Scientific Journal of Informatics
Vol. 10, No. 3, Aug 2023
p-ISSN 2407-7658 http://journal.unnes.ac.id/nju/index.php/sji e-ISSN 2460-0040

YOLOv8 Based on Data Augmentation for MRI Brain Tumor Detection

Rahma Satila Passa[1*], Siti Nurmaini[2], Dian Palupi Rini[3]
[1,2,3]Department of Computer Science, Faculty of Computer Sciences,
Universitas Sriwijaya, Indonesia

Abstract.
Purpose: This research aimed to detect meningioma, glioma, and pituitary brain tumors using the YOLOv8 architecture and data augmentations.
Methods: This research employed the YOLOv8 architecture with data augmentation techniques to detect meningioma, glioma, and pituitary brain tumors. The study collected a dataset of T1-weighted contrast-enhanced images. The dataset used for training, validation, and testing. Preprocessing and augmentation applied to enhance the training data.
Result: After applying data augmentation techniques, the performance of all tumor types improves significantly. Meningioma, Glioma, and Pituitary tumors demonstrate increased Precision, Recall, and mAP50 scores compared to the results before augmentation. The findings highlight the effectiveness of the proposed method in enhancing the model's ability to accurately detect brain tumors in MRI scans. The research conducted both with and without augmentation followed a similar procedure: data collection was first undertaken, followed by preprocessing and with or without augmentation. Subsequently, the collected data was partitioned into training and validation subsets for training with the YOLOv8 architecture. Finally, the model's performance was evaluated through testing to assess its effectiveness in detecting brain tumors.
Novelty: The novelty of this research lies in the YOLOv8 architecture and data augmentation techniques for MRI brain tumor detection. The study contributes to the existing knowledge by demonstrating the effectiveness of deep learning-based approaches in automating the detection process and improving the model's performance. By combining YOLOv8 with data augmentation, the proposed method enhances the model's accuracy and efficiency. The research findings emphasize the potential of this approach in facilitating early diagnosis and treatment planning, thereby improving patient care in the context of brain tumor detection.

Keywords: Deep learning, Object Detection, Brain Tumor, YOLOv8, Data Augmentation
Received July 2023 / **Revised** July 2023 / **Accepted** August 2023

INTRODUCTION

Brain tumors occur due to the emergence of uncontrolled and massive growth of abnormal cells. Tumors develop when there is uncontrolled cell division in the brain, leading to the formation of clusters of abnormal cells within the organ. These clusters can affect the normal brain function and destroy healthy cells. In 2016, brain tumors became the leading cause of cancer-related deaths among children (ages 0-14) in the United States, ranking higher than leukemia [1]. Brain tumors, along with Central Nervous System (CNS) tumors, are the third most common cancers among adolescents (ages 15-39).

Frequently occurring brain tumors often require medical services that have a basic understanding of diagnosis and management [2]. Diagnosis is typically performed by experts through various methods, including medical imaging such as Magnetic Resonance Imaging (MRI). MRI is known as one of the most accurate medical imaging techniques used for various diagnostic tasks, including brain tumor detection. It applies a biomarker-based neuroimaging technique utilizing magnetic field gradient to provide anatomical and physiological information for diagnosis [3]. MRI imaging can display the brain's anatomical structures and assist experts in detecting tumors.

*Corresponding author.
Email addresses: rahmasatilapassa@gmail.com (Passa)
DOI: 10.15294/sji.v10i3.45361

