IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Deep Learning Object Detection Algorithm for Brain Tumor Identification and Categorization

[1]Nitin Prajapati, [2]Raj Gupta, [3]Shivam Singh, [4]Dr. Bhawesh K. Thakur
[1]Student B.Tech. (IT-4th Yr.), [2] Student B.Tech. (IT-4th Yr.), [3] Student B.Tech. (IT-4th Yr.), [3] Professor and Head,
[1]Department of Information Technology,
[1] Babu Banarasi Das Northern India Institute of Technology, Lucknow, India

Abstract: Brain tumors represent a significant global health challenge, affecting approximately one million individuals worldwide in 2023. This study evaluates the efficacy of the object detection algorithm for the detection and categorization of brain tumors using MRI images. The evaluation utilized a dataset of MRI images from different patients, covering three primary brain tumor types: Pituitary, Meningioma, and Glioma. The model demonstrated robust performance, with general high precision, recall, and mAP values on the validation dataset. Notably, the model effectively differentiated between the various brain tumor types and background, as evidenced by the normalized confusion matrix. Practical testing on brain MRI images confirmed the model's ability to accurately identify different tumor regions. In conclusion, the object detection algorithm presents a promising automated method for identifying brain tumors and classification in MRI images, potentially aiding clinicians in precise diagnosis and treatment planning. Further research using larger datasets is recommended to enhance the model's clinical applicability and reliability.

***Index Terms* - brain tumor, object detection, MRI.**

I. INTRODUCTION

A brain tumor is characterized as an aberrant proliferation of cells within or in proximity to the cerebral region. Primary tumors arise from the brain tissue itself, whereas secondary or metastatic tumors originate from malignant cells that have disseminated from other bodily sites. These growths are categorized as either non-cancerous, with no harmful potential, or cancerous, possessing the capacity to infiltrate and compromise adjacent brain tissue.

Brain tumors are a pressing global health issue, affecting an estimated 1 million people worldwide. In 2023, there were 94,390 new primary brain tumor diagnoses, comprising 67,440 non-malignant and 26,940 malignant cases. Meningiomas are the most common benign tumors, representing 39.7% of all cases, while glioblastoma is the most prevalent malignant tumor, accounting for 14.2% of diagnoses. Over a five-year period, 76% of patients survive, with a median diagnostic age of 61. However, survival rates vary significantly by tumor type, with non-malignant tumors having a 91.8% survival rate and malignant tumors only 35.7%. With a median survival of eight months, glioblastoma has a low five-year survival rate of 6.9%. It was ranked 10th globally in terms of cancer fatalities in 2023.

Pediatrics brain tumors constitute 3.9% of all cases, with an estimated 3,920 new cases in 2023. They are the most diagnosed solid cancer in children and the top cause of cancer death in this age group. Childhood brain tumors have an 83.1% five-year survival rate, varying by tumor type, with pilocytic astrocytoma having the highest survival rates. Adolescents and young adults (AYA) represent 14.3% of primary brain tumors, with pituitary tumors being the most common and a five-year survival rate of 90.9%.

81.7% of primary brain tumors occur in adults 40 years of age and older; 79,340 new cases are predicted in 2023. The relative survival rate after five years is 72.5%, brain tumors rank sixth among tumor types and the seventh most common and leading cause of cancer-related mortality within this demographic.

Regarding race and ethnicity, Black individuals have slightly higher incidence rates, while White individuals have higher rates of specific tumor types like glioblastoma. Hispanic ethnicity is associated with improved survival rates. Brain tumor incidence rates are higher in females, with meningioma in females and glioblastoma more common in males. more common in males and meningioma in females. Despite higher incidence rates in females, males generally have worse survival outcomes from malignant brain tumors.

Brain tumors, affecting the brain's control center for movement, emotion, cognition, and sensory perception, can severely disrupt these vital functions.

Treatment methods for brain tumors focus on controlling tumors with minimal damage to healthy brain tissue through surgery, radiation therapy, and chemotherapy. Physical, occupational, and speech therapies are examples of rehabilitation therapies that help affected people regain or maximize their functional abilities and enhance their quality of life.

Recent interest in artificial intelligence has led to advancements in A computer vision approach called object detection locates and recognizes items within digital pictures or video frames. Identifying objects, establishing their locations using bounding boxes, and labelling them with class identifiers are all part of object detection. The You Only Look Once (YOLO) object identification approach is utilized in this study to identify, locate, and categorize brain tumors from MRI data. YOLO's efficiency, real-time performance, and high accuracy make it a suitable choice for this task.

II. LITERATURE REVIEW

This section examines the most recent developments in machine learning-based brain tumor detection.

Multi-layer perceptron (MLP) and naïve Bayes classification algorithms were investigated in a study by Sharma, K., Kaur, A., & Gujral, S. to distinguish malignant from benign brain tumors. The study focused on analysing textural data from medical pictures to obtain accurate tumor classification.

Xuan, X., & Liao, Q. developed a technique for extracting characteristics from brain images, such as texture, symmetry, and intensity. They utilized the AdaBoost algorithm to classify MR images as normal or abnormal, achieving a classification accuracy of 96.82%.

In a different study by Amin, S.E., & Mageed, M., neural networks were used to identify three different forms of brain cancer from MRI pictures: acoustic glioma, optic glioma, and astrocyte glioma. Using a dataset of 30 MRI images, the researchers were able to achieve an average detection rate of 78%.

Jafari, M., & Shafaghi, R. presented a hybrid method for classifying brain images that combines support vector machines (SVM) with genetic algorithms. Utilizing wavelet, statistical, and frequency transformation features, this method produced an average accuracy of 83.22%.

Mercaldo, Brunese, Martinelli, Santone, and Cesarelli proposed an automated method for detecting and localizing brain cancer using magnetic resonance imaging (MRI) analysis. They utilized the YOLOv8 (You Only Look Once) model for automated detection and localization. In their analysis of 300 brain images, they achieved a precision of 0.943 and a recall of 0.923 for brain cancer detection, along with a mean Average Precision (mAP) of 0.941 for brain cancer localization.

