



# การปรับปรุงประสิทธิภาพแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันในการจำแนก โรคเนื้องอกในสมองจากภาพ MRI ด้วยการปรับแต่งไฮเปอร์พารามิเตอร์

## Improving Performance of Convolutional Neural Network Models for Brain Tumor Classification from MRI Images through Hyperparameter Tuning

พงษ์ศธร เชิดสม (Pongsathorn Chedsom)\*

Received: June 3, 2024  
Revised: August 9, 2024  
Accepted: August 14, 2024

\* ผู้นิพนธ์ประสานงาน: พงษ์ศธร เชิดสม (Pongsathorn Chedsom) อีเมล: pongche@kku.ac.th

DOI:10.14416/j.it.2025.v1.008

### บทคัดย่อ

เนื้องอกในสมองเป็นโรคร้ายแรงที่พบได้บ่อย การวินิจฉัยและรักษาได้อย่างทันท่วงทีมีความสำคัญต่อการลดอัตราการเสียชีวิตและเพิ่มโอกาสในการรอดชีวิตของผู้ป่วย งานวิจัยนี้นำเสนอวิธีการจำแนกโรคเนื้องอกในสมองโดยใช้โครงข่ายประสาทเทียมเชิงลึก ซึ่งใช้ชุดข้อมูล Br35H จำนวน 3,000 ภาพ แบ่งภาพที่เป็นโรคและไม่เป็นโรคอย่างละ 1,500 ภาพ สำหรับใช้ในการเรียนรู้ของแบบจำลอง แบ่งข้อมูลออกเป็นชุดเรียนรู้ 70%: ชุดตรวจสอบ 20%: ชุดทดสอบ 10% และใช้ชุดข้อมูล Brain MRI Images จำนวน 253 ภาพ และ Brian Tumor จำนวน 4,600 ภาพ สำหรับการทดสอบความถูกต้องของแบบจำลองที่พัฒนาขึ้น แบ่งการทดลองออกเป็น 3 กลุ่ม กลุ่มที่ 1 นำสถาปัตยกรรมที่ได้รับความนิยมจากบทความต่าง ๆ โดยไม่มีการปรับแต่งและกำหนดค่าไฮเปอร์พารามิเตอร์พื้นฐานสำหรับการทดลอง ได้แก่ Optimizer = Adam Learning rate = 0.001 batch size = 32 นำ 5 สถาปัตยกรรมที่ได้ค่าความถูกต้องมากที่สุดไปใช้ในการทดลองถัดไปพบว่าสถาปัตยกรรม DenseNet201, Xception, InceptionResNetV2, MobileNetV2, NasNetMobile มีค่าความถูกต้องร้อยละ (Accuracy) 98.00, 98.00, 98.00, 97.67, 97.33 และค่าความสูญเสีย (Loss) 0.08, 0.06, 0.08, 0.11, และ 0.11 ตามลำดับ กลุ่มที่ 2 ปรับแต่งส่วนของไฮเปอร์พารามิเตอร์จำนวน 3 ตัว ได้แก่ Batch size, Optimizer, Learning rate โดยปรับแต่งไฮเปอร์พารามิเตอร์แต่ละตัว พบว่าสถาปัตยกรรม Xception (Batch size 16) ได้ค่าความถูกต้องมากที่สุดร้อยละ 98.34 และค่าความสูญเสีย 0.08 สถาปัตยกรรม

InceptionResNetV2 (Learning rate 0.01) ได้ค่าความถูกต้องมากที่สุดร้อยละ 98.67 และค่าความสูญเสีย 0.13 สถาปัตยกรรม Xception (Optimizer Adamax) ได้ค่าความถูกต้องมากที่สุดร้อยละ 98.67 และค่าความสูญเสีย 0.06 กลุ่มที่ 3 ปรับแต่งส่วนของไฮเปอร์พารามิเตอร์จำนวน 3 ตัว ได้แก่ Batch size, Optimizer, Learning rate โดยการจับคู่ไฮเปอร์พารามิเตอร์สองตัวเข้าด้วยกัน พบว่าสถาปัตยกรรม InceptionResNetV2 (Batch size 32, Optimizer Adam, Learning rate 0.01) ได้ค่าความถูกต้องสูงที่สุดร้อยละ 100.00 และค่าความสูญเสีย 0.001

**คำสำคัญ:** การจำแนกโรคเนื้องอกในสมอง โครงข่ายประสาทเทียมแบบคอนโวลูชัน ปรับแต่งไฮเปอร์พารามิเตอร์

### Abstract

Brain tumors are a common and severe disease, requiring timely diagnosis and treatment to reduce mortality rates and improve patient survival chances. This research presents a method for classifying brain tumors using deep neural networks. The Br35H dataset, consisting of 3,000 images (1,500 diseased and 1,500 non-diseased), was used for model training, divided into 70% training, 20% validation, and 10% testing sets. Additionally, 253 Brain MRI Images and 4,600 Brian Tumor images were used to test the model's accuracy. The experiments were divided into three groups: Group 1 employed popular architectures from existing literature without modification, using default hyperparameter settings.

\* งานวิชาการและวิจัย คณะสถาปัตยกรรมศาสตร์ มหาวิทยาลัยขอนแก่น

\* Academic Affairs and Research, Faculty of Architecture, Khon Kaen University.

Specifically, the configurations included the Adam optimizer with a learning rate of 0.001 and a batch size of 32. The top five architectures with the highest accuracy were selected for further experiments. The architectures DenseNet201, Xception, InceptionResNetV2, MobileNetV2, and NasNetMobile accuracy of 98.00%, 98.00%, 98.00%, 97.67%, and 97.33% and loss of 0.08, 0.06, 0.08, 0.11, and 0.11 respectively. Group 2: Tuning three hyperparameters separately: Batch size, Optimizer, and Learning rate. The best results were Xception (Batch size 16) accuracy of 98.34% and loss of 0.08, InceptionResNetV2 (Learning rate 0.01) accuracy of 98.67% and loss of 0.13, and Xception (Optimizer Adamax) accuracy of 98.67% and loss of 0.06. Group 3: Tuning pairs of hyperparameters: The best architecture was InceptionResNetV2 (Batch size 32, Optimizer Adam, Learning rate 0.01) accuracy of 100.00% and loss of 0.001.

**Keywords:** Brain Tumor Classification, Convolutional Neural Network, Hyperparameter Tuning.

## 1. บทนำ

เนื้องอกในสมองเป็นโรคร้ายแรงที่พบได้บ่อยการวินิจฉัยและรักษาโรคเนื้องอกในสมองได้อย่างทันท่วงทีมีความสำคัญต่อการลดอัตราการเสียชีวิตและเพิ่มโอกาสในการรอดชีวิตของผู้ป่วย ปัจจุบันวิธีการวินิจฉัยโรคเนื้องอกในสมองที่ใช้กันทั่วไปคือการตรวจเอกซเรย์คอมพิวเตอร์ (Computerized Tomography: CT) และการตรวจภาพด้วยคลื่นแม่เหล็กไฟฟ้า (Magnetic Resonance Imaging: MRI) ซึ่งวิธีการเหล่านี้ยังมีข้อจำกัด เช่น ความละเอียดของภาพที่จำกัดและความยากลำบากในการวิเคราะห์และคัดแยกเนื้องอกชนิดต่าง ๆ ในช่วงหลายปีที่ผ่านมาการประยุกต์ใช้เทคโนโลยีทางปัญญาประดิษฐ์ (Artificial Intelligence: AI) โดยเฉพาะการใช้โครงข่ายประสาทเทียมเชิงลึก (Convolutional Neural Networks: CNN) มาใช้ในการวิเคราะห์ภาพทางการแพทย์ ซึ่งโครงข่ายประสาทเทียมเชิงลึกเป็นหนึ่งในกลุ่มการเรียนรู้ของเครื่อง (Machine Learning: ML) ที่มีประสิทธิภาพสูงในการวิเคราะห์และจำแนกรูปแบบของภาพจึงทำให้มีความเหมาะสมและมีศักยภาพ ความสามารถในการจำแนกการวินิจฉัยโรคจากภาพทางการแพทย์รวมถึงโรคเนื้องอก

ในสมองนี้ด้วย การวินิจฉัยโรคเนื้องอกในสมองด้วยวิธีดั้งเดิมมีความล่าช้าส่งผลให้ผู้ป่วยเสียโอกาสในการรักษาและอาจจะเสียชีวิตได้ การพัฒนาวิธีการวินิจฉัยโรคเนื้องอกในสมองด้วยโครงข่ายประสาทเทียมเชิงลึกจึงมีความสำคัญอย่างยิ่ง เพราะอาจช่วยให้แพทย์สามารถวินิจฉัยโรคได้อย่างรวดเร็วและแม่นยำมากขึ้นหรือช่วยให้แพทย์รวมทั้งผู้ที่เกี่ยวข้องสามารถวิเคราะห์ภาพเบื้องต้นก่อนส่งต่อเพื่อรับการรักษา ซึ่งจะส่งผลดีต่อผู้ป่วยในแง่ของโอกาสในการรักษาและคุณภาพชีวิต

