Intracranial Tumor Detection and Classification Models Comparison Using YOLO Algorithms

Lim Cammy   
Lee Kong Chian Faculty of Engineering and Science (LKCFES)  
*Universiti Tunku Abdul Rahman (UTAR)*

Leon Siow Yi Hong  
Lee Kong Chian Faculty of Engineering and Science (LKCFES) *Universiti Tunku Abdul Rahman (UTAR)*  
  
Liow Ke Han  
Lee Kong Chian Faculty of Engineering and Science (LKCFES)  
*Universiti Tunku Abdul Rahman (UTAR)*

Wong Yu Chi  
Lee Kong Chian Faculty of Engineering and Science (LKCFES)  
*Universiti Tunku Abdul Rahman (UTAR)*

*Abstract*—Intracranial tumor detection is a critical task in medical diagnostics, requiring accurate classification of various tumor types such as glioma, meningioma, pituitary, and space-occupying lesions. This study evaluates the performance of Convolutional Neural Network (CNN)-based You Only Look Once (YOLO) object detection algorithms, which are YOLOv8, YOLOv11, and YOLOv12. Each model was trained and fine-tuned using hyperparameter tuning to ensure quality detection. The performance is evaluated using performance metrics, including precision, recall, F1 score, and mean average precision (mAP). While all models demonstrate high accuracy on the training dataset, significant performance drops on validation dataset indicated overfitting. Additionally, some challenges appear in this study, including dataset imbalance and hardware limitations. Despite these issues, the comprehensive analysis highlights room for improvement on datasets to train more accurate models for tumor detection.

1. INTRODUCTION

Intracranial tumor, also known as brain tumor, is referred to as a mass of abnormal tissue specifically around brain. According to Baptist Health (2021), an intracranial tumor can be cancerous or noncancerous, depending on whether the cells in the tumor growing uncontrollably. However, the brain function of a patient can be affected by intracranial tumor once it is large enough to compress nerves, regardless of cancerous or noncancerous. To detect the intracranial tumor, the Magnetic Resonance Imaging (MRI) is generally used to scan the patient's brain and display the brain images to identify whether there is a brain tumor.

While the proposed method of MRI scans can be useful in detecting brain tumors, there is still an overlying issue where some brain tumors are not easily detectable by the human eye, making some diagnosis of the patients unreliable. In which recent advancements in the neural network field have led to many emerging technologies aiding the medical field, namely the Convolutional Neural Network (CNN).

CNN is an advanced version of artificial neural networks (ANNs), which primarily extract information from grid-like matrix datasets. This enables CNN to perform well when extracting data from visual datasets such as images or videos, which makes it widely used in computer vision applications. According to Svitla Systems (2024), image classification can be easily achieved by CNNs as they are able to effectively detect local and spatial patterns in data by applying appropriate filters, which then classify the images based on the features that were detected using fully connected layers.

The objective of this report is to train the several You Only Look Once (YOLO) algorithm versions (YOLOv12, YOLOv11, YOLOv8) to be able to effectively and accurately detect brain tumors through images and live video footage as well as comparing the performance matrixes between the YOLO models to find the best performing model. The scope would encompass brain tumors such as glioma, meningioma, pituitary and space-occupying lesions.

1. LITERATURE REVIEW

The You Only Look Once (YOLO) is one of the famous algorithms that are used in object detection and classification based on convolutional networks (Kelta, 2024). Hence, it is also popular in the intracranial tumor detection and classification area.

According to a study, YOLOv8 has proved to have a higher speed and accuracy compared with YOLOv5, YOLOv6, and YOLOv7 in detecting, localizing, and classifying various types of intracranial tumors based on the COCO dataset (Patel *et al*, 2024). In this study, the review concluded that the YOLOv8 offers a significant improvement compared with the previous series of YOLO algorithms in terms of detection and classification accuracy and performance speed. However, this study also mentions a potential limitation of YOLOv8 in the detection and classification of intracranial tumors, is lack of support for 1280 resolution models. If high-resolution imaging is important for detailed tumor analysis, this constraint might affect the YOLOv8 model’s performance.

In addition, YOLOv11 has been utilized for real-time intracranial tumor detection. In a study, the processing speed for YOLOv11 reaches 34.16 frames per second while maintaining performance of 0.95 mAP50 and 0.65 mAP50-95 (Reis, 2024). In this study, the system was implemented and shows good performance on a cohort of 15 consecutively operated intracranial tumor patients. Hence, this shows a seamless integration in surgical workflow with YOLOv11 in the system, indicating a good ability of YOLOv11 in real-time detection and classification of intracranial tumors.

These two series of YOLO show good performance in the detection and classification of intracranial tumors. However, the application of the latest YOLO version, which is YOLOv12, in the detection and classification of intracranial tumors is currently limited. This research aims to bridge this gap by exploring the performance and utilization of YOLOv12 compared with YOLOv8 and YOLOv11 in detecting and classifying intracranial tumors.

# Methodology

## **Data Collection and Preprocessing**

The dataset was acquired from Roboflow.com in which 1986 images of brain x-ray scans were provided. The images were then separated into three different categories, which include 1370 images for the training dataset, 395 images for the validation dataset and 191 images for the testing dataset.

Image pre-processing is the process of manipulating and processing raw image data into useful information in which unwanted artifacts are removed while essential qualities are improved to be fed into machines vision applications. The dataset that was acquired has the following pre-processing methods applied:

* Auto-orientation of pixel data with Exchangeable Image File Format (EXIF)-orientation stripping.
* Resizing images to 640x640 pixel resolution with stretch.
* Filtering images that do not contain any annotations.

