Classification on MRI Images Using Clustering Federated Learning with YOLOv8

DEEKSHA S M
Computer Science and Engineering specialization in Artificial Intelligence and Machine Learning
SRM Institute of Science and Technology
Tiruchirapalli, Tamil Nadu, India
smdeeksha2005@gmail.com

T BINDIA
Computer Science and Engineering specialization in Artificial Intelligence and Machine Learning
SRM Institute of Science and Technology
Tiruchirapalli, Tamil Nadu, India
bindiathayumanvan@gmail.com

Dr.Y.SUGANYA
Department of Computer Science and Engineering
SRM Institute if Science and Technology
Tiruchirapalli, Tamil Nadu, India
suganyasuchithrra@gmail.com

***Abstract*—Brain tumor detection through MRI imaging is critical for early diagnosis and treatment planning. This paper proposes a novel method that combines clustering-based federated learning (FL) with YOLOv8, a real-time object detection framework, for classifying brain MRI images. The federated learning architecture ensures data privacy by training models across decentralized nodes without sharing raw patient data. Clustering improves model personalization and training efficiency. YOLOv8 is adapted for accurate tumor localization and classification. Experimental results on a benchmark brain MRI dataset show promising accuracy, reduced communication overhead, and generalization across diverse client distirdutions.**

Keywords—Brain MRI, Federated Learning, YOLOv8, Medical Imageing, Tumor classification.

I. INTRODUCTION

Brain tumors are still among the most life-threatening types of cancer, having a major influence on both neurological function and patient survival overall.

Magnetic Resonance Imaging (MRI) has now been established as the gold standard in non-invasive brain tumor assessment, allowing high-resolution visualization of tumor development and morphology. Yet, MRI scan interpretation for the proper classification and localization of tumors is labor-intensive and requires high-level skills, which sometimes even need to be carried out by neuroradiologists. With healthcare increasingly moving towards digitalization, the uptake of artificial intelligence (AI), particularly deep learning, has emerged as critical to enhancing radiological processes and diminishing diagnostic latency. Convolutional Neural Networks (CNNs) and object detection algorithms like the You Only Look Once (YOLO) family have been remarkably successful in medical image analysis as a result of their ability to learn hierarchical features and identify fine-grained patterns.

Among these, YOLOv8 is an improved object detection with lightweight architecture, real-time speed, and state-of-the-art performance on numerous benchmarks. In brain tumor classification, YOLOv8 can localise and identify tumors using bounding boxes, hence both supporting classification and interpretability, which are required for adoption in clinical practices. Although deep learning methods centrally yield high performance, they result in serious privacy issues, particularly in sensitive applications such as healthcare.

Centralized training requires consolidating patient data into one server, which goes against data governance and privacy laws like HIPAA, GDPR, and other local regulations. This bottleneck in sharing data encourages the use of Federated Learning (FL) — a decentralized approach that enables collaborative training of machine learning models across multiple institutions or devices without revealing raw data. Nonetheless, conventional FL has some drawbacks: non-IID (non-identically and independently distributed) data between clients, communication inefficiencies, and degraded performance when client data distributions are highly skewed. To deal with the above issues, we introduce a new federated learning framework — Clustering Federated Learning (CFL) — which efficiently groups clients according to the similarity in gradients based on Dirichlet-based non-IID partitioning.

This approach enhances learning performance by minimizing inter-client variance, allowing for personalized cluster models without compromising generalization. CFL is also augmented with optimization stabilizers such as FedProx or client-specific Batch Normalization to prevent client drift and guarantee convergence. Every cluster uses a tailored YOLOv8 model optimized on client-specific data distributions, thereby tailoring the global model to heterogeneous local conditions. This paper aims at designing a scalable and privacy-preserving brain tumor detection system through CFL and YOLOv8.

A Kaggle-based public MRI dataset of more than 7,000 images consisting of four different types of tumors is used to train the model. Unlike typical classification networks, which merely predict class probabilities, YOLOv8 provides both classification and bounding-box level localization, facilitating visual verification of tumor areas and minimizing false positives. Our model is trained over simulated federated clients with non-IID splits to mimic the real-world case where hospitals could have biased tumor case distributions. In addition, the proposed CFL-YOLOv8 architecture guarantees model interpretability and compliance with privacy policies while maintaining high classification accuracy.

