

Fast Brain Tumor Segmentation with Model Optimization

Mathematical Model and Architectural Overview for Reproducible Research*

Mostafizur RAHMAN

This document provides the architectural framework, preprocessing steps, mathematical formulations, and evaluation results for the proposed brain tumor segmentation model.

1. Data Preprocessing

Our preprocessing pipeline for brain tumor segmentation was designed to ensure consistency and standardization across MRI data. The initial scans underwent a series of preprocessing techniques to prepare them for segmentation.

The first step was the application of **N4ITK bias field correction**, which reduces intensity inhomogeneities caused by magnetic field fluctuations during image acquisition. Let I represent the original voxel intensity and B the estimated bias field. The corrected intensity, denoted as I_c , was computed as:

$$I_c(x, y, z) = \frac{I(x, y, z)}{B(x, y, z)} \quad (1)$$

The various stages of our preprocessing pipeline are shown in Figure 1. Specifically, (a) shows MRI modalities after applying N4ITK bias field correction, (b) depicts the original and cropped MRI slices along with their segmentation masks, and (c) visualizes the normalization process with voxel intensity mapping.

The original MRI scans consist of $155 \times 240 \times 240$ voxels, representing 155 slices per scan with each slice measuring 240×240 pixels. However, the intensity values are often inconsistent due to varying acquisition settings. The BraTS dataset developers applied standard preprocessing: interpolating to 1 mm^3 resolution, co-registering to an anatomical template, and skull stripping.

To ensure compatibility with the deep learning model, all modalities and segmentation masks were **resampled** to a uniform resolution of $128 \times 128 \times 128$ voxels. Image modalities were resampled using trilinear interpolation, while segmentation masks used nearest-neighbor interpolation. Let I_r represent the resampled intensity and f_x, f_y, f_z be the coordinate mapping functions. The resampling process is:

$$I_r(x', y', z') = I(f_x(x'), f_y(y'), f_z(z')) \quad (2)$$

To reduce intensity variability across datasets, voxel intensities were **normalized** to the range $[0, 1]$ as recommended by Patro et al. [?]. Let I_n denote the normalized intensity, and I_{\min} and I_{\max} be the minimum and maximum intensity values in the modality:

$$I_n = \frac{I_r - I_{\min}}{I_{\max} - I_{\min}} \quad (3)$$

*To support reproducible research, the mathematical model presented in this work, including the architectural framework and relevant formulations, has been made publicly available at <https://github.com/Rahman768/RMM>.

This ensures that all voxel intensities are normalized for consistent model input. The final output was a single 4D array, denoted as A , with dimensions $C \times H \times W \times D$, where C is the number of channels (modalities). The segmentation masks and 4D input arrays were stored in .h5 format using GZIP compression for efficiency.