Selvy, P.T., Dharani, V., & Indhuja, A. experimented with different segmentation algorithms and proposed a method to detect brain pictures connected to cancer. They used a two-step process for segmentation and classification.

Montalbo proposes transfer learning and fine-tuning for YOLOv4 to detect brain tumors from MRI scans. Trained on a dataset of 3064 MRI scans, the YOLOv4-Tiny model achieved a mean average precision of 93.14%, outperforming previous versions. This work aids in automating brain tumor detection to assist medical experts in diagnostics.

In their study, Arunachalam & Sethumathavan propose an enhanced YOLOV5 technique for brain tumor detection using MR images, achieving a notable mAP of 90.24%. They employ Hyperparameter Optimization and McCulloch's algorithm for segmentation accuracy evaluation, outperforming standard YOLOV5 models. Their method demonstrates potential for efficient brain tumor identification in medical image analysis.

Abdusalomov, A.B., Mukhiddinov, M., and Whangbo, T.K. addressed brain tumor detection in MRI scans using a YOLOv7 model fine-tuned through transfer learning. Their approach improved performance in identifying gliomas, meningioma, and pituitary tumors, achieving a remarkable 99.5% accuracy. Further research is warranted for detecting small tumors, aiming to enhance diagnostic capabilities and support medical practitioners in battling brain cancers.

III. METHODOLOGY

This section outlines our method for using brain MRI scans to automatically detect and locate brain tumors. Our focus is on brain MRIs, but the approach we suggest can be modified to analyze other biomedical images, such as those of the lungs or other organs. A high-quality dataset with accurate tumor region annotations is essential for a brain tumor detection and localization model from MRI scans. The foundation for training an accurate object detection model is this well-annotated dataset. It makes sure the model picks up on the ability to locate and identify cancer-specific areas in MRI scans, allowing for accurate predictions on fresh, unobserved data.

To create a reliable and accurate model that can forecast unseen images, we compiled an extensive image dataset of several forms of brain tumors, including gliomas, meningiomas, and pituitary tumors, taken from different perspectives and in varied lighting conditions. Resizing each image to a uniform size requires preprocessing to prepare them for additional analysis. Upon acquiring the pictures, we established three categories for the identification of the bounding boxes that are specifically focused in the presence of these Pituitary, Meningioma and Glioma brain tumor.

Annotation of every MRI picture with bounding boxes around any tumors that were found. The LabelImg tool, which was created especially for data annotation tasks, was used to carry out this annotation process.

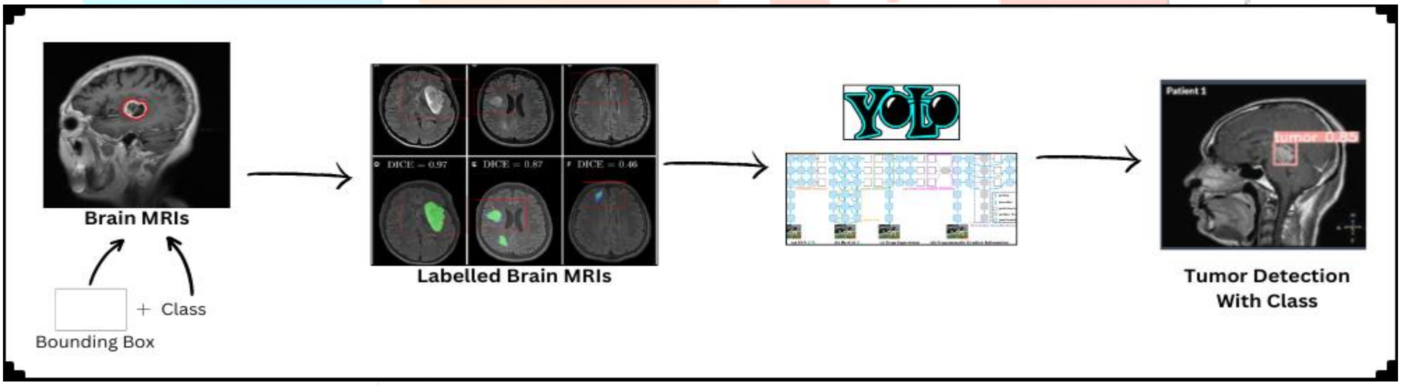


Figure 1 The work flow of the implemented method.

We use the YOLO model for object detection in this study. First presented in 2016 [24] by J. Redmon et al., YOLO is a ground-breaking one-stage deep learning detector. It is specifically made to be able to carry out accurate object localization and image classification at the same time.

One characteristic that distinguishes YOLO from other networks is its independent pipeline, which manages the entire process on its own. An image is used as the input in YOLO, and the output is made up of two primary parts: for each cell, a vector representing a bounding box and the corresponding class prediction.

Every image is separated into a S × S grid of cells during the analysis. When an object lies in the centre of a cell, that cell oversees identifying it. The five components of the bounding box prediction are (x, y, w, h, confidence). The (x, y) coordinates show the centre of the enclosing box with respect to the cell's grid location, which are normalized to values between 0 and 1. In relation to the image size, the box's size (w, h) is normalized to values between 0 and 1. In total, $S \times S \times B \times 5$ outputs are produced by the bounding box predictions, where B signifies the quantity of predicted bounding boxes for each cell.

Compared to current object detection models, YOLO has established exceptional speed. The main reason for this efficiency is that YOLO can complete the recognition task in one step, negating the need for additional phases. YOLO forecasts classifications of things present in the input image input image, bounding boxes, and object probabilities directly, in contrast to models that have distinct stages for object classification and region proposal. In contrast to alternative models for object detection, YOLO's speed is greatly increased by this simplified methodology. It is important to mention that the YOLO model has several versions. For this research, we focused on and implemented YOLOv9c using the ultralytics framework.