Image augmentation is the process of applying various transformation techniques to the original images. This enables more effective training for the YOLO model as it prevents image overfitting from occurring. The following image augmentations were directly applied to the dataset using the tools provided on Roboflow.com before downloading the dataset:

* 50% probability of horizontal flip
* Random crops between 0 and 20 percent of the image
* Random shear of between -10° to +10° horizontally and -10° to +10° vertically
* Applying Salt and pepper noise to 0.1 percent of pixels

## **Model Architecture Design**

* + 1. The Backbone

A diagram of a diagram

AI-generated content may be incorrect.

* + - 1. Backbone Blocks of YOLOv11

The main function of the backbone is to extract features from input images at multiple scales using convolutional layers. According to Rao (2024), YOLOv11 uses C3K2 blocks which is an evolution of the CSP (Cross Stage Partial) bottleneck introduced in earlier versions to handle feature extraction at different stages of the backbone.

The newer C3K2 blocks employ a smaller 3x3 C3K block which allows for more efficient computation while retaining the model’s ability to capture key features. By processing smaller, separate feature maps and merging them after several convolutions, the C3K2 block improves feature representation with fewer parameters compared to YOLOv8’s C2F blocks.

Functionally, the C3K2 block uses the C3K to process information in which a series of Conv blocks are used and concatenating the output of the blocks.

According to Rabbani (2025), YOLOv12 employs a newer convolutional block class which emphasizes lightweight operations and higher parallelization. These blocks utilize a series of smaller kernels.

By distributing the computation across multiple small convolutions instead of fewer large ones, YOLOv12 achieves faster processing without compromising feature extraction quality.

* + 1. The Neck

A diagram of a computer program

AI-generated content may be incorrect.

1. Spatial Pyramid Pooling Fast Block

The main function of the Neck is to aggregate features from different scales and transmits them to the head for predictions.

According to Rao (2024), YOLOv11 retains the SPFF module (Spatial Pyramid Pooling Fast), which was designed to pool features from different regions of an image at varying scales. This enhances the network's capacity to record items of all sizes, particularly small ones, which has proven difficult for previous iterations of YOLO.

The SPFF module features multiple max-pooling operations to aggregate multi-scale contextual information, which ensures that even small objects are recognized by the model, as it effectively combines information across different resolutions​.

According to Rabbani (2025), YOLOv12 innovates the Neck by introducing an area attention mechanism accelerated by FlashAttention, which enhances the model’s focus on critical regions in cluttered scenes.

By segmenting feature maps into areas and applying fast attention routines, YOLOv12 reduces memory transfers and computational overhead, enabling real-time inference even at higher input resolutions.

* + 1. The Head

The main function of the Head is to generate final predictions, including bounding box coordinates and class labels. According to Rao (2024), YOLOv11 uses a multi-scale prediction head to detect objects at different sizes in which it outputs detection boxes for three different scales (low, medium, high) using the feature maps generated by the backbone and neck.

According to Rabbani (2025), the head of YOLOv12 improves on the older models by including streamlined multi-scale detection pathways, and specialized loss functions that better balance localization and classification objectives. This enables better performance of YOLOv12 in real-time applications.

## **Model Training**

To train the model, hyperparameters and augmentations variable are set to effectively train the model based on the training dataset. Key training settings include batch size, learning rate and momentum which directly affect the speed and accuracy of the model. The values used are as follows:

|  |  |
| --- | --- |
| Hyperparameters | Values |
| epochs | 50 |
| batch | 16 |
| imgsz | 640 |
| workers | 8 |
| optimizer | Auto |
| lr0 | 0.001 |
| lrf | 0.00001 |
| momentum | 0.937 |

|  |  |
| --- | --- |
| Augmentations | Values |
| box | 0.2 |
| degrees | 0.0 |
| translate | 0.1 |
| shear | 0.0 |
| perspective | 0.0 |

After the model has succeeded in the training through all 50 training epochs, the model is then fed through a validation dataset to verify the accuracy and effectiveness of the model.

## **Model Evaluation**

After the model is trained and validated, an evaluation process is carried out to determine the accuracy and effectiveness of the trained model in which several performance parameters are taken into account, which according to Ultralytics (n.d.) include:

* P (Precision): The accuracy of the detected objects, stating how many detections were correct.
* R (Recall): The ability of the model to identify all instances of objects in the images.
* F1 Score: The evaluation of how well a classification model performs on a dataset.
* mAP50: Mean average precision calculated at an intersection over union (IoU) threshold of 0.50. It measures the accuracy of the model solely based on the ‘easy’ detections.
* mAP50-95: The average of the mean average precision calculated at varying IoU thresholds, ranging from 0.50 to 0.95. It gives a comprehensive view of the model's performance across different levels of detection difficulty.

The model is then compared to the different versions and iterations of the YOLO model, namely v11 and v8 with the same hyperparameters, augmentations and dataset to truly evaluate the accuracy of the model in detecting brain tumors.

## **Fine-tuning and Optimization**

When it comes to fine-tuning and optimizing the YOLO model, several modifications were made to the hyperparameters that were used during the training of the model, which are the lr0 (initial learning rate) and the lrf (final learning rate) of the model.