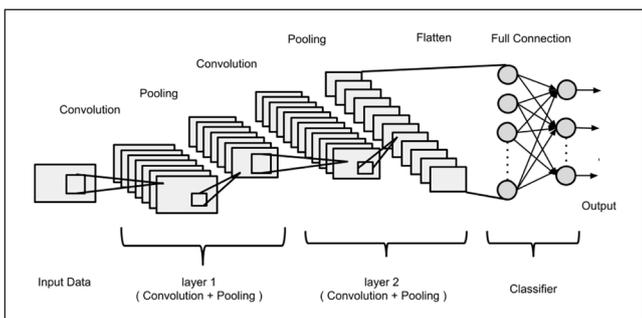
บทความนี้จึงนำเสนอแบบจำลองสำหรับการตรวจหาโรคเนื้องอกในสมองโดยใช้สถาปัตยกรรมที่ได้รับความนิยมในปัจจุบันนำมาปรับปรุงประสิทธิภาพด้วยการปรับแต่งไฮเปอร์พารามิเตอร์ โดยแบบจำลองที่พัฒนาขึ้นจะจำแนกภาพถ่ายจากคลื่นแม่เหล็กไฟฟ้า (MRI) ซึ่งเป็นเทคนิคที่นิยมใช้ในการตรวจวินิจฉัยโรคในสมอง ชุดข้อมูลที่น่าสนใจในการเรียนรู้และทดสอบแบบจำลองจำนวน 3 ชุดข้อมูลได้แก่ชุดข้อมูล Br35H - Brain Tumor Detection 2020 [1] จำนวน 3,000 ภาพ มาใช้ในการเรียนรู้และทดสอบแบบจำลอง ดังแสดงในภาพที่ 3 ตัวอย่างภาพจากชุดข้อมูล Br35H และการแบ่งชุดข้อมูลสำหรับการเรียนรู้และทดสอบแบบจำลองดังแสดงในภาพที่ 5 ชุดข้อมูล Brain MRI Images for Brain Tumor Detection [2] จำนวน 253 ภาพ และชุดข้อมูล Brian Tumor Dataset [3] จำนวน 4,600 ภาพ นำมาใช้ในการทดสอบประสิทธิภาพของแบบจำลอง แบ่งการทดลองออกเป็น 3 กลุ่ม กลุ่มที่ 1 นำสถาปัตยกรรมที่ได้รับความนิยมจากบทความต่าง ๆ มาจำแนกโรคโดยไม่มีการปรับแต่งและกำหนดไฮเปอร์พารามิเตอร์พื้นฐานสำหรับการทดลองไว้ นำสถาปัตยกรรมที่ได้ค่าความถูกต้องมากที่สุด 5 สถาปัตยกรรมมาใช้ในการทดลองถัดไปกลุ่มที่ 2 นำสถาปัตยกรรม 5 สถาปัตยกรรมมาปรับแต่งส่วนของไฮเปอร์พารามิเตอร์จำนวน 3 ตัว ซึ่งส่งผลกระทบต่อค่าความถูกต้องของการเรียนรู้ของแบบจำลองมากที่สุดในระดับต้น ๆ และค่าในแต่ละไฮเปอร์พารามิเตอร์นำมาจากบทความที่ได้รับความนิยมใช้ในการทดลอง อีกทั้งยังสามารถปรับแต่งไฮเปอร์พารามิเตอร์ได้อย่างอิสระเพื่อให้สอดคล้องกับลักษณะของข้อมูลที่น่าสนใจในแต่ละรูปแบบ ได้แก่ Batch size, Optimizer, Learning rate โดยปรับแต่งเพียงไฮเปอร์พารามิเตอร์แต่ละตัว กลุ่มที่ 3 นำสถาปัตยกรรม 5 สถาปัตยกรรมมาปรับแต่งส่วนของไฮเปอร์พารามิเตอร์

จำนวน 3 ตัว ได้แก่ Batch size, Optimizer, Learning rate โดยการจับคู่ไฮเปอร์พารามิเตอร์สองตัวเข้าด้วยกัน ดังแสดงสรุปการทดลองในภาพที่ 5 และกระบวนการทดลองการพัฒนาแบบจำลองในภาพที่ 6

## 2. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

### 2.1 ทฤษฎี

โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional neural networks: CNN) โครงข่ายนี้ได้รับความนิยมเป็นอย่างมากสำหรับการนำมาใช้ในการวิเคราะห์ภาพได้อย่างมีประสิทธิภาพซึ่งมีความสามารถในการสกัดคุณลักษณะเด่นของภาพ (Feature Extraction) โดยทั่วไปโครงสร้างประกอบด้วย 3 ชั้นหลัก ได้แก่ 1 ชั้นคอนโวลูชัน (Convolutional Layer) เป็นชั้นสำหรับการสกัดคุณลักษณะเด่นของข้อมูลโดยจะใช้ตัวกรองข้อมูลที่เรียกว่าฟิลเตอร์ (Filter) หรือคอร์เนล (Kernels) ชั้นที่ 2 ชั้นพูลลิ่ง (Pooling Layer) เป็นชั้นสำหรับการลดขนาดของข้อมูลโดยมีเทคนิคต่าง ๆ ได้แก่ การเลือกค่าเฉลี่ย (Average Pooling) การเลือกค่ามากที่สุด (Max pooling) เป็นเทคนิคที่ได้รับความนิยมมากที่สุด การเลือกน้อยที่สุด (Min pooling) และการเลือกผลรวม (Sum pooling) ชั้นสุดท้ายคือชั้นเชื่อมโยงแบบสมบูรณ์ (Fully Connected Layer) เป็นชั้นสำหรับการรวมข้อมูลทุก ๆ โหนดจากชั้นที่ผ่านมาก่อนหน้าให้มีลักษณะ 1 มิติ และนำไปใช้ในการจำแนกประเภท (Classification) ดังแสดงในภาพที่ 1 โครงสร้างประสาทเทียมแบบคอนโวลูชัน [4]

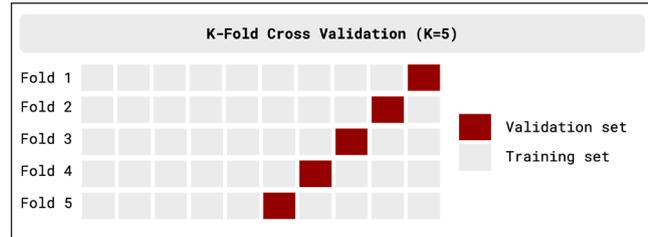


ภาพที่ 1 โครงสร้างประสาทเทียมแบบคอนโวลูชัน [4]

### การแบ่งข้อมูล K-Fold Cross Validation

เป็นเทคนิคการประเมินประสิทธิภาพของแบบจำลองที่ได้รับความนิยมในปัจจุบันจากอีกหลายเทคนิค นอกจากประสิทธิภาพของแบบจำลองแล้วยังเป็นเทคนิคที่ทำให้แบบจำลองที่พัฒนาขึ้นมีความน่าเชื่อถือและป้องกันการเกิด Bias

ในการพัฒนาอีกด้วย โดยวิธีนี้จะแบ่งข้อมูลออกเป็น K ชุด (folds) เท่า ๆ กัน จำนวน K ที่เหมาะสมสำหรับ K-Fold Cross Validation ขึ้นอยู่กับขนาดของข้อมูล และประเภทของแบบจำลอง โดยทั่วไปนิยมใช้ K = 5 หรือ K = 10 แต่ละชุดจะถูกแบ่งเป็นชุดประเมิน (Validation Set) หนึ่งส่วนและแบ่งชุดข้อมูลที่เหลือเป็นชุดฝึกอบรม (Training Set) ตามจำนวนที่กำหนดไว้ เช่น Training Set 70% Validation Set 30% ดังภาพที่ 2 การแบ่งข้อมูลด้วยเทคนิค K-Fold Cross Validation (K = 5)



ภาพที่ 2 การแบ่งข้อมูลด้วยเทคนิค K-Fold Cross Validation (K=5)

### การถ่ายโอนความรู้ (Transfer Learning)

เป็นเทคนิคที่ได้รับความนิยมและน่าสนใจในการแพทย์เนื่องจากเป็นเทคนิคที่นำค่าน้ำหนัก (Weight) ที่ผ่านการเรียนรู้ล่วงหน้ามาแล้วจากชุดข้อมูลขนาดใหญ่ เช่น ImageNet ซึ่งทำให้ลดระยะเวลาของการเรียนรู้ของแบบจำลอง นอกจากนั้นยังทำให้มีประสิทธิภาพเพิ่มมากขึ้นเป็นอย่างมาก [5] อีกทั้งยังมีความเหมาะสมกับชุดข้อมูลที่นำมาใช้ในการเรียนรู้ที่มีจำนวนไม่มากนัก

### 2.2 งานวิจัยที่เกี่ยวข้อง

การวินิจฉัยโรคเนื้องอกในสมองของ [5] ได้ใช้เทคนิคการถ่ายโอนความรู้ (Transfer Learning) จากแบบจำลอง ResNet152 VGG19 DenseNet169 และ MobileNetV3 และในการทดลองได้ใช้งาน Optimizer ได้แก่ Adaptive Moment Estimation (Adam) และกำหนด Learning rate 0.001 พบว่าแบบจำลอง MobileNetV3 มีค่าความถูกต้องมากที่สุดร้อยละ 99.75

การวินิจฉัยโรคเนื้องอกในสมองของ [6] จากการแบ่งส่วนหรือระบุตำแหน่งของโรค (Segmentation) และการจำแนกโรคโดยใช้ You Only Look Once version 5 และ 7 จากภาพ MRI ประกอบไปด้วย 3 ประเภท ได้แก่ เนื้องอกเมนิจีโอมา (Meningiomas) เนื้องอกไกลิโอมา (Gliomas) และเนื้องอกต่อมใต้สมอง (Pituitary tumors) จำนวน 827 ภาพ พบว่า Yolo V5 ได้ค่าความถูกต้อง ร้อยละ 90.60 และ Yolo V7 ได้ค่าความถูกต้อง ร้อยละ 93.60

การวินิจฉัยโรคเนื้องอกในสมองของ [7] โดยการปรับปรุงไฮเปอร์พารามิเตอร์ของแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชัน ได้ทำการปรับปรุงไฮเปอร์พารามิเตอร์ Dropout ตั้งแต่ 0.1 - 0.20 (เพิ่มขึ้นทีละหนึ่ง) Optimizer ได้แก่ Stochastic Gradient Descent (SGD) and Adaptive Moment Estimation (Adam) batch size ได้แก่ 64 - 256 โดยทดสอบกับ 2 ชุดข้อมูล จำนวน 7,023 ภาพ (จำนวน 4 คลาส) ได้ค่าความถูกต้องร้อยละ 96.00 และ 253 ภาพ (จำนวน 2 คลาส) ได้ค่าความถูกต้องร้อยละ 88.00 และจากการทดลอง Optimizer Adaptive Moment Estimation (Adam) ส่งผลต่อความถูกต้องของแบบจำลองมากกว่า Stochastic Gradient Descent (SGD)