We chose the YOLOv9c model because it significantly improves upon previous YOLO versions in several areas, including the following:

- ***Programmable Gradient Information (PGI):*** Designed to combat information loss in deep neural networks, PGI aids in preserving essential data across the network. This ensures reliable gradient generation, enhancing model convergence and overall performance.
- ***Reversible Functions:*** These are vital for maintaining an uninterrupted flow of information within deep learning models. By leveraging reversible functions, YOLOv9c prevents information degradation, particularly in the network's deeper layers, ensuring the preservation of critical data for accurate object detection.
- ***Generalized Efficient Layer Aggregation Network (GELAN):*** YOLOv9c includes the Generalized Efficient Layer Aggregation Network (GELAN), an architectural breakthrough that allows for better parameter utilization and computational efficiency. Because of its flexible design, which enables the integration of different computational blocks, the model is applicable to a variety of scenarios without compromising accuracy or speed.

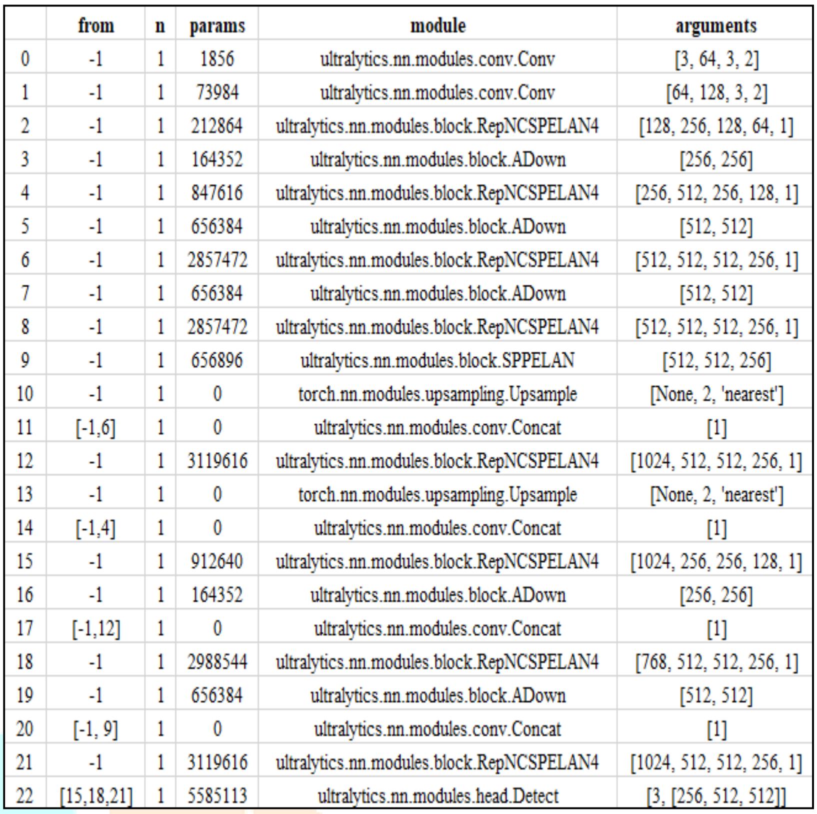
Together, these features—PGI, Reversible Functions and GELAN—enable YOLOv9c to achieve exceptional accuracy, efficiency, and adaptability in real-time object detection, setting new benchmarks in the field.

In classification of medical images, there are many advantages and benefits to using YOLO:

- ***Real-time Detection:*** Yolo makes it possible to quickly spot anomalies in medical images, which speeds up diagnosis and treatment preparation.
- ***Accurate Object Localization:*** YOLO helps with precise diagnosis by precisely localizing abnormalities in addition to classifying them.
- ***Multi-class handling:*** the ability to identify and categorize several medical disorders from an image.
- ***Effective Single-pass Method:*** YOLO's single-pass architecture improves computational efficiency and speed, making it perfect for big datasets of medical images.
- ***Adaptive Transfer Learning:*** YOLO uses a limited amount of labelled data to adapt general image features to medical specific images by using pre-trained models for extracting features.

The two primary parts of the YOLOv9c architecture are the head and the backbone, each of which is essential to the overall object detection procedure.

The key purpose of the backbone is to figure out crucial features from the input image. It begins with an initial placeholder layer that sets the foundation for subsequent operations. Following this, the architecture employs two convolutional down sampling layers, with the first down sampling the input image by a factor of two and the second further reducing its size. These down sampling stages are crucial for extracting hierarchical features of different scales from the input. Subsequently, the architecture incorporates the ELAN-1 block, a component that leverages the GELAN architecture to process the extracted features. This block is followed by an average and convolutional down sampling layer that further refines the feature map. The backbone then employs two ELAN-2 blocks, each with its specific channel configurations, to perform more intricate feature extraction and processing. Each of these blocks is followed by an average and convolutional down sampling layer, further reducing the feature map size, and enhancing its representational power.