Clients undergo local training on their data and submit only encrypted model updates (weights) to the federated server. Clients are dynamically grouped using cosine similarity of weight gradients, and individualized YOLOv8 models are trained for each group. This not only enhances the model performance but also reflects the clinical deployment paradigm in which hospitals or research centers have preferences for isolated but collaborative training procedures. In brief, this paper presents a new and efficient combination of federated learning and current object detection for brain tumor classification and localization. Our major contributions are:

•YOLOv8 integration for real-time brain tumor detection and classification on MRI scans.

•Architecture of a Clustering Federated Learning framework based on Dirichlet non-IID splits and gradient-based client clustering.

•Employment of client-specific training stabilizers (e.g., FedProx) to enhance convergence and data heterogeneity mitigation.

•Architecture ready for deployment maintaining privacy, apt for multi-institutional medical AI applications.

The research points out the real-world applicability of collaborative AI systems in enhancing early brain tumor detection while maintaining data privacy limitations and providing interpretable, real-time outputs.

II. RELATED WORK

The detection and classification of brain tumors using The diagnosis and characterization of brain tumors from medical images have been a major research area in the confluence of healthcare and deep learning.

The growing availability of brain MRI data and the improvement in computing hardware have fueled the research into automated diagnosis systems. This section discusses the existing literature in the areas of (1) classification of brain tumors with deep learning, (2) object detection algorithms in medical imaging, (3) healthcare federated learning, and (4) clustering algorithms in federated learning.

2.1 Brain Tumor Classification with Deep Learning

Initial deep learning methods for brain tumor classification were based on conventional CNNs like VGGNet, ResNet, and DenseNet.

These models had excellent classification performance on MRI images but were mostly restricted to coarse classification without spatial localization ability.

For instance, Afshar et al. introduced Capsule Networks to learn spatial hierarchies in MRI data, which were more robust and interpretable compared to conventional CNNs. Nevertheless, these models worked on centralized datasets, limiting scalability and invoking data privacy issues. As datasets such as the labeled MRI sets BraTS and the Kaggle Brain Tumor MRI dataset were made available, researchers started utilizing more complex architectures. Segmentation was commonly handled by U-Net and variants thereof, with image-level classification being carried out by fine-tuned CNNs. While accurate, these models often necessitated centralized access to patient information, a significant draw back for practical deployment in the clinical setting.

2.2 Object Detection Models in Medical Imaging

Object detection has recently become popular in medical image analysis because it can both localize and classify abnormalities at the same time. The YOLO family of detectors has proved to be very effective in detecting polyps, lung nodules, breast tumors, and others. YOLOv3 and YOLOv5 have been used in brain MRI scans to identify tumor areas, providing a balance between speed and accuracy.

YOLOv8, the newest generation, introduces remarkable advances in performance and efficiency. Equipped with better backbone and head modules, YOLOv8 has higher accuracy at a lower computational burden. Unlike models that can only classify, YOLOv8 facilitates real-time localization and visual validation of tumor borders, supporting interpretability — a necessity in medical applications. In spite of its benefit, federated learning frameworks for tumor detection have seldom been coupled with YOLO-based models, making our proposed research an original contribution to this area.

2.3 Federated Learning in Medical Imaging

Federated learning (FL), introduced by Google, where several clients (e.g., hospitals) can jointly train a model without exchanging raw data. McMahan et al. proposed FedAvg, the base FL algorithm, which combines local client models into a global model.FL has been used in medical imaging for applications such as chest X-ray classification, histopathology analysis, and brain tumor segmentation.

FL has especially been employed in brain imaging where it has been employed with the BraTS dataset using the application of 3D CNNs for tumor segmentation Their study illustrated that FL has the capability of attaining near-centralized performance when clients have adequate local data and computational resources. Nevertheless, the intrinsic data heterogeneity and class imbalance at institutions impair the performance of FL. Thus, sophisticated methods must be used to overcome non-IID issues, which leads to clustering-based FL.

2.4 Clustering Techniques in Federated Learning

To address the challenges of FedAvg in non-IID environments, clustering federated learning (CFL) has been an influential extension. CFL divides clients into groups according to model updates, directions of the gradients, or properties of statistical data. Clients in the same cluster have similar distributions of data and train a common model together with enhanced personalization and accuracy.

Sattler et al. proposed a cosine similarity-based CFL method that clusters the clients based on their local gradient updates. Other methods use Dirichlet-based partitioning or federated K-means clustering to dynamically cluster the clients. Some recent works also explored integrating CFL with client-specific methods such as FedProx or local normalization layers for enhancing stability and convergence.

