



เว็บแอปพลิเคชันสำหรับการตรวจนับนิวเคลียส
จากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง

นางสาวกัญจน์ชัยภรณ์ แซ่จุ่ง¹
นางสาวณิชาพัชร์ นบ Nob²
นายวายุ รักวงศ์ศิริ³

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาศาสตร์ข้อมูลสุขภาพ
คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
ปีการศึกษา 2565

เว็บแอปพลิเคชันสำหรับการตรวจนับนิวเคลียส
จากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง

นางสาวกัญจน์ชยาภรณ์ แซ่จุ่ง
นางสาวณิชาพัชร์ นบນอบ
นายวายุ รักษ์วงศ์ศิริ

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาศาสตร์ข้อมูลสุขภาพ
คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
ปีการศึกษา 2565

คณะกรรมการสอบโครงการ

.....
ดร. คิริษฐา แจ้งสามัคคี
(ดร. คิริษฐา แจ้งสามัคคี)

อาจารย์ที่ปรึกษา

.....
ดร. อัญมณี ชัยประسنคสุข
(ดร. อัญมณี ชัยประسنคสุข)
.....
ดร. ปณิธิ อัจฉราฤทธิ์
(ดร. ปณิธิ อัจฉราฤทธิ์)

อาจารย์ที่ปรึกษาร่วม

อาจารย์ที่ปรึกษาร่วม

.....
(ดร. ปิยนิตย์ เวปุลานนท์)

กรรมการ

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

Project Title	Web Application for Automatic Nucleus Counting Immunofluorescence Tissue Biopsies
Credits	3
Member(s)	Ms. Kanchayapond Seajoong Ms. Nichapat Nobnorb Mr. Wayu Ragwongsiri
Project Advisor	Khariththa Jangsamsi, Ph.D.
Co-Advisor	Anyamanee Chaiprasongsuk, Ph.D. Paniti Achararit, Ph.D.
Program	Bachelor of Science
Field of Study	Health Data Science
Department	Computer Engineering
Faculty	Engineering
Academic Year	2022

Abstract

Nucleus counting is important in pathology as it helps in disease diagnosis and treatment development. Normally, pathologists count nuclei using Fluorescence Microscopy machines to image tissue stained with DAPI fluorescent dye and use the Columbus platform to count nuclei in the image automatically. However, the platform cannot accurately count and locate nuclei when they overlap, cluster together, or have indistinct shapes due to overlapping fluorescent dyes. This requires pathologists to manually count nuclei, which is time-consuming and prone to errors. Therefore, researchers studied the use of image processing techniques and deep learning, including Mask R-CNN, YOLO, and U-Net, to develop the best model for counting nuclei and compared them to develop a user-friendly web application. The YOLO model achieved the highest accuracy in nuclear counting, with an F1 score of 90.2 percent. The researchers hope that the developed web application will help reduce time and increase accuracy in nuclear counting.

Keywords: Nucleus counting, Tissue immunofluorescence, Fluorochromes DAPI, Image Processing, Deep Learning, Web Application,

หัวข้อปริญญา呢พนธ์	เว็บแอปพลิเคชันสำหรับการตรวจนับนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง
หน่วยกิต	3
ผู้เขียน	นางสาวกัญจน์ชัยภรณ์ แซ่จุ่ง นางสาวนิชาพัชร์ นบวนอบ นายวายุ รักษาวงศ์สิริ
อาจารย์ที่ปรึกษา	ดร.คริษฐา แจ้งสามสี ดร.อัญมณี ชัยประสงค์สุข ดร.ปนิช อัจฉราฤทธิ์
หลักสูตร	วิทยาศาสตรบัณฑิต
สาขาวิชา	วิทยาศาสตร์ข้อมูลสุขภาพ
ภาควิชา	วิศวกรรมคอมพิวเตอร์
คณะ	วิศวกรรมศาสตร์
ปีการศึกษา	2565

บทคัดย่อ

การนับจำนวนนิวเคลียสมีความสำคัญทางพยาธิวิทยา ช่วยในการวินิจฉัยโรคหรือพัฒนาการรักษา โดยปกติการนับจำนวนนิวเคลียสนักพยาธิวิทยาจะใช้เครื่อง Fluorescence Microscopy ถ่ายภาพเนื้อเยื่อที่ผ่านการย้อมด้วยสารเรืองแสง DAPI และใช้แพลตฟอร์ม Columbus นับจำนวนนิวเคลียสในภาพถ่ายอัตโนมัติแต่แพลตฟอร์ม Columbus ไม่สามารถนับจำนวนและแสดงตำแหน่งนิวเคลียสได้กรณีที่นิวเคลียสทับซ้อนกัน เกาะกลุ่มติดกัน และมีรูปร่างที่ไม่แน่นอน นักพยาธิวิทยาจึงต้องนับจำนวนนิวเคลียสด้วยตนเอง ส่งผลให้ใช้เวลามากในการนับ และอาจเกิดความผิดพลาดได้ คณะผู้วิจัยศึกษากระบวนการนับจำนวนนิวเคลียสด้วยการประมวลผลภาพ (Image Processing) และ การเรียนรู้เชิงลึก (Deep Learning) ประกอบด้วย Mask R-CNN, YOLO และ U-Net เพื่อพัฒนาแบบจำลองที่ดีที่สุดสำหรับนับจำนวนนิวเคลียส และนำแบบจำลองที่ดีที่สุดไปพัฒนาเว็บแอปพลิเคชันที่ง่ายต่อการใช้งาน จากผลการทดลองพบว่าการเรียนรู้เชิงลึกโดยใช้แบบจำลอง YOLO มีความแม่นยำในการนับนิวเคลียสสูงที่สุดคือค่า F1 score ที่ร้อยละ 90.2 คณะผู้วิจัยหวังว่าเว็บแอปพลิเคชันที่ได้พัฒนาขึ้นมาจะสามารถช่วยลดเวลาและเพิ่มความแม่นยำในการนับจำนวนนิวเคลียสให้มีความแม่นยำเพิ่มมากขึ้น

คำสำคัญ: การนับจำนวนนิวเคลียส, ภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง, สารเรืองแสง DAPI, การประมวลผลภาพ, การเรียนรู้เชิงลึก, เว็บแอปพลิเคชัน

กิตติกรรมประกาศ

การทำโครงการนี้ประสบความสำเร็จลังได้ด้วยความช่วยเหลืออย่างดียิ่งของ ดร. คริษฐา แจ้งสามสี อาจารย์ที่ปรึกษา ตลอดจน ดร. อัญมณี ชัยประสงค์สุข และ ดร. ปนิธิ อัจฉราฤทธิ์ อาจารย์ที่ปรึกษาร่วม ที่กรุณาสละเวลาเพื่อให้คำปรึกษา เสนอแนะแนวทาง ที่ดี ให้ข้อมูล แนวความคิด และข้อเสนอแนะที่เป็นประโยชน์ ตลอดจนคอยติดตามดูแลเอาใจใส่ต่อการทำงานวิจัยนี้จนสำเร็จลุล่วงไป ได้ด้วย คณะผู้วิจัยจึงขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี่

ขอขอบพระคุณอาจารย์ ดร. อัญมณี ชัยประสงค์สุข ที่ให้ความช่วยเหลือในการเก็บรวบรวมข้อมูล และให้รายละเอียดของข้อมูล ที่มีความสำคัญสำหรับการจัดทำงานวิจัยนี้

ขอขอบพระคุณ Dr. Sreenivas Bhattiprolu, Glenn Jocher, Matterport และ Sven P. Voigt สำหรับหลักการทำงาน ความรู้ และตัวอย่างการทำงานของอัลกอริทึมหรือแบบจำลอง รวมไปถึงตัวอย่างการนำไปใช้งานจริง

ขอขอบพระคุณคณะกรรมการ ผู้ให้ข้อมูล แนวคิด คำแนะนำ และข้อเสนอแนะที่เป็นประโยชน์ต่อการพัฒนางานวิจัยให้ดียิ่งขึ้น

สารบัญ

หน้า

ABSTRACT	ii
บทคัดย่อ	iii
กิตติกรรมประกาศ	iv
สารบัญ	v
สารบัญตาราง	vii
สารบัญรูปภาพ	viii
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของงานวิจัย	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 ตารางการดำเนินงาน	2
บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง	3
2.1 ทฤษฎีที่เกี่ยวข้อง	3
2.1.1 ด้านพยาธิวิทยา	3
2.1.1.1 อิมมูโนฟลูออเรสเซนซ์ (Immunofluorescence)	3
2.1.1.2 สารรีองแสง (Fluorochromes)	3
2.1.1.3 ภาคถ่ายเนื้อเยื่อจากการข้อมตัวด้วยสารรีองแสง (Immunofluorescence Tissue)	3
2.1.2 ด้านคอมพิวเตอร์	4
2.1.2.1 การประมวลผลภาพ (Image Processing)	4
2.1.2.2 การเรียนรู้เชิงลึก (Deep Learning)	6
2.2 ภาษาคอมพิวเตอร์และเทคโนโลยี	12
2.2.1 Python	12
2.2.2 OpenCV	12
2.2.3 Scikit-image	12
2.2.4 Streamlit	13
2.2.5 PyTorch	13
2.2.6 TensorFlow	13
2.2.7 Docker	13
2.2.8 Google Cloud Platform (GCP)	13
2.2.9 Cloud Run	13
2.2.10 Firebase	13
2.3 ทบทวนวรรณกรรม	14
2.3.1 DeepHCS++: Bright-field to fluorescence microscopy image conversion using multi-task learning with adversarial losses for label-free high-content screening	14
2.3.2 Tox_(R)CNN: Deep learning-based nuclei profiling tool for drug toxicity screening	14
2.3.3 A Means of Assessing Deep Learning-Based Detection of ICOS Protein Expression in Colon Cancer	15
2.3.4 Breast Cancer Detection from Histopathology Images Based on YOLOv5	15
2.3.5 Automated segmentation of overlapped nuclei using concave point detection and segment grouping	16
2.3.6 Automated image processing workflow for morphological analysis of fluorescence microscopy cell images	16
2.3.7 Automated cell counting and cluster segmentation using concavity detection and ellipse fitting techniques	17

2.4 ผลิตภัณฑ์ที่เกี่ยวข้อง	17
2.4.1 An ImageJ plugin for the high throughput image analysis of in vitro scratch wound healing assays	17
2.4.2 Shape identification and particles size distribution from basic shape parameters using ImageJ	17
2.4.3 Columbus Image Data Storage and Analysis System	17
2.4.4 Validation and Automation of Phenotypic Profiling Across Multiple Cell Lines	19
บทที่ 3 วิธีการดำเนินงาน	20
3.1 รายละเอียดภาพรวมการดำเนินงานวิจัย	20
3.2 ความต้องการของนักพยาธิวิทยา	20
3.3 การออกแบบระบบของฟ์แลร์	21
3.3.1 การพัฒนาแบบจำลองในการนับจำนวนนิวเคลียส	21
3.3.1.1 การเรียนรู้เชิงลึก (Deep Learning)	21
3.3.1.2 การประมวลผลภาพ (Image Processing)	24
3.3.2 การประเมินผลแบบจำลอง (Evaluation plans)	25
3.3.2.1 ประเภทของผลลัพธ์จากการเรียนรู้เชิงลึก (Deep Learning) และ การประมวลผลภาพ (Image Processing)	25
3.3.2.2 วิธีการวัดผลที่นำมาใช้	25
3.4 การพัฒนาเว็บแอปพลิเคชัน	26
3.4.1 แผนภาพสถาปัตยกรรมของระบบ	26
3.4.2 ส่วนต่อประสานกับผู้ใช้ (User Interface Design)	26
3.4.3 UML Design	28
3.4.4 Activity Diagrams	29
3.4.5 แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชัน	30
3.4.5.1 ชุดคำถามสำหรับบุคคลทั่วไป	30
3.4.5.2 ชุดคำถามสำหรับนักพยาธิวิทยา	30
บทที่ 4 ผลการดำเนินงาน	33
4.1 การวิเคราะห์ข้อมูลและการทดสอบ	33
4.1.1 การทดสอบแต่ละแบบจำลองและปรับพารามิเตอร์	33
4.1.1.1 Mask R-CNN	33
4.1.1.2 YOLO	33
4.1.1.3 U-Net	33
4.1.1.4 การประมวลผลภาพ (Image Processing)	35
4.1.2 การเลือกแบบจำลอง	36
4.2 ความพึงพอใจในการใช้เว็บแอปพลิเคชัน	37
4.3 ผลการทดสอบระบบ	38
บทที่ 5 บทสรุป	42
5.1 สรุปผลงานวิจัย	42
5.2 สรุปผลการดำเนินงาน	42
5.3 ปัญหาที่พบและแนวทางการแก้ไข	42
5.4 ประโยชน์ที่ได้รับจากการดำเนินงาน	43
5.5 แนวทางการพัฒนาและประยุกต์ใช้ในอนาคต	43
หนังสืออ้างอิง	44
APPENDIX	48
A รายละเอียดวิธีการใช้งานเว็บแอปพลิเคชัน	49
B แบบประเมินความพึงพอใจ	52

สารบัญตาราง

ตารางที่	หน้า
2.1 Mask R-CNN architecture	8
2.2 YOLOv5 architecture	9
2.3 U-Net architecture	11
2.4 RA U-Net architecture	12
2.5 ตารางสรุปงานวิจัยที่ใช้ Deep Learning	18
2.6 ตารางสรุปงานวิจัยที่ใช้ Image Processing	18
2.7 ตารางสรุปผลิตภัณฑ์ที่เกี่ยวข้อง	19
3.1 ตารางการปรับพารามิเตอร์ของแบบจำลอง Mask R-CNN	22
3.2 ตารางการปรับพารามิเตอร์ของแบบจำลอง YOLO	22
3.3 ตารางการปรับพารามิเตอร์ของแบบจำลอง U-Net	23
4.1 ตารางผลการทดสอบของแบบจำลอง Mask R-CNN	33
4.2 ตารางผลการทดสอบของแบบจำลอง YOLO	35
4.3 ตารางผลการทดสอบของแบบจำลอง U-Net	35
4.4 ตารางผลการทดสอบการปรับพารามิเตอร์ของแบบจำลอง Image Processing	35
4.5 ตารางรวมผลการทดสอบที่ดีที่สุดของแต่ละแบบจำลอง	36
4.6 ตารางความพึงพอใจในการใช้เป็นแอปพลิเคชันของบุคคลทั่วไป	37
4.7 ตารางความพึงพอใจในการใช้เป็นแอปพลิเคชันของนักพยาธิวิทยา	38
4.8 ตารางสรุปผลการทดสอบระบบในแต่ละกรณี	38
4.9 การ Build และ Deploy บน Docker Desktop	39
4.10 การ Build ด้วย GCP Cloud Build	39
4.11 การ Deploy ด้วย GCP Cloud Run และใช้ Firebase Hosting	40
4.12 การอัปโหลดรูปภาพ	40
4.13 การแสดงรูปผลลัพธ์	40
4.14 การดาวน์โหลดรูปภาพผลลัพธ์	41
4.15 การดาวน์โหลดตารางผลลัพธ์	41
5.1 ตารางสรุปสถานะการดำเนินงาน	42

สารบัญ

รูปที่	หน้า
1.1 ตารางการดำเนินงานในภาคการศึกษาที่ 1	2
1.2 ตารางการดำเนินงานในภาคการศึกษาที่ 2	2
2.1 ภาพขยายของนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง	3
2.2 ตัวอย่างการกระจายค่า Histogram Equalization	4
2.3 หลักการกระจายค่า Histogram ของ CLAHE	5
2.4 Dilation และ Erosion	5
2.5 Opening และ Closing	6
2.6 Watershed	6
2.7 แบบจำลอง R-CNN	7
2.8 แบบจำลอง Mask R-CNN	7
2.9 แบบจำลอง YOLO	8
2.10 แบบจำลอง U-Net	10
2.11 A. Convolution block, B. Residual Convolution block	10
2.12 แบบจำลอง Residual Attention U-Net	10
2.13 Image Augmentation	11
2.14 รูปตัวอย่างจากเปเปอร์ DeepHCS	14
2.15 รูปตัวอย่างจากเปเปอร์ Tox_rcnn	15
2.16 รูปตัวอย่างจากเปเปอร์ A means	15
2.17 ตัวอย่างภาพถ่ายนิวเคลียสในงานวิจัยของ Wafaa Rajaa Drioua และคณะ	16
2.18 รูปตัวอย่างจากเปเปอร์ automated segment	16
3.1 แผนภาระละเอียดภาระรวมการดำเนินงานวิจัย	20
3.2 ภาพที่ใช้ในการพัฒนาแบบจำลอง	21
3.3 รูปตัวอย่างจาก augment	21
3.4 ค่าพื้นที่จริงของนิวเคลียส (ซ้าย), ผลลัพธ์จากแบบจำลอง U-Net (กลาง) และ ผลลัพธ์จาก Watershed (ขวา)	23
3.5 imks-process	24
3.6 โครงสร้างของเว็บแอปพลิเคชัน	26
3.7 ส่วนหน้าแรกของเว็บแอปพลิเคชัน	27
3.8 ส่วนอัปโหลดของเว็บแอปพลิเคชัน	27
3.9 ส่วนผลลัพธ์	28
3.10 UML Design	28
3.11 Activity Diagrams	29
3.12 แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันบุคคลทั่วไป	31
3.13 แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันของนักพยาธิวิทยา	32
4.1 ผลลัพธ์จากแบบจำลอง Mask R-CNN	33
4.2 ผลลัพธ์จากแบบจำลอง YOLO	34
4.3 ผลลัพธ์จากแบบจำลอง U-Net	34
4.4 ตาราง confusion matrix	36
A.1 ส่วนหน้าแรกของเว็บแอปพลิเคชัน	49
A.2 หน้าเว็บแอปพลิเคชัน	49
A.3 ผลลัพธ์หลังการประมวลผลเสร็จสิ้น	50
A.4 ผลลัพธ์หลังการขยายภาพ	50
B.1 แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันบุคคลทั่วไป	52
B.2 แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันของนักพยาธิวิทยา	53

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญ

นิวเคลียสเป็นส่วนประกอบหนึ่งของเซลล์ หน้าที่สำคัญของนิวเคลียสจะเป็นตัวควบคุมการทำกิจกรรมต่าง ๆ ของเซลล์ [1] เป็นหน่วยที่ใช้เก็บหัտพันธุกรรม และมีหน้าที่ในการผลิตโปรตีน[2] การนับจำนวนนิวเคลียสจะมีความสำคัญทางพยาธิวิทยา มีส่วนช่วยในการวินิจฉัยโรคหรือพัฒนาการรักษา โดยการนับจำนวนนิวเคลียสและปริมาณโปรตีนเป็นหนึ่งในขั้นตอนการทำงานในห้องปฏิบัติการ โดยการย้อมเนื้อเยื่อด้วยสารเรืองแสงแล้วถ่ายภาพด้วยเครื่อง Fluorescence Microscopy ซึ่งจะได้ผลลัพธ์เป็นภาพถ่ายของเนื้อเยื่อที่มีสีระดับตามที่แตกต่างกันไปตามการย้อมนั้น ๆ โดยสีที่มีอยู่ที่ติดบริเวณนิวเคลียสจะเป็น DAPI ซึ่งเป็นสีน้ำเงิน และสีที่ติดโปรตีนจะเป็น FITC เป็นสีเขียว และเนื่องจากนิวเคลียสจะทำหน้าที่ในการผลิตโปรตีน ทำให้ปริมาณของนิวเคลียสที่มีจำนวนมากในภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงก็จะทำให้มีปริมาณของโปรตีนที่มากตามไปด้วย

Columbus [3] เป็นแพลตฟอร์มที่ช่วยในการวิเคราะห์ภาพและนับจำนวนนิวเคลียสที่ทำงานได้อย่างแม่นยำเมื่อรูปเนื้อเยื่อที่ถูกย้อมสีนั้นมีนิวเคลียสที่แยกตัวกันเป็นเซลล์เดียวอย่างชัดเจน และลักษณะของเซลล์ที่เป็นรูปร่างคล้ายคลึงกัน นักพยาธิวิทยาจึงใช้โปรแกรมนี้ในการช่วยนับจำนวนของนิวเคลียสในเนื้อเยื่อนั้น ๆ ได้ แต่เมื่อเปลี่ยนเป็นภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงที่ซ้อนทับกันหลายชั้น จะไม่สามารถใช้งาน Columbus ได้ซึ่งชุดข้อมูลภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงที่คนละผู้วิจัยให้ความสนใจนั้นจะมีปัญหาเรื่องการซ้อนทับกัน ลักษณะของนิวเคลียสมีการรวมกลุ่มกันและติดกันเป็นจำนวนมาก และส่วนใหญ่นิวเคลียสมีลักษณะรูปร่างที่ไม่แน่นอน นอกจากนั้นบริเวณขอบของนิวเคลียสยังเบลอไม่ชัดเจนส่งผลให้สายตาของมนุษย์ไม่สามารถแยกนิวเคลียสออกจากกันได้แบบทันที ต้องใช้เวลาวิเคราะห์ผลที่ค่อนข้างนาน และผลลัพธ์ที่ได้อาจจะไม่แม่นยำหรือคลาดเคลื่อนได้

จากการศึกษาพบว่ามีการนำเทคโนโลยีมาใช้นับจำนวนและระบุตำแหน่งของนิวเคลียส ตัวอย่างเช่น งานวิจัยของ Gyuhyun Lee และคณะ [4] ศึกษาหาตำแหน่งของนิวเคลียส จากการถ่ายเรืองแสงจากกล้องจุลทรรศน์ของเนื้อเยื่อผิวหนังที่ถูกย้อมด้วยสารเรืองแสง DAPI, งานวิจัยของ Daniel Jimenez-Carretero และคณะ [5] ศึกษาการเรียนรู้ซึ่งลึกที่ทำให้สามารถจำแนกรูปภาพแบบที่มีความซับซ้อนได้อย่างมีประสิทธิภาพ และใช้โครงข่ายประสาทแบบคอนโว卢ชันในการทำนายความเป็นพิษของยาจากภาพนิวเคลียสที่ถูกย้อมด้วยสารเรืองแสง DAPI และ งานวิจัยของ Wafaa Rajaa Drioua และคณะ [6] ศึกษาหาวิวเคลียสซึ่งจำเป็นสำหรับการระบุลักษณะของมะเร็ง โดยทำการเรียนรู้ซึ่งลึกมาใช้เพื่อแยกจำนวนนิวเคลียส จากการถ่าย breast cancer histopathology จากการศึกษางานวิจัยดังที่กล่าวมาพบว่าส่วนใหญ่จะให้ความสนใจกับการนับจำนวนนิวเคลียสจากแบบจำลองการเรียนรู้ซึ่งผลลัพธ์ที่ได้เป็นที่น่าพอใจ แต่ด้วยรูปแบบข้อมูลของงานวิจัยเหล่านี้เป็นภาพเซลล์เดียวที่ไม่ซ้อนทับกัน คณะผู้วิจัยจึงนำแนวทางการเรียนรู้ซึ่งลึกมาเป็นตัวพัฒนาแบบจำลองสำหรับนับจำนวนนิวเคลียสที่มีการซ้อนทับกันต่อไป

ดังนั้นจากการศึกษาข้างต้น ทำให้คณาจารย์และนักวิจัยเล็งเห็นถึงปัญหานในการนับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงที่มีการซ้อนทับกัน จึงมีแนวคิดที่จะพัฒนาแบบจำลองที่สามารถตรวจจับตำแหน่งของนิวเคลียสและนับจำนวนของนิวเคลียส และพัฒนาให้ออกแบบเป็นเว็บแอปพลิเคชันที่สามารถใช้นับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงได้อัตโนมัติ เพื่อช่วยลดเวลาในการนับจำนวนนิวเคลียสของนักพยาธิวิทยา และเพื่อเพิ่มความแม่นยำในการนับจำนวนนิวเคลียสให้มีความแม่นยำมากยิ่งขึ้น

1.2 วัตถุประสงค์

- เพื่อพัฒนาแบบจำลองสำหรับนับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง
- เพื่อพัฒนาเว็บแอปพลิเคชัน ที่สามารถทำการนับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงได้

1.3 ขอบเขตของงานวิจัย

เป้าหมายของงานวิจัยคือต้องการพัฒนาแบบจำลองที่สามารถตรวจจับตำแหน่งของนิวเคลียสและนับจำนวนของนิวเคลียสได้ และพัฒนาเว็บแอปพลิเคชัน ที่สามารถทำการนับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงได้อย่างถูกต้อง และมีประสิทธิภาพ เพื่อช่วยลดเวลาในการนับจำนวนนิวเคลียสของนักพยาธิวิทยา และเพื่อเพิ่มความแม่นยำในการนับจำนวนนิวเคลียสให้มีความแม่นยำมากยิ่งขึ้น ดังนั้นขอบเขตของงานวิจัยจึงมีดังนี้

- ศึกษาวิธีการตรวจหาตำแหน่งและนับจำนวนนิวเคลียส ด้วยวิธีการประมวลผลภาพ และการเรียนรู้เชิงลึก เพื่อนำไปสู่การพัฒนาแบบจำลอง
- เพื่อพัฒนาแบบจำลองที่สามารถตรวจหาตำแหน่งและนับจำนวนนิวเคลียสในภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง
- เลือกวิธีการที่ดีที่สุดที่ได้จากการพัฒนาแบบจำลองทั้งหมด เพื่อนำมาใช้กับเว็บแอปพลิเคชัน
- พัฒนาเว็บแอปพลิเคชัน ที่สามารถนับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

จากการสร้างเว็บแอปพลิเคชันสำหรับการตรวจนับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง จะช่วยลดเวลาในการนับจำนวนนิวเคลียส เพิ่มความถูกต้องและความแม่นยำในการนับจำนวนนิวเคลียสสำหรับนักพยาธิวิทยาที่ทำเกี่ยวกับภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงและเพิ่มประสิทธิภาพในการแปลผลจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงได้มากขึ้น

1.5 ตารางการดำเนินงาน

ผลการดำเนินงานในภาคการศึกษาที่ 1 ทำการศึกษาข้อมูลและกระบวนการที่เกี่ยวกับอัลกอริทึมในการนับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง และจัดทำรายงานประจำภาคการศึกษาที่ 1 ดังรูปภาพที่ 1.1