ในการตรวจหาโรคเนื้องอกในสมอง [8] โดยการปรับแต่ง You Only Look Once version 7 (YOLOv7) ได้แก่การเพิ่มชั้น Convolutional Block Attention Module (CBAM) สำหรับการคัดแยกคุณสมบัติของข้อมูล และเพิ่มชั้น Spatial Pyramid Pooling Fast+ (SPPF+) และชั้น Bi-Directional Feature Pyramid Network (BiFPN) โดยทดสอบชุดข้อมูลจำนวน 7,023 ภาพ (จำนวน 4 คลาส) ได้ค่าความถูกต้องร้อยละ 99.50

การตรวจจับและการจำแนกเนื้องอกในสมองโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันของ [9] โดยทดสอบชุดข้อมูลจำนวน 7,023 ภาพ (จำนวน 4 คลาส) ได้ค่าความถูกต้องร้อยละ 94.82 อีกทั้งยังเสนอแนะใช้เทคนิคการถ่ายโอนความรู้ (Transfer Learning) จากสถาปัตยกรรมต่าง ๆ ได้แก่ VGG, ResNet50 ในการนำมาใช้ในการจำแนกโรคเนื้องอกในสมอง

การจำแนกโรคเนื้องอกของ [10] ได้ใช้วิธีการปรับแต่งไฮเปอร์พารามิเตอร์ ได้แก่ Dropout คือ 0.2, 0.5 Optimizer คือ Adam, Adamax, RMSprop, SGD และได้ทำการทดลองใน 3 รูปแบบ ได้แก่ รูปแบบที่ 1 โดยใช้โครงสร้างแบบจำลองที่พัฒนาขึ้นเอง รูปแบบที่ 2 ใช้โครงสร้างจากรูปแบบที่ 1 นำมาปรับแต่งโดยใช้ไฮเปอร์พารามิเตอร์ที่กำหนดไว้ และรูปแบบที่ 3 คือการนำโครงสร้างจากรูปแบบที่ 1 มาใช้ร่วมกับไฮเปอร์พารามิเตอร์ที่ได้ผลดีที่สุดในการทดลองในรูปแบบที่ 2 ทดสอบชุดข้อมูลจำนวน 3,264 ภาพ (จำนวน 4 คลาส) พบว่าการทดลองในรูปแบบที่ 3 มีประสิทธิภาพมากที่สุด ได้ค่าความถูกต้องมากที่สุดร้อยละ 96.00

การคัดแยกโรคเนื้องอกในสมองของ [11] โดยใช้แบบจำลอง Mobilenetv2 และทำการปรับแต่งโดยใช้เทคนิค fox optimization algorithm (MN-V2/CFO) ในการค้นหา

ไฮเปอร์พารามิเตอร์ที่เหมาะสม และเกิดประสิทธิภาพกับแบบจำลองมากที่สุด ทดสอบชุดข้อมูลจำนวน 3,064 ภาพ (จำนวน 3 คลาส) ซึ่งได้รับค่าความถูกต้องมากที่สุดคือร้อยละ 97.32

การตรวจหาเนื้องอกในสมองของ [12] โดยใช้สถาปัตยกรรม VGG, ResNet, DenseNet และ SqueezeNet มาใช้ในการค้นหาคุณสมบัติเด่นของข้อมูลและใช้การเรียนรู้ของเครื่อง ได้แก่ Support Vector Machines, K-Nearest Neighbors, Naive Bayes, Decision Tree และ Linear Regression Analysis ในการคัดแยกโรค ทดสอบชุดข้อมูลจำนวน 7,022 ภาพ พบว่าสถาปัตยกรรม ResNet ได้ผลดีมากที่สุด ซึ่งได้รับค่าความถูกต้องมากที่สุดคือร้อยละ 100

การตรวจหาและจำแนกเนื้องอกในสมองของ [13] โดยแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชัน ซึ่งใช้วิธีการถ่ายโอนความรู้จากแบบจำลองตระกูล EfficientNets ได้แก่ EfficientNetB0 - EfficientNetB4 จำนวน 5 แบบจำลอง ทดสอบร่วมกับชุดข้อมูล 3,064 ภาพ (จำนวน 3 คลาส) กำหนดไฮเปอร์พารามิเตอร์ได้แก่ optimizer คือ Adam learning rate คือ 0.001 dropout คือ 0.2 batch size คือ 32 พบว่ามีค่าความถูกต้องร้อยละ (Test accuracy) 97.43, 98.66, 99.06, 98.08, 97.90 ตามลำดับ (B0-B4)

การตรวจหาและจำแนกเนื้องอกในสมอง โดย [14] พัฒนาแบบจำลอง TumorDetNet มีชั้นของ Convolution สำหรับการสกัดคุณลักษณะของข้อมูล (feature extraction) จำนวน 48 ชั้น และมีการกำหนดไฮเปอร์พารามิเตอร์ในการทดลอง ได้แก่ Optimization คือ SGDM learning rate คือ 0.01 Epoch คือ 22 Dropout คือ 0.4 Batch Size สำหรับเรียนรู้ของแบบจำลอง คือ 80 Batch Size สำหรับทดสอบแบบจำลอง คือ 20 นำมาทดสอบกับชุดข้อมูล BT-D-MRI ได้ค่าความถูกต้องร้อยละ 99.83 และ BMI-BTD ได้ค่าความถูกต้องร้อยละ 96.08

การตรวจหาและจำแนกเนื้องอกในสมองของ [15] โดยแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชัน ซึ่งใช้วิธีการถ่ายโอนความรู้จากแบบจำลอง ResNet50 ร่วมกับ U-Net ซึ่งนำมาทดสอบกับชุดข้อมูล TCGA-LGG และ TCIA พบว่าแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันที่ไม่มีการปรับแต่ง มีค่าความถูกต้องร้อยละ 92.00 แบบจำลอง ResNet50 มีค่าความถูกต้องร้อยละ 94.00 และแบบจำลอง U-Net ร่วมกับ ResNet50 มีค่าความถูกต้องร้อยละ 95.00

การวินิจฉัยโรคเนื้องอกในสมองโดยแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันของ [16] ซึ่งใช้วิธีการถ่ายโอนความรู้จากสถาปัตยกรรม EfficientNetv2s ร่วมกับการปรับแต่ง optimizer ได้แก่ Ranger,SGD,RMSprop และ Adam จากการทดลองพบว่า EfficientNetv2 + Ranger มีความถูกต้องที่ดีที่สุดร้อยละ 99.85 และยิ่งพบอีกว่าการใช้งาน optimizer Ranger มีผลต่อความถูกต้องอย่างมีนัยสำคัญเมื่อเทียบกับแบบจำลองอื่น ๆ

การวินิจฉัยและการจำแนกโรคเนื้องอกในสมองใน [17] ได้พัฒนาแบบจำลอง 3 แบบจำลอง ในการจำแนกโรคเนื้องอกในสมอง ซึ่งในแบบจำลองที่ 1 เป็นการจำแนกโรคเนื้องอกในสมอง (เป็นหรือไม่เป็นโรค) ทดสอบร่วมกับชุดข้อมูลจำนวน 2,990 ภาพ โดยใช้แบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชัน และปรับแต่งไฮเปอร์พารามิเตอร์ที่สำคัญเช่น Batch Size ได้แก่ 4, 8, 16, 32, 64 Learning Rate ได้แก่ 0.0001, 0.0005, 0.001, 0.005 L2 Regularization ได้แก่ 0.0001, 0.0005, 0.001, 0.005 Momentum ได้แก่ 0.80, 0.85, 0.9, 0.95 ได้ด้วยความถูกต้องร้อยละ 99.33

การประเมินและเปรียบเทียบการตรวจหาโรคเนื้องอกในสมองโดยใช้สถาปัตยกรรม Inception-V3 ของ [18] มาใช้ในการพัฒนาแบบจำลอง และใช้ไฮเปอร์พารามิเตอร์ Optimizer คือ Adaptive Moment Estimation (ADAM) Epochs คือ 20 Learning Rate คือ 0.0001 Batch Size คือ 30 โดยใช้ชุดข้อมูลทั้งหมด 3,064 ภาพ แบ่งออกเป็น 4 คลาส ผลการทดลองพบว่าแบบจำลองจากสถาปัตยกรรม Inception-V3 ได้ค่าความถูกต้องร้อยละ 99.02 ซึ่งมากกว่า VGG16 และ ResNet-50

การจำแนกและการตรวจจับโรคเนื้องอกในสมองของ [19] โดยใช้เทคนิคการถ่ายโอนความรู้ (Transfer learning) และการเพิ่มข้อมูล (Data augmentation) ได้แก่สถาปัตยกรรม InceptionResNetV2, InceptionV3, DenseNet121, Xception, ResNet50V2, VGG19, and EfficientNetB7 สำหรับการจำแนกโรค และกำหนดไฮเปอร์พารามิเตอร์สำหรับการทดลอง ได้แก่ Optimizer คือ Adadelta Epochs คือ 50 Batch size คือ 4 ผลการทดลองพบว่าการใช้เทคนิค Transfer learning + Data augmentation มีค่าความถูกต้องร้อยละ 92.16 เทคนิค Data augmentation มีค่าความถูกต้องร้อยละ 86.27 เทคนิค Transfer learning มีค่าความถูกต้องร้อยละ 92.16 ไม่ใช้เทคนิคใด ๆ มีค่าความถูกต้องร้อยละ 82.35