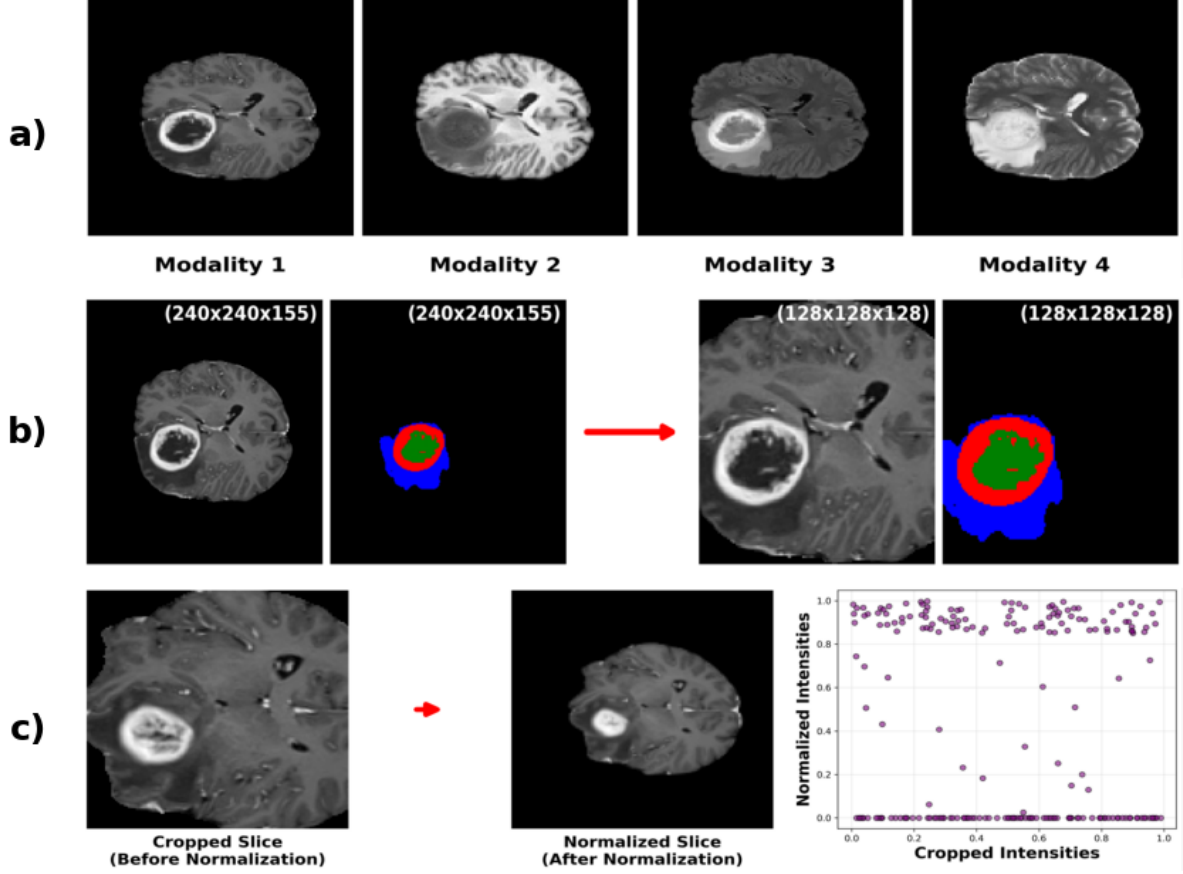


Figure 1: MRI Preprocessing Steps: (a) Bias Correction, (b) Cropping with Masks, (c) Intensity Normalization Mapping

2. Proposed Architecture Overview

2.1. Overview of the Framework

Figure 2 presents an overview of the proposed brain tumor segmentation framework, which integrates pseudo-3D convolutions, attention mechanisms, pruning strategies, quantization-aware training (QAT), and advanced loss functions to achieve high efficiency and accuracy.

The pipeline begins with the input stage, which accepts 4-channel 3D medical images of dimensions $4 \times 128 \times 128 \times 128$. This input is first processed by an initial pseudo-3D convolution layer. Instead of traditional 3D convolutions, it applies spatial $3 \times 3 \times 1$ and depthwise $1 \times 1 \times 3$ convolutions. This step reduces the feature map size to $32 \times 64 \times 64 \times 64$ with a stride of 2.

The encoder stage consists of two sets of Dilated Multi-Fiber (DMF) units enhanced with Efficient Channel Attention (ECA) modules and structured pruning for computational optimization. The first DMF set reduces the tensor size to $128 \times 32 \times 32 \times 32$ with pruning reducing group count from $g = 16$ to $g = 8$. The second DMF set further downsamples to $256 \times 16 \times 16 \times 16$, with additional pruning.

Each DMF unit employs pseudo-3D convolutions, and ECA modules refine the channel-wise importance. This is followed by the bottleneck, implemented as a DMF unit with adaptive

dilation. It dynamically adjusts dilation rates to enhance multi-scale feature capture, and reduces the dimensions from $384 \times 8 \times 8 \times 8$ to $256 \times 8 \times 8 \times 8$.