ผลการดำเนินงานในภาคการศึกษาที่ 2 เว็บแอปพลิเคชันสำหรับการใช้งานอัลกอริทึมในการนับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง และรายงานประจำภาคการศึกษาที่ 2 ดังรูปภาพที่ 1.2

Task	August				September				October				November				December				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1.ศึกษาอุปกรณ์ของหัวขอร่องคอกที่ได้รับ																					
2.จดท่าขอเสนอหัวขอร่องคอก (Project Idea)																					
3.จดท่าขอเสนอโครงการ (Project Proposal)																					
4.ศึกษาเทคโนโลยีและซอฟต์แวร์ที่จะใช้																					
a.การตรวจสอบความแม่นยำ																					
b.การทาร์วิบและอัลกอริทึม																					
c.การหาตัวอย่างและการสร้างไปร์ติน																					
5.จัดทำรายงานประจำภาคการศึกษาที่ 1																					
6.จดท่าเอกสารนำเสนอประกอบภาระเรียนศึกษาที่ 1																					

รูปที่ 1.1: ตารางการดำเนินงานในภาคการศึกษาที่ 1

Task	January				February				March				April				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
7.ให้ผู้เชี่ยวชาญหาด label ขอบเขตของนิวเคลียส																	
8.ปรับปรุงคุณภาพของรูปภาพ เพิ่มความคมชัดของรูป																	
9.ออกแบบและพัฒนาอัลกอริทึมในการนับจำนวนเซลล์																	
10.ทดสอบและปรับแต่งอัลกอริทึมในการนับจำนวนเซลล์																	
11.ออกแบบและพัฒนาเว็บแอปพลิเคชัน																	
12.ทดสอบและปรับปรุงเว็บแอปพลิเคชัน																	
13.จัดทำรายงานประจำภาคการศึกษาที่ 2																	
14.จัดทำเอกสารนำเสนอประกอบภาระเรียนศึกษาที่ 2																	

รูปที่ 1.2: ตารางการดำเนินงานในภาคการศึกษาที่ 2

บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงรายละเอียดทฤษฎีและความรู้ที่เกี่ยวข้องกับการนำการประมวลผลภาพ (Image Processing) และการเรียนรู้เชิงลึก (Deep Learning) มาใช้ในการวิเคราะห์ภาพถ่ายทางพยาธิวิทยา โดยจะกล่าวถึงหลักการ ทฤษฎีที่เกี่ยวข้องที่ใช้ในการทำงานและการศึกษาเพื่อบรรลุวัตถุประสงค์ การอธิบายภาษาทางคอมพิวเตอร์ที่ได้ก่อตัวถึงและใช้งานในการแก้ไขปัญหา และสุดท้ายคืองานวิจัยที่เกี่ยวข้องกับงานวิจัยนี้

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 ด้านพยาธิวิทยา

2.1.1.1 อิมมูโนฟลูอเรสเซนซ์ (Immunofluorescence)

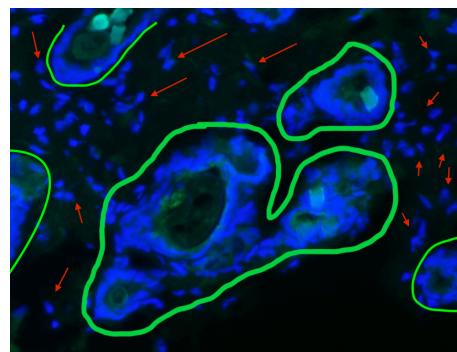
อิมมูโนฟลูอเรสเซนซ์เป็นเทคนิคการวินิจฉัยโดยใช้หลักการจับแอนติเจนและแอนติบอดี การตรวจหาจากฟลูอเรสเซนซ์ ใช้กันอย่างแพร่หลายเพื่อศึกษาตำแหน่งของโปรตีน และสถานะการทำงานของโปรตีนเป้าหมายในนิวเคลียสรึเนื้อเยื่อ เทคนิคนี้มีความไวและความจำเพาะต่อแอนติเจนจำเพาะสูง โดยแอนติบอดีปฐมภูมิจะจับกับโปรตีนที่สนใจ และแอนติบอดีทุติยภูมิจะจับกับแอนติบอดีปฐมภูมิอีกที่หนึ่ง ซึ่งแอนติบอดีปฐมภูมิจะทำหน้าที่จับกับสารเรืองแสงประเภทต่าง ๆ [7]

2.1.1.2 สารเรืองแสง (Fluorochromes)

สารเรืองแสง คือ สิ่งที่มีคุณสมบัติเปล่งแสงได้ สิ่งนี้มีคุณสมบัติที่จะเข้ามาปฏิริยَاกับนิวเคลียสหรือองค์ประกอบอยู่ข้างนิวเคลียสอย่างจำเพาะ ตัวอย่างเช่น 4',6-diamidino-2-phenylindole (DAPI) จะทำปฏิริยَاกับกรดนิวคลีิก (Nucleic Acids) โดยจับกับบริเวณ A-T base pair ของ DNA จึงเป็นที่นิยมใช้ในการบ่งบอกตำแหน่งของนิวเคลียส DAPI จะให้สีน้ำเงินเมื่อถูกกระตุ้นนักพยาธิวิทยาสามารถใช้สิ่ย้อมที่หลากหลายโดยมีความจำเพาะต่อส่วนต่าง ๆ ของนิวเคลียสในการบ่งชี้โครงสร้างของนิวเคลียสอย่างโครงสร้างพังผืด กัน ในการทดลองหลักสีพังผืด กัน ด้วยคุณสมบัติสเปกตรัมของ DAPI ที่เป็นสีน้ำเงินทำให้เหมาะสมสำหรับใช้ร่วมกับสีเขียว (เช่น Invitrogen Alexa Fluor 488, FITC, GFP) และสีแดง (เช่น Invitrogen Alexa Fluor 594, Rhodamine, Invitrogen Texas Red, mCherry, mKate-2) โดยทั่วไป DAPI จะใช้ย้อมสีนิวเคลียสเพื่อติดตามตำแหน่งของนิวเคลียสในเซลล์ ซึ่งจะบ่งบอกรูปักษณะของนิวเคลียส (Nucleus morphology) และจำนวนนิวเคลียส (Nucleus number) ได้ [8]

2.1.1.3 ภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง (Immunofluorescence Tissue)

ภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง คือภาพถ่ายที่ได้มาจากการย้อมสีด้วยสารเรืองแสง ซึ่งสามารถบ่งบอกตำแหน่งของนิวเคลียสในตัวอย่างเนื้อเยื่อด้วยการใช้สิ่ย้อม DAPI ซึ่งจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงนั้นจะพบว่ามีจำนวนนิวเคลียสร่วมตัวอยู่มาก ดังที่แสดงในรูปที่ 2.1 [9] ซึ่งจะประกอบด้วยสองส่วน คือ ส่วนของเซลล์เดียวที่ลอยแยกกันเดี่ยว ๆ ดังบริเวณที่ลูกศรสีแดงชี้อยู่ อีกส่วนคือเนื้อเยื่อ ซึ่งเป็นปัญหาที่ค่อนข้างวุ่นวายในที่จะแก้ไข ดังบริเวณกรอบสีเขียว



รูปที่ 2.1: ภาพขยายของนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง

2.1.2 ด้านคอมพิวเตอร์

2.1.2.1 การประมวลผลภาพ (Image Processing)

เป็นวิธีการดำเนินการประมวลผล ปรับปรุง และวิเคราะห์รูปภาพที่อยู่ในรูปแบบดิจิทัลโดยการใช้คอมพิวเตอร์ เพื่อให้ได้มาซึ่งรูปภาพที่มีลักษณะที่ต้องการ ทั้งในเชิงคุณภาพและปริมาณ [10] สามารถนำข้อมูลที่ได้จากรูปภาพไปใช้ประโยชน์ได้ต่อไป ในปัจจุบัน การประมวลผลภาพถูกนำมาใช้งานอย่างกว้างขวางในด้านงานวิจัยทางการแพทย์ ซึ่งจะช่วยทำให้สามารถวางแผนการรักษาได้อย่างมีประสิทธิภาพและเพิ่มความถูกต้องแม่นยำในการรักษาให้มากยิ่งขึ้น

- **RGB to Grayscale** คือการแปลงภาพสี RGB ให้เป็นภาพ Grayscale ที่มีระดับสี 256 แบบ จาก 0 (ดำ) ถึง 255 (ขาว) ซึ่งสามารถทำได้หลายวิธี เช่นการใช้สมการในการหาค่าเฉลี่ยของ 3 ค่าสีบน RGB [11] หรือวิธีการคูณค่าคงที่กับแต่ละค่าสีของ RGB [12] ดังสมการ 2.1

$$\text{Grayscale} = (w_1 * R) + (w_2 * G) + (w_3 * B) \quad (2.1)$$

เมื่อ $1 = w_1 + w_2 + w_3$ และ $0 < w_1, w_2, w_3$

Grayscale = รูปผลลัพธ์ในช่วงของสีเทา

R = ช่วงสีแดง G = ช่วงสีเขียว B = ช่วงสีน้ำเงิน

w_1 = น้ำหนักของสีแดง w_2 = น้ำหนักของสีเขียว และ w_3 = น้ำหนักของสีน้ำเงิน

- **Histogram Equalization (HE)** เป็นกระบวนการปรับปรุงค่า Contrast ในรูปภาพ โดยจะทำการยืดค่า Histogram ออกให้กระจายเท่า ๆ กันในแต่ละช่วง ดังนั้นค่าความเข้มแสง (Intensity) จึงกระจายเท่า ๆ กันทั้งช่วงของ Histogram ดังรูปที่ 2.2 ซึ่งการคำนวนค่า Equalization สามารถได้จากการคำนวนค่า Cumulative Distribution Function (CDF) เอียนได้ดังสมการ 2.2

$$H'(i) = \sum_{0 \leq j < i} H(j) \quad (2.2)$$

เมื่อ $H(i)$ = Histogram

$H'(i)$ = Cumulative Histogram

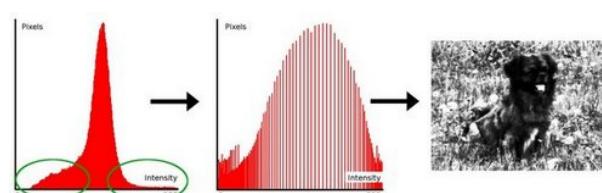
i = Max intensity (default is 255)

จากนั้นทำการ Normalize $H'(i)$ ให้อยู่ในช่วงค่าความเข้มแสง (Intensity) สูงที่สุดของรูปภาพ [13] เอียนได้ดังสมการ 2.3

$$\text{Equalized} = H'(i)(\text{src}) \quad (2.3)$$

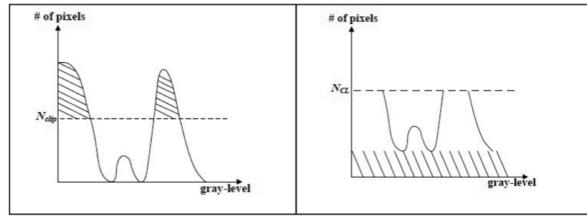
เมื่อ $\text{src}_{(x,y)}$ = Input Image

Equalized = ผลลัพธ์ของ HE



รูปที่ 2.2: ตัวอย่างการกระจายค่า Histogram Equalization [14]

- **Contrast Limited Adaptive Histogram Equalization (CLAHE)** เป็นกระบวนการหนึ่งในกลุ่มของ Adaptive Histogram Equalization (AHE) ซึ่งจะให้ความสนใจในการขยายสัญญาณที่มากเกินไปของ Contrast ช่วยปรับปรุงค่า Contrast ของรูปให้ดียิ่งขึ้น กระบวนการ CLAHE จะไม่ทำงานทั้งรูป แต่จะทำงานบนช่วงเล็ก ๆ ของรูปเรียกว่า Tile โดยชิ้นส่วน Tile อื่น ๆ จะถูกนำมารวมด้วย Bilinear Interpolation เพื่อลบ Artificial Boundaries [15] โดยค่า Histogram ที่มีระดับสูงกว่าค่าเฉลี่ยของพิกเซลจะถูกนำมากระจายให้ทุกพิกเซลในภาพ ดังรูปที่ 2.3. [16]



รูปที่ 2.3: หลักการกระจายค่า Histogram ของ CLAHE [17]

- **Thresholding** กระบวนการแปลงภาพเป็นภาพ Binary หรือภาพที่มีเฉพาะสีดำคือค่า Intensity เป็น 0 และสีขาวคือค่า Intensity เป็น 1 เพื่อแยกวัตถุออกจากพื้นหลังอย่างชัดเจน โดยพิจารณาจากค่า Intensity ของแต่ละ Pixel เปรียบเทียบกับ Threshold Value ถ้าค่า Intensity มากกว่า Threshold Value จะแปลงค่า Intensity เป็น 1 หรือสีขาว และถ้าค่า Intensity น้อยกว่า Threshold Value จะแปลงค่า Intensity เป็น 0 หรือสีดำ ซึ่งโดยทั่วไปจะทำกับภาพ Grayscale มีระดับสีหรือค่า Intensity 0 (ดำ) ถึง 255 (ขาว) [18] เนื่องจากภาพจะมีรูปแบบเก็บข้อมูลแบบ 2 ส่วน คือ Foreground(วัตถุ หรือสิ่งที่สนใจ) และ Background(พื้นหลังของภาพ) ซึ่งจะสามารถจำแนกวัตถุյ่างกว่าภาพสี [19]
- **Gaussian Blur** กระบวนการเบลอภาพด้วยการใช้ค่าเฉลี่ยแบบถ่วงน้ำหนัก ที่ใช้ฟังก์ชัน gauss เซียนในการคำนวณ แปลงค่าแต่ละพิกเซลเพื่อลบสัญญาณรบกวนในรูปภาพทำให้ความคมชัดของรูปภาพลดลง ซึ่งภาพจะเบลอน้อย และคงเส้นขอในภาพได้ โดยใช้สมการฟังก์ชัน gauss เซียน [20] ดังสมการ 2.4

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (2.4)$$

เมื่อ $G(x)$ = ผลลัพธ์กระบวนการเบลอภาพด้วย Gaussian Blur

x = ระยะทางจากจุดกำหนดในแนวอน

σ = ค่าเบี่ยงเบนมาตรฐานของการแจกแจงแบบ gauss เซียน

- **Morphological Operation** เทคนิคสำหรับการวิเคราะห์และประมวลผลภาพ โดยการเปลี่ยนแปลงลักษณะของรูปร่าง หรือโครงสร้างของภาพ ซึ่งอาศัยโครงสร้างทางเรขาคณิตบนพื้นฐานของทฤษฎีเซต ทฤษฎีตัวข่ายโครงสร้างของเครือข่าย และฟังก์ชันแบบสุ่ม การประมวลผลจะเปรียบเทียบพิกเซลในภาพกับพื้นที่ใกล้เคียงโดยการเลือกขนาดและรูปร่างของพื้นที่ใกล้เคียงจะมีตัวดำเนินการหลายแบบ เช่น Dilation การขยายภาพโดยมีสัดส่วนเท่ากันทั้งภาพตัวอย่างในภาพที่ 2.4b Erosion คือการกร่อนขนาดของบริเวณของของวัตถุตัวอย่างในภาพที่ 2.4c Skeleton จะเป็นการทำโครงสร้างหลักของวัตถุ Opening การนำข้อมูลภาพ ผ่านการขยายภาพ (Erosion) และตามด้วยการกร่อนภาพ (Dilation) โดยใช้ Template เมื่ອนกันตัวอย่างในภาพที่ 2.5a และ Closing จะทำการกร่อนภาพก่อนแล้วตามด้วยการขยายภาพ โดยใช้ template ชุดเดียวกันตัวอย่างในภาพที่ 2.5b [21]



(a) ภาพต้นฉบับ



(b) ภาพที่ผ่านกระบวนการ Dilatation



(c) ภาพที่ผ่านกระบวนการ Erosion

รูปที่ 2.4: Dilation และ Erosion [22]



รูปที่ 2.5: Opening และ Closing[22]

- **Watershed** เป็นวิธีหนึ่งในการทำ Image Segmentation โดยมีหลักพื้นฐานมาจากการที่มองว่ารูปภาพ Grayscale สามารถมองเป็นระดับความสูงได้จากค่า Intensity เรียกว่า Topographic surface ดังรูปภาพที่ 2.6a ดังนั้นวัตถุจะสูงขึ้นมาจากพื้นหลัง ถ้าเราระคาย ๆ เทhn้ำลงไปในรูปร่องอย่าง ๆ ระดับน้ำจะเพิ่มสูงขึ้นเรื่อยๆ จนเหลือเพียงแค่ส่วนยอดที่เหนือพื้นน้ำ ซึ่งจะได้ผลลัพธ์เป็นการแบ่งของวัตถุ ดังรูปภาพที่ 2.6b

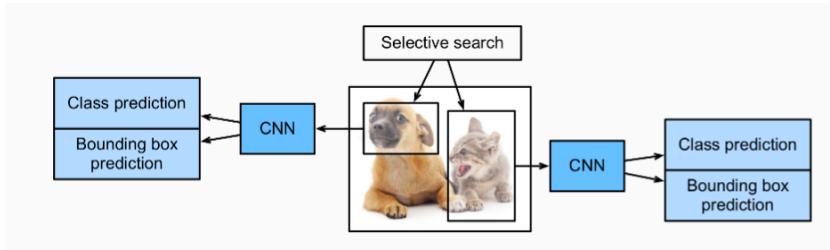


รูปที่ 2.6: Watershed [23]

2.1.2.2 การเรียนรู้เชิงลึก (Deep Learning)

การเรียนรู้เชิงลึก คือ การพัฒนาเทคโนโลยีคอมพิวเตอร์ให้สามารถเลียนแบบการทำงานของมนุษย์ ซึ่งการเรียนรู้เชิงลึกจะมีกระบวนการคิดคำนวนคล้ายกับระบบโครงข่ายประสาท (Neurons) ของสมองมนุษย์เรียกว่า โครงข่ายประสาทเทียม (Neural Network: NN) ความสามารถของการเรียนรู้เชิงลึกสามารถสร้างแบบจำลองและหาคำตอบได้ด้วยการนำโครงข่ายประสาทเทียมหลายชั้นเรียกว่า Hidden Layer มาใช้เคราะห์และหาคำตอบ ซึ่งคำว่าการเรียนรู้เชิงลึกก็มาจากการใช้โครงข่ายประสาทเทียมมากกว่า 2 ระดับขึ้นไปในการสร้างแบบจำลอง ดังนั้นจึงเปรียบได้ว่าจำนวนระดับชั้นของโครงข่ายประสาทเทียมยิ่งสูงใช้จำนวนมากในขั้นตอนการประมวลผล ยิ่งทำให้มีโครงสร้างการเรียนรู้ที่ลึกมากขึ้น ถัดไปจะเป็นการยกตัวอย่างแบบจำลองที่พบในงานวิจัยและมีการนำมาใช้งานกันอย่างแพร่หลาย ได้แก่

- **Region-based CNNs (R-CNN)** เป็นหนึ่งในแนวทางการนำการเรียนรู้เชิงลึกมาใช้กับการตรวจจับวัตถุ โดย R-CNN จะทำการแยกรูปภาพออกจากรูป input หลังจากนั้นจะใช้โครงข่ายประสาทคอนโวลูชัน (Convolutional Neural Network) ในการส่งต่อข้อมูลเพื่อทำการแยกคุณลักษณะต่าง ๆ ต่อไป โดยคุณสมบัติ R-CNN คือต้องการศึกษาการทำนายวัตถุที่ตรวจจับได้ของแต่ละ region โดยกระบวนการทำงานของ R-CNN สามารถแบ่งอยู่ได้เป็น 4 ขั้นตอนหลัก ๆ คือ ทำการเลือกขอบเขตที่สนใจจากรูป input โดยส่วนใหญ่ขอบเขตที่ได้เลือกนั้นจะมีรูปร่างหรือขนาดที่แตกต่างกัน โดยแต่ละขอบเขตจะทำการระบุตำแหน่งของ class และ ground-truth ไว้ ต่อมาจะนำ pretrained CNN ที่ได้ฝึกไว้มาทำการปรับขนาดของรูปที่ได้ก่อนที่จะทำการแยกคุณลักษณะต่าง ๆ ถัดมาจะมีการนำ support vector machine มาใช้ในการจำแนกวัตถุ และทำการฝึกอบรมแบบจำลอง linear regression เพื่อใช้ในการทำนาย region ตามความเป็นจริง โดยมีโครงสร้างดังที่รูปภาพที่ 2.7 [24]

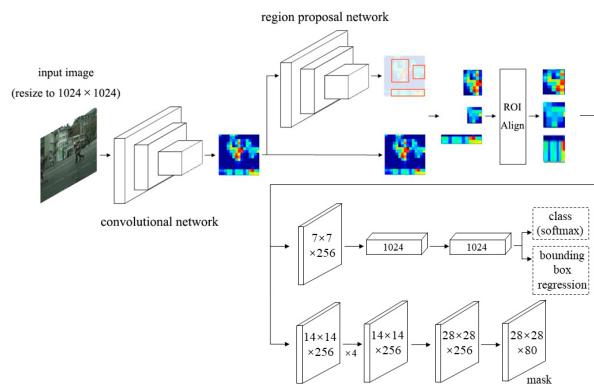


รูปที่ 2.7: แบบจำลอง R-CNN [25]

- Mask R-CNN Mask R-CNN ปรับปรุงจาก Faster R-CNN ซึ่งพัฒนามาจาก Region-based CNNs โดยการเพิ่มการทำนาย mask ของวัตถุ โดยจะทำงานควบคู่ไปกับ Faster R-CNN ที่จะจัดจำลองเขตของวัตถุ มีโครงสร้างดังรูปภาพที่ 2.8 มีพื้นฐานแนวคิดที่เรียบง่าย ยืดหยุ่น และทั่วไป และมีแนวทางตรวจจับวัตถุในภาพได้อย่างมีประสิทธิภาพสูงในการแบ่งแต่ละ mask นอกจากนี้ Mask R-CNN มีลักษณะการเรียนรู้ที่ง่ายและมีประสิทธิภาพ ช่วยลดความยากในงานวิจัยในอนาคต โดยสถาปัตยกรรมโครงสร้างของ Mask R-CNN แต่ละขั้นตอนสามารถอธิบายได้ ดังนี้

- ภาพชุดข้อมูลจะถูกส่งผ่านโคลงข่ายคอนโว Luisun
- ผลลัพธ์ที่ได้จากการทำโคลงข่ายคอนโว Luisun และจะถูกส่งผ่านไปยัง Region Proposal network (RPN) ซึ่งจะสร้าง anchor box (ตำแหน่งที่สนใจ) ที่แตกต่างกันตามการมีอยู่ของออบเจกต์นั้น ๆ ที่จะตรวจจับได้
- Anchor box จะถูกส่งไปยัง ROI Align stage (เป็นหนึ่งในฟีเจอร์หลักของ Mask RCNN สำหรับการป้องกันการจัดวางเชิงพื้นที่) ซึ่งจะแปลง ROI ให้เป็นขนาดเดียวกันกับที่จำเป็นสำหรับการประมวลผลเพิ่มเติม
- ผลลัพธ์นี้ถูกส่งไปยังレイเยอร์ที่เชื่อมต่อ กัน ซึ่งจะสร้างผลลัพธ์ของคลาสของวัตถุในพื้นที่นั้น ๆ และหาตำแหน่งของกล่องของวัตถุ
- ผลลัพธ์ของระยะ ROI Align จะถูกส่งไปยัง Conv Nets เพื่อสร้างหน้ากากของพิกเซลของวัตถุนั้น

ซึ่งแบบจำลองนี้จะมีพารามิเตอร์ทั้งหมดประมาณ 64 ล้านพารามิเตอร์ [26] มีโครงสร้างแบบจำลองดังตารางที่ 2.1 โดยคุณผู้วจัยได้เลือกวิธีการ Mask R-CNN ในการทดลองแทน R-CNN เนื่องจาก R-CNN ใช้ Support Vector Machine ในการแยกประเภทและต้องรู้จำนวนที่อยู่ทุกอันในลักษณะเหมือนริมจากศูนย์ทุกครั้ง นั่นคือถ้า region ถูกเสนอมา 2,000 ส่วน ก็ต้องทำสมือนการรู้จำนวนที่เป็นอิสระจากกัน 2,000 ครั้ง เท่ากับต้องคำนวน CNN ใหม่ 2,000 รอบ R-CNN จึงใช้ Computational Resource อุ่งมาก [27]

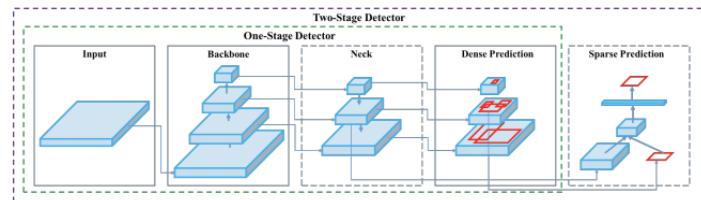


รูปที่ 2.8: แบบจำลอง Mask R-CNN [28]

ตารางที่ 2.1 Mask R-CNN architecture

layers	Description
Input Layers:	
Input layer:	shape = (batch_size, 512, 512, 3)
Backbone Layers:	
Convolutional Block1	7x7, 64, stride 2, padding same, activation ReLU
MaxPooling	3x3, stride 2
Residual Convolutional Block2	3x3, 64 filters, 3 blocks
Residual Convolutional Block3	4x4, 128 filters, 4 blocks
Residual Convolutional Block4	6x6, 256 filters, 23 blocks
Residual Convolutional Block5	3x3, 512 filters, 3 blocks
RPN Layers:	
RPN Convolutional Layers	when: k = Anchor boxes number 3x3, 512 filters, stride 1, padding same, activation ReLU
RPN Classification Layers	1x1, 2k filters, stride 1, padding valid, activation softmax
RPN Regression Layers	1x1, 4k filters, stride 1, padding valid, activation linear
FCN Layers:	
RoIAlign layer	
Convolutional Layers	3x3, 256 filters, stride 1, padding same, activation ReLU
Mask Head Layers:	
Convolutional Layers	3x3, 256 filters, stride 1, padding same, activation ReLU
Convolutional Layers	3x3, 256 filters, stride 1, padding same, activation ReLU
Convolutional Layers	3x3, 256 filters, stride 1, padding same, activation ReLU
Convolutional Layers	3x3, 256 filters, stride 1, padding same, activation ReLU
Convolutional Layers	2x2, 256 filters, stride 2, padding valid, activation ReLU
Convolutional Layers	1x1, num_classes filters, stride 1, padding valid, activation sigmoid