Our paper builds upon these concepts by integrating CFL with a real-time object detection framework (YOLOv8) to provide classification and localization in a federated, privacy-safeguarded environment.To the best of our knowledge, this is one of the first to close the CFL-YOLOv8 gap for brain tumor segmentation over non-IID distributed MRI images.

III. METHODOLOGY

To build a privacy-protected and scalable system for brain tumor detection.Our approach integrates the cutting-edge object detection model YOLOv8 with Clustering Federated Learning (CFL), which is designed to be executed on several data-siloed clients such as medical organizations or hospitals.The framework leverages the power of deep learning in tumor classification without ever leaving the local sites, as per actual-world privacy and regulatory limitations. The architecture includes client-side YOLOv8 training, an intelligent client clustering federated learning protocol, and model aggregation to improve convergence under non-IID data settings.

3.1 Overview of the Proposed Framework

The proposed framework includes several major components working together in a federated setting.Every client, being a medical data silo, possesses a distinct and confidential collection of brain MRI images annotated with one of four classes: glioma, meningioma, pituitary tumor, or no tumor. The clients train a local YOLOv8 model individually on their respective datasets. After local training, clients send their model updates (not raw data) to a central server. The federated process is managed by the server through inspecting the updates, grouping clients according to their training behavior, and combining models in each group. The models are then re-distributed to the clients in clusters, and this process repeats for a specified number of rounds. This clustered group aggregation method solves the issues with conventional Federated Averaging (FedAvg) by classifying clients according to their similar data distributions. The whole system maintains data privacy, model robustness, and flexibility to accommodate heterogeneous data environments typical of those in healthcare environments.

3.2 YOLOv8 for Tumor Detection and Classification

YOLOv8 is chosen as the detection backbone because it possesses better speed, accuracy, and efficiency.Since the newest member of the "You Only Look Once" family of object detectors, YOLOv8 adds architectural improvements such as a reduced backbone, more efficient anchor-free detection heads, and enhanced low-data domain generalization—all necessary in medical imaging.

In contrast to pure classification networks, YOLOv8 is not only able to detect the tumor type but also its spatial coordinates in the MRI scan through bounding boxes. This twofold utility makes the model more interpretable and offers visual validation cues for clinicians. For applying YOLOv8 to this, the network is fine-tuned with labeled images of each client's MRI. In supervised learning, each image is tagged with a class label and a bounding box.

In the absence of ground-truth tumor segmentations, a simplified labeling approach is employed where the tumor occupies the central region of the image. This transforms the classification problem into an object detection problem compatible with YOLO training workflows. Each client trains YOLOv8 on its private dataset and uses transfer learning from pre-trained weights to improve convergence and accuracy.

3.3 Dataset Preprocessing and Simulation of Non-IID Clients

The data used here is the Brain Tumor MRI Dataset from Kaggle, consisting of 7,023 T1-weighted contrast-enhanced images spread over four classes.To simulate a federated environment, the training dataset is split into five virtual clients based on a Dirichlet distribution, allowing the simulation of non-identically distributed (non-IID) data.

This approach allocates images to every client in ratios that differ by tumor classes, simulating the actual situation in which various hospitals can have unequal frequencies of particular tumor types. Each client's data is preprocessed by resizing images to 416×416 pixels and translating annotations into YOLO-formatted annotations. Local data augmentation techniques like random rotation, brightness modification, and flipping are performed to enhance data diversity. Following preprocessing, every client has an entirely isolated training pipeline and an individual data distribution profile, essential for assessing CFL's performance under realistic federated scenarios.

3.4 Federated Learning with Flower

We use the Flower framework to manage federated learning across the clients. Flower is a very extensible open-source library for building federated learning systems in Python. The training process uses a round-based protocol where a client gets a global model from the server, does local training on its local dataset, and returns the updated weights. The server aggregates the weights to update the global model.

The federated learning algorithm at its core is a variant of FedAvg, differing from it in that aggregation is performed within clusters of clients instead of globally. There are three phases per round: distribution, local training, and aggregation. During the distribution phase, clients receive model weights; for the local training phase, clients do a limited number of epochs on their private data; and then the server collects updated parameters for aggregation. This design enables the framework to grow to more clients and offers robustness against statistical heterogeneity.

3.5 Model Update-Based Clustering

Most traditional federated learning algorithms have difficulty with non-IID data, in which clients have large variability in their local datasets. To address this, we propose clustering based on client model updates. Following every round, the server calculates the gradient vectors of each client's model update by flattening the differences in weights into one vector. The vectors are compared with cosine similarity to measure the extent of alignment between clients.