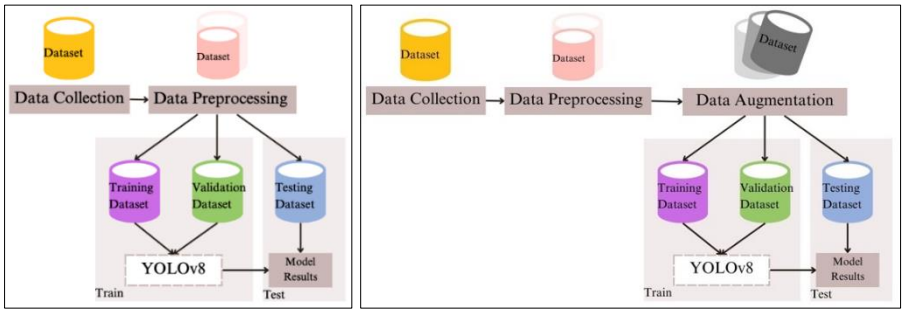
Currently, most tumor detection tasks rely on manual assessment by radiologists or pathologists, which can be time-consuming. Manual detection heavily relies on the level of expertise and experience to provide reliable and accurate results. Therefore, accurate and reliable detection is highly needed in the field of medicine for diagnosing brain tumors. Deep learning-based detection has proven to be highly effective in addressing this issue.

Deep learning utilizes various robust methods to achieve high levels of accuracy. Different approaches and transfer learning techniques are implemented to maximize results. Convolutional Neural Networks (CNNs) have proven to be reliable and popular for object detection tasks, achieving comparable results to other methods. Detection of brain tumors using CNNs has been intensively studied by researchers in recent years, employing different architectures such as YOLO (You Only Look Once) with various versions. Some of these studies include the use of YOLOv3 [4], YOLOv4 [5], and YOLOv5 [6] architectures for brain tumor detection. These approaches have demonstrated their capability to accurately detect brain tumors. While these models exhibit promising performance, the pursuit of enhancement propels the evolution to newer iterations like YOLOv8. YOLO has evolved from its original version to newer versions, including YOLOv8 [7].This dynamic progression underscores the constant drive for improved detection methodologies.

Therefore, the challenge at hand is brain tumor detection using deep learning techniques, which can save time in diagnosing tumor types and reduce the risk of misdiagnosis. This research focuses on brain tumor detection using the Convolutional Neural Network (CNN) method with the YOLOv8 architecture.

METHODS

This research aims to detect meningioma, glioma, and pituitary brain tumors using the YOLOv8 architecture based on data augmentations. The results obtained will be analyzed and evaluated. The research stages to achieve the final results of this study are outlined in the research framework shown in Figure 1.



(a) (b)

Figure 1. Research framework without augmentation (a) and with augmentation (b)

Data Collection

Data is the most crucial factor in methods such as artificial intelligence and machine learning. This is because data serves as the foundation for learning and adapting to problems in these methods. Training data is carefully selected from a dataset that represents the problem space in a homogeneous manner. The dataset contains a diverse set of relevant examples that are related to the problem at hand. By utilizing the right data, AI and machine learning models can learn, recognize patterns, and make accurate decisions [8]. Moreover, the selection of representative and homogeneous training data is vital to prevent bias and ensure good generalization to unseen data. In essence, data plays a fundamental role in the successful implementation of AI and machine learning methods for solving complex problems.

The first step in this research is data collection. Data collected form Jung Cheng [9] contains 3064 T1-weighted contrast-inhanced images with three kinds of brain tumor. The three types of brain tumor are meningioma, glioma, and pituitary.

Table 1. Dataset

Tumor Type	Training	Validation	Testing
Meningioma	496	141	71
Glioma	998	285	143
Pituitary	651	186	83

The table provides information on the distribution of data for different tumor types across the training, validation, and testing sets. The dataset is divided into three tumor types: Meningioma, Glioma, and Pituitary. For Meningioma, there are 496 images in the training set, 141 images in the validation set, and 71 images in the testing set. For Glioma, there are 998 images in the training set, 285 images in the validation set, and 143 images in the testing set. For Pituitary, there are 651 images in the training set, 186 images in the validation set, and 83 images in the testing set. These numbers indicate the number of images available for each tumor type in each respective dataset split, providing an overview of the data distribution for training, validation, and testing purposes.

Data Preprocessing

The obtained data still needs to be processed before it is ready for use. Preprocessing techniques are needed for data normalization and the elimination of redundant data within the dataset to make it ready for use. This preprocessing stage involves data conversion, which converts the data format from .mat to .jpg to proceed to the next stage. To facilitate the annotation process, we leveraged Roboflow [10], which required significant time and effort from our project team. The data was processed and labeled so that it can be understood and learned. The annotated objects are the identified tumor areas. An example of an annotated image is shown in Figure 2.