	from	n	params	module	arguments
0	-1	1	1856	ultralytics.nn.modules.conv.Conv	[3, 64, 3, 2]
1	-1	1	73984	ultralytics.nn.modules.conv.Conv	[64, 128, 3, 2]
2	-1	1	212864	ultralytics.nn.modules.block.RepNCSPELAN4	[128, 256, 128, 64, 1]
3	-1	1	164352	ultralytics.nn.modules.block.ADown	[256, 256]
4	-1	1	847616	ultralytics.nn.modules.block.RepNCSPELAN4	[256, 512, 256, 128, 1]
5	-1	1	656384	ultralytics.nn.modules.block.ADown	[512, 512]
6	-1	1	2857472	ultralytics.nn.modules.block.RepNCSPELAN4	[512, 512, 512, 256, 1]
7	-1	1	656384	ultralytics.nn.modules.block.ADown	[512, 512]
8	-1	1	2857472	ultralytics.nn.modules.block.RepNCSPELAN4	[512, 512, 512, 256, 1]
9	-1	1	656896	ultralytics.nn.modules.block.SPPELAN	[512, 512, 256]
10	-1	1	0	torch.nn.modules.upsampling.Upsample	[None, 2, 'nearest']
11	[-1,6]	1	0	ultralytics.nn.modules.conv.Concat	[1]
12	-1	1	3119616	ultralytics.nn.modules.block.RepNCSPELAN4	[1024, 512, 512, 256, 1]
13	-1	1	0	torch.nn.modules.upsampling.Upsample	[None, 2, 'nearest']
14	[-1,4]	1	0	ultralytics.nn.modules.conv.Concat	[1]
15	-1	1	912640	ultralytics.nn.modules.block.RepNCSPELAN4	[1024, 256, 256, 128, 1]
16	-1	1	164352	ultralytics.nn.modules.block.ADown	[256, 256]
17	[-1,12]	1	0	ultralytics.nn.modules.conv.Concat	[1]
18	-1	1	2988544	ultralytics.nn.modules.block.RepNCSPELAN4	[768, 512, 512, 256, 1]
19	-1	1	656384	ultralytics.nn.modules.block.ADown	[512, 512]
20	[-1, 9]	1	0	ultralytics.nn.modules.conv.Concat	[1]
21	-1	1	3119616	ultralytics.nn.modules.block.RepNCSPELAN4	[1024, 512, 512, 256, 1]
22	[15,18,21]	1	5585113	ultralytics.nn.modules.head.Detect	[3, [256, 512, 512]]

Figure 2 The architecture of YOLOv9c

The head of the YOLOv9 architecture processes the features extracted by the backbone to perform object detection. It begins with the ELAN-SPP block, which integrates the GELAN architecture with Spatial Pyramid Pooling (SPP) to capture features at multiple scales and improve the model's capacity to detect objects of varying sizes. This block is followed by an up sample and concatenation operation, which up samples the feature map and merges it with the corresponding feature map from the backbone. The head then incorporates two more ELAN-2 blocks, each with specific channel configurations, to perform further feature extraction and processing. The feature maps generated by these blocks are then averaged, down sampled, and fused with the features from the auxiliary branches at P4 and P5 levels. Finally, the detection head, using the features from both the backbone and the head, performs the final object detection, categorizing objects into the specified number of classes.

In summary, the backbone focuses on feature extraction and down sampling, extracting hierarchical and intricate features from the input image. In contrast, the head further refines these features, performs additional feature extraction, and carries out the final object detection, ultimately providing the final object detection results with improved accuracy and efficiency. The architecture of YOLOv9c is show in Figure 2.

The YOLOv9c model can perform the following tasks on unseen brain MRIs after it has been trained:

- Determine whether a brain tumor is present and where it is (i.e., localize).
- Identify the type of tumor: meningioma, glioma, or pituitary.
- Assign a tumor type probability.

IV. EXPERIMENTAL SETUP AND ANALYSIS

In this section, we present the experimental setup and outcomes of our experimental analysis, showcasing the ability of the suggested YOLO model to locate and identify brain tumor.

4.1 Dataset

For this investigation, a large set of data was used. Comprising 233 individuals' 3064 T1-weighted, contrast-enhanced MRI images. Three distinct forms of brain cancers were depicted in these images: a pituitary tumor (930 slices), a meningioma (708 slices), and a glioma (1426 sections). A total of 2124 images were selected at random from each tumor class to ensure a balanced representation. The MATLAB (.mat) files that collectively make up the dataset have a struct that has fields for each individual image. The MRI image data is stored in cjdata.image, cjdata.PID represents the patient ID, cjdata.tumorBorder stores the coordinates of discrete points outlining the tumor border, and cjdata.tumorMask is a binary image delineating the tumor region with 1s. These fields include cjdata.label, which indicates the tumor type with values 1 for meningioma, 2 for glioma, and 3 for pituitary tumor. 80% of the dataset is used for training, 10% for validation, and the remaining 10% for testing to ensure a strong and all-encompassing framework for brain tumor localization and detection during model training and evaluation. The original source of the dataset was Figshare. In Figure 3, some example brain MRIs has been shown.

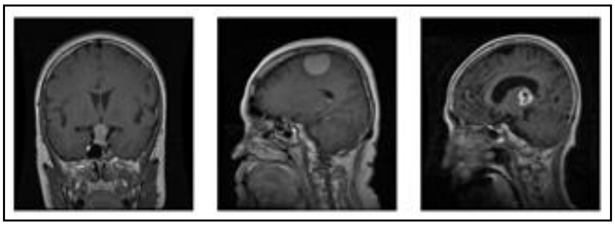


Figure 3 Brain MRIs with tumor

4.2 Data Preprocessing and Annotation

The original images in the dataset are of dimensions 512x512 pixels. To ensure compatibility with the training model, all images were resized to 640x640 pixels.

Each MRI image was annotated by delineating bounding boxes around the detected tumors. We employed the LabelImg tool, a specialized tool for data annotation tasks, to carry out this process. The annotations for each image were saved in .txt file format.

Each.txt file in the YOLO annotation format corresponds to an image and has a line for each object found in that image. The object class, the normalized height and width of the box, and the normalized x and y coordinates of the box's center make up the five space-separated values that make up the structure of the line. The integer indices for the object classes in the brain tumor detection task are as follows: 0 for pituitary, 1 for meningioma, and 2 for glioma tumor.

4.3 Model Training

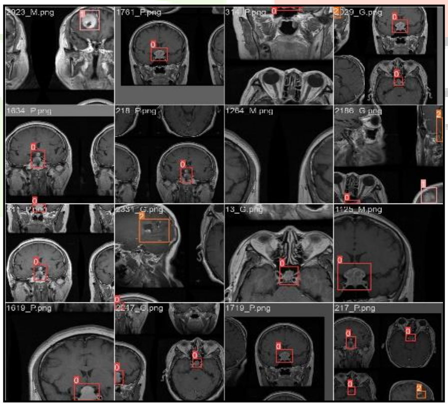


Figure 4 Training batch 1 image.