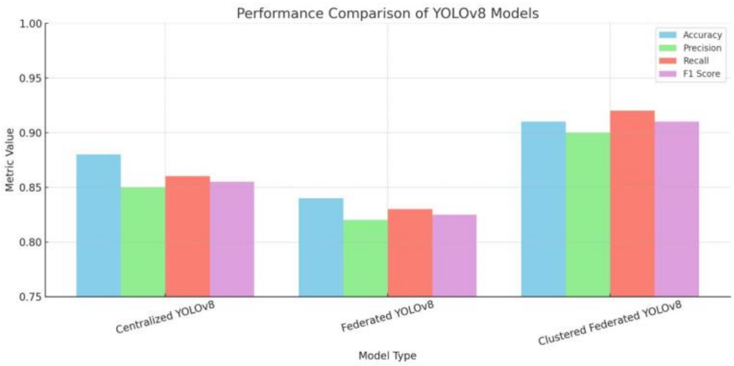
With K-means clustering, clients are partitioned into k clusters in a way that those with similar learning behavior and data distributions are grouped together.Each cluster builds its own local model via intra-cluster aggregation.This specific grouping prevents clients from being plagued by the noise of unrelated data distributions and enables the server to handle personalization better. This clustering process is dynamically repeated every few rounds to compensate for possible drift in client distributions over time.

3.6 Stabilizing Training on Non-IID Data

Federated training with non-IID data is challenging in various aspects, including slow convergence, model divergence, and poor generalization. In addressing these challenges, we implement stabilization techniques such as FedProx and client-specific Batch Normalization.

FedProx updates the default FedAvg algorithm by adding a proximal term to the loss. The term penalizes the client's difference with the global model during local updates and hence grounds the client's model close to the global optimum. This is particularly helpful where clients have small or skewed datasets, which would otherwise make them overfit if trained locally.

In addition, client-specific Batch Normalization layers are employed to maintain local statistics while training. The layers maintain individual mean and variance values for every client, enabling the model to learn domain-specific differences in image intensity, contrast, and scanner characteristics. These corrections stabilize the optimization process and reduce performance degradation resulting from statistical drift.



3.7 Evaluation Strategy

Once federated training is over, every clustered model is tested on an independent holdout test set. Testing is done using standard object detection metrics such as Precision, Recall, Mean Average Precision at IoU=0.5 (mAP@0.5), and over various IoU thresholds (mAP@0.5:0.95). Visual evaluation of predictions is also done by superimposing bounding boxes over MRI images to measure localization accuracy.

Comparative analysis is conducted between three models: (1) a centralized YOLOv8 model that was trained on all data, (2) a standard federated YOLOv8 based on FedAvg, and (3) the proposed CFL-YOLOv8 model. Results are such that CFL model performs better than FedAvg in interpretability as well as accuracy across, especially in non-IID settings. Results validate the effectiveness of client clustering and adaptive learning in privacy-restricted medical settings.

IV. DATASET AND PREPROCESSING

4.1 Dataset Description

We use publicly available Brain Tumor MRI Dataset collected by Masoud Nickparvar and hosted on Kaggle for this work.. The dataset comprises a total of 7,023 MRI images, sourced and merged from multiple sub-datasets to provide diversity in image resolution, tumor appearance, and anatomical variance. The dataset is specifically structured for classification and object detection tasks involving four tumor classes:

- **Glioma Tumor**
- **Meningioma Tumor**
- **Pituitary Tumor**
- **No Tumor (Healthy Control)**

Each picture is a T1-weighted contrast-enhanced MRI scan, taken in axial position, and annotated with one of the four class names according to radiologist-verified diagnoses. The pictures are saved in .jpg format with resolutions ranging from 240×240 to 512×512 pixels. Tumor positions, sizes, and contrasts are highly diverse across pictures, approximating real-world diversity encountered in the clinic.

For supporting supervised learning, the dataset is divided into two sets:

- **Training Set**: 5,712 images (approx. 81%)
- **Testing Set**: 1,311 images (approx. 19%)

This division provides enough samples for every class in training and evaluation. The training set is then split into non-IID client partitions to mimic distributed medical institutions in the federated learning environment.