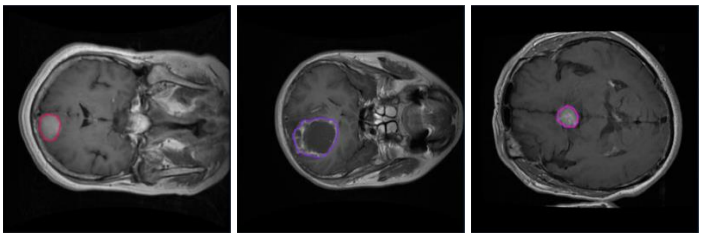


Figure 2. Meningioma, glioma, and pituitary tumor image anotation

Data Augmentation

Data augmentation is a crucial technique in image processing to increase the diversity of training data and enhance the generalization ability of machine learning models. In the context of this study, various augmentation operations were applied to the images. Augmentation techniques are carried out depend on the implementation within the augmentation tool, such as Roboflow. The tool would apply these transformations to original images, generating multiple variations for each image based on the defined augmentation factors and settings. These operations included describe in Table 2.

Table 2. Data augmentation

Technique	Applied
Flip	Horizontal
90° Rotate	Clockwise, Counter-Clockwise
Crop	0% Minimum Zoom, 20% Maximum Zoom
Rotation	Between -15° and +15°
Shear	±15° Horizontal, ±15° Vertical
Grayscale	Apply to 25% of images
Brightness	Between -40% and +40%
Exposure	Between -25% and +25%
Blur	Up to 2.5px
Noise	Up to 10% of pixels

By applying these data augmentation techniques, the dataset was augmented with diverse variations of the original images, resulting in a larger and more diverse training set [11].

The number of data used for training after the augmentation process varies based on the original dataset size, the types of augmentation techniques, and the extent of augmentation. The total augmented dataset includes the original images along with the newly generated augmented images. Consequently, the size of the training dataset grows to encompass the original dataset and the diverse set of augmented variations.

YOLOv8

YOLO, as a one-stage object detection algorithm with high computational efficiency, has become a prominent approach in the field [12]. In the realm of deep learning, YOLO has gained popularity due to its robustness, validity, and rapid detection capabilities [13].The advantages of the YOLO algorithm include its speed, ease of configuration, open-source nature, compatibility with various frameworks and libraries, and high accuracy. Over the past few years, the YOLO [14] [15] algorithm has undergone several iterations, including YOLOv2 [16], YOLOv3 [17],YOLOv4 [18], YOLOv5 [19], YOLOv6 [20], and YOLOv7 [21]. These iterations represent the evolution and improvement of the YOLO algorithm over time.

YOLOv8 [19]. capabilities and improvements in a computer vision model used for tasks such as object detection, classification, and segmentation. It mentions that YOLOv8 is easy to use and can be trained on large datasets. The architecture of YOLOv8 includes different scales of feature maps and utilizes structures like B1-B5, P3-P5, and N4-N5 in the backbone, FPN, and PAN [22].

The enhancements introduced in YOLOv8 compared to previous versions. These include the adoption of Feature Pyramid Network (FPN) and Path Aggregation Network (PAN) as part of the neural network architecture, as well as the development of a new labeling tool to simplify the annotation process [23].

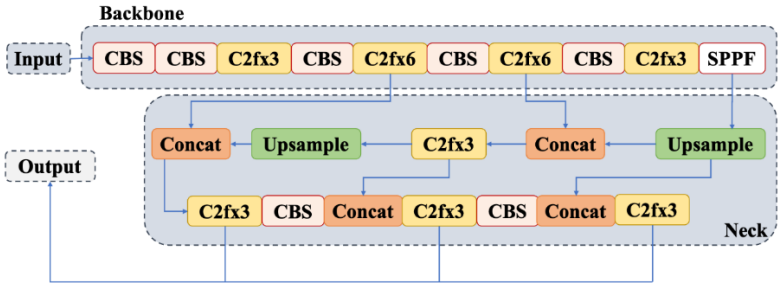


Figure 3. YOLOv8 architecture

The components of YOLOv8, such as the backbone and neck. The Backbone component of YOLOv8 is fundamentally similar to YOLOv5 [23]. The backbone consists of modules like C2f and SPPF, which extract features and ensure accuracy at different scales. The neck module incorporates feature fusion using PAN-FPN and incorporates the idea of separating the head for improved accuracy. Additionally, the C2f module enhances detection accuracy by combining high-level features with contextual information [24]. At the end of the backbone, the SPPF module is still utilized in YOLOv8 to extract features through three pooling operations, aiming to enhance the network's receptive field [25].