- YOLO อัลกอริทึม You Only Look Once (YOLO) [29] คือระบบตรวจจับภาพอัตโนมัติโดยใช้วิธีการตรวจจับวัตถุแบบเรียนรู้เชิงลึกขั้นสูง สามารถตรวจจับ จำกัด และ จำแนกวัตถุต่างๆในภาพ โดยอัลกอริทึมสามารถตรวจจับได้หลากหลายวัตถุในหนึ่งภาพ สรุปภาพรวมได้ดังแสดงดังรูปภาพที่ 2.9 YOLO เป็นที่รู้จักในด้านความเร็วและความแม่นยำสูง มีโครงสร้างดังรูปภาพที่ 2.9 และ YOLOv5 คืออัลกอริทึม YOLO ในเวอร์ชันที่ 5 เริ่มต้นโดยคุณ Glenn Joche ซึ่งสามารถเข้าถึงโค้ดแบบจำลองจาก GitHub [30] สถาปัตยกรรมของแบบจำลอง YOLOv5 สามารถทำ Image Detection ได้อย่างรวดเร็วและมีประสิทธิภาพ ซึ่งเขียนโดยใช้ภาษา Python และเฟรมเวิร์กที่ใช้คือ PyTorch โดยใน YOLOv5 จะมีแบบจำลองทั้งหมด 5 ขนาด เริ่มจาก YOLOv5 nano (เป็นแบบจำลองเล็กที่สุดและเร็วที่สุด) ไปจนถึง YOLOv5 extra-large (เป็นแบบจำลองใหญ่ที่สุด) ซึ่งจะมีพารามิเตอร์ตั้งแต่ 1.9 - 86.7 ล้านพารามิเตอร์[31] คุณผู้อ่านได้เลือกใช้ YOLOv5 medium โดยโครงสร้างแบบจำลองมีส่วนประกอบดังตารางที่ 2.2

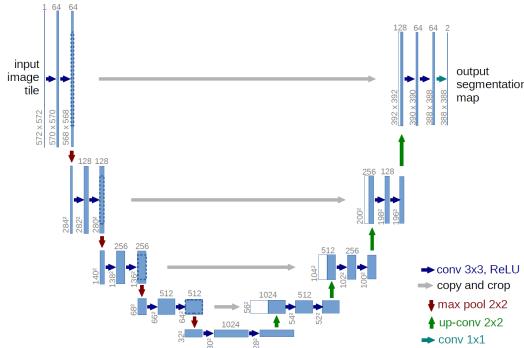


รูปที่ 2.9: แบบจำลอง YOLO [32]

ตารางที่ 2.2 YOLOv5 architecture

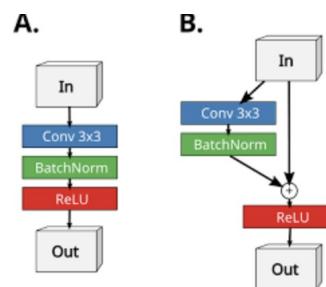
layers	Description
Input	
Input layer:	shape = (batch_size, 640, 640, 3)
Backbone	
Convolutional layers:	3x3 filters, stride=2, padding=1, LeakyReLU activation
CSP block (x3):	a series of convolutional layers followed by a concatenation with the input, followed by more convolutional layers
SPP module:	max pooling at multiple scales, followed by convolutional layers
CSP block (x15):	same as previous CSP block, repeated 15 times
Neck	
Convolutional layer:	1x1 filters, stride=1, padding=0, LeakyReLU activation
Upsample layer:	bilinear upsampling, scale factor=2
Concatenate layer:	concatenates the output of the corresponding CSP block from the backbone with the output of the previous layer
CSP block (x3):	same as previous CSP block, repeated 3 times
Dense prediction	
Convolutional layer:	3x3 filters, stride=1, padding=1, LeakyReLU activation
Convolutional layer:	1x1 filters, stride=1, padding=0, linear activation
YOLO detection layer:	predicts bounding boxes, objectness scores, and class probabilities for the corresponding scale
Sparse prediction	
Convolutional layer:	3x3 filters, stride=2, padding=1, LeakyReLU activation
Concatenate layer:	concatenates the output of the corresponding CSP block from the backbone with the output of the previous layer
CSP block (x3):	same as previous CSP block, repeated 3 times
Convolutional layer:	3x3 filters, stride=1, padding=1, LeakyReLU activation
Convolutional layer:	1x1 filters, stride=1, padding=0, linear activation
YOLO detection layer:	predicts bounding boxes, objectness scores, and class probabilities for the corresponding scale

- U-Net คือหนึ่งโครงข่ายประสาทเทียมที่พัฒนามาจากโครงข่ายประสาทคอนโวลูชัน (Convolutional Neural Network) แบบดั้งเดิม โดยรูปแบบสถาปัตยกรรมของ network นี้มีรูปแบบที่ยอดและขยายข้อมูลในแต่ละ layers ลักษณะคล้ายรูปตัว U ดังรูปภาพที่ 2.10 มีส่วนประกอบ 3 ส่วน โดยผังชั้ยของแบบจำลองจะเรียกว่า Encoder ตรงกลางของแบบจำลองจะเรียกว่า Bottleneck และผังชั้ยของแบบจำลองจะเรียกว่า Decoder ดังตารางที่ 2.3 โดยแบบจำลองนี้ สามารถระบุพื้นที่และแยกเส้นขอบได้ดี ในระดับ pixel space จึงเหมาะสมการโจทย์ Image Segmentation หรือ Image Classification โดย input และ output ของ U-Net เป็นขนาดเท่ากัน [33]

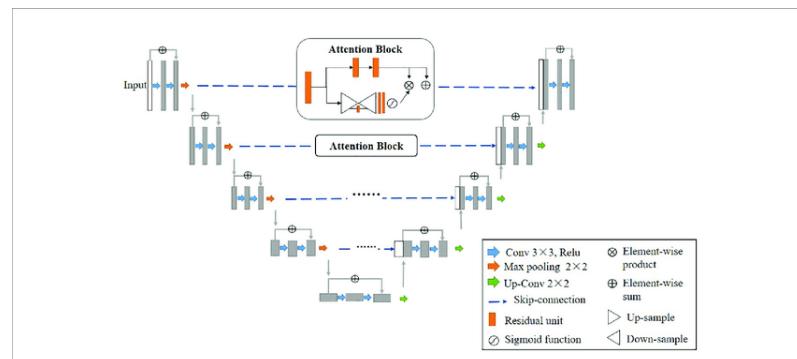


รูปที่ 2.10: แบบจำลอง U-Net [34]

- Residual Attention U-Net (RA U-Net) แบบจำลองที่นำคุณสมบัติของ Residual Convolution block แทนการใช้ Convolution block ใน Encoder โดยจะมีโครงสร้างแบบจำลองมีส่วนประกอบดังตารางที่ 2.4 และความแตกต่างจะแสดงดังรูปที่ 2.11 มาใช้ร่วมกับ Attention block แบบจำลองตามโครงสร้างจากงานวิจัยของ Qiangguo Jin และคณะ [35] แสดงดังรูปที่ 2.12



รูปที่ 2.11: A. Convolution block, B. Residual Convolution block [35]

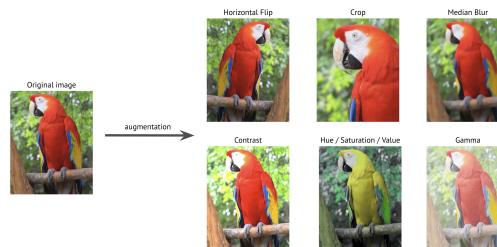


รูปที่ 2.12: แบบจำลอง Residual Attention U-Net [35]

ตารางที่ 2.3 U-Net architecture

layers	Description
Input Stage:	
Input layer:	shape = (batch_size, 256, 256, 3)
Encoding Stage:	
Convolutional Block:	64 filters, 3x3 kernel size, ReLU activation, and same padding
Max pooling layer:	2x2 pool size
Convolutional Block:	128 filters, 3x3 kernel size, ReLU activation, and same padding
Max pooling layer:	2x2 pool size
Convolutional Block:	256 filters, 3x3 kernel size, ReLU activation, and same padding
Max pooling layer:	2x2 pool size
Convolutional Block:	512 filters, 3x3 kernel size, ReLU activation, and same padding
Bottleneck Stage:	
Max pooling layer:	2x2 pool size
Convolutional Block:	1024 filters, 3x3 kernel size, ReLU activation, and same padding
Decoding Stage:	
Up-sampling layer:	2x2 stride
Concatenation layer:	corresponding encoding stage layer
Convolutional Block:	512 filters, 3x3 kernel size, ReLU activation, and same padding
Up-sampling layer:	2x2 stride
Concatenation layer:	corresponding encoding stage layer
Convolutional Block:	256 filters, 3x3 kernel size, ReLU activation, and same padding
Up-sampling layer:	2x2 stride
Concatenation layer:	corresponding encoding stage layer
Convolutional Block:	128 filters, 3x3 kernel size, ReLU activation, and same padding
Up-sampling layer:	2x2 stride
Concatenation layer:	corresponding encoding stage layer
Convolutional Block:	64 filters, 3x3 kernel size, ReLU activation, and same padding
Output layer:	appropriate number of filters and activation function

- **Image Augmentation** คือเทคนิคในการเพิ่มจำนวนชุดข้อมูลซึ่งจะช่วยแก้ปัญหาในกรณีที่ชุดข้อมูลมีขนาดน้อยเกินไป ส่งผลให้เกิดการ over-fit ได้ ในการทำ Image Augmentation สามารถทำได้หลากหลายแบบ เช่น Zoom, Shear, Rotate, Flip และ Preprocessing function อื่นๆ ดังรูปภาพที่ 2.13



รูปที่ 2.13: Image Augmentation [36]

ตารางที่ 2.4 RA U-Net architecture

layers	Description
Input Stage:	
Input layer:	shape = (batch_size, 256, 256, 3)
Encoding Stage:	
Residual Convolutional Block:	64 filters, 3x3 kernel size, and ReLU activation
Max pooling Layer:	2x2 pool size
Residual Convolutional Block:	128 filters, 3x3 kernel size, and ReLU activation
Max pooling Layer:	2x2 pool size
Residual Convolutional Block:	256 filters, 3x3 kernel size, ReLU activation, and same padding
Max pooling Layer:	max pooling with 2x2 pool size
Residual Convolutional Block:	512 filters, 3x3 kernel size, ReLU activation, and same padding
Bottleneck Stage:	
Residual Convolutional Block:	1024 filters, 3x3 kernel size, ReLU activation, and same padding
Decoding Stage:	
Up-sampling layer:	bilinear interpolation with 2x up-sampling factor
Concatenation layer:	corresponding encoding stage layer
Residual Convolutional Block:	512 filters, 3x3 kernel size, ReLU activation, and same padding
Up-sampling layer:	bilinear interpolation with 2x up-sampling factor
Concatenation layer:	corresponding encoding stage layer
Residual Convolutional Block:	256 filters, 3x3 kernel size, ReLU activation, and same padding
Up-sampling layer:	bilinear interpolation with 2x up-sampling factor
Concatenation layer:	corresponding encoding stage layer
Residual Convolutional Block:	128 filters, 3x3 kernel size, ReLU activation, and same padding
Up-sampling layer:	bilinear interpolation with 2x up-sampling factor
Concatenation layer:	corresponding encoding stage layer
Residual Convolutional Block:	64 filters, 3x3 kernel size, ReLU activation, and same padding
Output Stage:	
Convolutional layer:	appropriate number of filters and 1x1 kernel size
Output layer:	appropriate activation function

2.2 ภาษาคอมพิวเตอร์และเทคโนโลยี

2.2.1 Python

ภาษาการเขียนโปรแกรมที่ใช้อย่างแพร่หลายในเว็บแอปพลิเคชัน การพัฒนาซอฟต์แวร์ Data Science และการเรียนรู้ด้วยเครื่องเนื่องจากมีประสิทธิภาพสูง เรียนรู้ง่าย และสามารถทำงานบนแพลตฟอร์มต่าง ๆ ได้มากmany ทั้งนี้ซอฟต์แวร์ ไพทอนสามารถดาวน์โหลดได้ฟรี ผ่านการทำงานร่วมกับระบบทุกประเภท และเพิ่มความเร็วในการพัฒนา [37]

2.2.2 OpenCV

ไลบรารีพังก์ชันการเขียนโปรแกรมแบบข้ามแพลตฟอร์ม โดยจะมีป้าหมายไปที่การแสดงผลด้วยคอมพิวเตอร์แบบเรียลไทม์ โดย OpenCV ถูกพัฒนาขึ้นโดย Intel และสามารถใช้งานได้ฟรีภายใต้ ลิขสิทธิ์ของ BSD แบบ Open Source โดยตัว OpenCV ยังรองรับ Deep Learning Framework เช่น TensorFlow, Torch, PyTorch เป็นต้น [38]

2.2.3 Scikit-image

เป็นไลบรารีการประมวลผลภาพที่นำอัลกอริทึมและเครื่องมือต่างๆ ที่จำเป็นสำหรับการวิจัย การศึกษา และ การนำไปใช้จริง สามารถใช้งานได้ฟรีภายใต้ ลิขสิทธิ์ของ BSD แบบ Open Source [39]

2.2.4 Streamlit

Open-Source Python Library ที่ใช้ในการสร้าง Custom Web Applications ที่มีประสิทธิภาพ ที่ไม่จำเป็นต้องใช้ Libraries API (เช่น Flask หรือ Django) เป็นตัวช่วยในการติดต่อ ทำให้สามารถเขียน Web Application ได้โดยตรงทำให้เหมาะสมสำหรับ Data Science และ Machine Learning [40]

2.2.5 PyTorch

เป็นหนึ่งใน Library ของ Deep Learning ที่มีประสิทธิภาพสูง เน้นที่การใช้งานที่มีประสิทธิภาพและมีความเร็ว โดย PyTorch ถูกออกแบบจากหลักการเขียนโปรแกรมที่ imperative และ Pythonic เป็นรูปแบบการเขียนโค้ดที่เป็นส่วนเสริมให้ PyTorch ง่าย ต่อการตีบัก และง่ายต่อการเชื่อมกับไลบรารีคอมพิวเตอร์อื่นๆ โดย PyTorch ยังคงมีประสิทธิภาพและสนับสนุนการเพิ่มความเร็วของ ฮาร์ดแวร์ เช่น GPU [41]

2.2.6 TensorFlow

TensorFlow เป็นระบบการเรียนรู้ของเครื่องที่ทำงานได้หลากหลายและในสภาพแวดล้อมที่แตกต่างกัน แบบจำลองการคำนวณขั้นอยู่กับกราฟ data-flow ที่มีสถานะที่ไม่แน่นอน TensorFlow รองรับแอปพลิเคชันที่หลากหลาย แต่จะมีประสิทธิภาพสูงเมื่อนำไปใช้กับ train เฉพาะจุด และการใช้ deep neural networks โดย TensorFlow ทำหน้าที่เป็นแพลตฟอร์มสำหรับการวิจัย และการปรับใช้ระบบการเรียนรู้ของเครื่องในหลาย ๆ ด้าน เช่น การรู้จำเสียงพูด คอมพิวเตอร์วิทยุ หุ่นยนต์ การดึงข้อมูล และการประมวลผลภาษาธรรมชาติ [42]

2.2.7 Docker

Docker เป็นโครงการคอนเทนเนอร์แบบเบ็ดเตล็ดภาษา Go ระบบปฏิบัติการ Linux หลักทั้งหมดรองรับ Docker แนวคิดหลักของเทคโนโลยี Docker คือ Containerization หรือ การบรรจุนักถัง ”สร้าง จัดส่ง และเรียกใช้แบบได้ก็ได้ ทุกที่” นั่นคือ การจัดการจริงชีวิตของแอปพลิเคชันเป็นระยะๆ เพื่อให้บรรลุวัตถุประสงค์ของการบรรจุครั้งเดียวและทำงานทุกที่ [43]

2.2.8 Google Cloud Platform (GCP)

GCP เป็นผู้ให้บริการ Cloud Services เช่นเดียวกับ Amazon Web Services (AWS) และ Microsoft Azure ผู้ใช้งานสามารถเข้าใช้งานบริการต่างๆ เช่น พื้นที่จัดเก็บข้อมูล, databases, VM, AI platform และอื่นๆ [44]

2.2.9 Cloud Run

Cloud Run เป็นบริการหนึ่งใน GCP โดยมีวัตถุประสงค์เพื่อเป็น Serverless ด้วยความสามารถในการ Build และ Deploy แอปพลิเคชันที่มีพื้นฐานการพัฒนาแบบ Containerization [45]

2.2.10 Firebase

Firebase หรือ Google Firebase เป็นแพลตฟอร์มการพัฒนาแอปที่ช่วยให้สร้างและขยายแอปไปยังผู้ใช้ ซึ่งสนับสนุนโดย GCP ซึ่ง Firebase ช่วยให้นักพัฒนาสามารถพัฒนาเว็บแอปพลิเคชัน และแอปพลิเคชันทั้งใน iOS และ Android อีกทั้ง Firebase มีเครื่องมือสำหรับติดตามการวิเคราะห์ การรายงาน แก้ไขข้อขัดข้องของแอป การสร้างการทดสอบและการตลาดและผลิตภัณฑ์ นอกจากนี้ยังได้รับความไว้วางใจจากธุรกิจขนาดใหญ่โลก [46]

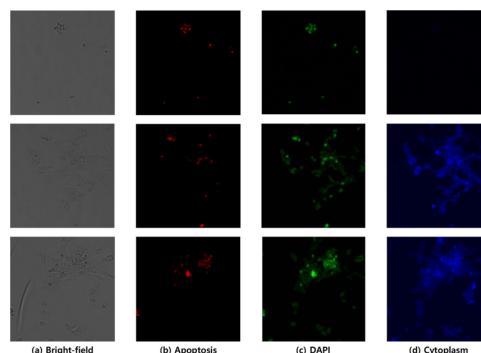
2.3 ทบทวนวรรณกรรม

โดยตารางสรุปงานวิจัยที่ใช้การเรียนรู้เชิงลึก (Deep Learning) สามารถดูได้ในตารางที่ 2.5 และในส่วนตารางสรุปงานวิจัยที่ใช้การประมวลผลภาพ (Image Processing) สามารถดูได้ในตารางที่ 2.6

2.3.1 DeepHCS++: Bright-field to fluorescence microscopy image conversion using multi-task learning with adversarial losses for label-free high-content screening

งานวิจัยของ Gyuhyun Lee และคณะ [4] ได้ศึกษาวิธีการแปลงภาพถ่ายเรืองแสงจากกล้องจุลทรรศน์ส่องสว่าง ที่แตกต่างกันสามภาพเพื่อสังเกตการตายของนิวเคลียส, นิวเคลียส และไซโตพลาสซึมของนิวเคลียส โดย end-to-end CNN สามารถสร้างภาพถ่ายเรืองแสงจากกล้องจุลทรรศน์ส่องสว่างให้มีความคล้ายกับภาพถ่ายกล้องจุลทรรศน์เรืองแสง โดย Gyuhyun Lee เพิ่มแบบจำลอง multi-task learning กับ adversarial losses เพื่อเป็นการสร้างภาพจำลองที่มีความแม่นยำและสมจริงยิ่งขึ้น และ Gyuhyun Lee ประเมินประสิทธิภาพของแบบจำลองโดยใช้ชุดภาพถ่ายจากกล้องจุลทรรศน์แบบส่องสว่างและภาพถ่ายจากกล้องจุลทรรศน์แบบเรืองแสง และตรวจสอบความถูกต้องของวิธีการด้วยเมตริกต่าง ๆ ดังนี้ cell number correlation (CNC), peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM), cell viability correlation (CVC), error maps, และ R^2 correlation.

งานวิจัยนี้ทำการวิจัยในภาพถ่ายเรืองแสงจากการแปลงภาพถ่ายจากการถ่ายจากกล้องจุลทรรศน์ส่องสว่าง ดังรูปภาพที่ 2.14 เพื่อให้ได้ภาพถ่ายเรืองแสงที่คล้ายกับภาพถ่ายที่ออกแบบจากกล้องจุลทรรศน์เรืองแสง จาก glioblastoma cells และ cell lines of lung adenocarcinoma ซึ่งแตกต่างกับงานของคณะผู้วิจัยที่จะตรวจจับจากภาพถ่ายเนื้อเยื่อผิวหนัง แต่งานวิจัยของ Gyuhyun Lee ภาพถ่ายจะถูกย้อมสีด้วย DAPI ซึ่งความคล้ายกับงานวิจัยของคณะผู้วิจัย

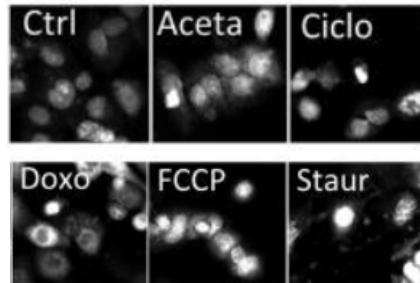


รูปที่ 2.14: ตารางแสดงรูปตัวอย่างภาพถ่ายสารเรืองแสง โดยแต่ละແລງจะแสดงฟื้นที่ที่สนใจ (a) ภาพที่ถ่ายด้วยกล้องจุลทรรศน์แบบไม่มีเครื่องหมายของโปรตีน (b) ภาพสารเรืองแสง apoptosis เพื่อตรวจจับเซลล์ที่ตายแล้ว (c) ภาพเรืองแสง DAPI สำหรับนิวเคลียสของเซลล์ และ (d) ภาพสารเรืองแสงของไซโตพลาสซึม สำหรับดูลักษณะรูปร่างของเซลล์

2.3.2 Tox_(R)CNN: Deep learning-based nuclei profiling tool for drug toxicity screening

งานวิจัยของ Daniel Jimenez-Carretero และคณะ [5] ได้ศึกษาความเป็นพิษของยาเป็นปัจจัยสำคัญในการพัฒนาฯ โดยนิวเคลียสจะถูกย้อมด้วยสารเรืองแสงเพื่อใช้ในการทวนยาระหว่างการจัดจำลักษณะรูปร่างนิวเคลียส ซึ่งการเรียนรู้เชิงลึกจะทำให้สามารถจำแนกรูปภาพแบบที่มีความซับซ้อนได้อย่างมีประสิทธิภาพมากยิ่งขึ้น และการใช้โครงข่ายประสาทแบบconvolutional neural networks (deep convolutional neural networks) ถูกนำมาใช้ทวนยาระหว่างการเป็นพิษจากภายนอกนิวเคลียสที่ย้อมด้วย DAPI และแบบจำลอง nuclei-cropping-based Tox_CNN สามารถนำมาใช้ในการจำแนกนิวเคลียส รวมถึงการดึงข้อมูลคุณลักษณะโดยอัตโนมัติและจัดกลุ่มสารประกอบตามกลไกการทำงานของยา นอกจากนี้ มีการใช้ fully automated region-based CNNs (RCNN) เพื่อตรวจจับและทำการจำแนกนิวเคลียส โดยให้ผลการทวนยาระหว่างการเป็นพิษของยาจากภาพได้ผลลัพธ์ที่มีความไว้สูงและมีความจำเพาะสูง แบบจำลองนี้สามารถนำมาใช้สำหรับการจัดลำดับความสำคัญของการแสดงออกของยา และอาจเพิ่มประสิทธิภาพในการค้นคว้าฯได้

งานวิจัยนี้ทำการวิจัยในภาพถ่ายนิวเคลียสที่ถูกย้อมด้วยสารเรืองแสง DAPI ดังรูปภาพที่ 2.15 รวมถึงการเลือกใช้แบบจำลองที่จะนำมาใช้เรียนรู้ให้สามารถจำแนกรูปภาพที่มีความซับซ้อนได้ ซึ่งงานวิจัยนี้มีความคล้ายคลึงกับวัตถุประสงค์ที่คณะผู้วิจัยสนใจ即จะทำการศึกษาคือมีภาพถ่ายนิวเคลียสที่ถูกย้อมด้วยสารเรืองแสง DAPI เช่นเดียวกัน

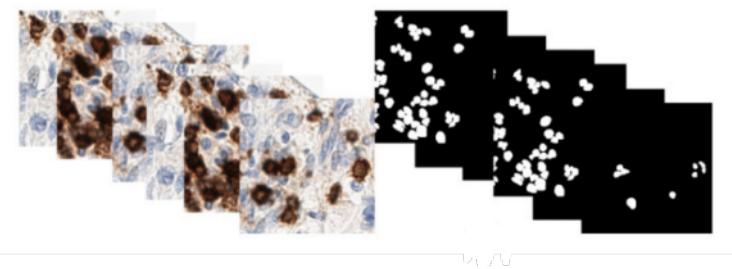


รูปที่ 2.15: ตัวอย่างภาพที่ได้จากการถ่ายด้วยกล้องจุลทรรศน์ฟลูออเรสเซนซ์ของเซลล์ที่ย้อมสีด้วย DAPI

2.3.3 A Means of Assessing Deep Learning-Based Detection of ICOS Protein Expression in Colon Cancer

งานวิจัยของ Md Mostafa Sark และคณะ [47] ได้ศึกษาตัวบ่งชี้ทางชีวภาพ Inducible T-cell COStimulator (ICOS) ที่สามารถระบุการตอบสนองของผู้ป่วยต่อการรักษา การใช้การเรียนรู้เชิงลึกสามารถวัดปริมาณของการตรวจตัวบ่งชี้ทางชีวภาพและการตรวจจับนิวเคลียสในสไลด์ IHC ดังรูปภาพที่ 2.16 เพื่อหาปริมาณตัวบ่งชี้ทางชีวภาพของการย้อมสีนิวเคลียส แบบจำลองการแบ่งนิวเคลียสและการตรวจจับ พิสูจน์ได้ว่ามีความแม่นยำมากขึ้นและใช้เวลาอย่างกว่าวิธีการแบบเดิม วิธีการนี้สามารถระบุความหมายเชิงพยากรณ์ของการแสดงออกของโปรตีน ICOS

งานวิจัยนี้ทำการวิจัยในภาพ Colorectal cancer เพื่อดูการแบ่งตัวของนิวเคลียสจากการตรวจหาโปรตีน immune-checkpoint ICOS ซึ่งแตกต่างกับงานของคณะผู้วิจัยที่จะตรวจจับนิวเคลียสจากภาพถ่ายเนื้อเยื่อผิวหนังที่ถูกย้อมสีด้วย DAPI ในส่วนที่เป็นนิวเคลียส