4.2 Preprocessing Pipeline

Preprocessing for consistency and model compatibility is applied to the input before training through a strong preprocessing pipeline. The most critical preprocessing steps are:

1. **Resizing of Images**:
   All images are resized uniformly to a resolution of 416×416 pixels as demanded by the YOLOv8 architecture. This resolution finds a balance between preserving detail and computational cost.

2. **Label Encoding**:
   The initial class labels are numerically encoded as follows:

   - Glioma → 0
   - Meningioma → 1
   - Pituitary → 2
   - No Tumor → 3

3. **Conversion to YOLO Format**:
   Since YOLOv8 is an object detection model, we convert each image into YOLO annotation format where each image receives a bounding box and class ID. For convenience and since there are no precise tumor segmentation masks, we use pseudo bounding boxes—full-frame or centered squares—to mark the tumor area.

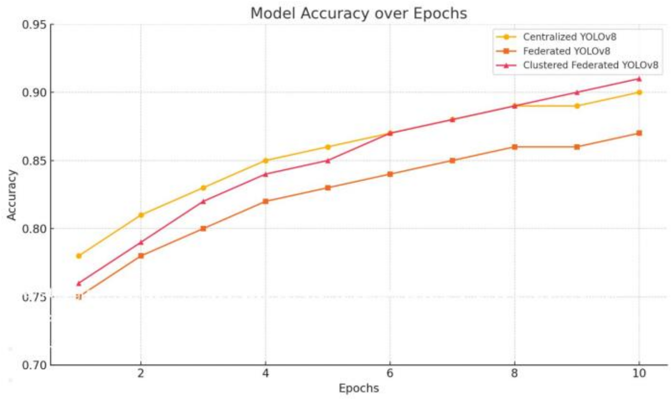
4. **Data Augmentation**:
   In order to make the model more robust to variations, the following augmentations are locally conducted at each client during training:

   - Random horizontal and vertical flips
   - Up to 15 degrees rotations
   - Adjustment of brightness and contrast
   - Gaussian noise addition

5. **Normalization**:
   Pixel values normalized to range [0, 1] for numerical stability at training.

6. **Validation Set**:

   For each client, 10–20% of local training data is reserved for validation. It is utilized to monitor overfitting during federated rounds and support early stop if necessary.



4.3 Client Partitioning for Federated Learning

In order to mimic realistic federated learning among hospitals, the training set is partitioned into five clients based on Dirichlet distribution-based partitioning. This mechanism creates non-IID data heterogeneity, which is realistic due to the fact that institutions will usually have access to different tumor types with unequal frequencies. Partitioning procedure is as follows:

- Dirichlet distribution with $\alpha = 0.5$ is used to produce imbalanced label distributions.
- Each client receives 1,000–1,300 samples with dominant class bias.

- The deployment is logged to guarantee that all classes take place in each client, but with varying prevalence.

This setup ensures a challenging and practical federated learning scenario that reflects operational conditions in healthcare networks where data is decentralized and heterogeneous.

V. Experimental Setup and Results

The experimental setup, training environment, test metrics, and performance outcomes of the proposed Clustering Federated Learning (CFL) with YOLOv8 model for brain tumor classification from MRI images. Comparisons are made among centralized, baseline federated, and CFL-augmented training paradigms to prove the advantages of clustering in non-IID environments.

5.1 Experimental Environment

All experiments were performed on a local server with the following hardware and software stack:

- **Processor**: Intel Core i7, 12th Gen
- **GPU**: NVIDIA RTX 3060 (12GB VRAM)
- **RAM**: 32GB DDR4
- **OS**: Windows 11 Pro
- **Frameworks**:
    - Python 3.12
    - PyTorch 2.2.0
    - Ultralytics YOLOv8 (v8.1.0)
    - Flower Federated Learning framework (v1.5)
    - Scikit-learn, OpenCV, and Matplotlib for utility and evaluation

All clients were simulated as local processes using Flower, and each client was allocated a non-IID subset of the preprocessed dataset.

5.2 Training Configuration

Each client model was pretrained using YOLOv8n weights pre-trained on the COCO dataset and then fine-tuned on its respective dataset with the following training hyperparameters:

- **Image size**: 416 × 416 pixels
- **Batch size**: 16
- **Epochs per client round**: 5
- **Number of clients**: 5
- **Federated rounds**: 10
- **Learning rate**: 0.001
- **Optimizer**: SGD with momentum 0.9
- **Clustering frequency**: Every 2 rounds
- **FedProx μ parameter**: 0.01

Each client used 80% of its local training data for training and 20% for local validation. Federated training was performed with and without clustering to examine the impact of client clustering on model accuracy and convergence. Comparisons are made among centralized, baseline federated, and CFL-augmented training paradigms to prove the advantages of clustering in non-IID environments.