SPPF is a element in YOLOv8's architecture. It uses pooling operations to extract features across various scales, enhancing the model's ability to capture context and spatial information effectively. SPPF is employed at the end of the backbone, improving the receptive field of the network. By aggregating information from different scales, SPPF ensures that YOLOv8 can accurately detect objects of varying sizes and complexities.

The C2f module involves dividing the feature maps of the base layer into two parts, merging them hierarchically, and using two convolutions. This approach optimizes computation by reducing gradient repetition and enhances the model's feature extraction capability. The C2f module plays a significant role in YOLOv8's backbone and contributes to its overall accuracy.

CBS, which forms the backbone of YOLOv8, employs cross-stage hierarchy by splitting and merging feature maps, improving gradient flow optimization. This innovative approach reduces computational burden while maintaining or enhancing accuracy. YOLOv8 integrates CBS-inspired C2f modules, enhancing the performance of its backbone and driving efficient feature extraction.

Performance

From the trained model, validation is performed using performance metrics based on the confusion matrix. In the confusion matrix, true positive refers to the model predicting a label correctly and matching the ground truth. False positive indicates the model predicting a label that is not part of the ground truth. True

negative means the model not predicting a label and it is not part of the ground truth. False negative signifies the model not predicting a label, but it is actually part of the ground truth. namely precision, recall, and mAP (mean Average Precision).

Precision is the measure of accuracy between user requests and system responses. It indicates the percentage of correct predictions among all predictions. Precision is used to assess the accuracy of the model being used. Recall is the ratio of the number of true positives to the total number of objects. For example, if there are 100 trees in an image and the model detects 75 trees, the recall would be 75%. Mean Average Precision (mAP) is the average of the Average Precision (AP) for each class. mAP is calculated by finding the AP for each class and averaging them across all classes.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (1)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (2)$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \quad (3)$$

RESULT AND DISCUSSION

The results of evaluating the performance of the YOLOv8 model, which has been implemented based on metrics performance such as precision, recall, and mAP, are analyzed for the detection of YOLOv7. The detection results from these models will be analyzed to obtain the best outcomes in detecting brain tumors, specifically meningioma, glioma, and pituitary tumors.

YOLOv8 offers five scaled versions: YOLOv8n (nano), YOLOv8s (small), YOLOv8m (medium), YOLOv8l (large), and YOLOv8x (extra large). These versions vary in terms of model size and complexity. This research use YOLOv8s with hyperparameter configuration here is an input size of 640 x 640, 100 epochs, and batch size 8. The configuration of model is described in Table 3.

Table 3. Hyperparameter

Configuration	Value
Model	YOLOv8s
Size	640x640
Epoch	100
Batch	8

Before Augmentation

The model was applied to the unaugmented data, and the training results are illustrated in Figure 4. The diagram depicts the values per epoch for box_loss, cls_loss, dfl_loss, precision, and recall (both train and validation). The box_loss represents the loss incurred in localizing the bounding boxes of the detected objects, while cls_loss reflects the loss associated with classifying the objects into different categories. The dfl_loss corresponds to the loss incurred in refining the predicted bounding boxes.