Altering these parameters would directly influence how rapidly model weights are updated and how the learning rate adjusts over time, in which case would make the learning process of the model faster or slower depending on the value set.

|  |  |
| --- | --- |
| Hyperparameters | Values |
| epochs | 50 |
| batch | 16 |
| imgsz | 640 |
| workers | 8 |
| optimizer | auto |
| lr0 | 0.01 |
| lrf | 0.01 |
| momentum | 0.937 |

|  |  |
| --- | --- |
| Augmentations | Values |
| box | 0.2 |
| degrees | 0.0 |
| translate | 0.1 |
| shear | 0.0 |
| perspective | 0.0 |

Note: Augmentations were not altered to maintain consistency in the dataset.

1. EXPERIMENTAL RESULTS

## **YOLO 8 Algorithm**

### Checking for Overfitting Using Performance Matrix

1. statistical summary of yolo 8 on training and validation datasets for default model for all classes

| Dataset | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| Training | 0.950 | 0.966 | 0.958 | 0.975 | 0.877 |
| Validation | 0.899 | 0.628 | 0.789 | 0.673 | 0.562 |

The YOLOv8 model with default hyperparameters shows a good performance on the training dataset, with a precision of 95% and recall of 96.6%, indicating that the model is able to detect most of the actual tumor cases and accurately identify the tumor without false positives during training. The F1 score of 95.8% shows a balance between precision and recall. In addition, the mean Average Precision at an Intersection over Union threshold of 0.5 (mAP@0.5) is 97.5%, while the mean Average Precision across Intersection over Union thresholds from 0.5 to 0.95 (mAP@0.5:0.95) is 87.7%. This indicates that the model performs well in localizing the tumor in the training dataset. However, the precision of this model decreases to 89.9% and recall decreases to 62.8%, suggesting that the model is missing some actual tumor cases while maintaining a high true positive percentage. The percentage of false negatives is high in the validation dataset. The 78.9% F1 score also shows that the imbalance between precision and recall occurs. Moreover, the mAP@0.5 decreases to 67.3% while the mAP@0.5:0.95 decreases to 56.2% during validation, indicating that the model is poor in localizing tumors in the validation dataset. Overall, the YOLOv8 model with default hyperparameters performs well in the training dataset but poorly in the validation dataset, especially in recall, mAP@0.5, and mAP@0.5:0.95. This might be due to the imbalanced classes of the original dataset.

1. statistical summary of yolo 8 on training and validation datasets for fine-tuned model for all classes

| Dataset | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| Training | 0.992 | 0.970 | 0.981 | 0.993 | 0.934 |
| Validation | 0.696 | 0.612 | 0.651 | 0.651 | 0.558 |

The YOLOv8 model after fine-tuning shows a good performance on the training dataset. The 95% precision and 97% recall scores show that the YOLOv8 model after fine-tuning is able to detect most of the actual tumor cases and accurately identify the tumor types, indicating that the model is able to minimize the false positive and false negative cases after fine-tuning. The F1 score of 98.1% shows a balance between precision and recall during training. In addition, mAP@0.5 is 99.3%, while the mAP@0.5:0.95 is 93.4% in the training dataset. This indicates that the model performs well in localizing the tumor in the training dataset. However, the precision of this model decreases to 69.6% and recall decreases to 61.2%, suggesting that the model is missing some actual tumor cases but remains highly correct in identifying the tumor types. The percentage of false negatives is high, but the percentage of false positives is low in the validation dataset. The 65.1% F1 score also shows that the imbalance between precision and recall occurs. Moreover, the mAP@0.5 decreases to 65.1% while the mAP0.5:0.95 decreases to 55.8% during validation, indicating that the model is poor in localizing tumors in the validation dataset. Overall, the YOLOv8 model after fine-tuning performs well in the training dataset but poorly in the validation dataset, especially in recall, mAP@0.5, and mAP@0.5:0.95. This might be due to the imbalanced classes of the original dataset.

### Compare Performance for Default Model and Fine-tuned Model

1. performance comparison between original and fine-tuned yolo 8 on testing dataset for all classes

| Dataset | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| Original | 0.570 | 0.660 | 0.612 | 0.618 | 0.484 |
| Finetuned | 0.694 | 0.600 | 0.644 | 0.635 | 0.508 |

After the fine-tuning, the YOLOv8 model shows improvements in overall performance in test dataset. First, the precision of the fine-tuned model is growing from 57% to 69.4%, showing an improvement of 12.4% during training compared with the model with the default hyperparameter. This indicates that the fine-tuning process effectively reduces false positives. Second, the recall of the fine-tuned model is dropping from 66% to 60% compared to the default model. This suggests that the ability to detect all tumor cases of the model is reduced after fine-tuning. The fine-tuned YOLOv8 model prioritizes specificity compared with sensitivity. However, the trade-off is common when optimizing for precision. To prove that, the F1 score of the fine-tuned model increases from 61.2% to 64.4% compared to the default mode, indicating a better balance between precision and recall. Third, the mAP@0.5 is increased from 61.8% to 63.5% after fine-tuning, proving that the fine-tuning effectively improves the performance of localization of tumors. Fourth, the mAP@0.5:0.95 also improved from 48.4% to 50.8% after fine-tuning. Overall, the overall performance of the YOLOv8 algorithm in intracranial tumor detection and classification is improved by the fine-tuning process.

### Confusion Matrix for Testing Dataset of Original YOLOv8 Model

A screenshot of a computer game

AI-generated content may be incorrect.

1. Confusion matrix for testing dataset of original YOLOv8 model

### Confusion Matrix for Testing Dataset of Original YOLOv8 Model

A screenshot of a computer game

AI-generated content may be incorrect.