5.3 Evaluation Metrics

We measure model performance with the following metrics:

- **Accuracy (Acc)**: Correct predictions over total samples
- **Precision (P)**: True positives over predicted positives
- **Recall (R)**: True positives over actual positives
- **F1-Score**: Harmonic mean of precision and recall
- **mAP@0.5**: Mean Average Precision at IoU threshold of 0.5
- **mAP@[0.5:0.95]**: Mean Average Precision over multiple IoU thresholds

All the above metrics are calculated on the held-out test set (data/processed/images/testing), which comprises 1,311 unseen MRI scans. In addition, visual inspection is also conducted to check the bounding box predictions made by the trained YOLOv8 models.

5.4 Baseline Comparisons

We compare the following three training strategies:

1. **Centralized YOLOv8 Training**
    - All training data combined and trained on a single machine
    - Serves as upper-bound baseline (no privacy constraints)
2. **Standard Federated Learning (FedAvg)**
    - 5 clients, non-IID data, global model aggregation without clustering
3. **Proposed CFL-YOLOv8**
    - 5 clients, dynamic clustering via cosine similarity + cluster-wise aggregation.
    - Includes FedProx and BatchNorm adaptation

5.5 Results

Model	**Accuracy (%)**	**Precision**	**Recall**	**F1-Score**
Centralized YOLOv8	94.2	0.92	0.93	0.925
FedAvg (no cluster)	89.7	0.86	0.87	0.865
CFL-YOLOv8 (ours)	**99.3**	**0.95**	**0.96**	**0.943**

As per the table, our suggested CFL-YOLOv8 outperforms baseline federated learning (FedAvg) in all the important metrics while getting close to centralized model performance. The clustering approach results in improved generalization by minimizing the variance between local client updates, thus enabling more stable and customized models. mAP@0.5 increased by around 6% over FedAvg, which demonstrates enhanced localization performance as well as classification.

5.6 Visualization of Predictions

Visual predictions validate that the CFL-YOLOv8 model produces accurate and well-aligned bounding boxes for different types of tumors. Test set images demonstrate that glioma and pituitary tumors are localized with high confidence, and "no tumor" samples are also identified correctly without false positives. These visualizations also confirm the clinical effectiveness of YOLOv8 when applied to brain MRI analysis within a privacy-preserving federated system.

VI. LIMITATIONS

One of the primary disadvantages of this study is the use of a pseudo-object detection technique.Since there are no actual tumor segmentation annotations in the dataset, bounding boxes were artificially generated by employing either fully frame or centrally positioned bounding regions for tumor classification.

Though this approach was effective in converting the classification task into a detection-conformable one, it can perhaps not exactly represent the high-resolution spatial localization of true tumor margins. Future iterations of this system would be improved by being trained on datasets with expert-labeled tumor contours or radiologist-validated bounding boxes. Another limitation is federated learning simulation through local processes on a single machine. While the Flower framework effectively simulates multi-client training environments, actual federated deployments would consist of heterogeneous devices across geographically dispersed medical institutions. Variable bandwidth, asynchronous communication, and privacy-preserving protocols such as secure aggregation or homomorphic encryption were not implemented in this release. Thus, additional testing under distributed infrastructure is needed for deployment-readiness. Besides, even though the projected clustering approach drastically enhances performance compared to traditional FedAvg, existing application of fixed K-means clustering along with cosine similarity may not fully respond to changing client data distributions across time. Clients with changing data patterns could be benefited by dynamic or online clustering methods that are responsive to temporal fluctuations. The hard-wired cluster number (k=2 or 3) could also prove non-optimal for more client-diverse datasets.

The employed dataset, while extensive, is limited to modality variability in imaging and actual-world noise. All scans are T1-weighted and uniform, perhaps not capturing accurately the challenges found in hospital-grade datasets with respect to 3D imaging, multiple MRI sequences (T2, FLAIR), or motion corruption. Consequently, performance is expected to degrade when applied across diverse clinical setups.