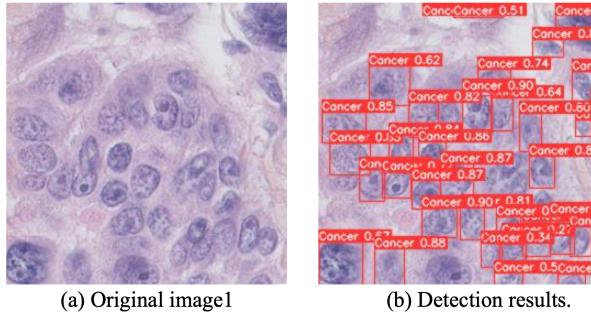


รูปที่ 2.16: ตัวอย่างภาพนิวเคลียสในสไลด์ IHC

2.3.4 Breast Cancer Detection from Histopathology Images Based on YOLOv5

งานวิจัยของ Wafaa Rajaa Drioua และคณะ [6] ได้ศึกษาการตรวจหานิวเคลียสของเซลล์ การวิเคราะห์ทางจุลพยาธิวิทยาที่แสดงถึงเซลล์ของตัวอย่างเนื้อเยื่อที่ให้ข้อมูล ซึ่งจำเป็นสำหรับการตรวจหาและระบุลักษณะของมะเร็ง และการเรียนรู้เชิงลึก (Deep learning) แสดงให้เห็นถึงประสิทธิภาพที่มีแนวโน้มในการวินิจฉัยมะเร็งเต้านม ดังนั้นในงานวิจัยนี้ จึงนำเสนออัลกอริทึมการตรวจหาเซลล์ตามแบบจำลอง YOLOv5 ที่เป็นเครือข่าย backbone และถูกนำไปใช้กับชุดข้อมูลสาธารณะของภาพเนื้อเยื่อย้อมสี hematoxylin และ eosin (H&E) ซึ่งสามารถช่วยแยกนิวเคลียร์ภายในเซลล์ เพื่อประเมินผลการแบ่งจากอัลกอริทึม

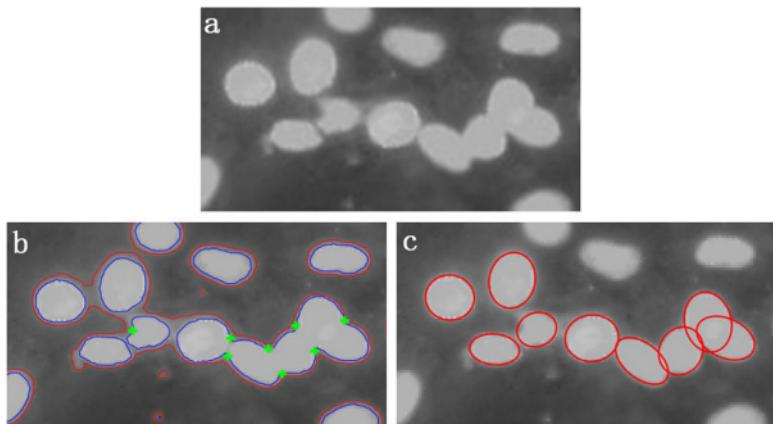
งานวิจัยนี้มีความคล้ายคลึงกับวัตถุประสงค์ที่คณะผู้วิจัยสนใจ即จะทำการศึกษาคือในส่วนของการตรวจจับนิวเคลียสภายในภาพถ่ายเนื้อเยื่อ แต่งานวิจัยนี้ทำการวิจัยในรูป breast cancer histopathology ที่ถูกย้อมสีด้วย hematoxylin และ eosin (H&E) ดังรูปภาพที่ 2.17 โดยงานของคณะผู้วิจัยจะตรวจจับจากภาพถ่ายเนื้อเยื่อผิวหนังที่ถูกย้อมสีด้วย DAPI



รูปที่ 2.17: ตัวอย่างภาพถ่ายนิวเคลียสในงานวิจัยของ Wafaa Rajaa Drioua และคณะ

2.3.5 Automated segmentation of overlapped nuclei using concave point detection and segment grouping

งานวิจัยของ Wanjun Zhang และคณะ [48] ได้ทำการประเมินหรือแบ่งส่วนของนิวเคลียสในงานทางชีววิทยา โดยวัดคุณลักษณะ คือการแบ่งนิวเคลียสที่มีการทับซ้อนกัน ในงานวิจัยนี้นำเสนอวิธีการทั้ง 4 ขั้นตอน ประกอบด้วย การสกัดหาขอบของวัตถุ (Contour Extraction), การหาส่วนร่องโค้ง (Concave Point Detection), การจัดกลุ่มของขอบวัตถุ (Contour Segment Grouping) และ การวง喋แน่นิวเคลียส (Ellipse Fitting) ดังรูปภาพที่ 2.18 ซึ่งการจะแบ่งภาพถ่ายนิวเคลียสที่มีความเบลอออกจากกันและภาพถ่ายนิวเคลียสที่หากขอบได้ยากเป็นปัญหาหลักในการแบ่งนิวเคลียสอยู่เสมอ โดยการลดระดับความเบลอด้วยพารามิเตอร์สามารถเพิ่มความแม่นยำในการแบ่งนิวเคลียสที่มีความเบลอออกจากกันได้ และมีการเสนอวิธีการแบ่งด้วยจุดเดียวโค้งในกลุ่มนิวเคลียส ในการสกัดหาส่วนเร้าโค้งของนิวเคลียสที่เห็นได้ชัดเจนและไม่ชัดเจน



รูปที่ 2.18: ตัวอย่างวิธีการแยกนิวเคลียสที่มีการซ้อนทับกัน โดย (a) รูปภาพต้นฉบับ (b) การสกัดรูปร่างและจุดเร้าของนิวเคลียส (c) ผลลัพธ์ที่ได้

2.3.6 Automated image processing workflow for morphological analysis of fluorescence microscopy cell images

งานวิจัย Sven P. Voigt และคณะ [49] การวิเคราะห์ภาพด้วยคอมพิวเตอร์ของนิวเคลียสและเนื้อเยื่อทางชีวภาพเป็นส่วนเสริม ที่จำเป็นสำหรับกล้องจุลทรรศน์ที่มีความเร็วสูง ที่จะช่วยให้นักวิจัยสามารถวิเคราะห์ข้อมูลภาพถ่ายนิวเคลียสที่มีปริมาณมากได้อย่างมีประสิทธิภาพ การศึกษานี้นำเสนอวิธีการวิเคราะห์แบบอัตโนมัติของภาพถ่ายจากกล้องจุลทรรศน์ฟลูออเรสเซนซ์ ซึ่งได้ประยุกต์จากการรันวิร์กโฟลว์ที่มีการแบ่งส่วนออกเป็นหลาย ๆ ส่วนและนำมาร่วมเข้าด้วยกันเพื่อสร้างการแบ่งส่วนสุดท้าย ที่ดีที่สุด ซึ่งมีการทดสอบโดยใช้ชุดข้อมูลของภาพนิวเคลียสที่ได้จากการจุลทรรศน์ฟลูออเรสเซนซ์ จำนวน 42 นิวเคลียส ประเมินเทียบกับชุดข้อมูลที่แบ่งส่วนด้วยมือ โดยทำการประเมินผลด้วยวิธี F1 score และเปรียบเทียบกับวิธีการแบ่งส่วนเดียวซึ่งทำ

หน้าที่เป็นตัวควบคุม ค่าความแม่นยำและความน่าเชื่อถือของเวิร์กโฟลว์ใหม่นั้นแสดงให้เห็นแล้วว่ามีความสามารถกว่าเวิร์กโฟลว์แบบควบคุม ซึ่งได้ค่า F1 score ที่ 0.845 และ 0.608 ตามลำดับ

2.3.7 Automated cell counting and cluster segmentation using concavity detection and ellipse fitting techniques

งานวิจัยของ onal Kothari และคณะ [50] การศึกษานี้เป็นการนำเสนอวิธีการใหม่ ที่มีความรวดเร็ว และก็อัตโนมัติสำหรับการแบ่งกลุ่มนิวเคลียสและการนับจำนวนนิวเคลียสของตัวอย่างภาพเนื้อเยื่อที่มีความแม่นยำ ซึ่งในทางพยาธิวิทยา ส่วนใหญ่จะพบกลุ่มนิวเคลียสที่มีความซับซ้อน ซึ่งพบได้ในตัวอย่างเนื้อเยื่อ การแบ่งกลุ่มของคลัสเตอร์เหล่านี้ที่เป็นเรื่องยากสำหรับการพัฒนาวิธีการนับจำนวนนิวเคลียสให้มีความแม่นยำ ดังนั้นเราจึงแก้ไขปัญหาการแบ่งกลุ่มคลัสเตอร์โดยทำการแบ่งตามขั้นตอนนี้ คือ ขั้นตอนแรกจะเกี่ยวข้องกับการประมวลผลล่วงหน้าที่จำเป็น เพื่อให้ได้ภาพของเขตของคลัสเตอร์นิวเคลียสที่เหมาะสมจากตัวอย่างเนื้อเยื่อ RGB ขั้นตอนที่สองเกี่ยวข้องกับการแบ่งส่วนที่เว้าเหล่านี้โดยใช้เทคนิคการวางรีแบบพอด เมื่อแบ่งกลุ่มแล้ว แต่ละนิวเคลียสจะถูกนิยามเป็นกลุ่มที่สามารถเก็บรวบรวมได้ ขั้นตอนที่สามเกี่ยวข้องกับการแบ่งส่วนที่เว้าเหล่านี้โดยใช้เทคนิคการวางรีแบบพอด เมื่อแบ่งกลุ่มแล้ว แต่ละนิวเคลียสจะถูกนิยามเป็นกลุ่มที่สามารถเก็บรวบรวมได้ ขั้นตอนที่สี่จะแสดงผลลัพธ์ที่มีเปอร์เซ็นต์ความผิดพลาดต่ำ true positive rate ที่สูง และ false discovery rate ต่ำ

2.4 ผลิตภัณฑ์ที่เกี่ยวข้อง

2.4.1 An ImageJ plugin for the high throughput image analysis of in vitro scratch wound healing assays

งานวิจัยของ C. Igathinath และคณะ [51] การศึกษานี้เป็นการพัฒนา plugin ที่ใช้สำหรับโปรแกรม ImageJ เพื่อจัดทำขนาดการรักษาบาดแผลแบบ อัตโนมัติ แก้ไขความกว้างของบาดแผลโดยเฉลี่ยโดยพิจารณาจากความเอียงและหาค่าพารามิเตอร์ที่สำคัญอื่นๆ เช่น พื้นที่ เศษส่วนของพื้นที่แหล่ง ความกว้างของแผลเฉลี่ย และส่วนเบี่ยงเบนความกว้างของบาดแผลที่ได้จากการทดสอบรอยขีดข่วน/การรักษาบาดแผล ในการพัฒนาอัลกอริทึมนี้มีการแบ่งส่วนการมองเห็นด้วย คอมพิวเตอร์แบบคลาสสิกที่เน้นการประเมินความแปรปรวนของความเข้มพิกเซลที่อยู่ใกล้เคียงด้วยการ ตรวจสอบอย่างรอบคอบของภาพระดับสีเทาที่ได้จากการทดสอบ การสมานของบาดแผล

2.4.2 Shape identification and particles size distribution from basic shape parameters using ImageJ

งานวิจัยของ H.S. Kim และคณะ [52] วัตถุประสงค์ของงานวิจัยนี้คือการวิเคราะห์การกระจายขนาดอนุภาคที่รวดเร็วและแม่นยำ ซึ่งเป็นสิ่งที่ต้องการวิเคราะห์ในด้านเทคนิคต่างๆ ที่จัดการกับวัตถุที่มีขนาดเล็กหรืออนุภาค รวมทั้งการลดขนาด นักวิจัยได้พัฒนาปลั๊กอิน ImageJ ที่แยกมิติออกจากภาพเพิ่มจิลลอกของวัตถุที่ไม่เป็นตัวเดียว หลังจากระบุรูปร่างและกำหนดการกระจายขนาดวัตถุแล้ว นักวิจัยจึงทำการกำหนดแกนหลักและแกนรองของ ImageJ พร้อมกับปัจจัยการแก้ไขที่พัฒนาขึ้นซึ่งกำหนดขนาดของวัตถุได้อย่างมีประสิทธิภาพ งานวิจัยนี้ได้อธิบายการพัฒนาปลั๊กอินและการประยุกต์ใช้กับอัญมณีอาหารและเชื้อมวนบด โดยใช้คอมพิวเตอร์สร้างรูปทรงเรขาคณิตเป็นวัตถุอ้างอิง กลยุทธ์การระบุรูปร่างที่กล่าวถึงรูปทรงเรขาคณิตทั่วไป เช่น สี่เหลี่ยมจัตุรัส สี่เหลี่ยมจัตุรัส เอียง สี่เหลี่ยมผืนผ้า สี่เหลี่ยมผืนผ้าเอียง วงกลม วงรี และวงรีเอียงได้รับการพัฒนา กลยุทธ์นี้ใช้พารามิเตอร์รูปร่างที่กำหนดใหม่ เพียงสามตัวในการระบุวัตถุ เช่น อัตราส่วนส่วนกลับ สี่เหลี่ยม และอัตราส่วนแกนหลัก Feret จากเอกสารพุฒนาตรฐานที่สร้างโดย ImageJ การประเมินผลกระทบของรูปร่างขนาด และทิศทางของวัตถุนบนความเบี่ยงเบนจากความยาวและความกว้างของวัตถุอ้างอิง บ่งชี้ว่าค่าเบี่ยงเบนสัมบูรณ์เฉลี่ยของปัจจัยทั้งหมดเหล่านี้อยู่ที่ 1.3% ปลั๊กอินที่พัฒนาแล้วถูกนำไปใช้อย่างประสบความสำเร็จในการวิเคราะห์ขนาดและการกระจายขนาดของเม็ดอาหารและภาพอนุภาคมิสแคนท์สบด ปลั๊กอินนี้สร้างการกระจายขนาดอนุภาคที่รวดเร็วและแม่นยำจากภาพดิจิทัล และสามารถนำไปใช้กับแอปพลิเคชันการวิเคราะห์อนุภาคต่างๆ

2.4.3 Columbus Image Data Storage and Analysis System

งานวิจัยของ PerkinElmer และคณะ [3] ได้อธิบายไว้ว่า Columbus เป็นแพลตฟอร์มที่ใช้ในการจัดเก็บและวิเคราะห์ภาพสำหรับภาพที่มีเนื้อหาหรือรายละเอียดสูง ซึ่งในปัจจุบันมีการใช้แบบจำลองของโรคที่มีความซับซ้อนและสัมพันธ์กันทางสรีรวิทยา มากขึ้น เครื่องมือที่มีความซับซ้อนมากขึ้นจะมีความจำเป็นในการอธิบายนิวเคลียสและฟีโนไทป์ของพากมันอย่างครอบคลุม แพลต-

ตารางที่ 2.5: ตารางสรุปงานวิจัยที่ใช้ Deep Learning

Papers	Method	Dataset	Accuracy
DeepHCS++: Bright-field to fluorescence microscopy image conversion using multi-task learning with adversarial losses for label-free high- content screening [4]	CNN	bright-field and fluorescence microscopy image datasets from patient-driven samples of a glioblastoma	0.91
Tox (R)CNN: Deep learning-based nuclei profiling tool for drug toxicity screening [5]	CNN, R-CNN	datasets of healthy and toxicity affected labeled nuclei	0.90
A Means of Assessing Deep Learning-Based Detection of ICCS Protein Expression in Colon Cancer[47]	U-Net, Mask R-CNN	immunohistochemistry-stained slides as a basis for quantifying nuclear staining biomarkers	0.98
Breast Cancer Detection from Histopathology Images Based on YOLOv5[47]	YOLOv5	the breast cancer histopathology image dataset (BNS) and the annotated dataset.	0.86

ตารางที่ 2.6: ตารางสรุปงานวิจัยที่ใช้ Image Processing

Papers	Method	Dataset	Accuracy
Automated segmentation of overlapped nuclei using concave point detection and segment grouping [48]	1. Three-phase level set formulation 2. Candidate point detection 3. Concave point detection 4. Segment grouping and Ellipse fitting	Nucleus images in two kinds of polyacrylamide substrates, there are lots of fuzzy nuclei.	0.98
Automated Image Processing Workflow for Morphological Analysis of Fluorescence Microscopy Cell Images [49]	1. smoothing intensity values and contrast enhancement 2. local average threshold and remove small objects 3. generate markers from peaks for separate touching nuclei 4.reassigning labels and generate relabeled markers	Fluorescence microscopy images of biological cells, with image variations arising from differences in cell phenotype	0.85
Automated cell counting and cluster segmentation using concavity detection and ellipse fitting techniques [50]	1. Preprocessing 2. Concavity or Notch Detection 3. Segmentation	H&E stained tissue images from three subtypes and IHC stained head and neck cancer tissue image	0.91

ฟอร์ม Columbus เป็นระบบจัดเก็บและวิเคราะห์ข้อมูลภาพเพียงระบบเดียวที่รองรับรูปแบบไฟล์ที่หลากหลาย โดยแพลตฟอร์ม Columbus สามารถเข้าถึงการจัดเก็บ และการสำรวจข้อมูล โดยสามารถนำเข้ารูปภาพจากเครื่องมือสร้างภาพที่มีรายละเอียดสูง สำหรับโซลูชันเดียว สำหรับการจัดเก็บข้อมูลและการวิเคราะห์รูปภาพ และการวิเคราะห์ภาพสำหรับการคัดกรองพีโนไทด์ สร้างข้อมูลที่มีนัยสำคัญทางสถิติ เชิงปริมาณ และแบบหลายพารามิเตอร์จากการนิวเคลียสเพื่อให้ได้ผลลัพธ์ที่มีประสิทธิภาพ [3]

2.4.4 Validation and Automation of Phenotypic Profiling Across Multiple Cell Lines

งานวิจัยของ D.Y. Eun และคณะ [53] จุดประสงค์ของการสัมมนานี้คือการนำเสนอการทดสอบ Autophagy ในกลุ่มนิวเคลียสมะเร็งทั้งหมด 3 กลุ่ม เพื่อทำการตรวจสอบและทำให้กระบวนการวิเคราะห์พีโนไทด์ HCS เป็นไปได้อย่างอัตโนมัติ โดยแพลตฟอร์ม Columbus และ Spotfire High Content Profiler โดยแพลตฟอร์ม Columbus จะเป็นแพลตฟอร์มที่ใช้ในการจัดเก็บข้อมูล จัดการข้อมูล และวิเคราะห์ภาพที่ได้จากเครื่องมือ HCS ทั้งหมด รวมถึงลำดับการวิเคราะห์ที่มีความซับซ้อนก็สามารถวิเคราะห์ได้โดยใช้แนวทางการ Building Block ทำให้ผลลัพธ์จากการวิเคราะห์รูปภาพนั้น สามารถเลือกได้โดยตรงภายใต้ Spofire High Content Profiler (HCP) สำหรับการวิเคราะห์และการแสดงภาพข้อมูล HCS แบบหลายพารามิเตอร์

ตารางที่ 2.7 ตารางสรุปผลิตภัณฑ์ที่เกี่ยวข้อง

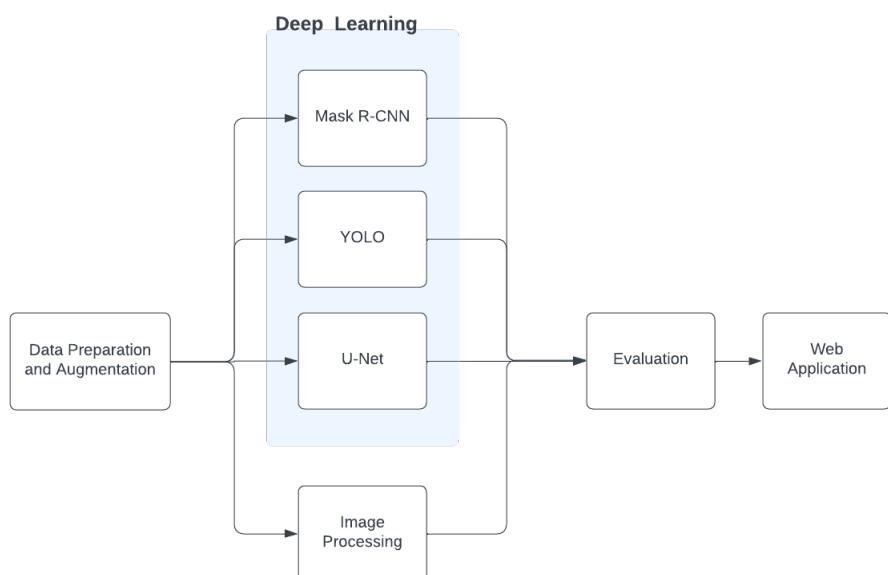
Papers	Product	Limitation
An ImageJ plugin for the high throughput image analysis of in vitro scratch wound healing assays [51]	ImageJ	ใช้เวลาในการประมวลผลนานและจำกัด ปริมาณรูปที่สามารถประมวลผลได้
Shape identification and particles size distribution from basic shape parameters using ImageJ [52]	ImageJ	ในการวัดค่าจะต้องทำการกำหนดขอบเขตที่จะให้โปรแกรมตรวจหาด้วยตัวเอง
Columbus Image Data Storage and Analysis System [3]	Columbus	ไม่สามารถหาระยะที่นิวเคลียสทรงกลม 2 อันซ้อนทับกันไม่สิบ เนื่องจากหัวรูปทรงที่มองเห็นไม่เป็นทรงกลมจริง ๆ
Validation and Automation of Phenotypic Profiling Across Multiple Cell Lines [53]	Columbus	ภาพ Immunofluorescence Tissue ไม่สามารถใช้งานโปรแกรม Columbus ได้

บทที่ 3 วิธีการดำเนินงาน

ในบทนี้จะกล่าวถึงภาพรวมการดำเนินงานวิจัยเรื่อง เว็บแอปพลิเคชันสำหรับการตรวจสอบนิวเคลียสจากภาพถ่ายเนื้อเยื่อจาก การย้อมด้วยสารเรืองแสง รวมถึงวิธีการดำเนินงาน กระบวนการที่ใช้ในการดำเนินงาน และการออกแบบงานวิจัยโดยมีเนื้อหา ดังนี้

3.1 รายละเอียดภาพรวมการดำเนินงานวิจัย

งานวิจัยในครั้งนี้มุ่งเน้นพัฒนาระบบที่สามารถนับจำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง เพื่อ ช่วยลดภาระของนักพยาธิวิทยาที่ต้องทำการนับจำนวนนิวเคลียสภายในภาพถ่ายด้วยตนเอง ซึ่งการนับจำนวนนิวเคลียสของนั้นต้อง ใช้ความพยายามในการคัดแยกและระยะเวลาต่อหนึ่งภาพที่สูงเป็นอย่างมาก ดังนั้นคณะผู้วิจัยจึงตัดสินใจที่จะออกแบบแนวทางการ นับจำนวนของนิวเคลียสภายในภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง โดยการดำเนินงานวิจัยจะเริ่มจากการเตรียมภาพและ label ตำแหน่งของนิวเคลียส หลังจากนั้นนำเข้าแบบจำลองที่ดีที่สุดที่นำมาใช้ในการพัฒนาเว็บ แอปพลิเคชัน ตามแผนภาพรายละเอียดภาพรวมการดำเนินงานวิจัยดังรูปภาพที่ 3.1



รูปที่ 3.1: แผนภาพรายละเอียดภาพรวมการดำเนินงานวิจัย

3.2 ความต้องการของนักพยาธิวิทยา

หลังจากที่คณะผู้วิจัยได้วิเคราะห์ความต้องการของนักพยาธิวิทยาที่ต้องการวิเคราะห์ภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรือง แสง มีรายละเอียดดังนี้

1. เว็บแอปพลิเคชันที่สามารถอัปโหลดรูปที่ต้องการวิเคราะห์
2. จำนวนนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง
3. ภาพผลลัพธ์ที่มีการ label ตำแหน่งของนิวเคลียส
4. ตารางสรุปตำแหน่งของนิวเคลียส
5. สามารถบันทึกรูปภาพผลลัพธ์และตารางสรุปตำแหน่งของนิวเคลียส

3.3 การออกแบบระบบซอฟต์แวร์

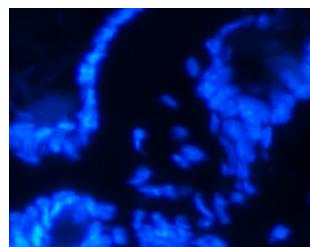
3.3.1 การพัฒนาแบบจำลองในการนับจำนวนนิวเคลียส

เนื่องจากงานวิจัยที่ได้ศึกษามาวิธีการที่ใช้ 2 แบบ ได้แก่ การเรียนรู้เชิงลึก (Deep Learning) และ การประมวลผลภาพ (Image Processing) คณะผู้วิจัยจึงทำการทดลองเปรียบเทียบผลลัพธ์ เพื่อเลือกวิธีที่ดีที่สุดมาใช้เป็นอัลกอริทึมของเว็บแอปพลิเคชัน

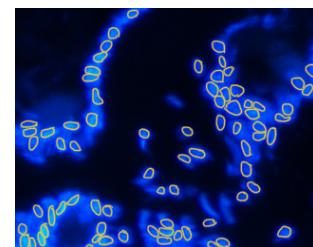
3.3.1.1 การเรียนรู้เชิงลึก (Deep Learning)

จากการศึกษางานวิจัยที่เกี่ยวข้องในตารางที่ 2.5 มีตัวอย่างของงานวิจัยที่ใช้แบบจำลอง Mask R-CNN, YOLO และ U-Net สำหรับนับจำนวนนิวเคลียส คณะผู้วิจัยจึงมีความสนใจที่จะทำการศึกษาและพัฒนาด้วยแบบจำลองเหล่านี้