1. Confusion matrix for testing dataset of original YOLOv8 model

### Recall-confidence Curve for Testing Dataset of Fine-tuned YOLOv8 Model

A graph of different colored lines

AI-generated content may be incorrect.

1. Recall-confidence curve for testing dataset of fine-tuned YOLOv8 model

### Precision-recall Curve for Testing Dataset of Fine-tuned YOLOv8 Model

A graph of a patient's curve

AI-generated content may be incorrect.

1. Precision-recall curve for testing dataset of fine-tuned YOLOv8 model

### Precision\_confidence Curve for Testing Dataset of Fine-tuned YOLOv8 Model

A graph of a graph showing the difference between a line and a line

AI-generated content may be incorrect.

1. Spatial Pyramid Pooling Fast Block

### F1-confidence Curve for Testing Dataset of Fine-tuned YOLOv8 Model

A graph of different colored lines

AI-generated content may be incorrect.

1. F1-confidence curve for testing dataset of fine-tuned YOLOv8 model

## **YOLO 11 Algorithm**

### Checking for Overfitting Using Performance Matrix

1. statistical summary of yolo 11 on training and validation datasets for default model for all classes

| Dataset | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| Training | 0.942 | 0.907 | 0.924 | 0.944 | 0.841 |
| Validation | 0.925 | 0.621 | 0.743 | 0.676 | 0.550 |

The YOLOv11 model with default hyperparameters shows good performance in the training dataset. The 94.2% precision and 90.7% recall show that the model is able to detect most tumor cases and correctly identify them during training, indicating low false negative and false positive rates. The F1 score of 92.4% suggests that the trade-off between precision and recall is balanced in the training dataset. In addition, mAP@0.5 is 94.4%, while the mAP@0.5:0.95 is 84.1% in the training dataset. This indicates that the model performs well in localizing the tumor in the training dataset. However, the model is performing badly overall during the validation. The precision of validation drops to 92.5% while the recall of validation dropping to 62.1% indicates that the model has failed to minimize the false negative in tumor detection and classification, although the false positive remains few. With an F1 score of 74.3%, the trade-off between precision and recall in the validation state is no longer balanced as it was during training. Both mAP@0.5 and mAP@0.5:0.95 are decreases to 67.6% and 55%, respectively. This shows that the YOLOv11 model is bad at localizing tumors during validation. Overall, the YOLOv11 model with default hyperparameters performs well during training but poorly in validation. This might be due to the imbalanced classes of the original dataset.

1. statistical summary of yolo 11 on training and validation datasets for fine-tuned model for all classes

| Dataset | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| Training | 0.961 | 0.972 | 0.966 | 0.985 | 0.916 |
| Validation | 0.866 | 0.649 | 0.742 | 0.667 | 0.554 |

The YOLOv11 model after fine-tuning shows good performance in the training dataset. The 96.1% precision and 97.2% recall show that the model is able to detect most tumor cases and correctly identify them during training, indicating low false negative and false positive rates. The fine-tuned YOLOv11 model prioritizes sensitivity compared with specificity. The F1 score of 96.6% suggests that the trade-off between precision and recall is balanced in the training dataset. In addition, mAP@0.5 is 98.5%, while the mAP@0.5:0.95 is 91.6% in the training dataset. This indicates that the model performs well in localizing the tumor in the training dataset. However, the model is performing badly overall during the validation. The precision of validation drops to 86.6% while the recall of validation dropping to 0.621 indicates that the model has failed to minimize the false negative in tumor detection and classification while increases the percentage of false positive cases. With an F1 score of 74.3%, the trade-off between precision and recall in the validation state is no longer balanced as it was during training. Both mAP@0.5 and mAP@0.5:0.95 are decreases to 66.7% and 55.4%, respectively. This shows that the YOLOv11 model is bad at localizing tumors during validation. Overall, the YOLOv11 fine-tuned model performs well during training but poorly in validation. This might be due to the imbalanced classes of the original dataset.

### Compare Performance for Default Model and Fine-tuned Model

1. performance comparison between original and fine-tuned yolo 11 on testing dataset for all classes

| Testing Dataset | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| Original | 0.816 | 0.629 | 0.710 | 0.637 | 0.501 |
| Fine-tuned | 0.650 | 0.643 | 0.6467 | 0.653 | 0.502 |

After fine-tuning, the overall performance of the model is increased compared to before during testing. Although the precision after fine-tuning drops from 81.6% to 65%, the other performance metrics increase. The recall score increases from 62.9% to 64.3%, showing an enhancement in accurately detecting all occurrences of tumor cases. Besides, the mAP@0.5 increases from 63.7% to 65.3%, while mAP@0.5:0.95 increases from 50.1% to 50.2%, indicating the improvement in localizing tumors after fine-tuning. Overall, the YOLOv11 fine-tuned model can provide a more accurate and reliable prediction compared to the model with default hyperparameters.

### Confusion Matrix for Testing Dataset of Original YOLOv11 Model

A screenshot of a computer screen

AI-generated content may be incorrect.

1. Precision-recall curve for testing dataset of fine-tuned YOLOv11 model

### Confusion Matrix for Testing Dataset of Fine-tuned YOLOv11 Model

A screenshot of a computer screen

AI-generated content may be incorrect.

1. Precision-recall curve for testing dataset of fine-tuned YOLOv11 model

### Recall\_confidence Curve for Testing Dataset of Fine-tuned YOLOv11 Model

A graph of a curve

AI-generated content may be incorrect.