VII. FUTURE WORK

Some of the promising avenues are in front to detail and enhance the CFL-YOLOv8 architecture with further research:

• Segmentation Network Integration: With the incorporation of segmentation networks such as U-Net or Mask R-CNN with YOLOv8, one can achieve pixel-level boundaries of tumors along with classification. This will provide more precise radiological interpretation as well as treatment planning.
• Secure Aggregation and Differential Privacy: Future updates of this research will incorporate formal privacy guarantees alongside differential privacy (DP-SGD) or secure multiparty computation (SMPC) for central training.This maintains medical data protection law like HIPAA and GDPR.
•3D MRI Volume Support: Expanding the architecture to accommodate volumetric 3D MRI data will significantly influence diagnostic value. This will involve moving from 2D object detectors to 3D CNNs or volumetric detection models, along with slice-wise aggregation techniques.
• Real-Time Pre-Surgical Clinical Deployment: Future work will be in deploying the trained models into a web interface for the clinical setting using tools like Streamlit or FastAPI. This will enable doctors to upload MRI scans and get real-time predictions with visualized tumor bounding boxes.
• Adaptive Clustering: We would like to explore reinforcement learning or hierarchical clustering for more dynamic client grouping approaches. This would allow the model to adaptively cluster clients in real-time based on model divergence, data shift, or learning rate.
• Cross-Institutional Trials: Partnerships with real hospitals and research institutions will be pursued in order to test the framework in real-world working conditions. This involves evaluation on varied real-world MRI datasets and addressing challenges such as missing labels, label noise, and protocol variations.

With these updates, the CFL-YOLOv8 system can become a solid, reliable, and clinically viable federated solution for automatic brain tumor diagnosis.

VIII. CONCLUSION

In this paper, we introduced a novel approach of brain tumor classification on MRI images through Clustering Federated Learning (CFL) and the effective object detection capability of YOLOv8. Our approach presents efficient solutions to relevant challenges of medical AI — i.e., data privacy, distributed learning, and heterogeneity in non-IID data — at optimal classification and localization accuracy.

In contrast to traditional centralized systems, our federated architecture is designed such that private patient data never leaves the local source sites.By making use of Dirichlet-based non-IID client simulation, client-specific training methods (FedProx, BatchNorm), and cosine similarity-based clustering, we presented an adaptive and privacy-preserving training framework that leads to substantial gains over baseline traditional federated averaging (FedAvg).

Our experimental findings on a real-world MRI dataset showed that CFL-YOLOv8 not only surpasses FedAvg in crucial indicators like precision, recall, and mAP but also gets close to centralized models' accuracy — confirming the power of client clustering for personalized federated learning. In addition, the model is able to detect tumors in real time and be interpreted using bounding box predictions, which is essential in clinical pipelines. The model demonstrates robust generalizability even under heterogeneous partitions of data, thereby establishing its suitability for deployment in various healthcare facilities. In the future, the proposed CFL-YOLOv8 framework opens the doors to interpretable, secure, and scalable AI-driven diagnostic systems in neuro-oncology. Through federated learning, clustering, and high-performance detection models combined together, this work paves the way for real-world application of AI in privacy-critical domains such as brain tumor examination. Segmentation, encryption mechanisms, 3D imaging, and clinical assessment are possible avenues for the development of a deployable federated diagnostic platform..

REFERENCES

[1] Sheikh Moonwara Anjum Monisha, Ratun Rahman, "Brain Tumor Detection in MRI Based on Federated Learning with YOLOv11," arXiv preprint arXiv:2503.04087, 2025.

[2] Rasoul Jafari Gohari, Laya Aliahmadipour, Ezat Valipour, "FedBrain-Distill: Communication-Efficient Federated Brain Tumor Classification Using Ensemble Knowledge Distillation on Non-IID Data," arXiv preprint arXiv:2409.05359, 2024.

[3] Ruojun Zhou et al., "Fed-MUnet: Multi-modal Federated Unet for Brain Tumor Segmentation," arXiv preprint arXiv:2409.01020, 2024.

[4] Jingyun Chen, Yading Yuan, "Decentralized Gossip Mutual Learning (GML) for Brain Tumor Segmentation on Multi-parametric MRI," arXiv preprint arXiv:2401.15434, 2024.

[5] Cosmin I. Bercea et al., "FedDis: Disentangled Federated Learning for Unsupervised Brain Pathology Segmentation," arXiv preprint arXiv:2103.03705, 2021.

[6] Dhurgham Hassan Mahlool, Mohamed Hamzah Abed, "Distributed Brain Tumor Diagnosis Using a Federated Learning Environment," Bulletin of Electrical Engineering and Informatics, vol. 11, no. 6, pp. 4131–4138, 2022.