In this study, we employed the Ultralytics YOLOv8.2.2 framework, implemented using Python version 3.10.12 and Torch version 2.2.1+cu121, for the task of object detection. The model was trained on an NVIDIA L4 GPU with 22,700MiB of memory and was configured using the yolov9c.yaml model configuration. The model was trained using a batch size of 32 on 500 epochs during the training phase and was trained on images resized to 640x640 pixels. The model utilized CUDA device 0 with 8 workers for data loading and processing (workers=8). To reduce overfitting, the optimization was carried out with the SGD algorithm at a momentum of 0.9, weight decay of 0.0005, and learning rate of 0.01. The initial three epochs were utilized for warm-up to stabilize the learning process. To ensure a balanced loss function, the following loss weights were incorporated: a box loss weight of 7.5, a class loss weight of 0.5, an objectness loss weight of 1.5 (DFL Loss), a pose loss weight of 12.0, an object loss weight of 1.0 (Kobj Loss), and a label smoothing of 0.0.

The training dataset comprised a collection of labelled images, while the validation set was distinct and utilized for assessing model performance. To enhance the robustness of the model, various data augmentation techniques were applied. These included HSV adjustments with h=0.015, s=0.7, and v=0.4, as well as 0.1 translation, 0.5 scaling, and a 0.4 erasing factor. The training process involved the use of Automatic Mixed Precision (AMP) to accelerate training while maintaining model accuracy. Additionally, the training process featured plotting and validation functionalities, with a confidence threshold set at 0.7, an IoU threshold of 0.7, and a maximum detection limit of 300. The validation set consisted of 211 images. The model architecture was detailed with 618 layers, 25,531,545 parameters, 25,531,529 gradients, and an operation efficiency of 103.7 GFLOPs.

Due to the lack of improvement seen in the final 100 epochs, early stopping was used, ending the training at 285 epochs.

4.4 Result

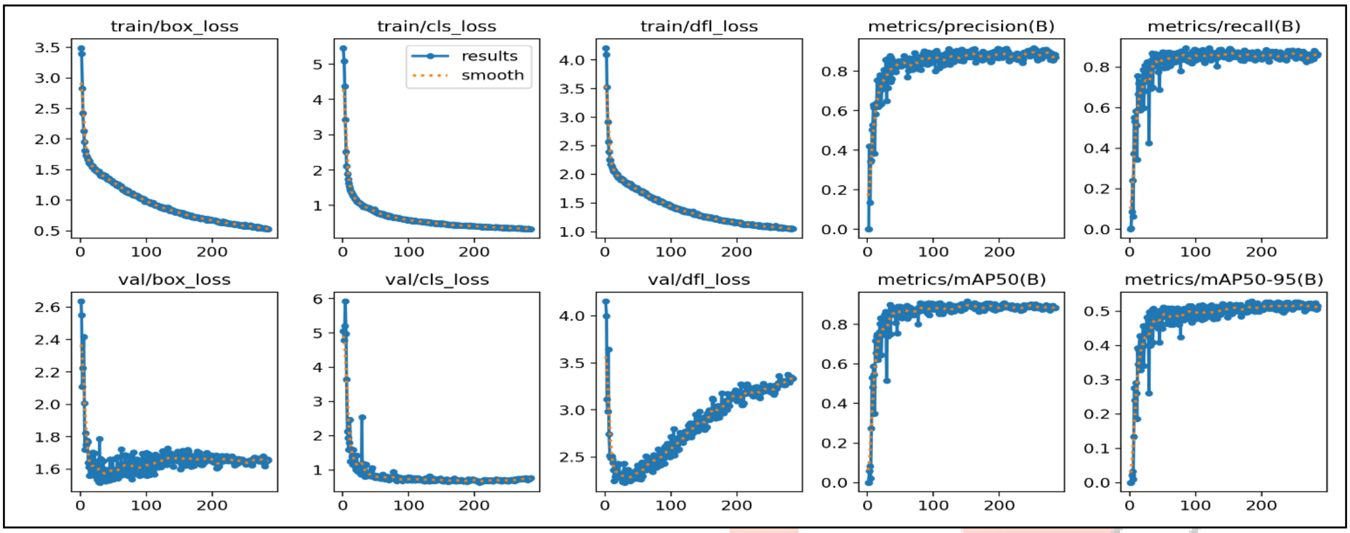


Figure 5 Result

The training process was stopped early because there was no progress seen in the final 100 epochs; epoch 185 produced the best results. In 4.529 hours, the 285 epochs of training were finished.

The validation of the best model (best.pt) was performed using the Ultralytics YOLOv8.2.2 framework, with the following summary: a fused architecture of 384 layers, 25,321,561 parameters, 0 gradients, and an operation efficiency of 102.3 GFLOPs. For the overall detection performance on the validation set comprising 211 images, the model reached a precision (P) of 0.871, recall (R) of 0.868, mAP50 of 0.906, and mAP50-95 of 0.529. When evaluated on individual classes, the Pituitary class showed a precision of 0.852, recall of 0.870, mAP50 of 0.900, and mAP50-95 of 0.492. The Meningioma class exhibited a precision of 0.955, recall of 0.986, mAP50 of 0.991, and mAP50-95 of 0.648. Lastly, the Glioma class had a precision of 0.804, recall of 0.750, mAP50 of 0.827, and mAP50-95 of 0.447. The processing speed per image during inference was recorded at 9.7ms, with 0.2ms for preprocessing and 1.4ms for post-processing.