[20] ได้พัฒนาแบบจำลองโดยใช้โครงข่ายประสาทเทียม

แบบคอนโวลูชัน จำนวน 4 แบบจำลองมีความแตกต่างกันชั้นของการสกัดคุณลักษณะของข้อมูล ในการตรวจจับและการจำแนกโรคเนื้องอกในสมอง ใช้ชุดข้อมูล 2 ชุดข้อมูล ชุดข้อมูลที่ 1 มีจำนวนข้อมูล 3,264 ภาพ ชุดข้อมูลที่ 2 มีจำนวนข้อมูล 7,023 ภาพ แบ่งข้อมูลออกเป็น 4 คลาส และได้กำหนดไฮเปอร์พารามิเตอร์ ได้แก่ Dropout-rate คือ 0.0, 0.2, 0.3, 0.4, 0.5, 0.6 Optimizer คือ Adam, SGD with Nesterov Learning rate คือ 0.001, 0.0001 ผลการทดลองพบว่าแบบจำลองที่ 3 Dropout คือ 0.4 Optimizer คือ SGD with Nesterov Learning rate คือ 0.0001 มีค่าความถูกต้องร้อยละ 98.01 ซึ่งสูงที่สุดเมื่อเทียบกับทั้ง 4 แบบจำลอง

การคัดแยกโรคเนื้องอกในสมองโดยใช้แบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันของ [21] ร่วมกับการกำหนด SoftMax ในชั้นของ Fully Connected เพียงอย่างเดียวพบว่ามีความแม่นยำร้อยละ 98.00 และนอกจากนั้นยังมีการใช้ร่วมกับตัวคัดแยก ได้แก่ Radial Basis Function (RBF) and Decision Tree (DT) พบว่ามีค่าความแม่นยำร้อยละ 98.24 และ 95.64 ตามลำดับ และยังได้นำเสนอวิธีการในการคัดแยก ซึ่งได้ค่าความแม่นยำร้อยละ 99.52 ซึ่งในการทดลองได้ใช้ชุดข้อมูลจำนวน 3,784 ภาพ

### 3. วิธีดำเนินการวิจัย

#### 3.1 เครื่องมือ

งานวิจัยนี้ใช้ไลบรารี Keras เวอร์ชัน 3 และ Tensorflow ในการพัฒนาแบบจำลองด้วยภาษา Python ซึ่งคอมพิวเตอร์สเปค CPU Intel Core i9-13900KS หน่วยความจำหลัก 32 GB หน่วยประมวลผลกราฟฟิก Nvidia RTX 4070 SUPPER (12GB) ระบบปฏิบัติการ Ubuntu เวอร์ชัน 22.04 LTS

#### 3.2 ชุดข้อมูลและการเตรียมข้อมูล

3.2.1 ชุดข้อมูล Br35H - Brain Tumor Detection 2020 [1] สำหรับการเรียนรู้ของแบบจำลองและใช้สำหรับการทดลองหลักในการวิจัยนี้ ประกอบด้วยข้อมูลจำนวน 3,000 ภาพ แบ่งออกเป็นภาพที่บ่งบอกว่าเป็นโรคเนื้องอกในสมองจำนวน 1,500 ภาพ และไม่เป็นโรคจำนวน 1,500 ภาพ ดังแสดงในตารางที่ 1 โดยแบ่งข้อมูลออกเป็นชุดสำหรับเรียนรู้ (Training set) 70% จำนวนภาพชุดสำหรับการประเมิน (Validation set) 20% จำนวนภาพและชุดข้อมูลสำหรับการทดสอบ (Test set) 10% จำนวนภาพและนำเทคนิคการแบ่งข้อมูลโดยใช้ K-Fold



Cross Validation (K=5) มาใช้ในการทดสอบประสิทธิภาพแบบจำลองและเพื่อให้ข้อมูลในพัฒนาแบบจำลองกระจายตัวอย่างสม่ำเสมอแสดงในภาพที่ 4 ภาพการแบ่งข้อมูลของชุดข้อมูลในการพัฒนาแบบจำลอง

3.2.2 ชุดข้อมูล Brain MRI Images for Brain Tumor Detection [2] สำหรับเป็นชุดข้อมูลที่ใช้ในการทดสอบแบบจำลอง (Unseen Data) ประกอบด้วยข้อมูลจำนวน 253 ภาพ แบ่งออกเป็นภาพที่บ่งบอกว่าเป็นโรคเนื้องอกในสมองจำนวน 155 ภาพ และไม่เป็นโรคจำนวน 98 ภาพ ดังแสดงในตารางที่ 1

3.2.3 ชุดข้อมูล Brian Tumor Dataset [3] สำหรับเป็นชุดข้อมูลที่ใช้ในการทดสอบแบบจำลอง (Unseen Data) ประกอบด้วยข้อมูลจำนวน 4,600 ภาพ แบ่งออกเป็นภาพที่บ่งบอกว่าเป็นโรคเนื้องอกในสมองจำนวน 2,513 ภาพ และไม่เป็นโรคจำนวน 2,087 ภาพ ดังแสดงในตารางที่ 1

ชุดข้อมูล	จำนวนข้อมูล		
	เป็นโรค	ไม่เป็นโรค	รวม
ชุดข้อมูล Brain MRI Images for Brain Tumor Detection [2]	155	98	253
ชุดข้อมูล Brian Tumor Dataset [3]	2,513	2,087	4,600

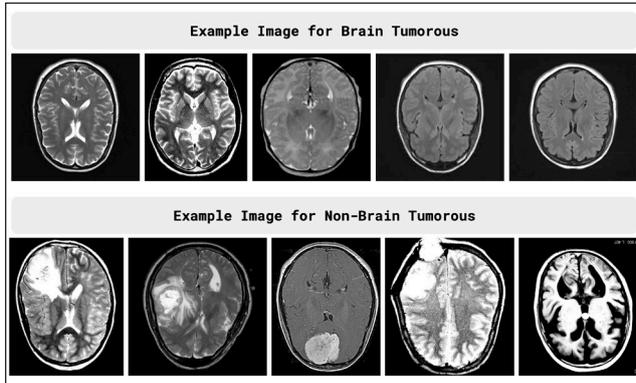
### 3.3 การประเมินประสิทธิภาพแบบจำลอง

การวัดประสิทธิภาพของแบบจำลอง ได้นำการวัดค่าความถูกต้อง (Accuracy) [22], [23] มาใช้

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

โดยที่

- TP (True Positive) คือ ผลบวกที่เป็นจริง
- FP (False Positive) คือ ผลบวกที่เป็นเท็จ
- TN (True Negative) คือ ผลลบที่เป็นจริง
- FN (False Negative) คือ ผลลบที่เป็นเท็จ



ภาพที่ 3 ตัวอย่างภาพจากชุดข้อมูล Br35H

Total 3,000 image	Training set (70%)	Validation set (20%)	Test set (10%)
Brain Tumorous 1,500 image	1,050	300	150
Non-Brain Tumorous 1,500 image	1,050	300	150

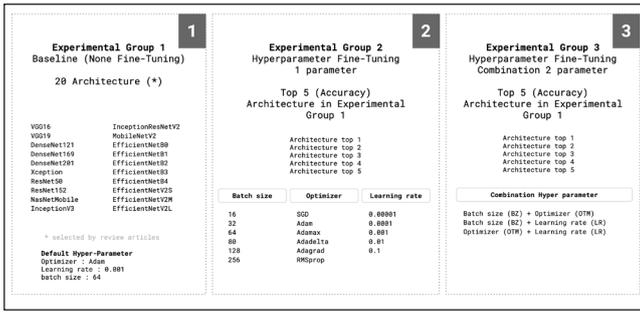
ภาพที่ 4 การแบ่งข้อมูลของชุดข้อมูลในการพัฒนาแบบจำลอง ตารางที่ 1 แสดงรายละเอียดของชุดข้อมูลที่ใช้ในการวิจัย

ชุดข้อมูล	จำนวนข้อมูล		
	เป็นโรค	ไม่เป็นโรค	รวม
ชุดข้อมูล Br35H - Brain Tumor Detection 2020 [1]	1,500	1,500	3,000

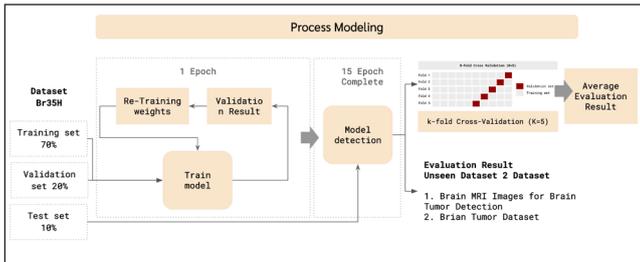
การวัดความสูญเสีย (Loss Function) คือ ฟังก์ชันที่ใช้วัดระยะห่างระหว่างค่าที่คาดการณ์ไว้กับค่าที่แท้จริงในแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning) ค่าการสูญเสียจะทำหน้าที่เป็นตัวบ่งชี้ประสิทธิภาพของแบบจำลองโดยค่าการสูญเสียต่ำจะหมายความว่าแบบจำลองทำงานได้แม่นยำมากขึ้น

### 3.4 การออกแบบการทดลอง

แบ่งการทดลองออกเป็น 3 กลุ่มตามลำดับดังแสดงในภาพที่ 6 และใช้ชุดข้อมูล Br35H - Brain Tumor Detection 2020 [1] สำหรับการเรียนรู้และพัฒนาแบบจำลอง นำชุดข้อมูล Brain MRI Images for Brain Tumor Detection [2] และชุดข้อมูล Brian Tumor Dataset [3] มาใช้ทดสอบความถูกต้องของแบบจำลอง (Unseen Dataset) ที่ได้พัฒนาขึ้น และกำหนดไฮเปอร์พารามิเตอร์พื้นฐานในการทดลองได้แก่ Optimizer = Adam Learning rate = 0.001 (ค่าพื้นฐานของ Optimizer Adam) และ batch size = 32 ดังแสดงในภาพที่ 6 กระบวนการทดลองและการพัฒนาแบบจำลอง