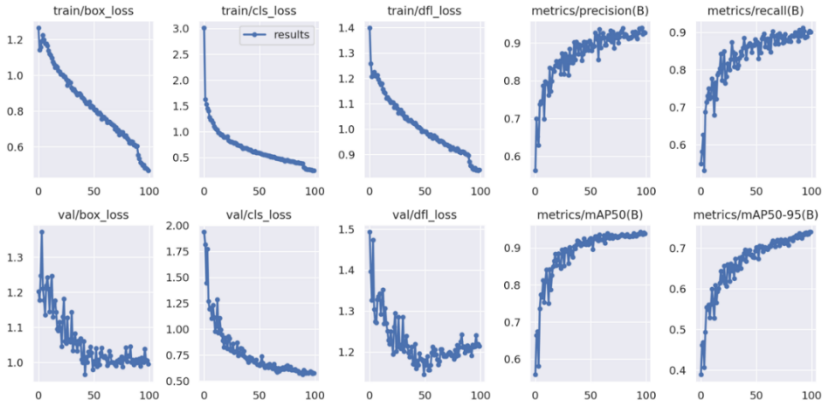


Figure 4. Results on training before augmentation

Table 4. Results on validation dataset before augmentation

Tumor Type	Precision	Recall	mAP50	mAP50-95
Meningioma	0,956	0,951	0,98	0,849
Glioma	0,866	0,816	0,866	0,596
Pituitary	0,956	0,939	0,97	0,773
All	0,926	0,902	0,938	0,739

Table 4 presents the results on the validation dataset before augmentation. It shows the performance metrics for different tumor types, including Precision, Recall, mAP50 (mean Average Precision at IoU threshold 0.50), and mAP50-95 (mean Average Precision from IoU threshold 0.50 to 0.95). For Meningioma tumors, the Precision is 0.956, the Recall is 0.951, the mAP50 is 0.98, and the mAP50-95 is 0.849. For Glioma tumors, the Precision is 0.866, the Recall is 0.816, the mAP50 is 0.866, and the mAP50-95 is 0.596. For Pituitary tumors, the Precision is 0.956, the Recall is 0.939, the mAP50 is 0.97, and the mAP50-95 is 0.773. The "All" category represents the overall performance across all tumor types. The Precision is 0.926, the Recall is 0.902, the mAP50 is 0.938, and the mAP50-95 is 0.739. Meningioma and Pituitary tumors achieved high Precision, Recall, and mAP50 scores. However, Glioma tumors had slightly lower performance scores compared to the other two tumor types.

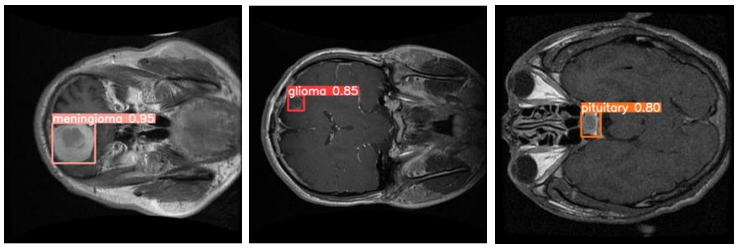


Figure 5. Results on testing dataset before augmentation

After Augmentation

The model was applied to the augmented data, and the training results are illustrated in Figure 6. By analyzing the values per epoch for box_loss, cls_loss, dfl_loss, precision, and recall (both train and validation), researchers can assess the training progress of the model. These metrics offer valuable information about the model's convergence, loss optimization, and performance in detecting and classifying objects accurately.

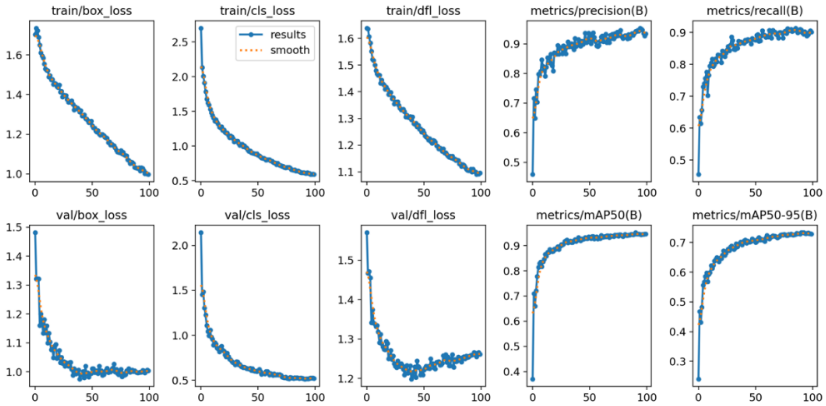


Figure 6. Results on training after augmentation

Table 5. Results on validation dataset after augmentation

Tumor Type	**Precision**	**Recall**	**mAP50**	**mAP50-95**
Meningioma	0,985	0,95	0,986	0,841
Glioma	0,891	0,831	0,894	0,599
Pituitary	0,95	0,942	0,975	0,758
All	0,942	0,908	0,952	0,733