1. Recall\_confidence curve for testing dataset of fine-tuned YOLOv11 model

### Precision-recall Curve for Testing Dataset of Fine-tuned YOLOv11 Model

A graph of a patient's disease

AI-generated content may be incorrect.

1. Precision-recall curve for testing dataset of fine-tuned YOLOv11 model

### Precision-confidence Curve for Testing Dataset of Fine-tuned YOLOv11 Model

A graph showing the difference between the different types of data

AI-generated content may be incorrect.

1. Precision-confidence curve for testing dataset of fine-tuned YOLOv11 model

### F1-confidence Curve for Testing Dataset of Fine-tuned YOLOv11 Model

A graph of different colored lines

AI-generated content may be incorrect.

1. F1-confidence curve for testing dataset of fine-tuned YOLOv11 model

## **YOLO 12 Algorithm**

### Checking for Overfitting Using Performance Matrix

1. statistical summary of yolo 12 on training and validation datasets for default model for all classes

| Dataset | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| Training | 0.950 | 0.966 | 0.958 | 0.975 | 0.877 |
| Validation | 0.899 | 0.628 | 0.739 | 0.673 | 0.562 |

The YOLO 12 default model performs well during training. The precision of 95% and the recall of 96.6% show that the model is able to detect and classify most tumor cases with high accuracy. The false negative and false positive rates are low. Besides, the 95.8% F1 score suggests that the trade-off between precision and recall is balanced for this model during training. Additionally, the mAP@0.5 has a score of 97.5%, while the mAP@0.5:0.95 has a score of 87.7%, showing that the model performs well in localizing the tumors. However, the overall performance of this model is poor during the validation. The precision decreases to 89.9% while the recall decreases to 62.8%, indicating that the failure to minimize the false negative and false positive cases in tumor detection and classification. The dropping of the F1 score to 73.9% during validation shows the trade-off between precision and recall is no longer as good as training. Moreover, the mAP@0.5 and mAP@0.5:0.95 are decreases to 67.3% and 56.2%, respectively, showing that the model performance in locating the tumor location is decreasing. Overall, the performance of the YOLO 12 model with default hyperparameters is good during training but poor during validation. This might be due to the imbalanced classes of the original dataset.

1. statistical summary of yolo 12 on training and validation datasets for fine-tuned model for all classes

| Dataset | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| Training | 0.900 | 0.850 | 0.874 | 0.905 | 0.772 |
| Validation | 0.674 | 0.631 | 0.652 | 0.665 | 0.538 |

The YOLOv12 fine-tuned model shows a good performance in the training dataset. The 90% precision and 85% recall indicate that the model is able to detect most of the actual tumor cases and accurately identify the tumor during training. This shows that the model is efficient in minimizing the false negatives and positives. The 87.4% F1 score suggests that the trade-off between precision and recall is balanced during training. Furthermore, the 90.5% mAP@0.5 and 53.8% mAP@0.5:0.95 indicate that the model is performing well in localizing tumors. However, during the validation stage, the precision and the recall are decreased to 67.4% and 63.1%, respectively. The model has failed to minimize the false negative and false positive. The F1 score drops to 65.2% during validation, showing that the trade-off between precision and recall is no longer as training. Moreover, the mAP@0.5 decreases to 66.5%, while the mAP@0.5:0.95 decreases to 53.8% during validation, indicating that the model is poor in localizing tumors in the validation dataset. Overall, the YOLOv12 fine-tuned model performs well in training but poorly in validation. This might be due to the imbalanced classes of the original dataset.

### Compare Performance for Default Model and Fine-tuned Model

1. performance comparison between original and fine-tuned yolo 12 on testing dataset for all classes

| Dataset | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| Original | 0.652 | 0.681 | 0.666 | 0.669 | 0.514 |
| Fine-tuned | 0.572 | 0.757 | 0.652 | 0.635 | 0.498 |

After fine-tuning, the YOLOv12 model shows slightly better results on the testing dataset. Although precision decreases from 65.2% to 57.2%, indicating a higher number of false positives, the recall increases from 68.1% to 75.7%, showing that the model becomes better at identifying more actual tumor cases after fine-tuning. This suggests that the model prioritizes sensitivity over specificity. The F1 score decreases from 66.6% to 65.2%, reflecting the trade-off between precision and recall. In terms of localization performance, the mAP@0.5 drops from 66.9% to 63.5%, and the mAP@0.5:0.95 drops from 51.4% to 49.8%, indicating a marginal reduction in localization accuracy after fine-tuning. Overall, the fine-tuned YOLOv12 model improves recall but sacrifices some precision and localization performance. This trade-off may be acceptable in medical contexts where detecting more tumor cases is preferred over missing potential cases even more false positive.

### Confusion Matrix for Testing Dataset of Original YOLOv12 Model

A screenshot of a computer game

AI-generated content may be incorrect.

1. Confusion matrix for testing dataset of original YOLOv12 model

### Confusion Matrix for Testing Dataset of Fine-tuned YOLOv12 Model

A screenshot of a computer game

AI-generated content may be incorrect.

1. Confusion matrix for testing dataset of fine-tuned YOLOv12 model

### Recall\_confidence Curve for Testing Dataset of Fine-tuned YOLOv12 Model

A graph of different colored lines

AI-generated content may be incorrect.

1. Recall\_confidence curve for testing dataset of fine-tuned YOLOv12 model

### Precision-recall Curve for Testing Dataset of Fine-tuned YOLOv12 Model

A graph of a patient's disease

AI-generated content may be incorrect.