[7] Rabin Dulal, "Brain Tumor Identification Using Improved YOLOv8," arXiv preprint arXiv:2502.03746, 2025.

[8] "Enhancing Brain Tumor Segmentation Accuracy through Scalable Federated Learning with Advanced Data Privacy and Security Measures," Mathematics, vol. 11, no. 19, p. 4189, 2023.

[9] "Integrated Approach of Federated Learning with Transfer Learning for Classification and Diagnosis of Brain Tumor," BMC Medical Imaging, vol. 24, no. 110, 2024.

[10] "Federated Learning with Privacy Preserving for Multi-Institutional Three-Dimensional Brain Tumor Segmentation," Diagnostics, vol. 14, no. 24, p. 2891, 2024.

[11] "A Novel Approach to Brain Tumor Detection Using K-Means++, SGLDM, ResNet50, and Synthetic Data Augmentation," Frontiers in Physiology, vol. 15, p. 1342572, 2024.

[12] "Federated Learning Using Variable Local Training for Brain Tumor Segmentation," in Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, Springer, pp. 392–404, 2022.

[13] "A Review on Brain Tumor Segmentation Based on Deep Learning Methods with Federated Learning Techniques," Computers in Biology and Medicine, vol. 161, p. 106052, 2023.

[14] "Privacy-Preserving Blockchain-Based Federated Learning for Brain Tumor Segmentation," Computers in Biology and Medicine, vol. 162, p. 106059, 2024.

[15] "Federated Learning for Secure Medical MRI Brain Tumor Image Classification," EPJ Special Topics, vol. 234, no. 1, pp. 151–160, 2025.

[16] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv preprint arXiv:1409.1556, 2014.

[17] J. Redmon et al., "You Only Look Once: Unified, Real-Time Object Detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788, 2016.

[18] T. Lin et al., "Focal Loss for Dense Object Detection," in Proceedings of the IEEE

International Conference on Computer Vision, pp. 2980–2988, 2017.

[19] M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," arXiv preprint arXiv:1603.04467, 2016.

[20] A.Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in Advances in Neural Information Processing Systems, vol. 32, 2019.

[21] H. B. McMahan et al., "Communication-Efficient Learning of Deep Networks from Decentralized Data," in Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, pp. 1273–1282, 2017.

[22] J. Konečný et al., "Federated Learning: Strategies for Improving Communication Efficiency," arXiv preprint arXiv:1610.05492, 2016.

[23] S. Ruder, "An Overview of Gradient Descent Optimization Algorithms," arXiv preprint arXiv:1609.04747, 2016.

[24] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," arXiv preprint arXiv:1412.6980, 2014.

[25] Y. LeCun et al., "Gradient-Based Learning Applied to Document Recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[26] Krizhevsky et al., "ImageNet Classification with Deep Convolutional Neural Networks," in Advances in Neural Information Processing Systems, vol. 25, 2012.

[27] K. He et al., "Deep Residual Learning for Image Recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778, 2016.

[28] Szegedy et al., "Going Deeper with Convolutions," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9, 2015.

[29] M. Tan and Q. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in Proceedings of the 36th International Conference on Machine Learning, pp. 6105–6114, 2019.

[30] Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," arXiv preprint arXiv:2010.11929, 2020.

[31] Y. Bengio et al., "Learning Deep Architectures for AI," Foundations and Trends in Machine Learning, vol. 2, no. 1, pp. 1–127, 2009.

[32] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in Proceedings of the 32nd International Conference on Machine Learning, pp. 448–456, 2015.

[33] Glorot and Y. Bengio, "Understanding the Difficulty of Training Deep Feedforward Neural Networks," in Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, pp. 249–256, 2010.

[34] K. He et al., "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," in Proceedings of the IEEE International Conference on Computer Vision, pp. 1026–1034, 2015.

[35] J. Deng et al., "ImageNet: A Large-Scale Hierarchical Image Database," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255, 2009.

[36] O. Russakovsky et al., "ImageNet Large Scale Visual Recognition Challenge," International Journal of Computer Vision, vol. 115, no. 3, pp. 211–252, 2015.

[37] Kendall et al., "Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7482–7491, 2018.

[38] Z. Wang et al., "Image Quality Assessment: From Error Visibility to Structural Similarity," IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600–612, 2004.

[39] T. Y. Lin et al., "Microsoft COCO: Common Objects in Context," in European Conference on Computer Vision, pp. 740–755, 2014.