- ชุดข้อมูลที่ใช้ในการพัฒนาแบบจำลอง จากชุดข้อมูลภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงทั้งหมด 197 รูป นักพยาธิวิทยาได้ทำการเลือก 5 รูปที่มีขนาด 1920×1536 พิกเซล จากนั้นทำการแบ่งเป็นรูปเล็ก ๆ ได้รูปขนาด 319×255 พิกเซล เพื่อให้ง่ายต่อการทำ label และลดขนาดของ Neural Network โดยจะแบ่งเป็นรูปคละ 36 รูปเล็ก เพราะฉะนั้นจะได้ทั้งหมด 180 รูป เจ้านั้นให้นักพยาธิวิทยา label นิวเคลียสภายในรูป ดังรูปที่ 3.2 ต่อมาผู้วิจัยได้นำรูปที่มีพื้นหลังสีดำซึ่งไม่มีนิวเคลียสภายในรูปออกจากชุดข้อมูลทั้งหมด ซึ่งเหลือผลลัพธ์ทั้งหมดจำนวน 122 รูป ค่าเฉลี่ยจำนวนนิวเคลียสในแต่ละรูปคือ 26 เซลล์ ผู้วิจัยได้ทำการแบ่งชุดข้อมูลจากรอบวนการก่อนหน้าออกเป็น 3 ชุด คือ ชุดข้อมูลที่ใช้ในการเรียนรู้ของแบบจำลอง (Train Dataset) จำนวน 87 รูป, ชุดข้อมูลที่ใช้ในการปรับปรุงประสิทธิภาพของแบบจำลอง (Validation Dataset) จำนวน 23 รูป และ ชุดข้อมูลที่ใช้ในการวัดผลความสามารถของแบบจำลอง (Test Dataset) จำนวน 12 รูป ตามอัตราส่วน 70:20:10 ตามลำดับ เนื่องจาก Train dataset มีขนาดที่น้อยเกินไป จึงจำเป็นที่ต้องทำ Image Augmentation ให้ผลลัพธ์รูปที่นำไปใช้ทั้งหมด 820 รูปซึ่งวิธีการประกอบด้วย ปรับขนาดรูป (Resize) เป็น 512×512 พิกเซล, พลิกแนวอน (Horizontal flip) ความน่าจะเป็นที่ 0.5, พลิกแนวตั้ง (Vertical flip) ความน่าจะเป็นที่ 0.5, การหมุน (Rotate) หมุน 45 องศา ที่ความน่าจะเป็นที่ 0.5 และ Contrast Limited Adaptive Histogram Equalization ความน่าจะเป็นที่ 0.5 เมื่อได้รูปผลลัพธ์จากการทำ Image Augmentation จึงนำ label ที่ได้จากนักพยาธิวิทยาแปลงเป็นรูปแบบไฟล์ (Annotation) ตามแต่ละแบบจำลอง ต้องการ ดังรูปที่ 3.3

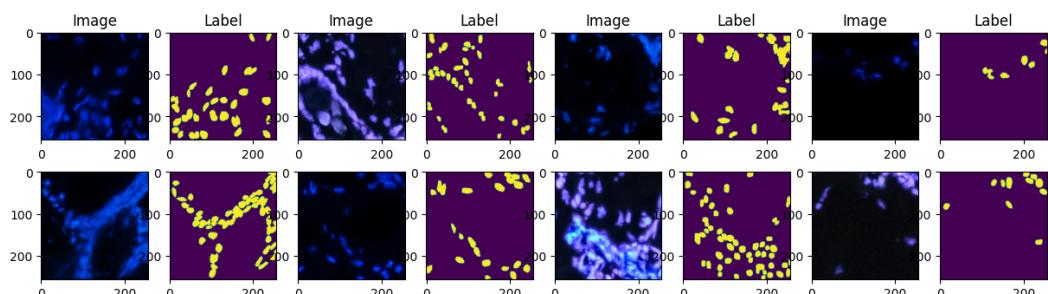


(a) ภาพต้นฉบับ



(b) ภาพ label ตำแหน่งนิวเคลียสโดยนักพยาธิวิทยา

รูปที่ 3.2: ภาพที่ใช้ในการพัฒนาแบบจำลอง



รูปที่ 3.3: ภาพผลลัพธ์จากการ Augmentation

- Mask R-CNN คณะผู้วิจัยได้ทำการทดลองกับแบบจำลอง Mask R-CNN มาใช้ฝึกกับชุดข้อมูลที่ใช้ในการพัฒนาแบบจำลองที่ได้กล่าวไว้ก่อนหน้านี้ โดยทำการปรับค่าพารามิเตอร์ ดังนี้ Backbone เท่ากับ Restnet101, Epochs เท่ากับ 100 ซึ่งจะเลือกเอาเฉพาะ epochs ที่ได้ค่าดีที่สุด และ Input size เท่ากับ 512×512 พิกเซล นอกจากนี้คณะผู้วิจัยทดลองปรับค่าพารามิเตอร์เพิ่มเพื่อให้ได้ค่า precision และ recall ที่สูงที่สุด โดยค่าของพารามิเตอร์ที่ทำให้ precision และ recall สูงที่สุดจะแสดงในตาราง 3.1 การทดลองนี้จะใช้ทั้งหมด 2 พารามิเตอร์คือ Non-maximum Suppression (NMS) และ ค่า Momentum โดยพารามิเตอร์ Non-maximum Suppression (NMS) ถูกปรับเพื่อให้การทำงานของ NMS เคลียร์ยิ่งขึ้น [54] โดยปรับจาก 0.3 เป็น 0.5 และปรับค่าพารามิเตอร์ momentum จาก 0.9 เป็น 0.3 ซึ่งการลดค่า momentum นี้จะเป็นการลดความเร็วในการเรียนรู้ของแบบจำลองลง [55]

ตารางที่ 3.1 ตารางการปรับพารามิเตอร์ของแบบจำลอง Mask R-CNN

Model	Parameter	Value
Mask R-CNN	Non-maximum Suppression (NMS)	0.5
	momentum	0.3

- YOLO คณะผู้วิจัยใช้แบบจำลองจาก Glenn Joche [30] มาใช้ฝึกกับชุดข้อมูลที่ใช้ในการพัฒนาแบบจำลองที่ได้กล่าวไว้ก่อนหน้านี้ โดยทำการปรับค่าพารามิเตอร์ ดังนี้ Batch size เท่ากับ 32, Epochs เท่ากับ 100 ซึ่งจะเลือกเอาเฉพาะ epochs ที่ได้ค่าดีที่สุด และ Input size เท่ากับ 640×640 พิกเซล นอกจากนี้คณะผู้วิจัยทดลองปรับค่าพารามิเตอร์เพิ่มเพื่อให้ได้ค่า precision และ recall ที่สูงที่สุดโดยค่าของพารามิเตอร์ที่ทำให้ precision และ recall สูงที่สุดจะแสดงในตาราง 3.2 การทดลองนี้จะใช้ทั้งหมด 6 พารามิเตอร์ พารามิเตอร์ที่ 1 Dropout ซึ่งคือการสุ่มลด node ของแบบจำลองโดยจะใส่ค่าความน่าจะเป็นเพื่อลดความ Overfit ของแบบจำลอง [56] พารามิเตอร์ที่ 2 Leaky เพื่อเปลี่ยนค่าสัมประสิทธิ์ความชันในช่วงติดลบ ตั้งแต่ 0 ลงไปซึ่งการลดค่าสัมประสิทธิ์ความชันจะทำให้แบบจำลองลดความซับซ้อนลง โดยแบบจำลองเดิมจะมีค่า Default 0.3 [57] พารามิเตอร์ที่ 3 Hardswish จะเป็นการเปิดใช้งานที่ใช้การคำนวณด้วยอะนาล็อกเชิงเส้นแบบแยกส่วน ซึ่งจะช่วยให้แบบจำลองคำนวนช้าลง [58] พารามิเตอร์ที่ 4 การปรับค่า momentum โดยจากแบบจำลองเดิมจะมีค่าอยู่ที่ 0.937 ซึ่งการลดค่าของ momentum คือการลดความเร็วในการเรียนรู้ของแบบจำลองลง [55] พารามิเตอร์ที่ 5 การปรับค่า scale image จากแบบจำลองเดิมจะตั้งค่าไว้ที่ 0.5 โดยการเพิ่มค่าให้เพิ่มขึ้นจะเพิ่มขนาดของภาพเพิ่มขึ้นก่อนเข้าแบบจำลอง และพารามิเตอร์ที่ 6 object loss การลดค่า object loss คือการลดค่า loss ในแต่ละพิกเซล โดยจากแบบจำลองเดิมตั้งค่าไว้ที่ 1.0 [59]

ตารางที่ 3.2 ตารางการปรับพารามิเตอร์ของแบบจำลอง YOLO

Model	Parameter	Value
YOLO	Dropout	0.1
	Leaky	0.1
	momentum	0.9
	scale image	0.8
	object loss	0.5

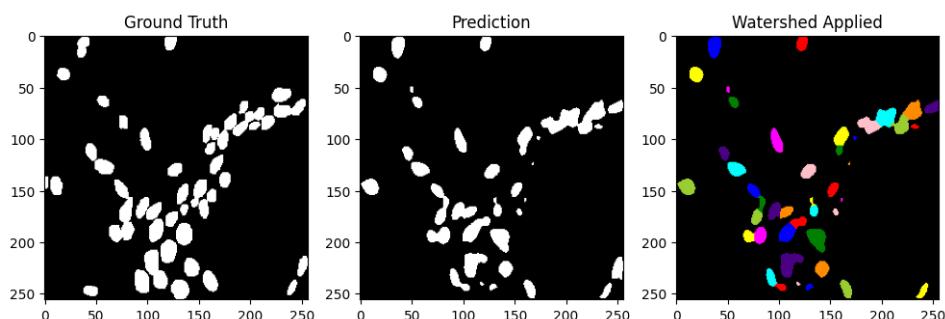
- U-Net

คณะผู้วิจัยได้แบ่งการทดลองออกเป็น 3 แบบจำลอง ประกอบด้วย U-Net คือ แบบจำลองตามโครงสร้างเดิมจากการวิจัยของ Olaf Ronneberger และคณะ [60], InceptionResNetV2 U-Net (IRNV2 U-Net) คือ แบบจำลองที่ใช้ส่วน Encoder ด้วย Pre-train weights ของ InceptionResNetV2 และคงส่วนของ Bottleneck และ Decoder ไว้เช่นเดิม และ Residual Attention U-Net (RA U-Net) ทั้ง 3 แบบจำลองคณะผู้วิจัยได้ทำการปรับค่าพารามิเตอร์ดังนี้ Batch size เท่ากับ 8, Epochs เท่ากับ 100 ซึ่งจะเลือกเอาเฉพาะ epochs ที่ได้ค่าที่สุด และ Input size 512x512 พิกเซล โดยตั้งค่า Earlystopper และใช้ Epochs ที่ให้ค่า Validation loss น้อยที่สุด ซึ่งคณะผู้วิจัยได้เลือกใช้ Adam และ Stochastic Gradient Descent (SGD) สำหรับ Optimizer ควบคู่กับ Binary Cross-Entropy (BCE) และใช้ Jaccard Coefficient ร่วมกับ Dice Coefficient สำหรับ Loss function สรุปได้ดังตารางที่ 3.3

ตารางที่ 3.3 ตารางการปรับพารามิเตอร์ของแบบจำลอง U-Net

Model	Optimizer	Loss Function
U-Net	Adam	BCE
		Dice+Jaccard
	SGD	BCE
		Dice+Jaccard
IRNV2 U-Net	Adam	BCE
		Dice+Jaccard
	SGD	BCE
		Dice+Jaccard
RA U-Net	Adam	BCE
		Dice+Jaccard
	SGD	BCE
		Dice+Jaccard

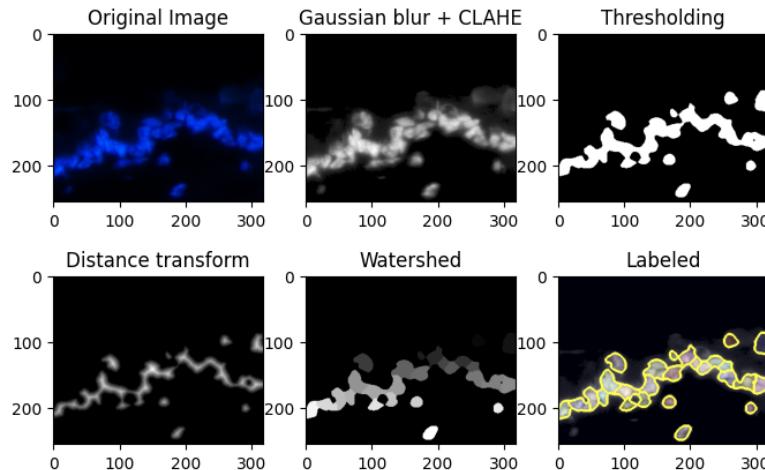
จากผลลัพธ์ของแบบจำลอง U-Net ทั้งหมดจะได้เป็นพื้นที่นิวเคลียสที่ไม่ได้แบ่งแยกกัน จึงจำเป็นต้องมีกระบวนการในการแบ่งนิวเคลียสที่ติดกันและนับจำนวนนิวเคลียส จากงานวิจัยของ Md Mostafa Sark และคณะ [47] ได้ใช้กระบวนการ Distance transform ควบคู่กับ Watershed เพื่อแบ่งนิวเคลียสที่ติดกันและ label นิวเคลียสออกจากกัน ทำให้ได้ผลลัพธ์เป็นตำแหน่งและจำนวนของแต่ละนิวเคลียส โดยตัวอย่างผลลัพธ์แสดงดังรูปภาพที่ 3.4



รูปที่ 3.4: ค่าพื้นที่จริงของนิวเคลียส (ซ้าย), ผลลัพธ์จากแบบจำลอง U-Net (กลาง) และ ผลลัพธ์จาก Watershed (ขวา)

3.3.1.2 การประมวลผลภาพ (Image Processing)

จากการศึกษางานวิจัยที่เกี่ยวข้องในตารางที่ 2.6 คณะผู้วิจัยสรุปขั้นตอนโดยแบ่งออกเป็น 3 ขั้นตอนหลัก ประกอบด้วย การแยกตัวถูกออกจากพื้นหลัง (Thresholding), การแบ่งนิวเคลียสที่ซ้อนทับกัน (Split touching Nucleus) และ การนับจำนวนนิวเคลียส (Nucleus counting) ตัวอย่างผลลัพธ์จากขั้นตอนแสดงดังรูปภาพที่ 3.5



รูปที่ 3.5: ผลลัพธ์จากการใช้กระบวนการ Image Processing ในแต่ละขั้นตอน

- Preprocessing กระบวนการก่อนการประมวลผลภาพนั้นประกอบด้วยการแปลงรูปภาพสีให้อยู่ในช่วงสีเทา (RGB to Grayscale) แล้วทำการลดสัญญาณรบกวนด้วย Gaussian Blur จากนั้นปรับปรุงค่า Contrast ของรูปด้วย Contrast Limited Adaptive Histogram Equalization เพื่อเพิ่มความสว่างของนิวเคลียสให้ชัดเจนยิ่งขึ้น
- Thresholding เราใช้การแยกตัวถูกออกจากพื้นหลังด้วย Short range & Long range Local Average Threshold [4] เนื่องจากรูป Datasets ประกอบด้วยช่วงความสว่างแตกต่างกัน โดย Short range Thresholding จะทำ Thresholding ที่ค่าต่ำ ซึ่งจะได้ผลลัพธ์เป็นขอบของนิวเคลียส แต่อาระมีวัตถุซึ่งเล็กๆ ที่ไม่ใช่นิวเคลียสกูก Thresholding มาด้วย ดังนั้นการทำ Long range Thresholding จะทำ Thresholding ที่ค่าสูงกว่า เมื่อนำค่าทั้งสองมารวมกันจะทำให้สามารถลบช่วงสว่างที่ไม่ใช่นิวเคลียสออกໄไปได้ เหลือเพียงขอบของนิวเคลียสที่สนใจ
- Split touching Nucleus ซึ่งทั้งงานวิจัยของ Wanjun Zhang [4] และ Sonal Kothari [47] ได้ใช้ขั้นตอนที่คล้ายกันนั้น คือการหา Concave point [4] ด้วย Harris Corner Detection และ Notch detection [47] เพื่อใช้ในการระบุจุดที่เป็นมนุษย์ กีดจากช้อนหักกัน ซึ่งสามารถนำไปสู่กระบวนการ Ellipse fitting เพื่อทำการวงขอรูปนิวเคลียส และในส่วนของงานวิจัยของ Sven P. Voigt [49] ได้ใช้กระบวนการที่คล้ายกับงานวิจัยที่กล่าวมาก่อนหน้านี้และเพิ่มกระบวนการ Distance Transform เพื่อนำไปใช้ Watershed ในการแบ่งส่วนนิวเคลียสที่หักช้อนกัน ซึ่งคณะผู้วิจัยได้ใช้วิธีของ Sven P. Voigt [49] ในขั้นตอนนี้ ซึ่งผลลัพธ์ของ Watershed คือ Label ของแต่ละนิวเคลียส
- Nucleus counting จากงานวิจัยของ Sven P. Voigt [49] ได้พัฒนาขั้นตอนของตัวเองจาก scikit-image โดยการนับจำนวน Label ซึ่งจะได้ผลลัพธ์จำนวนนิวเคลียสจากขั้นตอนก่อนหน้าและยังคำนวนพื้นที่ของแต่ละนิวเคลียสที่นับได้จากการคำนวนพื้นที่ของแต่ละ Label ซึ่งคณะผู้วิจัยได้ใช้วิธีนี้สำหรับขั้นตอนนี้

3.3.2 การประเมินผลแบบจำลอง (Evaluation plans)

3.3.2.1 ประเภทของผลลัพธ์จากการเรียนรู้เชิงลึก (Deep Learning) และ การประมวลผลภาพ (Image Processing)

ในกระบวนการวัดผลของทั้งการเรียนรู้เชิงลึก (Deep Learning) และ การประมวลผลภาพ (Image Processing) จะนำผลลัพธ์ที่ได้จากทั้ง 2 กระบวนการ มานับจำนวนนิวเคลียส โดยผลลัพธ์ที่ได้นั้นประกอบด้วย

- Ground Truth Number (GT) หรือ ค่าที่ได้จากการ label ด้วยนักพยาธิวิทยา
- True Positive (TP) หรือ ค่าที่ได้จากการทำนายว่า พื้นที่นิวเคลียสนั้นได้คำตอบเป็นพื้นที่นิวเคลียส
- False Positive (FP) หรือ ค่าที่ได้จากการทำนายว่า พื้นหลังนั้นได้คำตอบเป็นพื้นที่นิวเคลียส
- Flase Negative (FN) หรือ ค่าที่ได้จากการทำนายว่า พื้นที่นิวเคลียสนั้นได้คำตอบเป็นพื้นหลัง
- Estimate Number (EN) หรือ ค่าผลลัพธ์จำนวนนิวเคลียสทั้งหมดที่ได้มาจากการทำนาย

3.3.2.2 วิธีการวัดผลที่นำมาใช้

หลังจากที่ได้ผลลัพธ์ในรูปแบบข้างต้นจึงสามารถเลือกวิธีการวัดผลที่เหมาะสมได้ ซึ่งวิธีการวัดผลที่เราเลือกใช้จะประกอบด้วย

- Precision หรือ ค่าความแม่นยำของตำแหน่งพื้นที่นิวเคลียสริจเทียบกับผลการทำนายตำแหน่งพื้นที่นิวเคลียสทั้งหมด เมื่อ Precision เท่ากับ 1 หมายความว่าความแม่นยำของการทำนายสูงที่สุด หรือ ผลลัพธ์มีค่า FP เท่ากับ 0 และเมื่อ Precision เท่ากับ 0 หมายความว่าความแม่นยำของการทำนายต่ำที่สุด หรือ ผลลัพธ์มีค่า TP เท่ากับ 0 โดยค่า Precision สามารถคำนวณได้ดังสมการ 3.1

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3.1)$$

where: $0 \leq \text{Precision} \leq 1$

- Recall หรือ ค่าความครอบคลุมของตำแหน่งพื้นที่นิวเคลียสริจเทียบกับผลการทำนายตำแหน่งพื้นที่ทั้งหมด เมื่อ Recall เท่ากับ 1 หมายความว่าความครอบคลุมของการทำนายสูงที่สุด หรือ ผลลัพธ์มีค่า FN เท่ากับ 0 และเมื่อ Recall เท่ากับ 0 หมายความว่าความครอบคลุมของการทำนายต่ำที่สุด หรือ ผลลัพธ์มีค่า TP เท่ากับ 0 โดยค่า Recall สามารถคำนวณได้ดังสมการ 3.2

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3.2)$$

where: $0 \leq \text{Recall} \leq 1$

- F1 Score หรือ ค่าเฉลี่ย harmonic ของ Precision และ Recall เมื่อ F1 Score เท่ากับ 1 หมายความว่า Precision และ Recall เท่ากับ 1 ทั้งคู่ และเมื่อ F1 Score เข้าใกล้ 0 หมายความว่า Precision และ Recall เข้าใกล้ 0 ทั้งคู่ ในกรณีที่ Precision และ Recall เท่ากับ 0 จะไม่สามารถคำนวณค่า F1 Score ได้ ซึ่งอนุญาตได้ว่าแบบจำลองนั้นไม่มีความสามารถในการทำนายเลย โดยค่า F1 Score สามารถคำนวณได้ดังสมการ 3.3

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3.3)$$

where: $0 < F1 \leq 1$

and: $\text{Precision}, \text{Recall} \neq 0$

- Estimate Number ratio (ENr) หรือ อัตราส่วนระหว่าง EN เทียบกับ GT เมื่อ Estimate Number ratio เท่ากับ 1 หมายความว่าแบบจำลองทำนายผลลัพธ์ได้คำตอบเท่ากับค่าที่ได้จากการ label ด้วยนักพยาธิวิทยา ซึ่งอนุญาตได้ผลลัพธ์จำนวนนิวเคลียสถูกต้อง เมื่อ Estimate Number ratio มากกว่า 1 หมายความว่าแบบจำลองทำนายผลลัพธ์ได้คำตอบมากกว่าค่าที่ได้จากการ label ด้วยนักพยาธิวิทยา และเมื่อ Estimate Number ratio น้อยกว่า 1 หมายความว่าแบบจำลองทำนายผลลัพธ์ได้คำตอบน้อยกว่าค่าที่ได้จากการ label ด้วยนักพยาธิวิทยา โดยค่า Estimate Number ratio สามารถคำนวณได้ดังสมการ 3.4

$$ENr = \frac{EN}{GT} \quad (3.4)$$

where: $0 \leq ENr \leq \infty$

- Dice Similarity Coefficient (DSC) เป็นวิธีการวัดผลที่ใช้ในการวัดความเหมือนกันระหว่างตำแหน่งนิวเคลียสที่ได้จากนักพยาธิวิทยาและผลลัพธ์จากแบบจำลอง เป็นวิธีวัดผลที่ได้รับความนิยมในด้านการวิเคราะห์ภาพทางการแพทย์ [61] เมื่อ DSC เท่ากับ 1 หมายความว่า พื้นที่ที่นำมายื่นข้อบัญญัติที่จริงพอดี หรืออนุมานได้ว่า แบบจำลองที่นำมายื่นข้อบัญญัติท่องทั้งหมด และเมื่อ DSC เท่ากับ 0 หมายความว่า พื้นที่ที่นำมายื่นข้อบัญญัติที่จริงได้ๆ หรืออนุมานได้ว่า แบบจำลองไม่สามารถทำงานผลลัพธ์ได้ตำแหน่งที่ถูกต้องเลย สามารถคำนวณได้ดังสมการ 3.5

$$DSC = \frac{2|A \cap B|}{|A| + |B|} \quad (3.5)$$

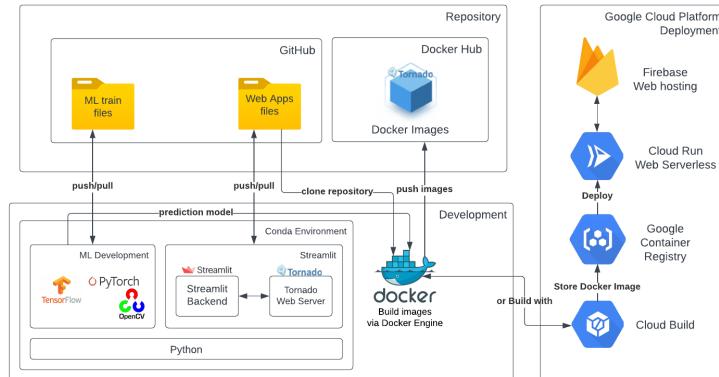
where: $0 \leq DSC \leq 1$

and: A = Area of Ground Truth, B = Area of Predicted

3.4 การพัฒนาเว็บแอปพลิเคชัน

3.4.1 แผนภาพสถาปัตยกรรมของระบบ

คณะกรรมการผู้วิจัยได้พัฒนาเว็บแอปพลิเคชันบนพื้นฐานของภาษา Python ด้วย Framework ต่าง ๆ เข้าด้วยกัน ซึ่งเว็บแอปพลิเคชันสามารถแปลงให้อยู่ในรูปแบบของ Containerization ได้และ deploy ได้บน Web Serverless services ซึ่งคณะกรรมการผู้วิจัยได้แยกการพัฒนาออกเป็นสองส่วน ประกอบด้วย การพัฒนาแบบจำลองในการนับจำนวนนิวเคลียสที่กล่าวในหัวข้อก่อนหน้า และ การพัฒนาเว็บแอปพลิเคชัน แล้วนำอัลกอริทึมที่ดีที่สุดเป็นเบื้องหลังของการนับจำนวนนิวเคลียส จากนั้นแปลง source code ให้อยู่ในรูปแบบของ Docker Image เพื่อนำไป deploy บน Firebase ซึ่งทำหน้าที่เป็น Web hosting ทำงานร่วมกับ Cloud Run ซึ่งทำหน้าที่เป็น Web Serverless services โดยโครงสร้างของเว็บแอปพลิเคชันแสดงดังรูปที่ 3.6

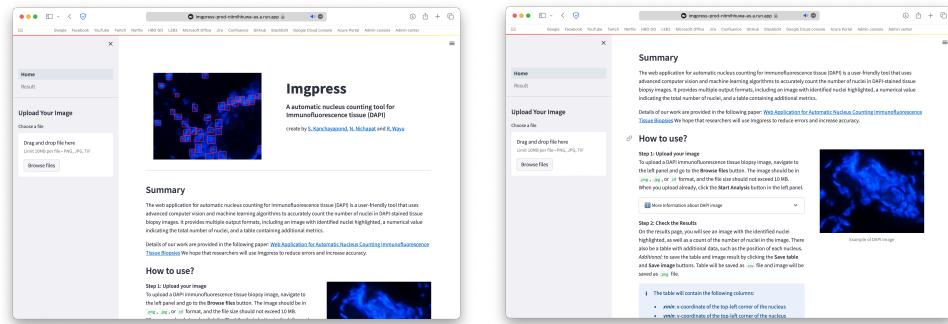


รูปที่ 3.6: โครงสร้างของเว็บแอปพลิเคชัน

3.4.2 ส่วนต่อประสานกับผู้ใช้ (User Interface Design)