1. Precision-recall curve for testing dataset of fine-tuned YOLOv12 model

### Precision-confidence Curve for Testing Dataset of Fine-tuned YOLOv12 Model

A graph of different colored lines

AI-generated content may be incorrect.

1. Precision-confidence Curve for Testing Dataset of Fine-tuned YOLOv12 model

### F1-confidence Curve for Testing Dataset of Fine-tuned YOLOv12 Model

A graph of different colored lines

AI-generated content may be incorrect.

1. F1-confidence curve for testing dataset of fine-tuned YOLOv12 model

## Compare Performance of YOLOv8, YOLOv11, and YOLOv12 algorithms

1. statistical summary of testing dataset performance across different fine-tuned models for all classes

| Fine-tuned Model | Metrics | | | | |
| --- | --- | --- | --- | --- | --- |
| P | R | F1 score | mAP50 | mAP50-95 |
| YOLOv8 | 0.694 | 0.600 | 0.644 | 0.635 | 0.508 |
| YOLOv11 | 0.650 | 0.643 | 0.6467 | 0.653 | 0.502 |
| YOLOv12 | 0.572 | 0.757 | 0.652 | 0.635 | 0.498 |

According to Table X, the YOLOv8 fine-tuned model has the highest precision among these three series of YOLO models, indicating that it has the lowest false positive rate. However, the YOLOv8 fine-tuned model will be missing the most actual tumor cases since it has the lowest recall. For the YOLOv11 fine-tuned model, it has the most balanced precision and recall, as it has the highest F1 score among these three models. However, the precision of the YOLOv11 fine-tuned model is slightly lower than that of YOLOv8, indicating that the percentage of false positive cases among all detected positive cases is higher in YOLOv11 than in YOLOv8. On the other hand, the YOLOv12 fine-tuned model achieves the highest recall, suggesting that it is the most effective at detecting all tumor cases. However, it also has the lowest precision, meaning it produces the highest rate of false positives. In summary, the YOLOv8 fine-tuned model is the most precise, the YOLOv11 fine-tuned model offers the most balanced performance between precision and recall, and the YOLOv12 fine-tuned model is the most sensitive. Compared to YOLOv8 and YOLOv11, YOLOv12 is more suitable for detecting critical or high-risk tumor cases in clinical contexts, where missing a tumor could have more severe consequences than a false alarm.

1. Discussion

### Strengths and Weaknesses of the Proposed Methodology and Model

Data preprocessing is used on the dataset to ensure quality object detection. This is because data preprocessing ensures a clean, consistent, and structured dataset for model training. The data preprocessing techniques applied to the dataset include auto-orientation of pixel data with EXIF-orientation stripping, resizing image to a 640x640 pixel resolution with stretching, and filtering out images that do not contain any annotations. The strength of auto-orientation of pixel data with EXIF-orientation stripping is that it ensures a consistent display of images, regardless of whether the images were captured in landscape or portrait format. This is achieved by reading the EXIF tag and physically rotating the pixel so that the images are oriented in the same way. After rotation, the EXIF tag will be removed (Dwyer, 2020). However, the EXID tag has its weaknesses because it may remove useful metadata. The EXIF tag holds information like camera settings, exposure information, and timestamps (Mohith, 2023). Deletion this metadata may cause loss of essential information that could be useful for analysis. For the resizing technique, it is applied because Convolutional Neural Network (CNN) models usually require a fixed input size. A size of 640x640 is chosen so that the images can be seen clearly and ensure quality tumor detection. However, if a high-resolution image is downscaled, it may lose critical tumor features and make it harder to distinguish between tumors and noise. Besides, filtering images that do not contain any annotations is important to ensure the models only learn from relevant data, ensuring quality model training. Images without annotations may confuse the models during model learning. Nevertheless, dataset size and diversity of the training dataset are important to ensure the models are well-trained and provide accurate object detection results. Filtering out images without annotation may reduce dataset size and reduce dataset diversity, potentially affecting the models' generalization ability.

Data augmentation is applied to the training dataset to prevent overfitting. It enhances the robustness of image-based machine learning models and generalization ability. The data augmentation techniques applied to the dataset include horizontal flip, random zoom cropping (0% minimum to 20% maximum), horizontal and vertical shear (up to ±10º), and pixel-level noise affecting up to 0.1% of the image. During model training, horizontal flip is a useful technique to help models develop invariance to horizontal orientation, especially in tasks where the horizontal axis does not carry semantic meaning (SERP AI, n.d.). In real-world scenarios, input images may appear in various orientations, and horizontal flip helps the model generalize better to these variations. However, horizontal flip is not always applicable because it may cause semantic changes, especially in object detection tasks where the direction or orientation of an object is important. Furthermore, random zoom cropping is applied to help models generalize better. This is because the object(s) of interest to be learned by the models may not always be completely visible or appear at the same scale in the training data (Nelson, 2020). By exposing the models to different levels of zoom and partial views during training, they become more robust to different object sizes and framing conditions. However, if a cropped image contains an annotation that falls completely outside the frame, that annotation will be dropped (Nelson, 2020). This can weaken the models' learning process because every training data is valuable. Moreover, horizontal and vertical shear transformations are applied to the dataset to enhance models' robustness when dealing with distorted images. These augmentations simulate real-world scenarios where objects may appear rotated or skewed due to camera angles or perspective shifts (Ruman, 2023). On the other hand, horizontal and vertical shear have some limitations as it might cause objects to appear unnaturally in the images, increasing annotation complexity. Additionally, noise is introduced into the images to increase noise resilience. When dealing with imperfect or noisy data in the real world, the models can produce better performance. Yet, applying noise at 0.1% has some weaknesses. 0.1% is considered low noise level, as it will not make great contribution to the accuracy of object detection tasks (Li & Xu, 2025).