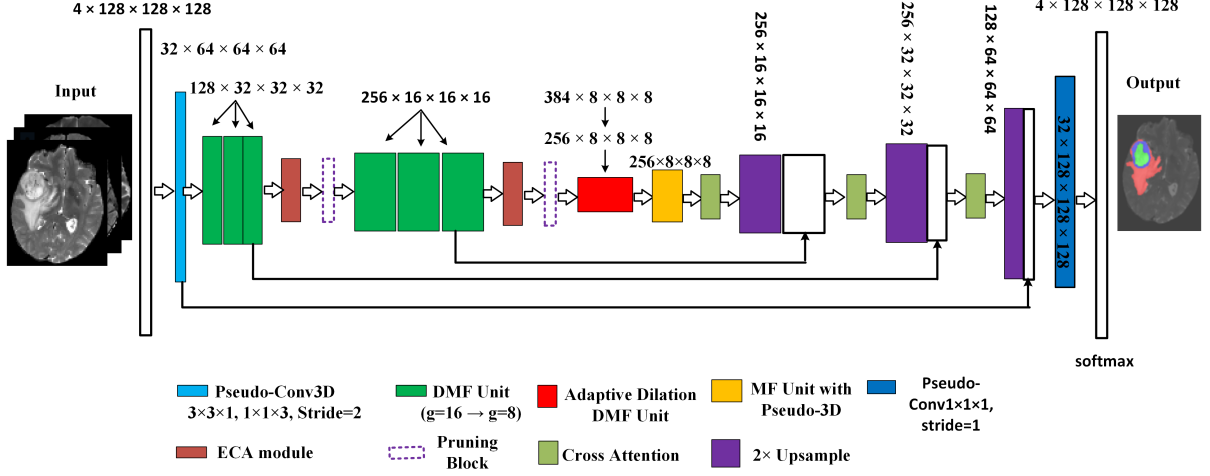


Figure 2: Overview of the Proposed Brain Tumor Segmentation Framework

The decoder reconstructs the segmentation map by progressively upsampling feature maps while integrating skip connections from the encoder:

- The first decoder stage is a Multi-Fiber (MF) unit with cross-attention, outputting $256 \times 8 \times 8 \times 8$.
- The first upsampling block doubles the spatial resolution to $256 \times 16 \times 16 \times 16$ and incorporates skip connections from the second DMF set.
- The second upsampling block outputs $256 \times 32 \times 32 \times 32$ and includes skip connections from the first DMF set.
- The final block outputs $128 \times 64 \times 64 \times 64$ using skip connections from the initial pseudo-3D convolution layer.

A final pseudo-3D convolution layer reduces the tensor to $32 \times 128 \times 128 \times 128$ using a $1 \times 1 \times 1$ convolution. This is followed by the output stage, consisting of an upsampling block and a softmax layer, both outputting the final segmentation map of dimensions $4 \times 128 \times 128 \times 128$, representing tumor regions for enhancing tumor (ET), tumor core (TC), and whole tumor (WT).

Additional architectural optimizations include:

- Quantization-Aware Training (QAT) to convert the model to INT8 precision for faster inference while preserving accuracy.
- Use of a combination of Dice Loss and Boundary Loss to improve boundary precision and detect small lesions effectively.
- Adaptive dilation within the bottleneck, replacing fixed rates to enhance multi-scale decoding.
- Use of sub-pixel convolution instead of trilinear interpolation in the decoder, improving output segmentation detail.

Overall, the integration of pseudo-3D convolutions, lightweight attention mechanisms, pruning, QAT, and refined loss functions results in a highly efficient and accurate segmentation architecture.

2.2. Key Components of the Proposed Architecture

This section describes the core modules that make up the proposed framework. These include Residual Units with Pseudo-3D Convolutions, Multi-Fiber (MF) and Dilated Multi-Fiber (DMF) Units, and adaptive fusion mechanisms.

(1) Residual Unit with Pseudo-3D Convolutions The Residual Unit with Pseudo-3D Convolutions, shown in Figure 3(a), enhances computational efficiency while preserving feature representation. The input channel C_{in} undergoes a spatial convolution ($3 \times 3 \times 1$) producing an intermediate C_{mid} , followed by a depthwise convolution ($1 \times 1 \times 3$) producing the output C_{out} . A residual shortcut connects C_{in} directly to C_{out} using element-wise addition. This promotes gradient flow and efficient feature propagation, all while maintaining low computational cost.

(2) Multi-Fiber Design with Grouped Residual Units As depicted in Figure 3(b), the multi-fiber block introduces three parallel residual units with pseudo-3D convolutions. Each fiber processes $\frac{C_{\text{in}}}{g}$ channels, where g is the number of groups. Each fiber performs a spatial ($3 \times 3 \times 1$) convolution followed by a depthwise ($1 \times 1 \times 3$) convolution. Residual connections within each fiber ensure preservation of information. This design reduces parameter count by a factor of g , improving efficiency.

(3) MF Unit with Adaptive Fusion The MF Unit (Figure 3(c)) gathers features from all fibers and applies adaptive fusion through a learnable weighted summation. Each fiber output F_i is assigned a scalar weight w_i , enabling dynamic feature integration:

$$F_o = w_1 \cdot F_1 + w_2 \cdot F_2 + w_3 \cdot F_3 \quad (4)$$

This mechanism allows the network to prioritize fiber responses based on task-specific importance. Additionally, an auxiliary loss branch stemming from the output provides intermediate supervision during training, helping accelerate convergence and enhance learning.