ภาพที่ 5 แสดงสรุปการทดลองกลุ่มที่ 1 - 3



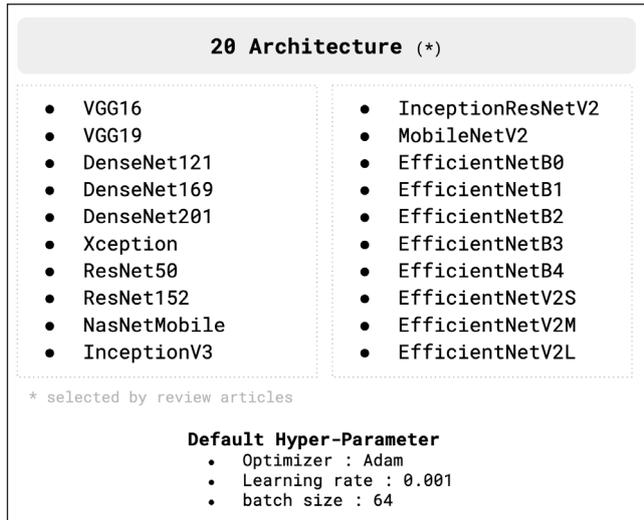
ภาพที่ 6 กระบวนการทดลองและการพัฒนาแบบจำลอง

3.4.1 การทดลองกลุ่มที่ 1 โดยใช้สถาปัตยกรรมจำนวน 20 สถาปัตยกรรม โดยได้คัดเลือกจากบทความอื่น ๆ ที่ได้รับความนิยมดังแสดงในภาพที่ 7 ขั้นตอนการทดลองกลุ่มที่ 1 ได้แก่

- VGG16-19 [5], [14], [18]
- DenseNet121-169-201 [5], [18]
- Xception [18]
- ResNet50-152 [4], [5]
- NasNetMobile [5], [13]
- InceptionV3 [18]
- InceptionResNetV2 [18]
- MobileNetV2 [11]
- EfficientNetB0 - 4 [21]
- EfficientNetV2S - M - L [15]

3.4.2 การทดลองกลุ่มที่ 2 ด้วยการปรับแต่งไฮเปอร์พารามิเตอร์จำนวน 3 ตัว และนำมาทดสอบกับสถาปัตยกรรมในกลุ่มการทดลองที่ 1 ที่ได้ค่าความถูกต้องมากที่สุด 5 ลำดับดังแสดงในภาพที่ 8 ขั้นตอนการทดลองกลุ่มที่ 2 ได้แก่

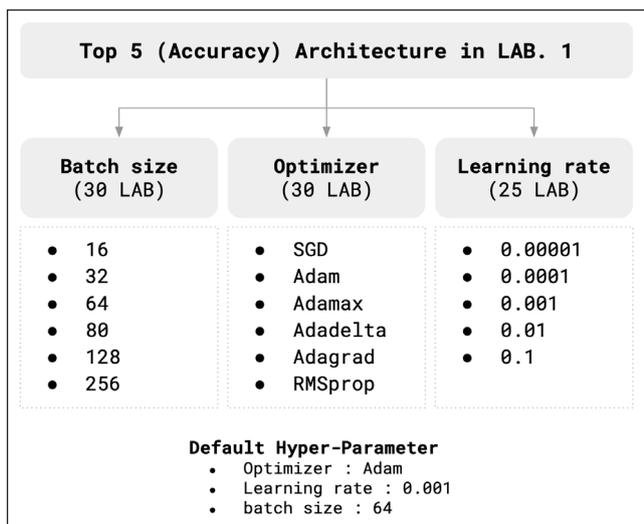
- Batch size โดยกำหนดค่า ได้แก่ 16, 32, 64, 80, 128, 256 [8], [14], [16] - [18], [21]
- Optimizer โดยกำหนดค่า ได้แก่ SGD, Adam, Adamax, Adadelata Adagrad, RMSprop [4], [5], [8], [15], [19], [21]
- Learning rate โดยกำหนดค่า ได้แก่ 0.00001, 0.0001, 0.001, 0.01, 0.1 [5], [14], [16], [17], [19], [21]



ภาพที่ 7 ขั้นตอนการทดลองกลุ่มที่ 1

3.4.2 การทดลองกลุ่มที่ 2 ด้วยการปรับแต่งไฮเปอร์พารามิเตอร์จำนวน 3 ตัว และนำมาทดสอบกับสถาปัตยกรรมในกลุ่มการทดลองที่ 1 ที่ได้ค่าความถูกต้องมากที่สุด 5 ลำดับดังแสดงในภาพที่ 8 ขั้นตอนการทดลองกลุ่มที่ 2 ได้แก่

- Batch size โดยกำหนดค่า ได้แก่ 16, 32, 64, 80, 128, 256 [8], [14], [16]-[18], [21]
- Optimizer โดยกำหนดค่า ได้แก่ SGD, Adam, Adamax, Adadelata Adagrad, RMSprop [8], [4], [5], [15], [19], [21]
- Learning rate โดยกำหนดค่า ได้แก่ 0.00001, 0.0001, 0.001, 0.01, 0.1 [5], [14], [16], [17], [19], [21]

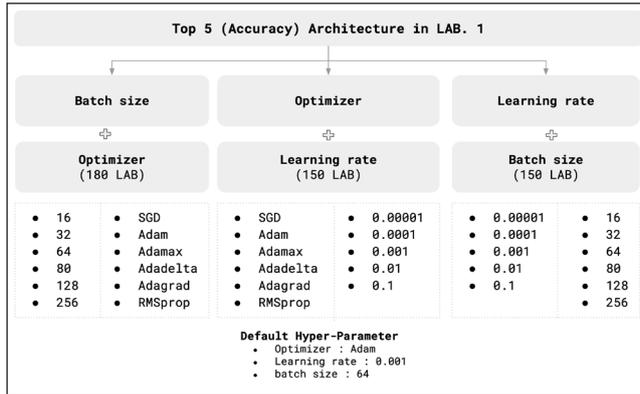


ภาพที่ 8 ขั้นตอนการทดลองกลุ่มที่ 2



3.4.3 การทดลองกลุ่มที่ 3 ด้วยการจับคู่ไฮเปอร์พารามิเตอร์ 2 ตัว และนำมาทดสอบกับสถาปัตยกรรมในกลุ่มการทดลองที่ 1 ที่ได้ค่าความถูกต้องมากที่สุด 5 ลำดับดังแสดงในภาพที่ 9 ขั้นตอนการทดลองกลุ่มที่ 3 ได้แก่

- Batch size (BZ) และ Optimizer (OTM)
- Batch size (BZ) และ Learning rate (LR)
- Optimizer (OTM) และ Learning rate (LR)



ภาพที่ 9 ขั้นตอนการทดลองกลุ่มที่ 3

#### 4. ผลการดำเนินการวิจัย

##### 4.1 ผลการทดลองกลุ่มที่ 1

นำสถาปัตยกรรมจำนวน 20 สถาปัตยกรรม มาใช้ในการทดลองได้ผลการทดลองดังตารางที่ 2 ผลการทดลองกลุ่มการทดลองที่ 1

สถาปัตยกรรมที่ได้ค่าความถูกต้องมากที่สุด 5 อันดับเพื่อนำมาใช้สำหรับการทดลองในกลุ่มที่ 2 และกลุ่มการทดลองที่ 3 คือ (เรียงจากมากไปน้อย) (ค่าความถูกต้อง ค่าความสูญเสีย)

- 1) DenseNet201 (98.00,0.08)
- 2) Xception (98.00,0.06)
- 3) InceptionResNetV2 (98.00,0.08)
- 4) MobileNetV2 (97.67,0.11)
- 5) NasNetMobile (97.33,0.11)

##### 4.2 ผลการทดลองกลุ่มที่ 2

นำสถาปัตยกรรมจากกลุ่มการทดลองที่ 1 จำนวน 5 สถาปัตยกรรม มาปรับแต่งไฮเปอร์พารามิเตอร์จำนวน 3 พารามิเตอร์ ซึ่งแต่ละไฮเปอร์พารามิเตอร์ได้กำหนดค่าที่แตกต่างกันดังแสดงในตารางที่ 3 – 5 และตารางที่ 6 ที่ได้สรุปผลพารามิเตอร์ที่ได้ผลดีที่สุดในกลุ่มการทดลองที่ 2

4.2.1 การทดลองปรับแต่งไฮเปอร์พารามิเตอร์ Batch size (BZ) ซึ่งผลการทดลองแสดงในตารางที่ 3

โดยแสดงสถาปัตยกรรมละ 3 การทดลองที่ได้ค่าความถูกต้องมากที่สุด จำนวน 5 อันดับ (เรียงจากมากไปน้อย) (ค่าความถูกต้อง ค่าความสูญเสีย)

- 1) Xception + BZ: 80 (98.34, 0.08)
- 2) MobileNetV2 + BZ: 32 (98.01, 0.08)
- 3) InceptionResNetV2 + BZ: 32 (97.67, 0.05)
- 4) NasNetMobile + BZ: 32 (97.67, 0.10)
- 5) DenseNet201 + BZ: 16 (97.67, 0.31)