YOLOv8 shows its strengths for brain tumor detection. Firstly, its anchor-free detection approach allows for less complex model architecture and enhances performance on small objects (Chen et al., 2023). YOLOv8 unifies object localization and classification into a single, end-to-end differentiable neural network, balancing both speed and accuracy. In other words, it simplifies the prediction process, improves efficiency of hyperparameter tuning, and enhances the models' adaptability to objects with different ratios and scales. Furthermore, compared to its predecessors, YOLOv8 incorporates new data augmentation techniques that improve model generalization. The models can be exposed to varying object scales, orientations, and spatial configuration to increase their robustness to produce accurate object detection results (Yaseen, 2024). Besides, YOLOv8 supports a wide variety of visual AI tasks and introduces features, which are object tracking, pose estimation, and oriented bounding box (Ultralytics, n.d.). Moreover, YOLOv8 integrates bottleneck module, which reduces computational complexity and enhances feature reusability. This leads to reduced inference time and improved detection accuracy (Yaseen, 2024). However, YOLOv8 has its limitations when dealing with tumor detection tasks. The YOLOv8 model is highly dependent on the quality and variety of its training data and achieving comprehensive coverage is still a challenge. If the training dataset is imbalanced or biased, it may affect the accuracy of the prediction, leading to biased results. In addition, using YOLOv8 model may cause lack of contextual understanding between different regions because the model processes the entire image at once. Hence, this can result in misinterpretation and incorrect analysis, especially in tasks where the relationships between objects are essential for accurate detection (Nadeem, 2024).

YOLOv11 demonstrates significant strengths for brain tumor detection. To begin with, it includes the C3K2 (Cross Stage Partial with kernel size 2) block to preserve rich information by merging outputs from multiple bottlenecks. This can lead to better feature representation and gradient flow (Hui, 2025). Furthermore, its C2PSA (Cross-Stage Partial with Spatial Attention) attention mechanism contributes to the high spatial awareness and precision (Ali and Zhang, 2024). Hence, the model can focus on critical image regions within an image, such as small, overlapping, or partially occluded tumors. This is achieved by incorporating two PSA (Partial Spatial Attention) modules on separate branches of the feature map and are later concatenated. It uses multi-head attention and feedforward networks to prioritize positional aspects, strengthening spatial relationship understanding of the model (Hui, 2025). Moreover, its restructured backbone and neck architecture with smaller kernel sizes and the SPPF (Spatial Pyramid Pooling-Fast) module ensure fast and accurate feature aggregation for real-time medical diagnostics (Ali and Zhang, 2024). Next, it is highly adaptable to a wide range of applications, including cloud platforms, edge devices, and other systems with NVIDIA GPUs (Boesch, 2025).

Despite these advantages, YOLOv11 tends to face difficulty in detecting low-resolution or rotated tumors and overfit on limited medical datasets. On top of that, it can be challenging to deploy efficiently in resource-constrained environments, despite its lightweight design (Hui, 2025). This may limit its performance in more complex diagnostic scenarios.

YOLOv12 introduces a newer convolutional block class that enhances its tumor detection performance. One of its strengths lies in its lightweight and parallelized convolutional block that utilizes a series of smaller kernel (Alif & Hussain, 2025). This improves image processing efficiency while ensuring details of image are captured. This is extremely important in medical imaging, where it is going to detect small and irregularly shaped tumors. YOLOv12 also ensures quality feature extraction, allowing low-contract or inapparent tumors to be effectively identified, regardless of their location and size in the images.

Hyperparameter tuning is applied to the dataset to produce accurate predictions. It will also help balance speed and accuracy. By adjusting the setting when training models, the models can become more robust and can generalize better (Yolov8-Architecture). However, hyperparameter tuning usually has high time complexity and high computational complexity. It might have high requirements regarding the substantial processing power and hardware specifications used to perform effectively.

### Challenges and Potential Improvements

Firstly, the imbalanced dataset on certain classes could contribute to the low overall performance metrics for the models. There is a total of four different classes in our dataset, namely “NO\_tumor”, “glioma”, “meningioma”, “pituitary”, and “space-occupying lesion-”. However, dataset for training, validation, and testing have significantly lesser images for the class of “glioma” and “space-occupying lesion-”. To address this issue, one of the potential improvements is to explore class-weighted loss functions for the imbalanced classes according to the frequency of the classes. According to the standard practice, classes with higher frequency are assigned lower weights, while classes with lower frequency are assigned higher weights. In this way, the detection and classification of these minority classes might be performed better.

Secondly, the limitations of the free version of Google Colab, particularly the restricted access to GPUs like the G4 is one of the challenges. It significantly hinders the efficiency of training models and evaluating them on validation and testing datasets. The possible solution will be switching to alternatives which can offer longer GPU usage time without abrupt disconnections. Moreover, investing in Google Colab Pro or using free cloud credits from platforms like AWS can offer more powerful GPUs with extended runtimes.

Thirdly, best model selection is another major challenge. Choosing the right YOLO variant (n, s, m, l) for different versions of models is quite difficult, especially without in-depth knowledge or sufficient time to test each option. To tackle this, training each variant for a small number of epochs and then evaluating their validation and testing performance can greatly reduce the process time. Besides that, referring to the official YOLO documentation and performance benchmarks can help in guiding the selection of the most suitable model which strikes a balance between speed and accuracy.