Table 4.4 Model Performance

Metric	**Overall**	**Pituitary**	**Meningioma**	**Glioma**
Precision (P)	0.871	0.852	0.955	0.804
Recall (R)	0.868	0.870	0.986	0.750
mAP50	0.906	0.900	0.991	0.827
mAP50-95	0.529	0.492	0.648	0.447

In Figure 5, the first plot, known as the Box plot, depicts the trend of the box loss metric during training for brain tumor localization and classification into Pituitary, Meningioma, and Glioma. The plot shows the training steps, with the epochs on the x-axis and the loss values on the y-axis. In the context of brain tumor detection, which involves both the localization and classification of tumors within medical images, bounding boxes are utilized as the primary method for pinpointing the tumor locations. The employed loss function calculates the disparity between predicted and true bounding boxes, striving to minimize it across epochs for improvement.

In Figure 5, the second plot illustrates the objectness loss specific to the task of brain tumor localization and classification into Pituitary, Meningioma, and Glioma. This plot depicts the model's confidence in identifying the existence of a tumor within a specified bounding box, known as objectness. Concurrently, the class score reflects the likelihood of the tumor belonging to a particular class—Pituitary, Meningioma, or Glioma. Multiplying the objectness and class score yields the overall confidence score for each class. As training progresses, it is preferable for the objectness to diminish towards zero, indicating the model's increasing confidence in precisely localizing the tumors. Thus, the plot delineates the trend of objectness scores across epochs during the training phase.

The third plot in Figure 5 focuses on the classification feature of the brain tumor detection task. During the training process, this plot highlights the classification of the detected tumors into the specific categories of Pituitary, Meningioma, and Glioma. The primary goal of classification is two-fold: firstly, to ascertain the presence of a tumor within the image, and secondly, to correctly identify the type of tumor. The plot displays the loss associated with this classification, which evaluates the accuracy of classifying each predicted bounding box into one of the tumor categories or as "background". For classification tasks, the cross-entropy loss is typically used. The plot shows how this loss changes over epochs, reflecting the model's learning progress in accurately classifying tumors.

For the validation dataset, trends in box, objectness, and classification losses are displayed in validation plots. As the number of epochs rises, we expect these metrics to decline similarly to the training plots. The precision and recall metrics for each epoch are plotted in the fourth and fifth plots, Recall and Precision, which offer information about the accuracy and completeness of the model in identifying and categorizing pituitary, meningioma, and glioma brain tumors.

Precision is measured to assess how well the model predicts positive outcomes when classifying brain tumors as Glioma, Meningioma, and Pituitary. It considers false positives, or cases that are mistakenly classified as a particular type of tumor. Precision can be calculated as:

$$Precision = TP / (TP + FP) \qquad (4.1)$$

In this context, TP represents True Positives, while FP stands for False Positives.

Recall, also known as sensitivity or the true positive rate. Assesses the percentage of real positive brain tumor cases – like pituitary, meningioma, and glioma – that the model correctly identified. False negative are cases that are mistakenly classified as positive. It takes this into account. Recall can be calculated as:

$$Recall=TP / (TP + FN) \qquad (4.2)$$

In this context, TP represents True Positives, while FP stands for False Negatives.

Recall and precision should both trend upward as the number of epochs rises. The graphs showing these metrics clearly show the anticipated trend. Notably, when categorising brain tumors into pituitary, meningioma, and glioma categories, precision and recall values between 0 and 1 are thought to be ideal. The deep learning model has acquired knowledge and performed well in accurately classifying the various types of brain tumors, as evidenced by higher values for recall and precision in the last epochs.

A crucial metric for evaluating object detection models' accuracy is Average Precision (AP). It uses recall values which varies between 0 to 1 to compute the average precision. This metric is useful for assessing the efficacy of object detection algorithms because it provides a thorough analysis of the model's functionality at various recall levels.

The composite metric known as Mean Average Precision (mAP) comprises multiple essential elements, such as Precision, recall, precision-recall curve, intersection over union (IoU), and average precision (AP).

The standardized metric known as IoU is used to assess the overlap between actual and predicted bounding boxes. It is calculated as the ratio of two bounding boxes’ union area to intersection area, given by the formula:

$$IoU = Area\ of\ Intersection\ /\ Area\ of\ Union \qquad (4.3)$$

This measure is essential for assessing how well bounding box predictions perform in object detection tasks.

The trade-off between precision and recall across various classification thresholds is illustrated by the Precision-Recall curve in Figure 5. This curve provides information about how well the model performs at different operating points.

The mean precision, or mAP, is determined at every point on the Precision–Recall curve. It gives an overview of the model's general object detection performance.
.

The mAP values for two distinct IoU thresholds—0.5 and a range from 0.5 to 0.95—are displayed in Table 4.4. The precision obtained by applying an IoU threshold of 0.5 is denoted by mAP@0.5. This indicates that the detection is considered successful if there is a greater than 50% overlap between the predicted and ground truth bounding boxes. Accurate bounding box detection becomes more difficult as the IOU threshold rises. Therefore, compared to mAP@0.5, a higher IOU threshold typically yields a lower mAP value, suggesting that accurate bounding box detection will be more challenging to achieve with a stricter overlap requirement.

From Table 4.4, the validation results reveal compelling precision and recall metrics. Specifically, the overall precision is recorded at 0.871, with the individual class values being 0.852 for Pituitary, 0.955 for Meningioma, and 0.804 for Glioma. The overall recall stands at 0.868, distributed as 0.870 for Pituitary, an impressive 0.986 for Meningioma, and 0.750 for Glioma. Additionally, the mAP@0.5 demonstrates strong performance, registering at 0.906 overall, with class-specific values of 0.900 for Pituitary, 0.991 for Meningioma, and 0.827 for Glioma. The mAP@0.5:0.95, however, is comparatively lower at 0.529 overall, with values of 0.492 for Pituitary, 0.648 for Meningioma, and 0.447 for Glioma.

Figure 6 shows the precision-recall plot which provide a closer look at the performance that was attained in terms of precision and recall. We can assess the accuracy and comprehensiveness of the model's performance by looking at these values. A low false positive rate is indicated by a high precision value, and a low false negative rate is implied by a high recall value. As a result, creating an object detection model that works well requires striking a balance between recall and precision.