Table 5 presents the results on the validation dataset after augmentation. For Meningioma tumors, the Precision is 0.985, the Recall is 0.95, the mAP50 is 0.986, and the mAP50-95 is 0.841. For Glioma tumors, the Precision is 0.891, the Recall is 0.831, the mAP50 is 0.894, and the mAP50-95 is 0.599. For Pituitary tumors, the Precision is 0.95, the Recall is 0.942, the mAP50 is 0.975, and the mAP50-95 is 0.758. Overall, performance all tumor types with the Precision is 0.942, the Recall is 0.908, the mAP50 is 0.952, and the mAP50-95 is 0.733. Meningioma, Glioma, and Pituitary tumors all showed increased Precision, Recall, and mAP50 scores compared to the results before augmentation.

The augmentation techniques implemented effectively enhanced the model's ability. The changes in precision and recall stand out as significant contributors to the observed improvements. The results before augmentation compared to after augmentation reveals a consistent increase in precision across tumor types. Notably, the Meningioma, Glioma, and Pituitary categories all experienced noticeable precision enhancements. Furthermore, the recall values after augmentation across different tumor types exhibit meaningful improvements too.

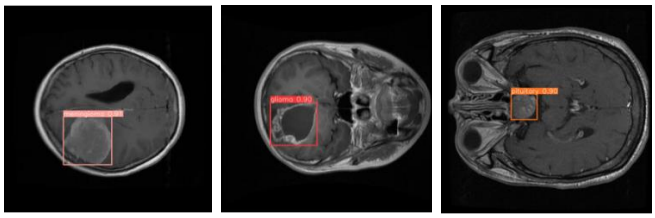


Figure 7. Results on testing dataset before augmentation

CONCLUSION

This paper presents a YOLOv8-based approach combined with data augmentation for MRI brain tumor detection. The results show that the proposed method achieves accurate and efficient detection of brain tumors in MRI scans. The integration of YOLOv8 and data augmentation techniques enhances the model's performance, improving its ability to handle diverse tumor variations and imaging conditions. The findings demonstrate the potential of deep learning-based approaches in automating brain tumor detection and facilitating early diagnosis and treatment planning for improved patient care. Future work may involve further optimization and validation of the proposed method using larger datasets and exploring other data pre-processing.

REFERENCES

[1] N. Abiwinanda, M. Hanif, S. T. Hesaputra, A. Handayani, and T. R. Mengko, "Brain Tumor Classification Using Convolutional Neural Network." *World Congr. Med. Phys. Biomed. Eng. 2018*, 2019, pp. 183–189, doi: 10.1007/978-981-10-9035-6_33.

[2] J. R. McFaline-Figueroa and E. Q. Lee, "Brain Tumors," *Am. J. Med.*, vol. 131, no. 8, pp. 874–882, 2018, doi: 10.1016/j.amjmed.2017.12.039.

[3] B. Khagi and G.-R. Kwon, "3D CNN based Alzheimer's Diseases Classification Using Segmented Grey Matter Extracted from Whole-Brain MRI," *JOIV Int. J. Informatics Vis.*, vol. 5, no. 2, p. 200, 2021, doi: 10.30630/joiv.5.2.572.

[4] F. Ali *et al.*, "A Two-Tier Framework Based on GoogLeNet and YOLOv3 Models for Tumor Detection in MRI," *Comput. Mater. Contin.*, vol. 72, no. 1, pp. 73–92, 2022, doi: 10.32604/cmc.2022.024103.