ตารางที่ 2 แสดงผลการทดลองกลุ่มการทดลองที่ 1

สถาปัตยกรรม	ผลการทดลอง	
	ความถูกต้อง (Accuracy)	ความสูญเสีย (Loss)
VGG16	96.00	0.10
VGG19	95.67	0.12
DenseNet121	97.67	0.11
DenseNet169	96.67	0.13
<b>DenseNet201</b>	<b>96.67</b>	<b>0.13</b>
<b>Xception</b>	<b>98.00</b>	<b>0.08</b>
ResNet50	98.00	0.06
ResNet152	91.00	0.23
<b>NasNetMobile</b>	<b>92.33</b>	<b>0.21</b>
InceptionV3	97.33	0.11
<b>InceptionResNetV2</b>	<b>96.67</b>	<b>0.11</b>
<b>MobileNetV2</b>	<b>98.00</b>	<b>0.08</b>
EfficientNetB0	97.67	0.11
EfficientNetB1	73.67	0.58
EfficientNetB2	63.67	0.62
EfficientNetB3	74.00	0.55
EfficientNetB4	64.33	0.68
EfficientNetV2S	77.33	0.52
EfficientNetV2M	90.00	0.24
EfficientNetV2L	73.33	0.58

ตารางที่ 3 แสดงผลการปรับแต่งไฮเปอร์พารามิเตอร์

Batch size (BZ) ในกลุ่มการทดลองที่ 2

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์	ผลการทดลอง	
		BZ	ความถูกต้อง
DenseNet201	16	97.67	0.31
	32	97.34	0.11
	64	97.34	0.12
Xception	16	98.34	0.14
	80	98.34	0.08
	32	97.67	0.08
Inception ResNetV2	16	97.67	0.11
	32	97.67	0.05
	128	97.67	0.07
MobileNetV2	32	98.01	0.08
	64	97.67	0.09
	80	97.34	0.08
NasNetMobile	32	97.67	0.10
	64	97.67	0.11
	80	97.67	0.11

ตารางที่ 4 แสดงผลการปรับแต่งไฮเปอร์พารามิเตอร์

Optimizer (OTM) ในกลุ่มการทดลองที่ 2

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์	ผลการทดลอง	
		OTM	ความถูกต้อง
DenseNet201	Adamax	98.01	0.09
	Adam	97.67	0.10
	RMSprop	96.34	0.24
Xception	Adam	98.34	0.06
	SGD	97.67	0.27
	Adamax	97.67	0.06
Inception ResNetV2	SGD	97.67	0.18
	Adam	97.34	0.06
	Adamax	97.34	0.06
MobileNetV2	Adam	97.34	0.08
	Adamax	97.01	0.08
	Adagrad	97.01	0.12
NasNetMobile	Adamax	97.34	0.08
	Adam	97.01	0.14
	Adagrad	96.67	0.10

4.2.2 การทดลองปรับแต่งไฮเปอร์พารามิเตอร์ Optimizer (OTM) ซึ่งผลการทดลองแสดงในตารางที่ 4 โดยแสดงสถาปัตยกรรมละ 3 การทดลองที่ได้ค่าความถูกต้องมากที่สุด จำนวน 5 อันดับ (เรียงจากมากไปน้อย) (ค่าความถูกต้อง ค่าความสูญเสีย)

- 1) Xception + OTM: Adamax (97.67, 0.06)
- 2) DenseNet201 + OTM: Adam (97.67, 0.10)
- 3) InceptionResNetV2 + OTM: SGD (97.67, 0.18)
- 4) MobileNetV2+ OTM: Adam (97.34, 0.08)
- 5) DenseNet201 + OTM: Adamax (97.34, 0.08)

4.2.3 การทดลองปรับแต่งไฮเปอร์พารามิเตอร์ Learning rate (LR) ซึ่งผลการทดลองแสดงในตารางที่ 5 โดยแสดงสถาปัตยกรรมละ 3 การทดลองที่ได้ค่าความถูกต้องมากที่สุด จำนวน 5 อันดับ (เรียงจากมากไปน้อย) (ค่าความถูกต้อง ค่าความสูญเสีย)

- 1) InceptionResNetV2 + LR: 0.01 (98.67, 0.13)
- 2) MobileNetV2+ LR: 0.1 (98.34, 3.58)
- 3) Xception + LR: 0.001 (98.01, 0.06)
- 4) DenseNet201 + LR: 0.001 (97.67, 0.10)
- 5) NasNetMobile + LR: 0.01 (97.67, 1.25)



ตารางที่ 5 แสดงผลการปรับแต่งไฮเปอร์พารามิเตอร์ Learning rate (LR) ในกลุ่มการทดลองที่ 2

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์	ผลการทดลอง	
	LR	ความถูกต้อง	ความสูญเสีย
DenseNet201	<b>0.001</b>	<b>97.67</b>	<b>0.10</b>
	0.1	97.34	8.03
	0.0001	97.01	0.08
Xception	0.0001	98.01	0.07
	<b>0.001</b>	<b>98.01</b>	<b>0.06</b>
	0.01	97.34	0.38
Inception ResNetV2	<b>0.01</b>	<b>98.67</b>	<b>0.13</b>
	0.001	98.01	0.06
	0.0001	97.34	0.06
MobileNetV2	<b>0.1</b>	<b>98.34</b>	<b>3.58</b>
	0.01	97.67	0.98
	0.001	97.34	0.08
NasNetMobile	<b>0.01</b>	<b>97.67</b>	<b>1.25</b>
	0.001	97.34	0.10
	0.0001	97.01	0.09

ตารางที่ 6 ไฮเปอร์พารามิเตอร์ที่ได้ผลดีที่สุด ในกลุ่มการทดลองที่ 2

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์		
	BZ	OTM	LR
DenseNet201	16	Adam	0.001
Xception	80	Adamax	0.001
InceptionResNetV2	32	Adam	0.01
MobileNetV2	32	Adam	0.1
NasNetMobile	32	Adamax	0.01

### 4.3 ผลการทดลองกลุ่มที่ 3

นำสถาปัตยกรรมจากกลุ่มการทดลองที่ 1 จำนวน 5 สถาปัตยกรรมมาปรับแต่งไฮเปอร์พารามิเตอร์จำนวน 3 พารามิเตอร์ ซึ่งแต่ละไฮเปอร์พารามิเตอร์ได้กำหนดค่าที่แตกต่างกัน โดยการจับคู่ไฮเปอร์พารามิเตอร์ 2 ตัว

4.3.1 การทดลองปรับแต่งไฮเปอร์พารามิเตอร์ Batch size (BZ) และ Optimizer (OTM) ซึ่งผลการทดลองแสดงในตารางที่ 7 โดยแสดงสถาปัตยกรรมละ 3 การทดลองที่ได้ค่าความถูกต้องมากที่สุด จำนวน 5 อันดับ (เรียงจากมากไปน้อย) (ค่าความถูกต้อง ค่าความสูญเสีย)

- 1) InceptionResNetV2 + BZ: 16 + OTM: RMSprop (97.01, 0.12)
- 2) Xception + BZ: 16 + OTM: Adam (96.67, 0.11) และ Xception + BZ: 32 + OTM: RMSprop (96.67, 0.11) และ DenseNet201 + BZ: 16 + OTM: Adam (96.67, 0.11)
- 3) NasNetMobile + BZ: 16 + OTM: RMSprop (96.34, 0.10)
- 4) MobileNetV2 + BZ: 16 + OTM: RMSprop (95.67, 0.13)

ตารางที่ 7 แสดงผลการจับคู่ไฮเปอร์พารามิเตอร์ Batch size (BZ) และ Optimizer (OTM)

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์		ผลการทดลอง	
	BZ	OTM	ความถูกต้อง	ความสูญเสีย
DenseNet201	16	Adam	96.67	0.11
	16	RMSprop	96.34	0.10
	32	RMSprop	96.00	0.11
Xception	16	Adam	96.67	0.11
	32	RMSprop	96.67	0.11
	16	RMSprop	95.34	0.11
Inception ResNetV2	16	Adam	97.34	0.13
	16	RMSprop	97.01	0.12
	32	Adam	97.01	0.15

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์		ผลการทดลอง	
	BZ	OTM	ความถูกต้อง	ความสูญเสีย
MobileNetV2	<b>16</b>	<b>RMSprop</b>	<b>95.67</b>	<b>0.13</b>
	16	Adam	95.00	0.14
	32	RMSprop	94.67	0.17
NasNet Mobile	<b>16</b>	<b>RMSprop</b>	<b>96.34</b>	<b>0.10</b>
	32	RMSprop	96.34	0.13
	16	Adam	95.67	0.12

4.3.2 การทดลองปรับแต่งไฮเปอร์พารามิเตอร์ Batch size (BZ) และ Learning rate (LR) ซึ่งผลการทดลองแสดงในตารางที่ 8 โดยแสดงสถาปัตยกรรมละ 3 การทดลองที่ได้ค่าความถูกต้องมากที่สุด จำนวน 5 อันดับ (เรียงจากมากไปน้อย) (ค่าความถูกต้อง ค่าความสูญเสีย)

- 1) Xception + BZ: 80 + LR: 0.0001 (98.67, 0.37)
- 2) MobileNetV2 + BZ: 16 + LR: 0.0001 (98.34, 0.07)
- 3) InceptionResNetV2 + BZ: 32 + LR: 0.1 (98.34, 1.90)
- 4) NasNetMobile + BZ: 16 + LR: 0.01 (98.34, 4.53)
- 5) DenseNet201 + BZ: 16 + LR: 0.0001 (98.01, 0.08)

ตารางที่ 8 แสดงผลการจับคู่ไฮเปอร์พารามิเตอร์ Batch size (BZ) และ Learning rate (LR)

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์		ผลการทดลอง	
	BZ	LR	ความถูกต้อง	ความสูญเสีย
DenseNet201	<b>16</b>	<b>0.0001</b>	<b>98.01</b>	<b>0.08</b>
	16	0.001	97.67	0.32
	64	0.001	97.67	0.12
Xception	<b>80</b>	<b>0.01</b>	<b>98.67</b>	<b>0.37</b>
	128	0.001	98.34	0.07
	16	0.001	98.34	0.12