Because precision and recall are inherently trade-offs in object detection tasks, the precision-recall plot usually shows a monotonically decreasing trend. Since only the most certain predictions are included, increasing the threshold for classification as a positive prediction has the potential to improve precision at the expense of recall. Conversely, by including more false positives, recall may rise if the threshold is lowered to incorporate more predictions at the expense of decreasing precision. However, there may be anomalies to this pattern brought about by things like imbalances in the classes, anomalies, or peculiarities in the dataset. Even though a monotonically declining trend is typical, each situation calls for a careful examination of the unique characteristics of the precision-recall graph.

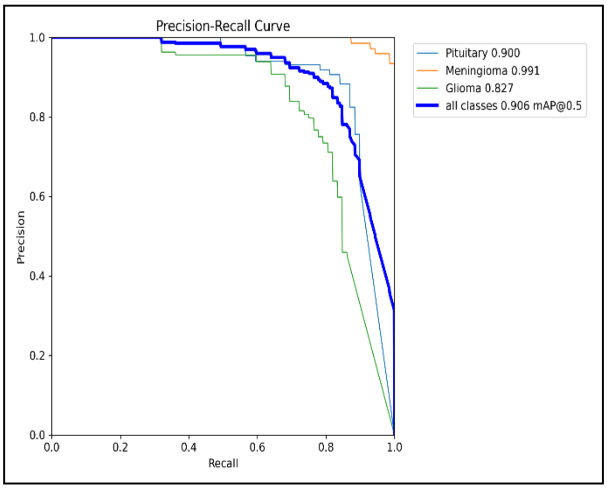


Figure 6 Precision-Recall curve

Examining Figure 6, the precision-recall plot indeed showcases a decreasing trend, aligning with the typical behaviour expected. The following are the area under the curve (AUC) values for the precision-recall curve associated with brain tumor detection: Meningioma (0.991), Glioma (0.827), Pituitary (0.900), and an overall average (0.906) (with an AUC mAP@0.5).

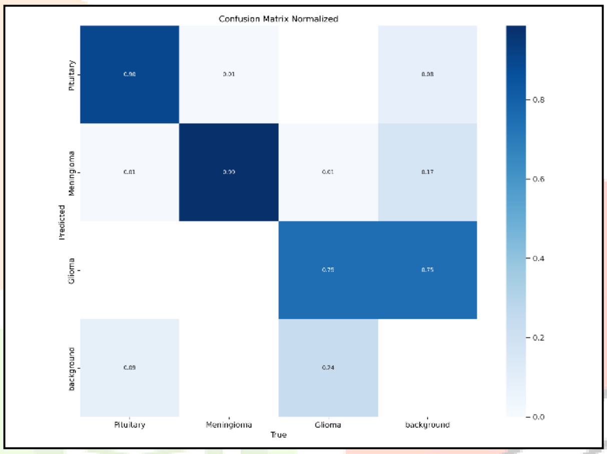


Figure 7 Normalized confusion matrix

Figure 7 displays the normalized confusion matrix for the YOLO model, offering insights into its performance across classes and identifying misclassification patterns to guide model refinement in object detection tasks. We can confirm the model's efficacy in identifying and localizing brain tumors, as well as differentiating between tumor types and the brain background, based on the normalized confusion matrix. The predicted classes are represented by the columns in the YOLO confusion matrix, which have been normalized to add up to 1. Specifically, the classes are the various tumor types (Pituitary, Meningioma, Glioma) and the brain background, with the latter not being predicted by the model according to YOLO developers.

4.5 Prediction Example

To illustrate how the suggested approach can be used in practice, Figure 8 illustrates the results of processing sixteen brain images with the proposed model, where bounding boxes are automatically added to highlight areas associated with cancer. A prediction percentage is also included for each bounding box, enabling an evaluation of the model's confidence in its predictions.

The figure demonstrates that the model can generalize the identification of brain tumors across various sizes and types, accurately identifying cancerous areas irrespective of their size, coloration, or the surrounding area. The tumors, classified into Pituitary, Meningioma, and Glioma, are depicted in different shades, including white, grey, or a combination of both. This showcases the model's robust ability to effectively identify cancerous regions with diverse characteristics and classify them into specific tumor types.

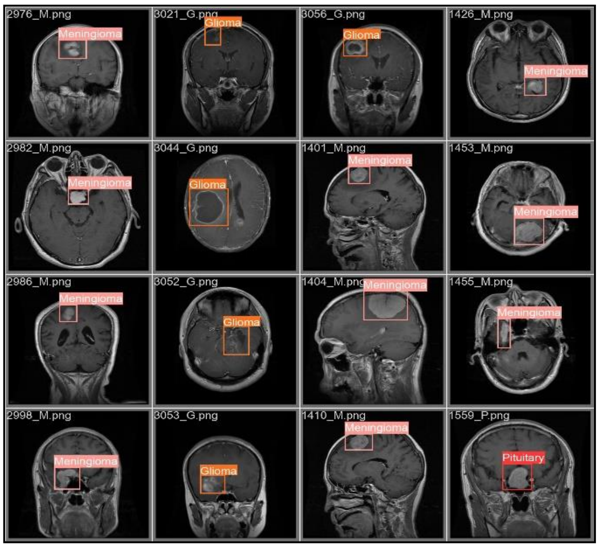


Figure 8 Processed MRI images by trained YOLO model.

V. Conclusion

In this study we evaluated the efficiency of the object detection model known as You Only Look Once, or YOLO for MRI image-based brain tumor localization and classification. Utilizing a dataset of 3064 MRI images from 233 patients with three primary brain tumor types, the Ultralytics YOLOv8.2.2 framework demonstrated strong performance. Within the validation dataset, the model attained a precision (P) of 0.871, recall (R) of 0.868, mAP50 of 0.906, and mAP50-95 of 0.529. Individual class performance was also notable: Pituitary (P: 0.852, R: 0.870), Meningioma (P: 0.955, R: 0.986), and Glioma (P: 0.804, R: 0.750). The precision-recall curve analysis validated the model's robustness, with an average precision (AP) of 0.906. The YOLO model effectively distinguished between various brain tumors and background, as shown in the normalized confusion matrix. Practical application on sixteen brain images confirmed the model's ability to identify diverse cancerous regions accurately.