[5] F. J. P. Montalbo, "A Computer-Aided Diagnosis of Brain Tumors Using a Fine-Tuned YOLO-based Model with Transfer Learning," *KSII Trans. Internet Inf. Syst.*, vol. 14, no. 12, 2020, doi: 10.3837/tiis.2020.12.011.

[6] H. Chegraoui *et al.*, "Object Detection Improves Tumour Segmentation in MR Images of Rare Brain Tumours," *Cancers (Basel).*, vol. 13, no. 23, pp. 6113, 2021, doi: 10.3390/cancers13236113.

[7] G. Jocher, A. Chaurasia, and J. Qiu, "YOLO by Ultralytics. 2023," *URL https//github. com/ultralytics/ultralytics*, 2023.

[8] M. Horvat and G. Gledec, "A Comparative Study of YOLOv5 Models Performance for Image Localization and Classification," in *Cent. Eur. Conf. Inf. Intell. Syst.*, 2022, pp. 349–356.

[9] J. Cheng, "Brain Tumor dataset," *Figshare*, 2017. https://figshare.com/articles/dataset/brain_tumor_dataset/1512427.

[10] B. Dwyer, J. Nelson, and J. Solawetz, "Roboflow (version 1.0)[software]." 2022, [Online].

Available: https://roboflow.com/research.
[11] J. Fan, L. Cui, and S. Fei, "Waste Detection System Based on Data Augmentation and YOLO_EC," *Sensors*, vol. 23, no. 7, pp. 3646, 2023, doi: 10.3390/s23073646.
[12] S. Pan, J. Liu, and D. Chen, "Research on License Plate Detection and Recognition System based on YOLOv7 and LPRNet," *Acad. J. Sci. Technol.*, vol. 4, no. 2, pp. 62–68, 2023, doi: 10.54097/ajst.v4i2.3971.
[13] M. U. Khan, M. Dil, M. Misbah, and F. A. Orakazi, "Deep Learning Empowered Fast and Accurate Multiclass UAV Detection in Challenging Weather Conditions," *Preprints.org*, 2022, doi: 10.20944/preprints202212.0049.v1.
[14] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.
[15] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *2017 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6517–6525, doi: 10.1109/CVPR.2017.690.
[16] J. Redmon and A. Farhadi, "YOLO v.3," *Tech Rep.*, pp. 1–6, 2018, [Online]. Available: https://pjreddie.com/media/files/papers/YOLOv3.pdf.
[17] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv Prepr.*, 2020, [Online]. Available: https://arxiv.org/abs/2004.10934.
[18] G. Jocher *et al.*, "Ultralytics/yolov5: v7. 0-yolov5 Sota Realtime Instance Segmentation," *Zenodo*, 2022, doi: 10.5281/zenodo.7347926.
[19] C. Li *et al.*, "YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications," *arXiv Prepr.*, 2022, [Online]. Available: http://arxiv.org/abs/2209.02976.
[20] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 7464–7475.
[21] Y. Li, Q. Fan, H. Huang, Z. Han, and Q. Gu, "A Modified YOLOv8 Detection Network for UAV Aerial Image Recognition," *Drones*, vol. 7, no. 5, pp. 304, 2023, doi: 10.3390/drones7050304.
[22] D. Reis, J. Kupec, J. Hong, and A. Daoudi, "Real-Time Flying Object Detection with YOLOv8," *arXiv Prepr.*, 2023, doi: 10.48550/arXiv.2305.09972.
[23] H. Lou *et al.*, "DC-YOLOv8: Small-Size Object Detection Algorithm Based on Camera Sensor," *Electronics*, vol. 12, no. 10, pp. 2323, 2023, doi: 10.3390/electronics12102323.
[24] J. Terven and D. Cordova-Esparza, "A Comprehensive Review of YOLO: From YOLOv1 and Beyond," *arXiv Prepr.*, pp. 1–34, 2023, [Online]. Available: http://arxiv.org/abs/2304.00501.
[25] H. Lou *et al.*, "CS-YOLO: A New Detection Algorithm for Alien Intrusion on Highway," *Res. Sq.*, 2023, doi: 10.21203/rs.3.rs-2795266/v1.