ใช้ Streamlit และภาษา Python ในการออกแบบสร้างเว็บแอปพลิเคชัน โดยหน้าเว็บแอปพลิเคชันจะมีส่วนต่าง ๆ ประกอบด้วยสองส่วน

ส่วนหน้าแรก ส่วนหน้าแรกของเว็บแอปพลิเคชัน จะมีรายละเอียดโดยย่อ ดังแสดงในรูปที่ 3.7a และขั้นตอนใช้งานของเว็บแอปพลิเคชัน ดังแสดงในรูปที่ 3.7b



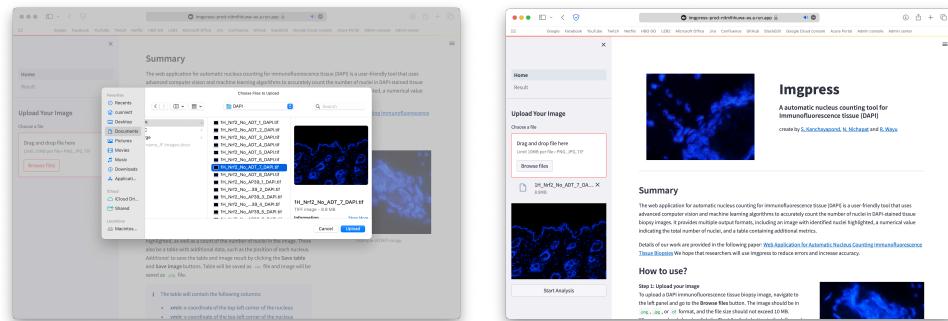
(a) ส่วนหน้าแรกของเว็บแอปพลิเคชัน

(b) ขั้นตอนใช้งานของเว็บแอปพลิเคชัน

รูปที่ 3.7: ส่วนหน้าแรกของเว็บแอปพลิเคชัน

ส่วนอัปโหลดรูป

ส่วนอัปโหลดรูปจะมีคำอธิบายเกี่ยวกับรูปที่จะสามารถอัปโหลดและรูปตัวอย่างบริเวณด้านล่างของปุ่ม Browse files ดังแสดงในรูปหน้าแรกของเว็บแอปพลิเคชันรูปที่ 3.7a ซึ่งปุ่ม Browse files จะใช้ในการอัปโหลดรูปที่ต้องการทราบตำแหน่งของนิวเคลียสมีอยู่กี่ปุ่มแล้วจะขึ้นโพลเดอร์เพื่อให้เลือกรูปที่จะอัปโหลดดังแสดงในรูปที่ 3.8a และเมื่ออัปโหลดรูปที่ต้องการแล้วรูปที่อัปโหลดจะแสดงที่บริเวณทางด้านล่างของปุ่ม Browse files ดังแสดงในรูปที่ 3.8b

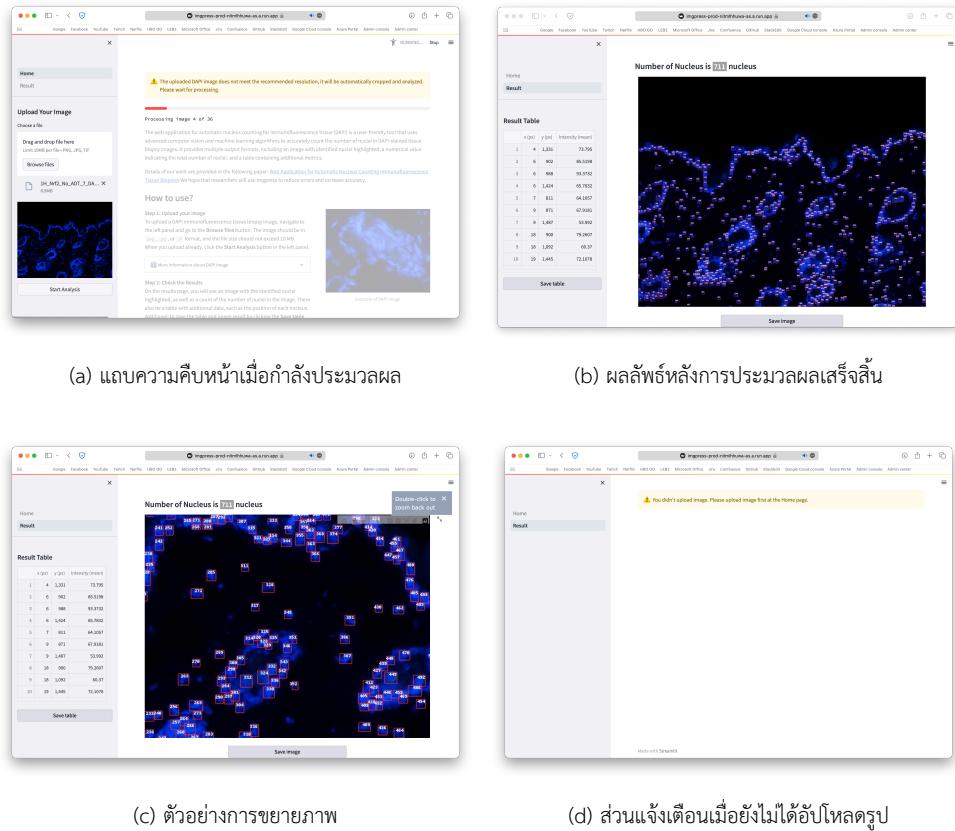


(a) หลังจากกดปุ่ม Browse files จะให้เลือกไฟล์ภาพจาก

(b) รูปที่อัปโหลดจะแสดงที่บริเวณทางด้านล่าง

รูปที่ 3.8: ส่วนอัปโหลดของเว็บแอปพลิเคชัน

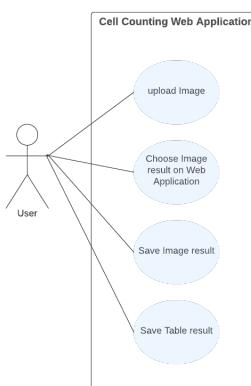
ส่วนผลลัพธ์ หลังจากอัปโหลดรูป จะแสดงແນບความคืบหน้าของการทำงานของเว็บ ดังแสดงดังรูปที่ 3.9a เมื่อประมวลผลเสร็จสิ้นจะนำผู้ใช้งานเข้าสู่หน้าผลลัพธ์ ซึ่งมีการบอกจำนวนของนิวเคลียสที่นับได้ทั้งหมดจากแบบจำลองอยู่บริเวณด้านบนของรูปภาพผลลัพธ์ โดยภาพผลลัพธ์จะแสดงด้านขวา และ ตารางผลลัพธ์จะแสดงในด้านซ้าย ซึ่งจะประกอบด้วย พิกัด (x, y) และ intensity ของแต่ละนิวเคลียส ดังแสดงในรูปที่ 3.9b ซึ่งผู้ใช้งานสามารถย่อขยายในกรณีที่รูปมีขนาดใหญ่ สามารถครอบเลือกเพื่อขยายขนาดรูป ดังแสดงในรูปที่ 3.9c ผู้ใช้งานสามารถบันทึกผลลัพธ์ทั้งหมดได้ จากปุ่ม Save table และ Save Image โดยภาพผลลัพธ์จะเป็นสกุลไฟล์ png และตารางจะเป็นสกุลไฟล์ csv แต่ถ้าหากผู้ใช้งานยังไม่ได้อัปโหลดรูปในหน้าแรกของเว็บแอปพลิเคชันหน้าผลลัพธ์จะขึ้นแจ้งเตือนให้กลับไปอัปโหลดรูปที่หน้าแรกดังแสดงในรูปที่ 3.9d



รูปที่ 3.9: ส่วนผลลัพธ์

3.4.3 UML Design

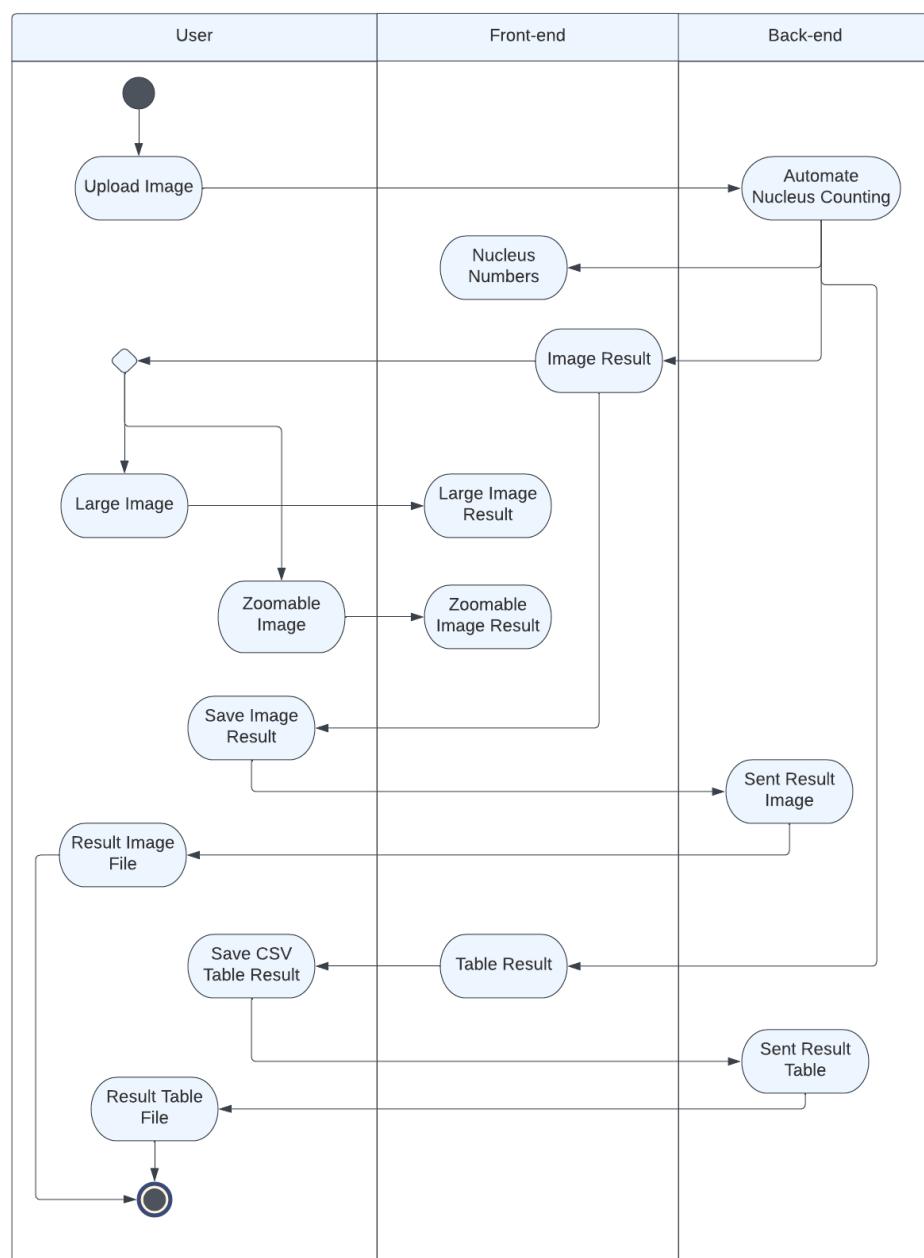
หลังจากกำหนดพังก์ชัน และรายละเอียดการทำงานของระบบบันทึกนิวเคลียสแล้ว สามารถเขียนแผนภาพโดยแกรมเพื่อให้สามารถเข้าใจการทำงานของระบบมากยิ่งขึ้น ด้วยแผนภาพ Use Case Diagram รูปที่ 3.10 แสดง Use Case การทำงานของระบบทั้งหมด ซึ่งผู้ใช้งาน (User) คือ นักพยาธิวิทยาที่ต้องการจำนวนนิวเคลียส จะสามารถอัปโหลดรูปภาพ Immunofluorescence Tissue ปรับรูปแบบแล้วบันทึกนิวเคลียส และบันทึกผลลัพธ์บนหน้าเว็บแอปพลิเคชัน และบันทึกผลลัพธ์ที่มีการ label ตำแหน่งนิวเคลียส และตารางสรุปพื้นที่ของแต่ละนิวเคลียส และเว็บแอปพลิเคชันนี้เป็นแบบ Web Automation ทำให้มีต้องมีในส่วนของ Admin ในการกำกับดูแลเนื้อหาในเว็บแอปพลิเคชันนี้



รูปที่ 3.10: UML Design

3.4.4 Activity Diagrams

ต่อมาเป็นส่วนของ Activity Diagrams ซึ่งจะแสดงความสัมพันธ์ระหว่างผู้ใช้งานกับ Front-end และ Back-end ซึ่งจะมีพังค์ชันและกระบวนการตามรูป รูปที่ 3.11 โดยจะเริ่มจากการรับภาพจากผู้ใช้งาน แล้วจะส่งไปยัง Back-end เพื่อทำการประมวลผลภาพแล้วส่งข้อมูล回來เป็นจำนวนนิวเคลียส รูปที่มีการ label ตำแหน่งนิวเคลียส และตารางสรุปที่ของแต่ละนิวเคลียส ซึ่งผลลัพธ์ที่มีการ label ตำแหน่งนิวเคลียส ผู้ใช้งานสามารถเลือกรูปดูภายในรูปผลลัพธ์ ต่อจากนั้นผู้ใช้งานสามารถบันทึกตาราง โดยสามารถเลือกนามสกุลไฟล์ได้ 2 แบบ คือ csv และ xlsx และ Back-end จะทำการส่งตารางกลับมาสู่ผู้ใช้งาน อีกทั้งผู้ใช้งานยังสามารถบันทึกภาพ ใหญ่ท่าหน้าเว็บแอปพลิเคชัน และแบบที่ผู้ใช้งานสามารถเลือกซูมดูภายในรูปผลลัพธ์ ต่อจากนั้นผู้ใช้งานสามารถบันทึกตาราง โดยสามารถเลือกนามสกุลไฟล์ได้ 2 แบบ คือ csv และ xlsx และ Back-end จะทำการส่งตารางกลับมาสู่ผู้ใช้งาน



รูปที่ 3.11: Activity Diagrams

3.4.5 แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชัน

การประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันจากการใช้แบบสอบถามดังแสดงในรูปที่ 3.12 และ 3.13 จากการสัมภาษณ์ผู้ใช้งาน ได้แก่ บุคคลทั่วไป จำนวน 24 คน และนักพยาธิวิทยา จำนวน 2 คน โดยการประเมินผลความพึงพอใจต่อการใช้งานเว็บแอปพลิเคชันนั้น จะถูกแบ่งออกเป็น 3 ด้านในการประเมิน ได้แก่ ด้านการออกแบบและการจัดการรูปแบบ ด้านการใช้งาน และ ด้านการแสดงผลลัพธ์การนับจำนวนนิวเคลียส ซึ่งในแต่ละหัวข้อจะมีคำตอบได้ 5 รูปแบบ ประกอบด้วย น้อยที่สุด เท่ากับ 1 คะแนน, น้อย เท่ากับ 2 คะแนน, ปานกลาง เท่ากับ 3 คะแนน, มาก เท่ากับ 4 คะแนน และ มากที่สุด เท่ากับ 5 คะแนน โดยสรุปผลคะแนนจะนำทุกคำตอบของแบบสอบถามมาคำนวนค่าเฉลี่ย โดยหัวข้อคำามจะผู้แบ่งออกเป็น 2 ส่วน สำหรับบุคคลทั่วไป และสำหรับนักพยาธิวิทยา มีดังนี้

3.4.5.1 ชุดคำถามสำหรับบุคคลทั่วไป

1. ด้านการออกแบบและการจัดการรูปแบบ

- ความเหมาะสมในการเลือกใช้รูปแบบ ขนาด สีตัวอักษรบนเว็บแอปพลิเคชัน
- ความเหมาะสมในการเลือกใช้ข้อความเพื่ออธิบายการใช้เว็บแอปพลิเคชัน
- ความเหมาะสมในการใช้สัญลักษณ์หรือรูปภาพในการสื่อความหมาย
- ความสวยงามและความทันสมัยของเว็บแอปพลิเคชัน
- ความน่าสนใจของหน้าเว็บแอปพลิเคชัน
- ความเหมาะสมของสีพื้นหลังที่ง่ายต่อการใช้เว็บแอปพลิเคชัน

2. ด้านการแสดงผลลัพธ์การนับจำนวนนิวเคลียส

- รูปภาพผลลัพธ์และตารางมีความชัดเจนและเข้าใจง่าย

3. ด้านการใช้งานได้ตามที่ต้องการ

- การใช้คำสั่งในส่วนการอัปโหลดรูปภาพ ที่ง่ายต่อการใช้งาน
- การใช้คำสั่งในการบันทึกผลลัพธ์ที่ได้ลงอุปกรณ์ของตนเอง ที่ง่ายต่อการใช้งาน
- การใช้คำสั่งการย่อ/ขยายรูปผลลัพธ์ ที่ง่ายต่อการใช้งาน
- ความเร็วในการประมวลผลของเว็บแอปพลิเคชัน
- ประโยชน์ในการใช้งานเว็บแอปพลิเคชัน
- ภาพรวมของเว็บแอปพลิเคชันตรงต่อความต้องการของผู้ใช้

3.4.5.2 ชุดคำถามสำหรับนักพยาธิวิทยา

1. ด้านการออกแบบและการจัดการรูปแบบ

- ความเหมาะสมในการเลือกใช้รูปแบบ ขนาด สีตัวอักษรบนเว็บแอปพลิเคชัน
- ความเหมาะสมในการเลือกใช้ข้อความเพื่ออธิบายการใช้เว็บแอปพลิเคชัน
- ความเหมาะสมในการใช้สัญลักษณ์หรือรูปภาพในการสื่อความหมาย
- ความสวยงามและความทันสมัยของเว็บแอปพลิเคชัน
- ความน่าสนใจของหน้าเว็บแอปพลิเคชัน
- ความเหมาะสมของสีพื้นหลังที่ง่ายต่อการใช้เว็บแอปพลิเคชัน

2. ด้านการแสดงผลลัพธ์การนับจำนวนนิวเคลียส

- ผลลัพธ์ที่ได้จากการทำงานมีความชัดเจนและตรงกับความต้องการของผู้ใช้

- ผลลัพธ์มีความถูกต้อง
- ขนาดของรูปผลลัพธ์มีขนาดที่ความเหมาะสมต่อการมองเห็น

3. ด้านการใช้งานได้ตามพัฟ์ชันการทำงาน

- การใช้คำสั่งในส่วนการอป็อกเกอร์รูปภาพ ที่ง่ายต่อการใช้งาน
- การใช้คำสั่งในการบันทึกผลลัพธ์ที่ได้ลงอุปกรณ์ของตนเอง ที่ง่ายต่อการใช้งาน
- การใช้คำสั่งการย่อ/ขยายรูปผลลัพธ์ ที่ง่ายต่อการใช้งาน
- ความเร็วในการประมวลผลของเว็บแอปพลิเคชัน
- ประโยชน์ในการใช้งานเว็บแอปพลิเคชัน
- ภาพรวมของเว็บแอปพลิเคชันตรงต่อความต้องการของผู้ใช้

แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันของบุคคลทั่วไป

	มากที่สุด	มาก	ปานกลาง	น้อย	น้อยที่สุด
1. ด้านการออกแบบและการจัดการรูปแบบ					
• ความเหมาะสมในการเลือกใช้รูปแบบ ขนาด สีตัวอักษรบนเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมในการเลือกใช้ข้อความเพื่ออธิบายการใช้เงินและเพิ่มความเข้าใจ	<input type="checkbox"/>				
• ความเหมาะสมในการใช้สัญลักษณ์หรือรูปภาพในการสื่อความหมาย	<input type="checkbox"/>				
• ความสวยงามและความทันสมัยของเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความน่าสนใจของหน้าเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมของสีพื้นหลังที่ง่ายต่อการใช้เว็บแอปพลิเคชัน	<input type="checkbox"/>				
2. ด้านการแสดงผลลัพธ์การนับจำนวนบุคคล					
• รูปภาพผลลัพธ์และตารางมีความชัดเจนและเข้าใจง่าย	<input type="checkbox"/>				
3. ด้านการใช้งานได้ตามพัฟ์ชันการทำงาน					
• การใช้คำสั่งในส่วนการอป็อกเกอร์รูปภาพ ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• การใช้คำสั่งในการบันทึกผลลัพธ์ที่ได้ลงอุปกรณ์ของตนเอง ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• การใช้คำสั่งการย่อ/ขยายรูปผลลัพธ์ ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• ความเร็วในการประมวลผลของเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ประโยชน์ในการใช้งานเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ภาพรวมของเว็บแอปพลิเคชันตรงต่อความต้องการของผู้ใช้	<input type="checkbox"/>				

รูปที่ 3.12: แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันบุคคลทั่วไป

แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันของนักพยาธิฯ

	มากที่สุด	มาก	ปานกลาง	น้อย	น้อยที่สุด
1. ด้านการออกแบบและการจัดการรูปแบบ					
• ความเหมาะสมสมใน การเลือกใช้รูปแบบ ขนาด สีด้วยอักษรบนเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมสมในการเลือกใช้รูปภาพเพื่ออธิบายการใช้เว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมสมในการใช้สัญลักษณ์หรือรูปภาพในการเลือกความหมาย	<input type="checkbox"/>				
• ความสวยงามและความทันสมัยของเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความน่าสนใจของหน้าเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมสมของสีที่ให้หลังที่ง่ายต่อการใช้เว็บแอปพลิเคชัน	<input type="checkbox"/>				
2. ด้านการแสดงผลลัพธ์การบันทึกข้อมูลเดียวกัน					
• ผลลัพธ์ที่ได้จากการทำนายมีความถูกต้องและตรงกับความต้องการของผู้ใช้	<input type="checkbox"/>				
• ผลลัพธ์มีความถูกต้อง	<input type="checkbox"/>				
• ขนาดของรูปผลลัพธ์มีขนาดที่ถูกต้องและเหมาะสมต่อการมองเห็น	<input type="checkbox"/>				
3. ด้านการใช้งานได้ด้วยฟังก์ชันการทำงาน					
• การใช้คำสั่งในส่วนการอัปโหลดรูปภาพ ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• การใช้คำสั่งในการบันทึกผลลัพธ์ที่ได้ลงอุปกรณ์ของตนเอง ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• การใช้คำสั่งการย่อ/ขยายรูปผลลัพธ์ ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• ความเร็วในการประมวลผลของเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ประโยชน์ในการใช้งานเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ภาพรวมของเว็บแอปพลิเคชันตรงต่อความต้องการของผู้ใช้	<input type="checkbox"/>				

รูปที่ 3.13: แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันของนักพยาธิฯ

บทที่ 4 ผลการดำเนินงาน

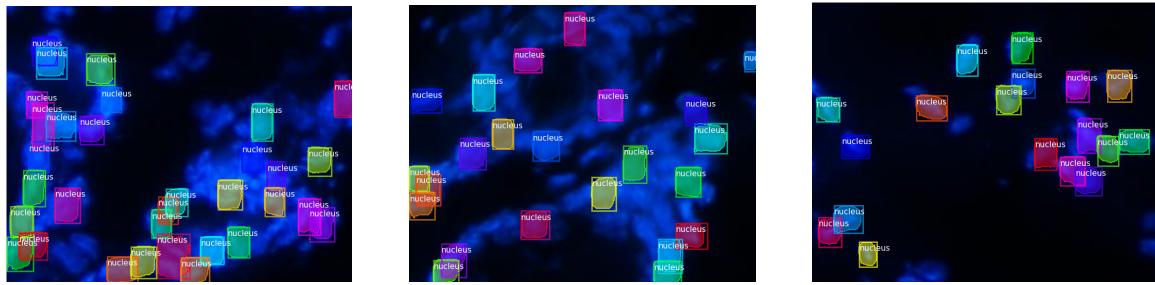
4.1 การวิเคราะห์ข้อมูลและการทดลอง

4.1.1 การทดลองแต่ละแบบจำลองและปรับพารามิเตอร์

โดยการทดลองนี้ได้ทำการทดลองกับแบบจำลองการเรียนรู้เชิงลึกทั้งหมด 3 แบบจำลองได้แก่ Mask R-CNN, YOLO, U-Net และการประมวลผลภาพ (Image Processing)

4.1.1.1 Mask R-CNN

การทดลองปรับพารามิเตอร์ของแบบจำลอง Mask R-CNN คณะผู้จัดได้ทำการทดลองทั้งหมด 2 พารามิเตอร์ โดยในแต่ละพารามิเตอร์จะทำการทดลองปรับค่าพารามิเตอร์เพื่อให้ได้ค่าพารามิเตอร์ที่ทำให้แบบจำลองมีผลลัพธ์ดีที่สุดซึ่งคือ Precision, Recall, F1 Score, ENr และ DSC ได้ค่าการวัดผลที่ดีที่สุด ซึ่งจากตารางที่ 4.1 พบว่าการปรับพารามิเตอร์ momentum เท่ากับ 0.3 ได้ผลลัพธ์ดีที่สุด ตัวอย่างผลลัพธ์จากแบบจำลอง Mask R-CNN แสดงดังรูปที่ 4.1



รูปที่ 4.1: ผลลัพธ์จากแบบจำลอง Mask R-CNN

ตารางที่ 4.1 ตารางผลการทดสอบของแบบจำลอง Mask R-CNN

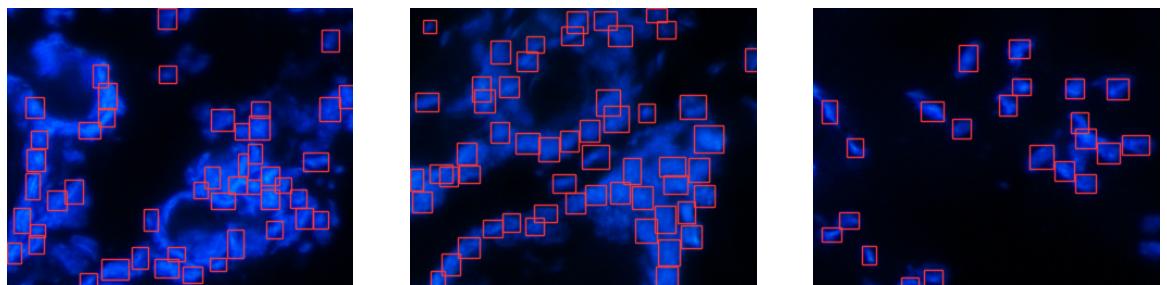
Method	Precision	Recall	F1 Score	ENr	DSC
Base	0.904	0.465	0.601	0.502	0.537
Non-maximum Suppression(NMS)= 0.5	0.941	0.409	0.547	0.456	0.509
momentum = 0.3	0.857	0.581	0.693	0.659	0.571