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์		ผลการทดลอง	
	BZ	LR	ความถูกต้อง	ความสูญเสีย
InceptionResNetV2	<b>32</b>	<b>0.1</b>	<b>98.34</b>	<b>1.90</b>
	16	0.1	98.01	9.31
	32	0.01	98.01	0.85
MobileNetV2	<b>16</b>	<b>0.0001</b>	<b>98.34</b>	<b>0.07</b>
	128	0.01	98.01	0.41
	32	0.01	98.01	0.80
NasNet Mobile	<b>16</b>	<b>0.01</b>	<b>98.34</b>	<b>4.53</b>
	32	0.001	98.01	0.09
	32	0.01	98.01	1.60

4.3.3 การทดลองปรับแต่งไฮเปอร์พารามิเตอร์ Optimizer (OTM) และ Learning rate (LR) ซึ่งผลการทดลองแสดงในตารางที่ 9 โดยแสดงสถาปัตยกรรมละ 3 การทดลองที่ได้ค่าความถูกต้องมากที่สุด จำนวน 5 อันดับ (เรียงจากมากไปน้อย) (ค่าความถูกต้อง ค่าความสูญเสีย)

- 1) InceptionResNetV2 + OTM: Adam + LR: 0.01 (100.0, 0.001)
- 2) Xception + OTM: Adam + LR: 0.01 (98.34, 0.56)
- 3) DenseNet201 + OTM: Adamax + LR: 0.001 (98.01, 0.07)
- 4) NasNetMobile + OTM: Adadelta + LR: 0.1 (98.01, 0.08)
- 5) MobileNetV2 + OTM: RMSprop + LR: 0.01 (98.01, 1.38)

ตารางที่ 9 แสดงผลการจับคู่ไฮเปอร์พารามิเตอร์ Optimizer (OTM) และ Learning rate (LR)

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์		ผลการทดลอง	
	OTM	LR	ความถูกต้อง	ความสูญเสีย
DenseNet201	Adam	0.001	98.01	0.08
	<b>Adamax</b>	<b>0.001</b>	<b>98.01</b>	<b>0.07</b>
	Adadelta	0.1	97.67	0.08

สถาปัตยกรรม	ไฮเปอร์พารามิเตอร์		ผลการทดลอง	
	OTM	LR	ความถูกต้อง	ความสูญเสีย
Xception	Adam	0.01	98.34	0.56
	Adagrad	0.01	98.01	0.10
	Adam	0.001	98.01	0.06
Inception ResNetV2	<b>Adam</b>	<b>0.01</b>	<b>100.00</b>	<b>0.001</b>
	Adam	0.0001	97.67	0.06
	Adam	0.001	97.67	0.07
Mobile NetV2	<b>RMSprop</b>	<b>0.01</b>	<b>98.01</b>	<b>1.38</b>
	Adadelta	0.1	97.67	0.09
	Adam	0.001	97.67	0.09
NasNet Mobile	Adadelta	0.1	98.01	0.08
	Adam	0.0001	98.01	0.09
	Adam	0.001	97.67	0.11

## 5. สรุปผลการวิจัย

จากการทดลองจำแนกโรคเนื้องอกในสมองโดยใช้โครงข่ายประสาทเทียมเชิงลึก ซึ่งใช้ชุดข้อมูล Br35H จำนวน 3,000 ภาพ แบ่งภาพที่เป็นโรคและไม่เป็นโรคอย่างละ 1,500 ภาพ สำหรับใช้ในการเรียนรู้ของแบบจำลอง แบ่งข้อมูลออกเป็น ชุดเรียนรู้ 70%: ชุดตรวจสอบ 20%: ชุดทดสอบ 10% พบว่า

กลุ่มที่ 1 นำสถาปัตยกรรมที่ได้รับความนิยมจากบทความต่าง ๆ โดยไม่มีการปรับแต่งและกำหนดค่าไฮเปอร์พารามิเตอร์ สำหรับใช้ในการทดลองไว้ แล้วนำ 5 สถาปัตยกรรมที่มีค่าความถูกต้องมากที่สุดไปใช้ในการทดลองถัดไป สถาปัตยกรรม Xception, ResNet50, InceptionResNetV2, MobileNetV2, DenseNet201 มีประสิทธิภาพสูงที่สุด และเหมาะสำหรับงานจำแนกประเภทเนื้องอกในสมอง โดยมีค่าความถูกต้องมากกว่า 98.00% และค่าความสูญเสียต่ำกว่า 0.10 สถาปัตยกรรม ResNet152 มีค่าความถูกต้องร้อยละ 91.00 และค่าความสูญเสียสูงสุด 0.23 สถาปัตยกรรม EfficientNetB1, EfficientNetB, EfficientNetB, EfficientNetB, EfficientNetV2L มีประสิทธิภาพต่ำ โดยมีค่าความถูกต้อง

น้อยกว่าร้อยละ 80.00 และค่าความสูญเสียมากกว่า 0.50 สถาปัตยกรรม EfficientNetV2S, EfficientNetV2M มีประสิทธิภาพปานกลาง โดยมีค่าความถูกต้องมากกว่า 70.00% และค่าความสูญเสียต่ำกว่า 0.60

กลุ่มที่ 2 ปรับแต่งส่วนของไฮเปอร์พารามิเตอร์จำนวน 3 ตัว ได้แก่ Batch size, Optimizer, Learning rate โดยปรับแต่งเพียงไฮเปอร์พารามิเตอร์แต่ละตัว พบว่าสถาปัตยกรรมที่มีค่าความถูกต้องมากที่สุดคือ Xception ความถูกต้องร้อยละ 98.34 MobileNetV2 ความถูกต้องร้อยละ 98.34 เมื่อใช้ Learning Rate 0.1 InceptionResNetV2 ความถูกต้องร้อยละ 98.67 เมื่อใช้ Learning Rate 0.01 จะเห็นได้ว่าการปรับ Optimizer มีผลกระทบต่อความถูกต้องมากที่สุด โดยเฉพาะการใช้ระหว่าง Adam และ Adamax และ Learning Rate ที่มีค่าต่ำ จะส่งผลต่อความถูกต้องที่สูงขึ้นไปด้วย

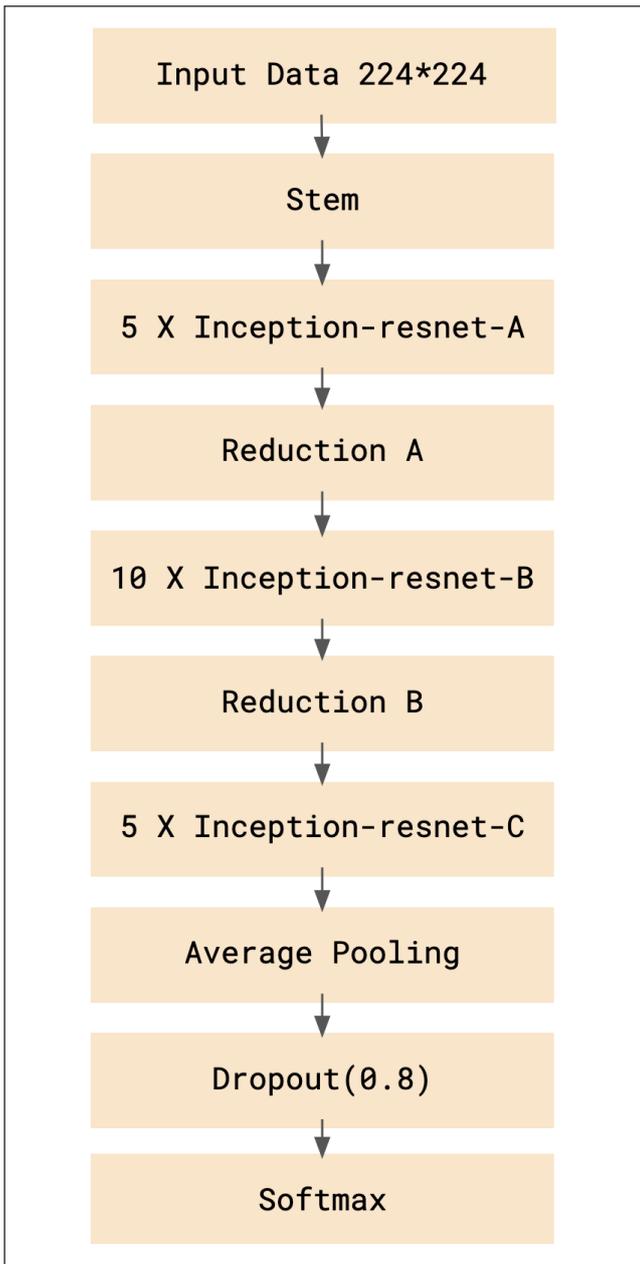
กลุ่มที่ 3 ปรับแต่งส่วนของไฮเปอร์พารามิเตอร์จำนวน 3 ตัว ได้แก่ Batch size, Optimizer, Learning rate โดยการจับคู่ไฮเปอร์พารามิเตอร์สองตัวเข้าด้วยกัน พบว่าสถาปัตยกรรมที่มีค่าความถูกต้องมากที่สุดคือ InceptionResNetV2 ค่าความถูกต้องร้อยละ 100.00 เมื่อใช้ Adam และ Learning Rate 0.01 Xception ค่าความถูกต้องร้อยละ 98.67 เมื่อใช้ Batch Size 80 และ Learning Rate 0.01 DenseNet201 ค่าความถูกต้องร้อยละ 98.01 เมื่อใช้ Batch Size 16 และ Learning Rate 0.0001 หรือ Adam/Adamax และ Learning Rate 0.001 จะเห็นได้ว่า Batch Size ที่เหมาะสมจะอยู่ที่ค่าต่ำถึงปานกลาง ได้แก่ 16 และ 32