(4) DMF Unit with Adaptive Weighting The DMF Unit, illustrated in Figure 3(d), captures multi-scale features through three parallel branches with distinct dilation rates ($d = 1, 2, 3$). Each branch uses pseudo-3D convolutions (spatial and depthwise), followed by adaptive weights w_1, w_2, w_3 to combine outputs. This weighted sum effectively aggregates multi-scale features while limiting FLOPs.

(5) Schematic of 3D Dilated vs. Pseudo-3D Convolution Operations To highlight the computational benefits of our architecture, Figure 4 compares conventional 3D dilated convolutions (left panel) with our pseudo-3D decomposition (right panel). The dilated convolutions use increasing dilation rates ($d = 1, 2, 3$) to expand the receptive field. However, they require expensive 3D kernels, increasing computational complexity.

In contrast, the pseudo-3D strategy replaces the $3 \times 3 \times 3$ kernel with a sequential application of $3 \times 3 \times 1$ (spatial) and $1 \times 1 \times 3$ (depthwise) kernels. This approach reduces FLOPs while retaining spatial and volumetric context — ideal for volumetric medical image segmentation.

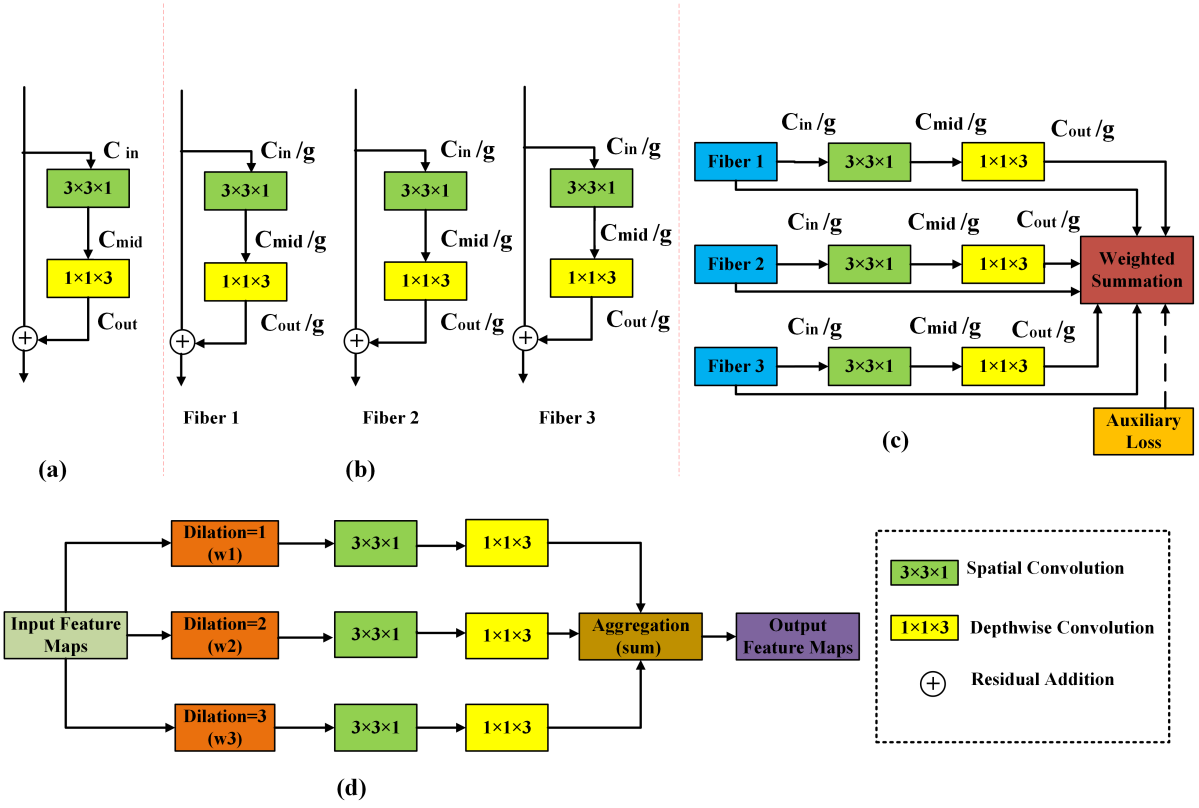


Figure 3: Built Blocks in the Proposed Architecture: (a) Residual Unit with Pseudo-3D Convolutions, (b) Multi-Fiber with Grouped Residual Units, (c) MF Unit with Adaptive Fusion, (d) DMF Unit with Adaptive Weighting