4.1.1.2 YOLO

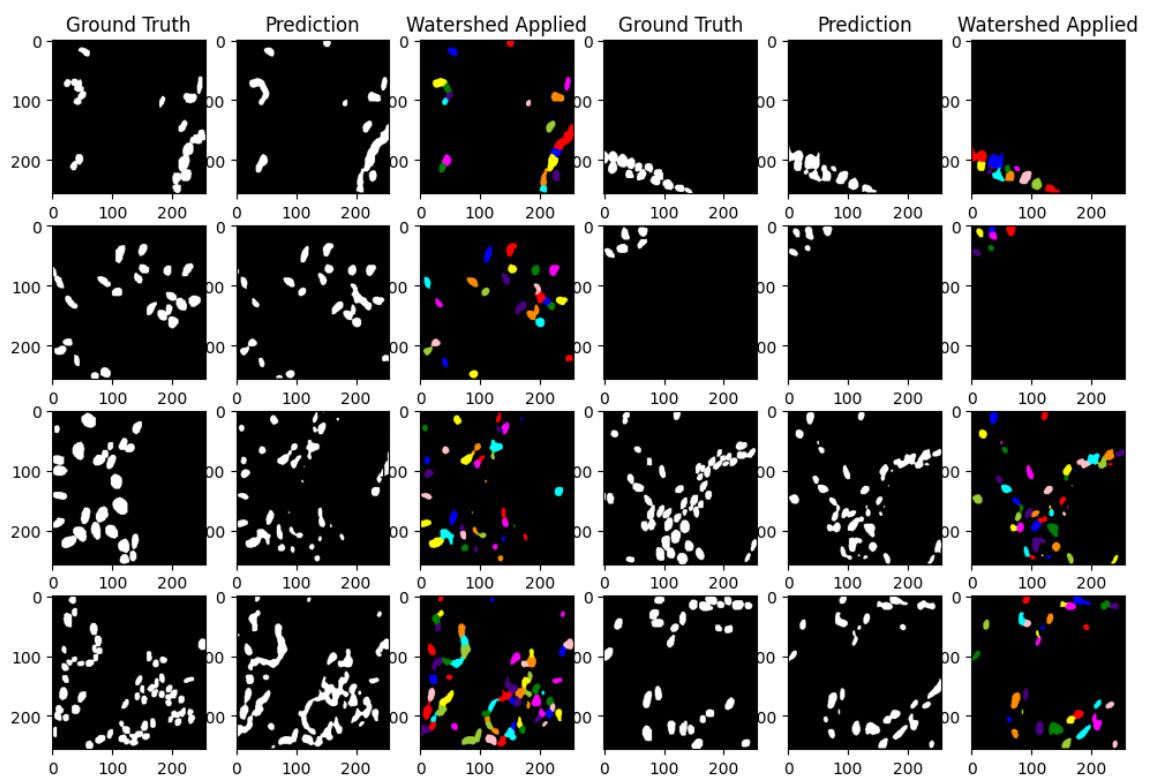
การทดลองปรับพารามิเตอร์ของแบบจำลอง YOLO คณะผู้จัดได้ทำการทดลองทั้งหมด 6 พารามิเตอร์และการทดลองกับพื้นที่ 6 ฟังก์ชันในหนึ่งการทดลอง โดยในแต่ละฟังก์ชันทำการทดลองปรับค่าพารามิเตอร์เพื่อให้ได้ค่าพารามิเตอร์ที่ทำให้แบบจำลองมีผลลัพธ์ค่าการวัดผลที่ดีที่สุดซึ่งคือ Precision, Recall, F1 Score, ENr และ DSC ได้ค่ามากที่สุด ซึ่งจากตารางที่ 4.2 พบว่าการใส่พื้นที่ 6 พารามิเตอร์ได้ค่าการวัดผลได้ผลลัพธ์ดีที่สุด ตัวอย่างผลลัพธ์จากแบบจำลอง YOLO แสดงดังรูปที่ 4.2

4.1.1.3 U-Net

การทดลองปรับพารามิเตอร์ของแบบจำลอง Image Processing คณะผู้จัดได้ปรับค่าพารามิเตอร์เพื่อให้แบบจำลองสามารถทำงานตามที่ต้องการได้โดยใช้ Adam ในการอัปเดตค่าพารามิเตอร์ IRNV2 U-Net with Adam, BCE ได้ค่าการวัดผลได้ผลลัพธ์ดีที่สุด ตัวอย่างผลลัพธ์จากแบบจำลอง U-Net แสดงดังรูปที่ 4.3



รูปที่ 4.2: ผลลัพธ์จากแบบจำลอง YOLO



รูปที่ 4.3: ผลลัพธ์จากแบบจำลอง U-Net

ตารางที่ 4.2 ตารางผลการทดสอบของแบบจำลอง YOLO

Method	Precision	Recall	F1 Score	ENr	DSC
Base	0.802	0.862	0.831	0.907	0.710
Dropout = 0.1	0.802	0.862	0.831	0.918	0.708
Leaky = 0.1	0.831	0.890	0.859	0.881	0.687
Hardswish	0.802	0.862	0.831	0.918	0.708
momentum = 0.90	0.840	0.813	0.826	0.826	0.708
scale image = 0.8	0.859	0.888	0.873	0.859	0.708
object loss = 0.5	0.826	0.805	0.815	0.719	0.683
All of changed parameters	0.896	0.908	0.902	0.823	0.713

ตารางที่ 4.3 ตารางผลการทดสอบของแบบจำลอง U-Net

Method	Precision	Recall	F1 Score	ENr	DSC
U-Net with Adam					
BCE	0.816	0.874	0.835	0.908	0.722
Dice+Jaccard	0.793	0.817	0.780	0.905	0.706
U-Net with SGD					
BCE	0.763	0.926	0.824	1.506	0.730
Dice+Jaccard	0.812	0.884	0.834	0.936	0.724
IRNV2 U-Net with Adam					
BCE	0.840	0.926	0.881	0.908	0.740
Dice+Jaccard					
IRNV2 U-Net with SGD					
BCE	0.719	0.896	0.780	1.090	0.645
Dice+Jaccard	0.761	0.924	0.823	0.941	0.732
RA U-Net with Adam					
BCE	0.803	0.858	0.814	0.962	0.733
Dice+Jaccard	0.804	0.899	0.840	0.895	0.733
RA U-Net with SGD					
BCE	0.802	0.920	0.849	0.954	0.728
Dice+Jaccard	0.800	0.881	0.827	0.897	0.725

4.1.1.4 การประมวลผลภาพ (Image Processing)

การทดลองปรับพารามิเตอร์ของการประมวลผลภาพ คณะผู้วิจัยได้ปรับค่าพารามิเตอร์เพื่อให้แบบจำลองสามารถทำงานได้ใกล้เคียงมากที่สุด ซึ่งจากการที่ 4.4 แสดงให้เห็นว่าแบบจำลองหลังจากปรับพารามิเตอร์แล้วสามารถทำได้ดีขึ้น

ตารางที่ 4.4 ตารางผลการทดลองการปรับพารามิเตอร์ของแบบจำลอง Image Processing

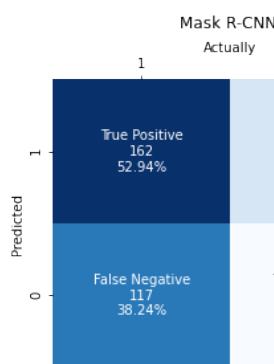
Method	Precision	Recall	F1 Score	ENr	DSC
Image Processing	0.657	0.953	0.778	1.083	0.670

4.1.2 การเลือกแบบจำลอง

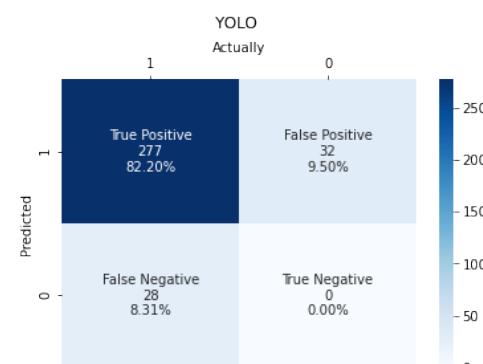
คณะกรรมการวิจัยทำการโดยเลือกผลการทดลองที่ดีที่สุดจากทุกแบบจำลอง แสดงในตารางที่ 4.5 พบว่าแต่ละแบบจำลองมีค่า F1 Score เป็นหลัก เนื่องจากเป้าหมายของงานวิจัยนี้คือการนับจำนวนให้ถูกต้องที่สุด การเลือกใช้เพียงค่า Precision จะสามารถวัดผลได้เฉพาะสัดส่วนความถูกต้องจากการทำงาน และค่า Recall จะสามารถวัดผลสัดส่วนความถูกต้องจากการจริง ดังนั้นคณะกรรมการวิจัยจึงเลือกค่า F1 Score เป็นหลัก เนื่องจากครอบคลุมทั้งความสัดส่วนความถูกต้องจากการผลการทำงาน และสัดส่วนความถูกต้องจากการจริง ดังนั้น แบบจำลองที่ดีที่สุดคือ YOLO เนื่องจากมีค่า F1 Score สูงที่สุด

ตารางที่ 4.5 ตารางรวมผลการทดลองที่ดีที่สุดของแต่ละแบบจำลอง

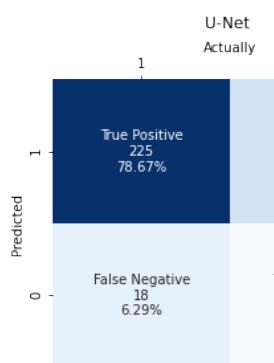
Model	Precision	Recall	F1 Score	ENr	DSC
Mask R-CNN	0.857	0.581	0.693	0.659	0.509
YOLO	0.896	0.908	0.902	0.823	0.713
U-Net	0.840	0.926	0.881	0.908	0.740
Image Processing	0.657	0.953	0.778	1.083	0.670



(a) ตาราง confusion matrix ของ MaskR-CNN



(b) ตาราง confusion matrix ของ YOLO



(c) ตาราง confusion matrix ของ U-Net



(d) ตาราง confusion matrix ของ Image Processing

รูปที่ 4.4: ตาราง confusion matrix

จากผลลัพธ์ Mask R-CNN มีค่า False Negative สูงจึงทำให้มีค่า Recall ต่ำ เช่นเดียวกันกับ Image Processing มีค่า False Positive สูงจึงทำให้มีค่า Precision ต่ำ

4.2 ความพึงพอใจในการใช้เว็บแอปพลิเคชัน

จากแบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชัน ดังแสดงในรูปที่ 3.12 และ 3.13 พบว่ามีจำนวนผู้ทำแบบประเมินทั้งหมด 26 คน โดยแบ่งเป็น บุคคลทั่วไป จำนวน 24 คน และนักพยาธิวิทยา จำนวน 2 คน ซึ่งในแต่ละหัวข้อจะมีคำตอบได้ 5 รูปแบบ ประกอบด้วย น้อยที่สุด เท่ากับ 1 คะแนน, น้อย เท่ากับ 2 คะแนน, ปานกลาง เท่ากับ 3 คะแนน, มาก เท่ากับ 4 คะแนน และ มากที่สุด เท่ากับ 5 โดยผลคะแนนที่ได้นั้นมาจากการนำทุกคำตอบของแบบสอบถามมาคำนวนหาค่าเฉลี่ย ซึ่งได้ผลลัพธ์ดังนี้

ตารางที่ 4.6 ตารางความพึงพอใจในการใช้เว็บแอปพลิเคชันของบุคคลทั่วไป

หัวข้อ	คะแนนเต็ม (5 คะแนน)
1. ด้านการออกแบบและการจัดการรูปแบบ	
• ความเหมาะสมในการเลือกใช้รูปแบบ ขนาด สีตัวอักษรบนเว็บแอปพลิเคชัน	4.75
• ความเหมาะสมในการเลือกใช้ข้อความเพื่ออธิบายการใช้เว็บแอปพลิเคชัน	4.17
• ความเหมาะสมในการใช้สัญลักษณ์หรือรูปภาพในการสื่อความหมาย	4.17
• ความสวยงามและความทันสมัยของเว็บแอปพลิเคชัน	4.50
• ความน่าสนใจของหน้าเว็บแอปพลิเคชัน	4.20
• ความเหมาะสมของสีพื้นหลังที่ง่ายต่อการใช้เว็บแอปพลิเคชัน	4.83
คะแนนเฉลี่ย	4.47
2. ด้านการแสดงผลลัพธ์การนับจำนวนนิวเคลียส	
• รูปภาพผลลัพธ์และตารางมีความชัดเจนและเข้าใจ	4.54
คะแนนเฉลี่ย	4.54
3. ด้านการใช้งานได้ตามฟังก์ชันการทำงาน	
• การใช้คำสั่งในส่วนการอัปโหลดรูปภาพ ที่ง่ายต่อการใช้งาน	4.83
• การใช้คำสั่งในการบันทึกผลลัพธ์ที่ได้ลงอุปกรณ์ของตนเอง ที่ง่ายต่อการใช้งาน	4.83
• การใช้คำสั่งการย่อ/ขยายรูปผลลัพธ์ ที่ง่ายต่อการใช้งาน	4.33
• ความเร็วในการประมวลผลของเว็บแอปพลิเคชัน	4.16
• ประโยชน์ในการใช้งานเว็บแอปพลิเคชัน	4.70
• ภาระรวมของเว็บแอปพลิเคชันตรงต่อความต้องการของผู้ใช้	4.54
คะแนนเฉลี่ย	4.56

จากตาราง 4.6 พบว่า ด้านการออกแบบและการจัดการรูปแบบมีคะแนนเฉลี่ยอยู่ที่ 4.47, ด้านการแสดงผลลัพธ์การนับจำนวนนิวเคลียสที่ได้คะแนนเฉลี่ยอยู่ที่ 4.54 และด้านการใช้งานได้ตามฟังก์ชันการทำงานได้คะแนนเฉลี่ยอยู่ที่ 4.56 ซึ่งคะแนนทั้ง 3 ด้าน เมื่อนำมาเทียบกับเกณฑ์การประเมิน สามารถแปลผลได้ว่า ผู้ใช้งานมีความพึงพอใจต่อการใช้เว็บแอปพลิเคชันอยู่ในระดับมาก

ตารางที่ 4.7 ตารางความพึงพอใจในการใช้เว็บแอปพลิเคชันของนักพยาธิวิทยา

หัวข้อ	คะแนนเต็ม (5 คะแนน)
1. ด้านการออกแบบและการจัดการรูปแบบ	
• ความเหมาะสมในการเลือกใช้รูปแบบ ขนาด สีตัวอักษรบนหน้าจอแอปพลิเคชัน	4.50
• ความเหมาะสมในการเลือกใช้ข้อความเพื่ออธิบายการใช้เว็บแอปพลิเคชัน	4.50
• ความเหมาะสมในการใช้สัญลักษณ์หรือรูปภาพในการสื่อความหมาย	4.50
• ความสวยงามและความทันสมัยของเว็บแอปพลิเคชัน	4.50
• ความน่าสนใจของหน้าเว็บแอปพลิเคชัน	4.50
• ความเหมาะสมของสีพื้นหลังที่ง่ายต่อการใช้เว็บแอปพลิเคชัน	4.00
คะแนนเฉลี่ย	4.42
2. ด้านการแสดงผลลัพธ์การนับจำนวนนิวเคลียส	
• ผลลัพธ์ที่ได้จากการคำนวณมีความซัดเจนและตรงกับความต้องการของผู้ใช้	4.50
• ผลลัพธ์มีความถูกต้อง	4.50
• ขนาดของรูปผลลัพธ์มีขนาดที่คุณภาพเหมาะสมต่อการมองเห็น	4.50
คะแนนเฉลี่ย	4.50
3. ด้านการใช้งานได้ตามฟังก์ชันการทำงาน	
• การใช้คำสั่งในส่วนการอัปโหลดรูปภาพ ที่ง่ายต่อการใช้งาน	4.00
• การใช้คำสั่งในการบันทึกผลลัพธ์ที่ได้ลงอุปกรณ์ของตนเอง ที่ง่ายต่อการใช้งาน	4.50
• การใช้คำสั่งการย่อ/ขยายรูปผลลัพธ์ ที่ง่ายต่อการใช้งาน	4.50
• ความเร็วในการประมวลผลของเว็บแอปพลิเคชัน	4.00
• ประโยชน์ในการใช้งานเว็บแอปพลิเคชัน	4.50
• ภาพรวมของเว็บแอปพลิเคชันตรงต่อความต้องการของผู้ใช้	4.50
คะแนนเฉลี่ย	4.33

จากตาราง 4.7 พบร่วมกับ ด้านการออกแบบและการจัดการรูปแบบมีคะแนนเฉลี่ยอยู่ที่ 4.42, ด้านการแสดงผลลัพธ์การนับจำนวนนิวเคลียสที่ได้คะแนนเฉลี่ยอยู่ที่ 4.50 และด้านการใช้งานได้ตามฟังก์ชันการทำงานได้คะแนนเฉลี่ยอยู่ที่ 4.33 ซึ่งคะแนนทั้ง 3 ด้าน เมื่อนำมาเทียบกับเกณฑ์การประเมิน สามารถแปลผลได้ว่า ผู้ใช้งานมีความพึงพอใจต่อการใช้เว็บแอปพลิเคชันอยู่ในระดับมาก

4.3 ผลการทดสอบระบบ

การทดสอบระบบจะจัดทำเมื่อเว็บแอปพลิเคชันพัฒนาเสร็จสมบูรณ์ การทดสอบทั้งหมดจะทำในรูปแบบการทดสอบด้วยตนเองจากการทดสอบระบบในแต่ละกรณีได้ผลลัพธ์สรุปได้ดังตารางที่ 4.8

ตารางที่ 4.8 ตารางสรุปผลการทดสอบระบบในแต่ละกรณี

หัวข้อ	ผลลัพธ์	
	ผ่าน	ไม่ผ่าน
1. การทดสอบการทำงานของระบบ		
• การ Build และ Deploy บน Docker Desktop	✓	
• การ Build ด้วย GCP Cloud Build	✓	
• การ Deploy ด้วย GCP Cloud Run และใช้ Firebase Hosting	✓	
2. การทดสอบการทำงานของส่วนต่อประสานกับผู้ใช้		
• การอัปโหลดรูปภาพ	✓	
• การแสดงรูปผลลัพธ์	✓	
• การดาวน์โหลดรูปภาพผลลัพธ์	✓	
• การดาวน์โหลดตารางผลลัพธ์	✓	

ผลการทดสอบระบบทั้งหมด ผ่านการทดสอบทุกส่วน โดยรายละเอียดของการทดสอบระบบในแต่ละหัวข้อที่ได้สรุปดังตารางที่ 4.8 สามารถแสดงรายละเอียดการทดสอบระบบในแต่ละหัวข้อได้ดังตารางที่ 4.9 ถึง ตารางที่ 4.15

ตารางที่ 4.9 การ Build และ Deploy บน Docker Desktop

Test Case: การ Build และ Deploy บน Docker Desktop

Steps	Input	
	Type	Example Input
1. สร้าง Dockerfile	-	-
2. สร้าง Docker Image ด้วย Docker Build	-	-
3. สร้าง Container จาก Image เพื่อทดสอบการ Deploy	-	-
4. เช็คเมล์เว็บแอปพลิเคชันผ่าน localhost port 8080	-	-

Constraint: -

Expected Output: สามารถเข้าชมเว็บแอปพลิเคชันผ่าน localhost port 8080 และเว็บแอปพลิเคชันมีการทำงานที่สมบูรณ์

Status: ผ่านการทดสอบ

Comment: -

ตารางที่ 4.10 การ Build ด้วย GCP Cloud Build

Test Case: การ Build ด้วย GCP Cloud Build

Steps	Input	
	Type	Example Input
1. ติดตั้ง gcloud CLI และ Login	-	-
2. สร้าง Project บน GCP	-	-
3. สร้าง Image ด้วย GCP Cloud Build	-	-
4. จัดเก็บที่ GCP Container Registry	-	-

Constraint: จำเป็นต้อง Activate GCP Account เพื่อใช้งาน GCP Container Registry

Expected Output: พับ Image ใน GCP Container Registry ของ Project ที่ได้สร้างไว้

Status: ผ่านการทดสอบ

Comment: การใช้งาน GCP Container Registry มีค่าใช้จ่ายตามปริมาณการใช้งาน

ตารางที่ 4.11 การ Deploy ด้วย GCP Cloud Run และใช้ Firebase Hosting

Test Case: การ Deploy ด้วย GCP Cloud Run และใช้ Firebase Hosting

Steps	Input	
	Type	Example Input
1. ตรวจสอบ Image ใน GCP Container Registry	-	-
2. Deploy Image บน Cloud Run	-	-
3. เข้ามายังแอปพลิเคชันผ่าน URL ที่ขึ้นสร้างอัตโนมัติ	-	-
4. ตั้งค่า firebase.json เพื่อเตรียม Deploy Hosting	-	-
5. Deploy Hosting URL ที่เข้าสู่ระบบ Redirect ไปยัง URL จาก Cloud Run	-	-

Constraint: การใช้งาน GCP Cloud Run จะไม่สามารถกำหนด URL ได้เอง

Expected Output: สามารถเข้าใช้งาน URL จาก Firebase ที่ทำหน้าที่ Redirect ไปยัง URL ของ Cloud Run

Status: ผ่านการทดสอบ

Comment: Firebase จะสร้าง URL ตามชื่อของ Project บน GCP

ตารางที่ 4.12 การอัปโหลดรูปภาพ

Test Case: การอัปโหลดรูปภาพ

Steps	Input	
	Type	Example Input
1. กดปุ่ม Browse Files ใน Sidebar	Image	.png, .jpg, .tif ขนาดไม่เกิน 10MB
2. เลือกรูปที่จะอัปโหลด	-	-
3. แสดงผลรูปที่อัปโหลดใน Sidebar	-	-
4. กดปุ่ม Start Analyze เพื่อเริ่มการนับจำนวน	-	-

Constraint: ผู้ใช้งานจะไม่สามารถเลือกไฟล์ที่มีสกุลที่ไม่ตรงกับที่ระบุ

Expected Output: สามารถอัปโหลดรูปได้ เท็ງภาพตัวอย่างที่อัปโหลด และสามารถกดปุ่ม Start Analyze

Status: ผ่านการทดสอบ

Comment: กรณีที่ขนาดไฟล์มากกว่าที่กำหนด จะอัปโหลดไม่สมบูรณ์

ตารางที่ 4.13 การแสดงรูปผลลัพธ์

Test Case: การแสดงรูปผลลัพธ์

Steps	Input	
	Type	Example Input
1. รูปผลลัพธ์จากการนับจะแสดงตรงกลางและเติมพื้นที่เว็บ	-	-
2. คลิกลากเพื่อกำหนดขอบเขตที่ต้องการขยายภาพ	-	-

Constraint: -

Expected Output: ผู้ใช้งานสามารถขยายภาพผลลัพธ์ที่ต้องการได้ เพื่อตรวจสอบความถูกต้องของการนับ

Status: ผ่านการทดสอบ

Comment: ต้องเพิ่มคำอธิบายเพื่อให้ผู้ใช้งานเข้าใจวิธีการขยายภาพผลลัพธ์

ตารางที่ 4.14 การดาวน์โหลดรูปภาพผลลัพธ์

Test Case: การดาวน์โหลดรูปภาพผลลัพธ์

Steps	Input	
	Type	Example Input
1. กดปุ่ม Save Image ใต้รูปผลลัพธ์	-	-
2. กำหนดชื่อไฟล์เพื่อดownload ให้เป็น .png	-	-

Constraint: รูปผลลัพธ์จะอยู่ในสกุลไฟล์ .png เท่านั้น

Expected Output: ผู้ใช้งานสามารถดาวน์โหลดรูปผลลัพธ์ได้

Status: ผ่านการทดสอบ

Comment: -

ตารางที่ 4.15 การดาวน์โหลดตารางผลลัพธ์

Test Case: การดาวน์โหลดตารางผลลัพธ์

Steps	Input	
	Type	Example Input
1. ตารางผลลัพธ์จะแสดงตรงกลางและเติมพื้นที่เว็บต่อจากรูปผลลัพธ์	-	-
2. กดปุ่ม Save Image ใต้รูปผลลัพธ์	-	-
3. กำหนดชื่อไฟล์เพื่อดownload ให้เป็น .csv	-	-

Constraint: ตารางผลลัพธ์จะอยู่ในสกุลไฟล์ .csv เท่านั้น

Expected Output: ผู้ใช้งานสามารถดาวน์โหลดตารางผลลัพธ์ได้

Status: ผ่านการทดสอบ

Comment: -

บทที่ 5 บทสรุป

5.1 สรุปผลงานวิจัย

งานวิจัยเว็บแอปพลิเคชันสำหรับการตรวจสอบนิวเคลียสจากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง ทดลองหาตำแหน่งนิวเคลียสโดยการใช้แบบจำลองต่าง ๆ เพื่อหาแบบจำลองที่ได้ค่าการวัดผลที่สูงที่สุด จากชุดข้อมูลภาพถ่ายเนื้อเยื่อบริเวณผิวนังที่ผ่านการย้อมด้วยสารเรืองแสง DAPI สรุปได้ว่าแบบจำลอง YOLO มีประสิทธิภาพดีที่สุดโดยมีค่า F1-score เท่ากับ 0.902 หลังจากนั้นนำแบบจำลองมาใช้ในการนับจำนวนของนิวเคลียสในรูปแบบเว็บแอปพลิเคชัน โดยเว็บแอปพลิเคชันพัฒนาบนแพลตฟอร์ม Python ด้วยเครื่องมือ Streamlit ที่ช่วยจัดการเรื่อง Front-end และ Back-end เข้าด้วยกัน และนำไป deploy บน Firebase ซึ่งทำหน้าที่เป็น Web hosting ทำงานร่วมกับ Cloud Run