สถาปัตยกรรมที่ได้ค่าความถูกต้องมากที่สุด คือ สถาปัตยกรรม InceptionResNetV2 [19] ดังภาพที่ 10 โครงสร้างของสถาปัตยกรรมในการทดลองได้กำหนด Batch size คือ 32 กำหนด Optimizer คือ Adam กำหนด Learning rate คือ 0.01 ได้ค่าความถูกต้องสูงที่สุดร้อยละ 100.00 และค่าความสูญเสีย 0.001 ได้แสดงในภาพที่ 11 แสดงค่าความถูกต้อง (ซ้าย) และค่าความสูญเสีย (ขวา) ของสถาปัตยกรรม InceptionResNetV2 ในการเรียนรู้ของแบบจำลอง และภาพที่ 12 แสดง Confusion Matrix ของการพยากรณ์ อีกทั้งเมื่อนำมาเปรียบเทียบกับผลการทดลองที่ 1 ที่ไม่มีการปรับแต่งใด ๆ พบว่ามีค่าความถูกต้องเพิ่มขึ้นร้อยละ 2.00 และค่าความสูญเสียลดลง 0.07

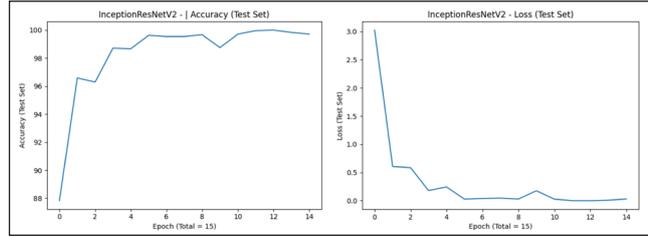
เมื่อนำมาทดสอบกับชุดข้อมูลที่ไม่เกี่ยวข้องกันชุดข้อมูล



การเรียนรู้ (Unseen Dataset) จำนวน 2 ชุดข้อมูล พบว่าชุดข้อมูล Brain MRI Images for Brain Tumor Detection ว่ามีค่าความถูกต้องร้อยละ 98.41 และค่าความสูญเสีย 0.37 และได้แสดง Confusion Matrix การพยากรณ์ชุดข้อมูล Brain MRI Images for Brain Tumor Detection ดังภาพที่ 13 และชุดข้อมูล Brian Tumor Dataset พบว่ามีค่าความถูกต้องร้อยละ 97.96 และค่าความสูญเสีย 0.24 และได้แสดง Confusion Matrix การพยากรณ์ชุดข้อมูล Brian Tumor Dataset ดังภาพที่ 14 และตารางที่ 10



ภาพที่ 10 โครงสร้างสถาปัตยกรรม InceptionResNetV2



ภาพที่ 11 แสดงค่าความถูกต้อง (Accuracy) (ซ้าย) และค่าความสูญเสีย (Loss) ของสถาปัตยกรรม InceptionResNetV2

	ปกติ (Normal)	โรคเนื้องอกในสมอง (Brain Tumor)
ปกติ (Normal)	<b>150</b>	<b>0</b>
โรคเนื้องอกในสมอง (Brain Tumor)	<b>0</b>	<b>150</b>

ภาพที่ 12 แสดง Confusion Matrix การพยากรณ์ของชุดข้อมูล Br35H [1] กับสถาปัตยกรรม InceptionResNetV2

	ปกติ (Normal)	โรคเนื้องอกในสมอง (Brain Tumor)
ปกติ (Normal)	<b>154</b>	<b>1</b>
โรคเนื้องอกในสมอง (Brain Tumor)	<b>3</b>	<b>95</b>

ภาพที่ 13 แสดง Confusion Matrix การพยากรณ์ของชุดข้อมูล Brain MRI Images [2] กับสถาปัตยกรรม InceptionResNetV2

	ปกติ (Normal)	โรคเนื้องอกในสมอง (Brain Tumor)
ปกติ (Normal)	2479	34
โรคเนื้องอกในสมอง (Brain Tumor)	60	2027

ภาพที่ 14 แสดง Confusion Matrix การพยากรณ์ของชุดข้อมูล Brain Tumor Dataset [3] กับสถาปัตยกรรม InceptionResNetV2

ตารางที่ 10 แสดงค่าความถูกต้องและค่าความสูญเสียชุดข้อมูล Brain MRI Images [2] และ Brian Tumor Dataset [3] ของสถาปัตยกรรม InceptionResNetV2

สถาปัตยกรรม	Brain MRI Images [2]		Brian Tumor Dataset [3]	
	ความถูกต้อง	ความสูญเสีย	ความถูกต้อง	ความสูญเสีย
Inception ResNetV2	98.41	0.37	97.96	0.24

## 6. เอกสารอ้างอิง

- [1] Kaggle, *Br35H: Brain Tumor Detection 2020*. Available Online at <https://kaggle.com/datasets/ahmedhamada0/brain-tumor-detection>, accessed on 13 April 2024.
- [2] Kaggle, *Brain MRI Images for Brain Tumor Detection*. Available Online at <https://kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection>, accessed on 13 April 2024.
- [3] Kaggle, *Brian Tumor Dataset*. Available Online at <https://kaggle.com/datasets/preetviradiya/brian-tumor-dataset>, accessed on 13 April 2024.
- [4] S. Sengupta, S. Basak, P. Saikia, S. Paul, V. Tsalavoutis, F. Atiah, and A. Peters. "A review of deep learning with special emphasis on architectures, applications and recent trends." *Knowledge-Based Systems*, Vol. 194, April, 2020.
- [5] S. Mathivanan, S. Sonaimuthu, and S. Murugesan. "Employing deep learning and transfer learning for accurate brain tumor detection." *Scientific Reports*, Vol. 14, 2024.
- [6] M. Almufareh, M. Imran, A. Khan, M. Humayun, and M. Asim. "Automated Brain Tumor Segmentation and Classification in MRI Using YOLO-Based Deep Learning." *IEEE Access*, Vol. 12, pp. 16189-16207, 2024.
- [7] A. Asiri, A. Shaf, T. Ali, M. Aamir, M. Irfan, and S. Alqahtani. "Enhancing brain tumor diagnosis: an optimized CNN hyperparameter model for improved accuracy and reliability." *PeerJ. Computer science*, 2024.
- [8] A. Abdulalomov, B. Mukhiddinov, and M. Whangbo. "Brain Tumor Detection Based on Deep Learning Approaches and Magnetic Resonance Imaging." *Cancers*, Vol. 15, No. 16, pp. 4172, 2023.
- [9] A. Banerjee, K. Jaiswal, T. Biswas, V. Sharma, M. Bal, and S. Mishra. "Brain Tumor Detection and Classification Using a Hyperparameter Tuned Convolutional Neural Network." *Proceedings of International Conference on Contemporary Computing and Informatics (IC3I)*, Gautam Buddha Nagar, India, pp. 502-506, 2023.
- [10] A. Minarno, E. Hazmi, M. Munarko, and H. Hariyady. "Convolutional Neural Network with Hyperparameter Tuning for Brain Tumor Classification." *Kinetik: Game Technology Information System Computer Network, Computing, Electronics and Control*, Vol. 6, No. 2, pp. 127-132, May, 2021.
- [11] L. Xu and M. Mohammadi. "Brain tumor diagnosis from MRI based on Mobilenetv2 optimized by contracted fox optimization algorithm." *Heliyon*, Vol. 10, No. 1, January, 2023.



- [12] M. Güle and E. Naml. "Brain Tumor Detection with Deep Learning Methods' Classifier Optimization Using Medical Images." *Applied Sciences*, Vol. 14, No. 2, 2024.
- [13] B. Bab, S. Srinivasan, S. Mathivanan, S. Mahalakshmi, P. Jayagopal, and G. Dalu. "Detection and classification of brain tumor using hybrid deep learning models." *Scientific Reports*, Vol. 13, No. 1, 2023.
- [14] N. Ullah, A. Javed, A. Alhazmi, S. Hasnain, M. Tahir, and A. Ashraf. "TumorDetNet: A unified deep learning model for brain tumor detection and classification." *Plos one*, Vol. 18, No. 9, 2023.
- [15] A. Asiri, A. Shaf, A. Ali, T. Aamir, M. Irfan, M. Alqahtani, and S. Alqhtani. "Brain tumor detection and classification using fine-tuned CNN with ResNet50 and U-Net model: A study on TCGA-LGG and TCIA dataset for MRI applications." *Life*, Vol. 13, No. 7, 2023.
- [16] Y. Anagun. "Smart brain tumor diagnosis system utilizing deep convolutional neural networks." *Multimedia Tools and Applications*, Vol. 82, pp. 44527-44553, 2023.
- [17] E. Irmak. "Multi-classification of brain tumor MRI images using deep convolutional neural network with fully optimized framework." *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, Vol. 45, No.3, pp. 1015-1036, 2021.
- [18] V. Kavitha and K. UlagaPriya. "Comparative Evaluation for Brain Tumor Detection Using Inception-V3 Architecture." *International Journal of Intelligent Systems and Applications in Engineering*, Vol. 12, No. 1, pp. 277-283, 2024.
- [19] A. Anaya-Isaza, L. Mera-Jiménez, L. Verdugo-Alejo, and L. Sarasti. "Optimizing MRI-based brain tumor classification and detection using AI: A comparative analysis of neural networks, transfer learning, data augmentation, and the cross-transformer network." *European Journal of Radiology Open*, Vol. 10, 2023.
- [20] A. Mahir. "Bayesian Optimization-based CNN Framework for Automated Detection of Brain Tumors." *Balkan Journal of Electrical and Computer Engineering*, Vol. 11, No. 4, pp. 395-404, 2022.
- [21] J. Miah, D. Cao, M. Sayed, A. Taluckder, S. Haque, and S. Mahmud. "Advancing Brain Tumor Detection: A Thorough Investigation of CNNs, Clustering, and SoftMax Classification in the Analysis of MRI Images." *arXiv:2310.17720*, 2023.
- [22] A. Šimundić. "Measures of diagnostic accuracy: basic definitions." *EJIFCC*, Vol. 19, No. 4, January, 2009.
- [23] D.M.W. Powers. "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation." *arXiv:2010.16061*, 2020.