1. Conclusion

In conclusion, this study has implemented various YOLO versions to compare and identify the best model for intracranial tumor detection. The evaluation showed that YOLOv8 has the most precise results, suitable for scenarios where minimizing false positives is critical. YOLOv11 offered the most balanced performance between precision and recall, making it suitable for tumor detection tasks where both false positives and false negatives must be controlled. For YOLOv12, it has the highest sensitivity, making it suitable for critical clinical applications, provided that the training dataset is balanced.

1. Contribution

|  |  |
| --- | --- |
| Name | Contributions |
| Lim Cammy | * Wrote code for YOLOv8 and YOLOv12 * Wrote the challenges part of the report |
| Liow Ke Han | * Wrote the abstract, strengths and weaknesses of methodology, and conclusion part of the report |
| Wong Yu Chi | * Wrote code for YOLOv11 * Wrote the results and discussion part of the report |
| Leon Siow Yi Hong | * Fine-tuned hyperparameters. * Wrote the methodology and introduction part of the report |

1. References

Baptist Health. (n.d.). Benign vs. Malignant Tumors. [online] Available at: https://www.baptisthealth.com/blog/cancer-care/benign-vs-malignant-tumors

Boesch, G. (2025) *Yolo11: A new iteration of ‘You only look once’*, *viso.ai*. [online] Available at: https://viso.ai/computer-vision/yolov11/.

Chen, Y., Yuan, X., Wu, R., Wang, J., Hou, Q. and Cheng, M.-M. (2023). YOLO-MS: Rethinking Multi-Scale Representation Learning for Real-time Object Detection. *arXiv (Cornell University)*. doi:https://doi.org/10.48550/arxiv.2308.05480.

detection, brain tumor (2024). *Tumor Detection Dataset*. [online] Roboflow. Available at: https://universe.roboflow.com/brain-tumor-detection-wsera/tumor-detection-ko5jp.

Hui, O.Y. (2025) *Walkthrough: Training yolov11 for brain tumor detection*, *Medium*. Available at: https://medium.com/data-and-beyond/walkthrough-training-yolov11-for-brain-tumor-detection-a194575f3339.

Kelta, Z. (2022). *YOLO Object Detection Explained: A Beginner’s Guide*. [online] www.datacamp.com. Available at: https://www.datacamp.com/blog/yolo-object-detection-explained

Li, H. and Xu, K. (2025). Innovative adaptive edge detection for noisy images using wavelet and Gaussian method. *Scientific Reports*, [online] 15(1). doi:https://doi.org/10.1038/s41598-025-86860-9.

Dwyer, B. (2020). *When Should I Auto-Orient My Images?* [online] Roboflow Blog. Available at: https://blog.roboflow.com/exif-auto-orientation/.

Mohith (2023). *Understanding EXIF tags and fields - EXIF Viewer by Fluntro*. [online] EXIF Viewer by Fluntro. Available at: https://exifviewerapp.com/understanding-exif-tags-and-fields/

Nadeem, M.A. (2024). *Unraveling YOLO v8: Benefits and Challenges in Object Detection*. [online] Medium. Available at: https://medium.com/@masadnadeem23/unraveling-yolo-v8-benefits-and-challenges-in-object-detection-1ec762debefd.

Nelson, J. (2020). *Why and How to Implement Random Crop Data Augmentation*. [online] Available at: https://blog.roboflow.com/why-and-how-to-implement-random-crop-data-augmentation/.

Patel, S., Kadam, Y., Abhishek Thombare, Salvi, K. and Jadhav, D.D. (2024). A Review of Brain Tumor Detection Techniques Using YOLOv8. *International Journal for Research in Applied Science and-Engineering Technology*, 12(3), pp.1075–1078. doi:https://doi.org/10.22214/ijraset.2024.58666

Rabbani, A. and Hussain, M. (2025). YOLOv12: A Breakdown of the Key Architectural Features. *arXiv (Cornell University)*. doi:https://doi.org/10.48550/arxiv.2502.14740.

Rao, S.N. (2024). *YOLOv11 Architecture Explained: Next-Level Object Detection with Enhanced Speed and Accuracy*. [online] Medium. Available at: https://medium.com/@nikhil-rao-20/yolov11-explained-next-level-object-detection-with-enhanced-speed-and-accuracy-2dbe2d376f71

Ruman (2023). *YOLO Data Augmentation Explained*. [online] Medium. Available at: https://rumn.medium.com/yolo-data-augmentation-explained-turbocharge-your-object-detection-model-94c33278303a

SERP AI. (2025). *Random Horizontal Flip: Data Augmentation Technique for Image-Based Machine Learning | SERP AI*. [online] Available at: https://serp.ai/posts/random-horizontal-flip/

Svitla Systems. (2024). *Understanding CNN for Image Processing | Svitla Systems*. [online] Available at: https://svitla.com/blog/cnn-for-image-processing/

Ultralytics (n.d.). *YOLO Performance Metrics*. [online] docs.ultralytics.com. Available at: https://docs.ultralytics.com/guides/yolo-performance-metrics/#class-wise-metrics.

Yaseen, M (2015). *What is YOLOv8: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector*. [online] Available at: https://arxiv.org/html/2408.15857.

Yolov8-Architecture. (2025). YOLOv8 Hyperparameter Tuning | Optimize Model Performance. [online] Available at: https://yolov8architecture.com/yolov8-hyperparameter-tuning/.