In summary, the YOLO object detection model presents a viable automated method for identifying and categorizing brain tumors in magnetic resonance imaging. These findings suggest its potential to aid clinicians in precise diagnosis, contributing to improved treatment planning and patient outcomes. Further research on larger datasets is recommended to enhance the model's clinical applicability and reliability.

References

[1] Brain tumor - Symptoms and causes - Mayo Clinic. (2023, April 21). Mayo Clinic. https://www.mayoclinic.org/diseases-conditions/brain-tumor/symptoms-causes/syc-20350084

[2] Brain Tumor Facts - National Brain Tumor Society. (2024, February 20). National Brain Tumor Society. https://braintumor.org/brain-tumors/about-brain-tumors/brain-tumor-facts/

[3] Di Giammarco, M.; Iadarola, G.; Martinelli, F.; Mercaldo, F.; Ravelli, F.; Santone, A. Explainable Deep Learning for Alzheimer Disease Classification and Localisation. In Proceedings of the International Conference on Applied Intelligence and Informatics; Springer: Cham, Switzerland, 2022; pp. 129–143.

[4] Fomchenko, E.I.; Holland, E.C. Stem cells and brain cancer. Exp. Cell Res. 2005, 306, 323–329.

[5] Mercaldo, F.; Di Giammarco, M.; Apicella, A.; Di Iadarola, G.; Cesarelli, M.; Martinelli, F.; Santone, A. Diabetic retinopathy detection and diagnosis by means of robust and explainable convolutional neural networks. Neural Comput. Appl. 2023, 35, 17429–17441.

[6] Tandel, G.S.; Biswas, M.; Kakde, O.G.; Tiwari, A.; Suri, H.S.; Turk, M.; Laird, J.R.; Asare, C.K.; Ankrah, A.A.; Khanna, N.; et al. A review on a deep learning perspective in brain cancer classification. Cancers 2019, 11, 111.

[7] Sharma, K.; Kaur, A.; Gujral, S. Brain tumor detection based on machine learning algorithms. Int. J. Comput. Appl. 2014, 103, 7–11

[8] Ramteke, R.; Monali, Y.K. Automatic medical image classification and abnormality detection using K-Nearest Neighbour. Int. J. Adv. Comput. Res. 2012, 2, 190–196

[9] Xuan, X.; Liao, Q. Statistical structure analysis in MRI brain tumor segmentation. In Proceedings of the 2007 Fourth International Conference on Image and Graphics, ICIG 2007, Chengdu, China, 22–24 August 2007; pp. 421–426.

IJCRT

[10] Jafari, M.; Shafaghi, R. A hybrid approach for automatic tumor detection of brain MRI using support vector machine and genetic algorithm. Glob. J. Sci. Eng. Technol. 2012, 3, 1–8.
[11] Amin, S.E.; Mageed, M. Brain tumor diagnosis systems based on artificial neural networks and segmentation using MRI. In Proceedings of the 2012 8th International Conference on Informatics and Systems (INFOS), Giza, Egypt, 14–16 May 2012; pp. 15–25.
[12] Mercaldo, F.; Brunese, L.; Martinelli, F.; Santone, A.; Cesarelli, M. Object Detection for Brain Cancer Detection and Localization. Appl. Sci. 2023, 13, 9158. https://doi.org/ 10.3390/app13169158
[13] Selvy, P.T.; Dharani, V.; Indhuja, A. Brain tumour detection using deep learning techniques. Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol. 2019, 169, 175.
[14] Sivapathi Arunachalam & Gopalakrishnan Sethumathavan (2022) An effective tumor detection in MR brain images based on deep CNN approach: i-YOLOV5, Applied Artificial Intelligence, 36:1, 2151180, DOI: 10.1080/08839514.2022.2151180
[15] Montalbo, F.J.P. (2020). A Computer-Aided Diagnosis of Brain Tumors Using a Fine-Tuned YOLO-based Model with Transfer Learning. College of Informatics and Computing Sciences, Batangas State University, Batangas City, Batangas 4200, Philippines. [Email: francismontalbo@ieee.org]. *Corresponding author: Francis Jesmar P. Montalbo. Received August 5, 2020; revised November 4, 2020; accepted November 23, 2020; published December 31, 2020.
[16] Abdusalomov, A.B.; Mukhiddinov, M.; Whangbo, T.K. Brain Tumor Detection Based on Deep Learning Approaches and Magnetic Resonance Imaging. Cancers 2023, 15, 4172. https:// doi.org/10.3390/cancers15164172
[17] labelImg. (2021, October 11). PyPI. https://pypi.org/project/labelImg/
[18] Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91. keywords: {Computer architecture;Microprocessors;Object detection;Training;Real-time systems;Neural networks;Pipelines}
[19] U. (2024, April 18). YOLOv9. Ultralytics YOLOv8 Docs. https://docs.ultralytics.com/models/yolov9/
[20]W. (n.d.). yolov9/models/detect/yolov9-c.yaml at main · WongKinYiu/yolov9. GitHub. https://github.com/WongKinYiu/yolov9/blob/main/models/detect/yolov9-c.yaml
[21] brain tumor dataset. (2017, April 2). Figshare. https://figshare.com/articles/dataset/brain_tumor_dataset/1512427
[22] ultralytics. (2024, April 19). PyPI. https://pypi.org/project/ultralytics/
[23] Python Release Python 3.10.12. (n.d.). Python.org. https://www.python.org/downloads/release/python-31012/ torch. (2024, February 22). PyPI. https://pypi.org/project/torch/2.2.1/