5.2 สรุปผลการดำเนินงาน

ตารางที่ 5.1 ตารางสรุปสถานะการดำเนินงาน

การดำเนินงาน	สถานะ
การออกแบบภาพรวมการดำเนินงานวิจัย	
การออกแบบภาพรวมการทดลองเพื่อหาแบบจำลองที่ดีที่สุด	เสร็จสมบูรณ์
การออกแบบภาพรวมการพัฒนาเว็บแอปพลิเคชัน	เสร็จสมบูรณ์
การนำเข้าและจัดเตรียมข้อมูล	
การทำ label ตำแหน่งนิวเคลียส	เสร็จสมบูรณ์
การแบ่งรูปเป็นรูปอย่าง ๆ	เสร็จสมบูรณ์
การทดลองเพื่อหาแบบจำลองที่ดีที่สุด	
การเรียนรู้เชิงลึก: Mask R-CNN	เสร็จสมบูรณ์
การเรียนรู้เชิงลึก: YOLO	เสร็จสมบูรณ์
การเรียนรู้เชิงลึก: U-Net	เสร็จสมบูรณ์
การประมวลผลภาพ	เสร็จสมบูรณ์
การประมวลผลแบบจำลอง	เสร็จสมบูรณ์
สรุปผลและคัดเลือกแบบจำลอง	เสร็จสมบูรณ์
การพัฒนาเว็บแอปพลิเคชัน	
การออกแบบสถาปัตยกรรมของระบบ	เสร็จสมบูรณ์
การพัฒนาส่วนต่อประสานกับผู้ใช้	เสร็จสมบูรณ์
การทดสอบระบบของเว็บแอปพลิเคชัน	เสร็จสมบูรณ์
การประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชัน	เสร็จสมบูรณ์

5.3 ปัญหาที่พบและแนวทางการแก้ไข

- ประเด็นปัญหา: รูปจากชุดข้อมูลมีขนาดที่ใหญ่เกินไป ส่งผลต่อประสิทธิภาพของการพัฒนาแบบจำลอง
แนวทางการแก้ไข: แบ่งรูปออกเป็นจำนวน 36 รูปต่อ 1 รูปจริง ตามขนาดที่เหมาะสมกับการพัฒนาแบบจำลอง
- ประเด็นปัญหา: รูปจากชุดข้อมูลไม่มีการ label ตำแหน่งของนิวเคลียส
แนวทางการแก้ไข: นำรูปบางส่วนจากขั้นตอนการแบ่งรูป ให้นักพยาธิวิทยา label ตำแหน่งของนิวเคลียส
- ประเด็นปัญหา: รูปจากชุดข้อมูลมีจำนวนการ label ที่ไม่เที่ยง泊 เนื่องจากข้อจำกัดทางด้านเวลา
แนวทางการแก้ไข: ทำการ augmentation เพื่อเพิ่มจำนวนรูปภาพก่อนนำเข้าแบบจำลอง
- ประเด็นปัญหา: ผลจากการนับจำนวนมีกรณีที่น้อยกว่าจากผลการวัดผลด้วย Confusion Matrix
แนวทางการแก้ไข: เพิ่มวิธีการวัดผลด้วย DSC เพื่อให้ครอบคลุมเงื่อนไขการซ้อนทับของผลการนับนิวเคลียส

5.4 ประโยชน์ที่ได้รับจากการดำเนินงาน

1. ได้เรียนรู้ลักษณะพื้นฐานของนิวนิวเคลียร์ที่ได้จากภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง การประเมินคุณภาพ และการวิเคราะห์ข้อมูล
2. ได้เรียนรู้เกี่ยวกับภาษาโปรแกรมคอมพิวเตอร์อื่น ๆ รวมถึงเครื่องมือและซอฟต์แวร์ที่เกี่ยวข้อง
3. ได้เรียนรู้พื้นฐานการทำงานของแต่ละอัลกอริทึมที่ใช้ในการวิเคราะห์ภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสง
4. ได้พัฒนาทักษะการแปลผล การตีความข้อมูล และการหาข้อสรุปจากผลลัพธ์ที่ได้ รวมถึงการพัฒนาทักษะการวางแผนเพื่อให้สามารถดำเนินการออกแบบการทดสอบกับภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงที่มีความซับซ้อนให้ได้ผลลัพธ์ที่น่าพึงพอใจที่สุดภายในระยะเวลาที่กำหนด

5.5 แนวทางการพัฒนาและประยุกต์ใช้ในอนาคต

1. พัฒนาแบบจำลองให้มีประสิทธิภาพเพิ่มขึ้น โดยการเพิ่มภาพที่ใช้ในการเรียนรู้เชิงลึกของแบบจำลอง หรือทดลองนำการประมวลผลภาพมาใช้งานควบคู่กับการเรียนรู้เชิงลึก
2. พัฒนาเว็บแอปพลิเคชันสามารถประมวลผลภาพถ่ายเนื้อเยื่อจากการย้อมด้วยสารเรืองแสงชนิดอื่นควบคู่ไปกับตำแหน่งนิวนิวเคลียร์ เช่นการหาตำแหน่งร่วมกันของโปรตีนและนิวนิวเคลียร์ การหาสัดส่วนระหว่างโปรตีนต่อจำนวนนิวนิวเคลียร์
3. ปรับปรุงและพัฒนาเว็บแอปพลิเคชัน สามารถประมวลผลภาพได้มากกว่าหนึ่งรูป และใช้งานง่าย
4. ปรับปรุงและพัฒนาเว็บแอปพลิเคชัน สามารถประมวลผลภาพได้นอกเหนือจากเนื้อเยื่อบริเวณผิวนิ้วหนัง

หนังสืออ้างอิง

1. J. Smith, 2010, "Nucleus and its function in the cell," *Cell Biology International*, vol. 34, no. 3, pp. 267–271, 2010.
2. John Doe, 2023, "Nucleus: The Unit for Genetic Code Storage and Protein Synthesis," *Journal of Cellular Biology*, vol. 123, no. 1, pp. 1–10, 2023.
3. PerkinElmer, "Columbus Columbusplus," Available at <https://www.perkinelmer.com/product/image-data-storage-and-analysis-system-columbus> (2022/10/09).
4. Gyuhyun Lee, Jeong-Woo Oh, Nam-Gu Her, and Won-Ki Jeong, 2021, "DeepHCS : Bright-field to fluorescence microscopy image conversion using multi-task learning with adversarial losses for label-free high-content screening," *Medical Image Analysis*, vol. 70, pp. 101995, 2021.
5. Daniel Jimenez-Carretero, Vahid Abrishami, Laura Fernández-de Manuel, Irene Palacios, Antonio Quílez-Álvarez, Alberto Díez-Sánchez, Miguel A Del Pozo, and María C Montoya, 2018, "Tox_(R)CNN: Deep learning-based nuclei profiling tool for drug toxicity screening," *PLoS Comput. Biol.*, vol. 14, no. 11, pp. e1006238, Nov. 2018.
6. Wafaa Rajaa Drioua, Nacéra Benamrane, and Lakhdar Sais, 2022, "Breast Cancer Detection from Histopathology Images Based on YOLOv5," in **2022 7th International Conference on Frontiers of Signal Processing (ICFSP)**, 2022, pp. 30–34.
7. Ben N.G. Giepmans, Andrew C. York, and G. Ulrich Nienhaus, 2015, "Fluorescence Microscopy: High-Resolution Imaging with Photons," in *Annual Review of Biochemistry*, vol. 84, pp. 709–743.
8. John P. Bilezikian, T. John Martin, Thomas L. Clemens, and Clifford J. Rosen, 2020, *Principles of Bone Biology*, Elsevier/Academic Press.
9. Mei-Shang Ho, Jen-Ren Wang, and Kuan-Jen Chen, 1992, "Immunofluorescence in diagnostic virology," *Journal of virological methods*, vol. 39, no. 3, pp. 261–274, 1992.
10. Aurélie Baronnet, "The importance of image processing," Available at <https://www.blog.i2s.fr/en/the-importance-of-image-processing/> (2022/10/15).
11. BSTSoft, 2015, "Image processing: convert RGB to gray scale (part 1)," Available at <http://www.babsoftsoft.blogspot.com/2015/10/image-processing-convert-gray-scale> (2022/09/27).
12. "Image Processing," Available at http://digital_collect.lib.buu.ac.th/dcms/files/50910453/chapter2.pdf (2022/10/03).
13. OpenCV, "Histogram Equalization," Available at https://docs.opencv.org/3.4/d4/d1b/tutorial_histogram_equalization.html (2022/11/20).
14. OpenCV, Accessed 2023, "Histogram Equalization," https://docs.opencv.org/3.4/d4/d1b/tutorial_histogram_equalization.html.
15. 2021, "Clahe histogram equalization - opencv," Available at <https://www.geeksforgeeks.org/clawe-histogram-equalization-opencv/> (2022/11/23).
16. Pichet Wayalum, 2013, "Evaluation of automated cheomosome G-band Images Enhancement using histogram equalization," 2013.
17. Jing Xu and Yongqiang Liu, 2014, "Fog Removal from Color Images using Contrast Limited Adaptive Histogram Equalization and Normalized Cut Segmentation," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 7, no. 3, pp. 199–208, 2014.
18. Anjing Guo, Yirui Wang, Lijun Guo, Rong Zhang, Yang Yu, and Shangce Gao, 2023, "An adaptive position-guided gravitational search algorithm for function optimization and image threshold segmentation," *Engineering Applications of Artificial Intelligence*, vol. 121, pp. 106040, 2023.
19. Guoying Qi, Jian Sun, and Jia Jia, 1999, "Watershed based image segmentation using hierarchical image representation," *Pattern Recognition*, vol. 32, no. 8, pp. 1289–1299, 1999.
20. Advance Innovation Centre, 2020, "Gaussian blur," Available at <https://docs.aic-eec.com/artificial-intelligence-ai/computer-vision-for-python/lab-2-basic-cv/gaussian-blur> (2022/09/30).

21. "Morphological Image Processing," Available at <http://fivedots.coe.psu.ac.th/~montri/Teaching/240-373/Chap6.pdf> (2022/10/03).
22. Ryan Gibson, Ali Ahmadinia, Scott Mcmeekin, Niall Strang, and Gordon Morison, 2013, "A reconfigurable real-time morphological system for augmented vision," *EURASIP Journal on Applied Signal Processing*, vol. 2013, pp. 134–, 12 2013.
23. Amandeep Kaur, 2014, "Image Segmentation Using Watershed Transform," 2014.
24. "Region-based CNNs," Available at https://d2l.ai/chapter_computer-vision/rcnn.html(2022/11/23).
25. "14.8. region-based cnns (R-cnns) colab [pytorch] open the notebook in colab colab [mxnet] open the notebook in colab colab [jax] open the notebook in colab colab [tensorflow] open the notebook in colab sagemaker studio lab open the notebook in SageMaker Studio Lab," .
26. Yan Huang, Lingling Wang, Xu Chen, Yi Zhang, Yujie Zhang, and Huajun Chen, 2022, "Comparison of Optical and Radio Frequency Ranging Technologies for Indoor Localization: A Review," *Sensors*, vol. 22, no. 14, pp. 4312, 2022.
27. Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick, 2017, "Mask R-CNN," in **Proceedings of the IEEE International Conference on Computer Vision (ICCV)**, Oct 2017.
28. Yiqing Zhang, Jun Chu, Lu Leng, and Jun Miao, 2020, "Mask-refined R-CNN: A network for refining object details in instance segmentation," .
29. Turki Alanazi-Abdulrahman Senyor Naif Alanezi Nader Al-Qubaydhi, Abdulrahman Alenezi and Saudi Arabia Bandar Alotaibi, 2022, "Detection of Unauthorized Unmanned Aerial Vehicles Using YOLOv5 and Transfer Learning," 2022.
30. Glenn Jocher, "YOLOv5 by Ultralytics," Available at <https://github.com/ultralytics/yolov5> (2020/10/15).
31. Jianyuan Li, Chunna Liu, Xiaochun Lu, and Bilang Wu, 2022, "CME-YOLOv5: An Efficient Object Detection Network for Densely Spaced Fish and Small Targets," *Water*, vol. 14, pp. 2412, 08 2022.
32. Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao, 2020, "Yolov4: Optimal Speed and accuracy of object detection," .
33. "U-Net," Available at <https://blog.paperspace.com/unet-architecture-image-segmentation/> (2022/11/21).
34. Olaf Ronneberger, Philipp Fischer, and Thomas Brox, 2015, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in **Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015**, Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, Eds., Cham, 2015, pp. 234–241, Springer International Publishing.
35. Xiao-Hu Zhou Zeng-Guang Hou Xiao-Liang Xie Chen Wang Yan-Jie Zhou Rui-Qi Li Zhen Li Zhen-Liang Ni, Gui-Bin Bian, 2019, "RAUNet: Residual Attention U-Net for Semantic Segmentation of Cataract Surgical Instruments," *Computer Vision and Pattern Recognition*, 2019.
36. Mubashir Habib, Mutawarra Hussain, Babar Raza, and Saba Khalid, 2020, "Survey of Image Data Augmentation Techniques," *Information*, vol. 11, no. 2, pp. 125, 2020.
37. Pam Scheunemann, 2002, "What is Python?," Available at <https://aws.amazon.com/th/what-is/python/> (2022/9/27).
38. 2021, "OpenCV & Python | Computer vision," Available at <https://www.youtube.com/watch?v=fLDJJXEykvo> (2022/09/25).
39. Stéfan van der Walt1, Johannes L. Schönberger2, Juan Nunez-Iglesias3, François Boulogne4, Joshua D. Warner5, Neil Yager6, Emmanuelle Gouillart7, Tony Yu8, The Scikit-Image Contributors, Behnel S, and et al., 2014, "Scikit-image: Image processing in python," .
40. Streamlit, "Streamlit docs," Available at <https://docs.streamlit.io/> (2022/09/27).

41. Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala, 2019, “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” in **Advances in Neural Information Processing Systems**, H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, Eds. 2019, vol. 32, Curran Associates, Inc.
42. Martín Abadi, 2016, “TensorFlow: Learning Functions at Scale,” **SIGPLAN Not.**, vol. 51, no. 9, pp. 1, sep 2016.
43. Suping Wang, Ligu Zhu, and Mengke Cheng, 2019, “Docker-based Web Server Instructional System,” in **2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS)**, 2019, pp. 285–289.
44. “What is Google Cloud Platform (GCP)?,” Available at <https://acloudguru.com/blog/engineering/what-is-google-cloud-platform-gcp> (2022/11/21).
45. “Cloud Run,” Available at <https://cloud.google.com/run> (2022/11/21).
46. Google Firebase, 2021, “Firebase and Google Cloud,” <https://firebase.google.com/firebase-and-gcp>, Accessed: 2022-03-31.
47. Md Mostafa Sarker, Yasmine Makhlof, Stephanie G. Craig, Matthew P. Humphries, Maurice Loughrey, Jacqueline A. James, Manuel Salto-Tellez, Paul O'Reilly, and Perry Maxwell, 2021, “A means of assessing deep learning-based detection of ICOS protein expression in colon cancer,” **Cancers**, vol. 13, no. 15, pp. 3825, 2021.
48. Wanjun Zhang and Huiqi Li, 2017, “Automated segmentation of overlapped nuclei using concave point detection and segment grouping,” **Pattern Recognition**, vol. 71, pp. 349–360, 2017.
49. Sven P. Voigt, K. Ravikumar, Bikramjit Basu, and Surya R. Kalidindi, 2021, “Automated image processing workflow for morphological analysis of fluorescence microscopy cell images,” **JOM**, 2021.
50. Sonal Kothari, Qaiser Chaudry, and May D. Wang, 2009, “Automated cell counting and cluster segmentation using concavity detection and ellipse fitting techniques,” **2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro**, 2009.
51. Alejandra Suarez-Arnedo, Felipe Torres Figueroa, Camila Clavijo, Pablo Arbeláez, Juan C. Cruz, and Carolina Muñoz-Camargo, 2020, “An image J plugin for the high throughput image analysis of in vitro scratch wound healing assays,” 2020.
52. C. Igathinathane, L.O. Pordesimo, E.P. Columbus, W.D. Batchelor, and S.R. Methuku, 2008, “Shape identification and particles size distribution from basic shape parameters using imagej,” **Computers and Electronics in Agriculture**, vol. 63, no. 2, pp. 168–182, 2008.
53. PerkinElmer, “Validation and automation of phenotypic profiling across multiple cell lines,” Available at <https://www.perkinelmer.com/library/validation-and-automation-of-phenotypic-profiling-across-multiple-cell-lines.html> (2022/10/25).
54. Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick, 2017, “Mask R-CNN,” **arXiv preprint arXiv:1705.02950**, 2017.
55. Kai Zhang, Wangchunshu Zhou, Yuchen Guo, Tong Zhang, and Jianping Shi, 2021, “Revisiting the Importance of Label Smoothing for Model Robustness and Uncertainty,” 2021.
56. Surapong Kanoktipsatharporn, 2020, “Dropout คืออะไร และทำการใช้ dropout ลด overfit ใน Deep Neural Network - regularization EP.2.” .
57. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, 2015, “Faster r-cnn: Towards real-time object detection with region proposal networks,” **IEEE transactions on pattern analysis and machine intelligence**, vol. 39, no. 6, pp. 1137–1149, 2015.
58. Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam, 2019, “Searching for MobileNetV3.” .
59. Jirka Borovec, 2021, “Yolov5/hyp.scratch-low.yaml at master · ultralytics/yolov5.” .

60. Thomas Brox Olaf Ronneberger, Philipp Fischer, 2015, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” **Computer Vision and Pattern Recognition**, 2015.
61. Annika Reinke, Matthias Eisenmann, Minu Dietlinde Tizabi, Carole Sudre, Tim Rädsch, Michela Antonelli, Tal Arbel, Spyridon Bakas, Manuel Jorge Cardoso, Veronika Cheplygina, Keyvan Farahani, Ben Glocker, Doreen Heckmann-Nötzel, Fabian Isensee, Pierre Jannin, Charles Kahn, Jens Kleesiek, Tahsin Kurc, Michal Kozubek, and Alican Noyan, 2021, “Common Limitations of Image Processing Metrics: A Picture Story,” 04 2021.

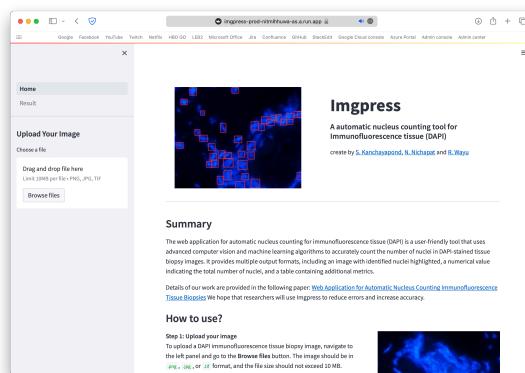
ภาคผนวก A

รายละเอียดวิธีการใช้งานเว็บแอปพลิเคชัน

ขั้นตอนการใช้งานเว็บแอปพลิเคชัน

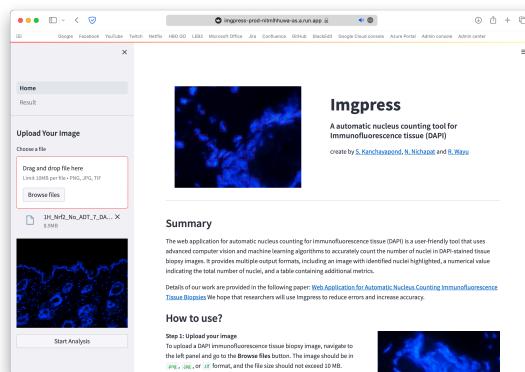
1. เข้าเว็บของเว็บแอปพลิเคชัน <https://automatic-nucleus-counting.web.app/>

2. อัปโหลดภาพที่ต้องการบริโภนซ้ายมือของผู้ใช้งานที่ปุ่ม Browse files



รูปที่ A.1: ส่วนหน้าแรกของเว็บแอปพลิเคชัน

3. กดปุ่ม Start Analysis บริโภนทางด้านล่างของภาพที่อัปโหลด หรือกดปุ่น Result ที่อยู่ด้านล่างของปุ่ม Home



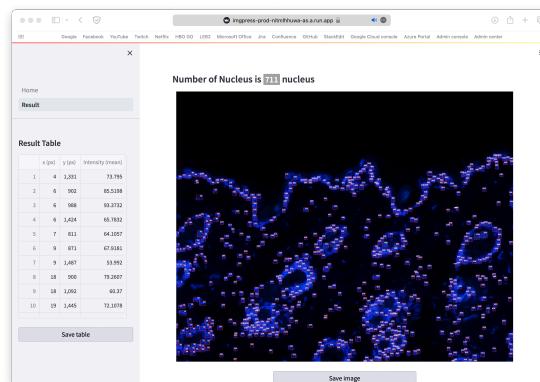
รูปที่ A.2: หน้าเว็บแอปพลิเคชัน

4. เว็บแอปพลิเคชันจะแสดงหน้าผลลัพธ์ โดยจะมีส่วนผลลัพธ์ต่าง ๆ ดังนี้

(a) จำนวนของนิวเคลียสที่แนบได้ทั้งหมดจากแบบจำลอง

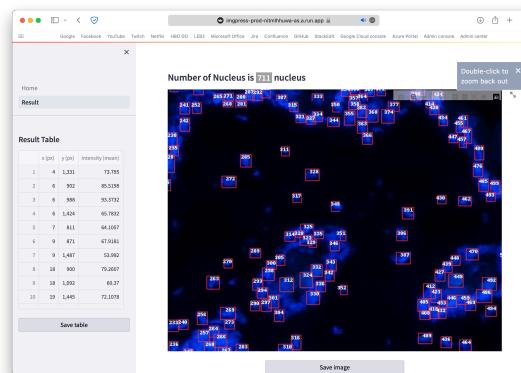
(b) ภาพผลลัพธ์

(c) ตารางผลลัพธ์



รูปที่ A.3: ผลลัพธ์หลังการประมวลผลเสร็จสิ้น

5. ครอบเลือกเพื่อลด/ขยายขนาดภาพ ในกรณีที่ภาพมีขนาดใหญ่



รูปที่ A.4: ผลลัพธ์หลังการขยายภาพ

6. กด Save table บริเวณด้านล่างตารางเพื่อบันทึกตารางเป็นสกุลไฟล์ csv

7. กด Save Image บริเวณด้านล่างภาพผลลัพธ์เพื่อบันทึกตารางเป็นสกุลไฟล์ png

ภาคผนวก B

แบบประเมินความพึงพอใจ

ชุดคำถามสำหรับบุคคลทั่วไป

แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันของบุคคลทั่วไป

	มากที่สุด	มาก	ปานกลาง	น้อย	น้อยที่สุด
1. ด้านการออกแบบและการจัดการรูปแบบ					
• ความเหมาะสมในการเลือกใช้รูปแบบ ขนาด สีตัวอักษรบนเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมในการเลือกใช้ข้อความที่อธิบายการใช้เว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมในการใช้สัญลักษณ์หรือรูปภาพในการสื่อความหมาย	<input type="checkbox"/>				
• ความสวยงามและความทันสมัยของเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความน่าสนใจของหน้าเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมของสีพื้นหลังที่ง่ายต่อการใช้เว็บแอปพลิเคชัน	<input type="checkbox"/>				
2. ด้านการแสดงผลลัพธ์การนับจำนวนนิเวศสีียส					
• รูปภาพผลลัพธ์และตารางมีความชัดเจนและเข้าใจง่าย	<input type="checkbox"/>				
3. ด้านการใช้งานได้ตามฟีก์ชันการทำงาน					
• การใช้คำสั่งในส่วนการอัปโหลดรูปภาพ ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• การใช้คำสั่งในการบันทึกผลลัพธ์ที่ได้ลงอุปกรณ์ของตนเอง ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• การใช้คำสั่งการย่อ/ขยายรูปผลลัพธ์ ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• ความเร็วในการประมวลผลของเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ประโยชน์ในการใช้งานเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ภาพรวมของเว็บแอปพลิเคชันตรงต่อความต้องการของผู้ใช้	<input type="checkbox"/>				

รูปที่ B.1: แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันบุคคลทั่วไป

ชุดคำถามสำหรับนักพยาธิวิทยา

แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันของนักพยาธิวิทยา

	มากที่สุด	มาก	ปานกลาง	น้อย	น้อยที่สุด
1. ด้านการออกแบบและการจัดกรรูปแบบ					
• ความเหมาะสมในการเลือกใช้รูปแบบ ขนาด สีตัวอักษรบนเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมในการเลือกใช้ข้อความเพื่อเรียบง่ายการใช้เว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมในการใช้สัญลักษณ์หรือรูปภาพในการสื่อความหมาย	<input type="checkbox"/>				
• ความสวยงามและความทันสมัยของเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความน่าสนใจของหน้าเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ความเหมาะสมของสีพื้นหลังที่ง่ายต่อการใช้เว็บแอปพลิเคชัน	<input type="checkbox"/>				
2. ด้านการแสดงผลลัพธ์การบันทึกงานนิเวศลีส					
• ผลลัพธ์ที่ได้จากการทำงานมีความชัดเจนและตรงกับความต้องการของผู้ใช้	<input type="checkbox"/>				
• ผลลัพธ์มีความถูกต้อง	<input type="checkbox"/>				
• ขนาดของรูปผลลัพธ์มีขนาดที่ความเหมาะสมต่อการมองเห็น	<input type="checkbox"/>				
3. ด้านการใช้งานได้ตามที่เก็บข้อมูลการทำงาน					
• การใช้คำสั่งในส่วนการบันทึกผลลัพธ์ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• การใช้คำสั่งในการบันทึกผลลัพธ์ที่ได้ลงทะเบียนของตนเอง ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• การใช้คำสั่งการย่อ/ขยายรูปผลลัพธ์ ที่ง่ายต่อการใช้งาน	<input type="checkbox"/>				
• ความเร็วในการประมวลผลของเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ประโยชน์ในการใช้งานเว็บแอปพลิเคชัน	<input type="checkbox"/>				
• ภาพรวมของเว็บแอปพลิเคชันตรงต่อความต้องการของผู้ใช้	<input type="checkbox"/>				

รูปที่ B.2: แบบประเมินความพึงพอใจในการใช้เว็บแอปพลิเคชันของนักพยาธิวิทยา