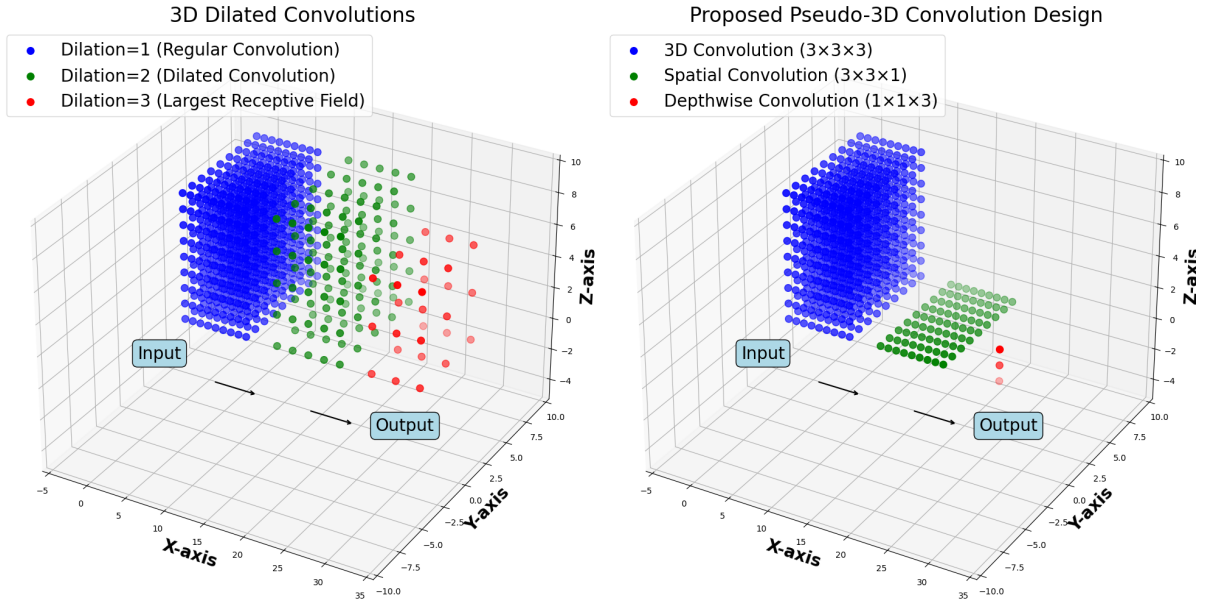


Figure 4: Comparison of 3D Dilated vs. Pseudo-3D Convolution Operations

3. Mathematical Formulations

3.1. Loss Functions

The training used a compound loss function that combines Dice Loss and Boundary Loss, applied to both the main and auxiliary outputs of the network. The total loss $\mathcal{L}_{\text{total}}$ is defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{main}} + \lambda_{\text{aux}} \cdot \mathcal{L}_{\text{aux}}$$

where:

$$\begin{aligned}\mathcal{L}_{\text{main}} &= \alpha \cdot \mathcal{L}_{\text{Dice}}(\hat{y}, y) + \beta \cdot \mathcal{L}_{\text{Boundary}}(\hat{y}, y) \\ \mathcal{L}_{\text{aux}} &= \alpha \cdot \mathcal{L}_{\text{Dice}}(\hat{y}_{\text{aux}}, y) + \beta \cdot \mathcal{L}_{\text{Boundary}}(\hat{y}_{\text{aux}}, y)\end{aligned}$$

Here:

- \hat{y} and \hat{y}_{aux} are the main and auxiliary predictions.
- y is the ground truth segmentation.
- $\mathcal{L}_{\text{Dice}}$ measures volumetric overlap.
- $\mathcal{L}_{\text{Boundary}}$ penalizes discrepancies along tumor borders.
- α, β are weights for the Dice and Boundary components.
- λ_{aux} is the weight for the auxiliary output (in our case, $\lambda_{\text{aux}} = 0.4$).

