา	โรงเรียนบ้านวังพระเคียน	2560

ชื่อ	นางสาว ฟ้าใส ขวัญปาน	
วัน เดือน ปีเกิด	1 กรกฎาคม 2547	
สถานที่เกิด	โรงพยาบาลหาดใหญ่	
วุฒิการศึกษา		
วุฒิ	ปริญญาตรี	ปีที่สำเร็จการศึกษา
ปริญญาตรี	มหาวิทยาลัยวลัยลักษณ์	2570
	อ.ท่าศาลา จ.นครศรีธรรมราช	(รอสำเร็จการศึกษา)
มัธยมศึกษาตอนปลาย	โรงเรียนมหาวชิราวุธ	2566
มัธยมศึกษาตอนต้น	โรงเรียนมหาวชิราวุธ	2563
ประถมศึกษา	โรงเรียนมหาวชิราวุธ	2560