3.2. Adaptive Fusion in MF and DMF Units

Let F_i denote the output of the i -th parallel branch in the MF or DMF unit. The final output is computed as a weighted sum:

$$F_{\text{fused}} = \sum_{i=1}^n \alpha_i \cdot F_i \quad \text{where} \quad \sum_{i=1}^n \alpha_i \text{ are learnable and task-dependent}$$

This adaptive fusion mechanism enables the model to dynamically weigh contributions from different receptive fields (in DMF) or fibers (in MF), improving multi-scale feature representation without excessive parameter cost.

4. Experimental Results and Analysis

4.1. Experimental Setup

To evaluate the effectiveness of the proposed brain tumor segmentation framework, a series of experiments were conducted. The experiments were performed on a high-performance system featuring an Intel Core i9-14900K processor with 24 cores at 6.0 GHz, an Nvidia GeForce RTX 3090 GPU (24 GB VRAM), 1TB HDD, 500GB SSD, and 48 GB RAM. The model was developed using Python 3.9, CUDA 11.8, and PyTorch 2.0.0. Additionally, Keras with TensorFlow 2.15 was used for architecture design and testing.

4.2. Dataset and Training Settings

The model was trained and evaluated using the BraTS 2020 [1] and BraTS 2021 [2] benchmark datasets. BraTS 2021 includes 1251 patient cases with both high-grade gliomas (HGG) and low-grade gliomas (LGG), while BraTS 2020 includes 369 cases.

Each case consists of four MRI modalities: T1, T1ce, T2, and FLAIR, with original resolution $155 \times 240 \times 240$. Figure 5 shows an example of multi-modal scans and corresponding expert-labeled ground truth.

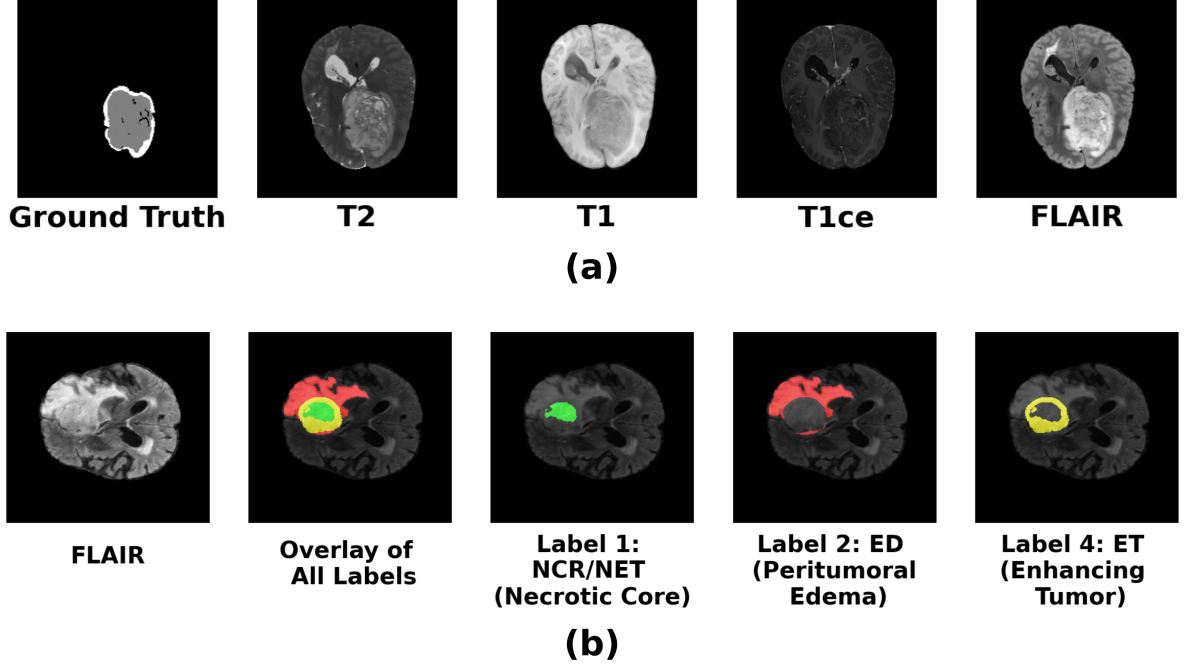


Figure 5: Visual Examples of the Dataset: (a) Multi-modal MRI scans and expert-labeled ground truth for a patient. (b) Ground truth segmentation: green = NCR/NET (label 1), red = ED (label 2), yellow = ET (label 4).

Each segmentation mask includes four labels: background (0), NCR/NET (1), edema (2), and enhancing tumor (4). For performance evaluation, these are grouped into three key subregions:

- **Whole Tumor (WT)** = NCR/NET + ED + ET
- **Tumor Core (TC)** = NCR/NET + ET
- **Enhancing Tumor (ET)** = ET

Five-fold cross-validation was performed, with a 4:1 training-validation split. Training was performed for 400 epochs using the Adam optimizer with an initial learning rate of 1×10^{-4} . A batch size of 2 was used with gradient accumulation to simulate larger batch training. L2 regularization with factor 0.02 was applied to convolutional kernels. The hybrid loss function combined Dice Loss and Boundary Loss.

Quantization-Aware Training (QAT) was applied to convert the model to INT8 precision. Collapsible structured pruning was used to remove redundant parameters. Data augmentation included random flips, spatial shifts, and Gaussian noise (std = 0.01).

4.3. Evaluation Metrics

Segmentation quality was assessed using the following metrics:

Dice Coefficient (also F₁-score):

$$\text{Dice} = \frac{2 \cdot |P \cap T|}{|P| + |T|} \quad (5)$$

Since ET, TC, and WT segmentation are binary sub-tasks, the Dice score is equivalent to the F₁-score:

$$F_1 = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \quad (6)$$

Sensitivity (Recall):

$$\text{Sensitivity} = \frac{|P \cap T|}{|T|} \quad (7)$$

Specificity:

$$\text{Specificity} = \frac{|P^c \cap T^c|}{|T^c|} \quad (8)$$

95th Percentile Hausdorff Distance (HD95):

$$\text{HD95}(A, B) = \max \left(\sup_{a \in A} \inf_{b \in B} d(a, b), \sup_{b \in B} \inf_{a \in A} d(a, b) \right) \quad (9)$$

GFLOPs (Computational Efficiency):

$$\text{GFLOPs} = \sum_{i=1}^L C_i \cdot H_i \cdot W_i \cdot K_i^2 \quad (10)$$

where L is the number of layers, C_i is the number of input channels, H_i, W_i are the spatial dimensions, and K_i is the kernel size.

Inference Time: Measured in seconds per 3D volume, it reflects the model’s real-time applicability in clinical environments.

5. Summary of Results

5.1. Quantitative Results

To evaluate the effectiveness of the proposed model, we present a detailed comparison with state-of-the-art brain tumor segmentation methods using the BraTS 2021 and BraTS 2020 datasets. Tables 1 and 2 show segmentation performance in terms of Dice score, sensitivity, specificity, and HD95 for WT, TC, and ET tumor regions. Table 3 compares computational efficiency in terms of parameters and FLOPs.

Table 1: Comparison of Segmentation Results of Different Models on BraTS 2021 Validation Set

Model	Mean Dice (%)			Sensitivity (%)			Specificity (%)			HD95 (mm)		
	WT	TC	ET	WT	TC	ET	WT	TC	ET	WT	TC	ET
3D U-Net [3]	88.53	86.73	82.23	90.23	86.17	82.05	99.85	99.72	99.67	6.23	9.73	9.67
TransBTS [4]	88.75	85.51	83.42	90.31	84.17	82.33	99.87	99.68	99.85	5.73	9.82	5.44
TransUNet [5]	89.23	83.31	81.54	92.66	85.73	84.76	99.82	99.77	99.62	6.10	7.43	9.14
Swin UNETR [6]	91.05	88.11	84.63	94.32	89.56	87.45	99.87	99.82	99.72	6.42	7.17	9.76
Proposed	91.85	88.52	85.55	94.88	90.76	89.56	99.95	99.72	99.97	2.58	3.53	3.65

Table 2: Comparison of Segmentation Results of Different Models on BraTS 2020 Validation Set

Model	Mean Dice (%)			Sensitivity (%)			Specificity (%)			HD95 (mm)		
	WT	TC	ET	WT	TC	ET	WT	TC	ET	WT	TC	ET
3D U-Net [3]	87.76	83.22	80.21	90.21	85.42	81.77	99.81	99.72	99.85	5.17	9.33	9.45
TransBTS [4]	88.10	86.22	82.23	91.78	89.34	85.75	99.87	99.84	99.81	4.97	9.06	8.41
TransUNet [5]	88.66	84.54	80.22	92.77	89.32	82.86	99.92	99.86	99.81	6.67	8.13	9.42
Swin UNETR [6]	91.10	86.51	83.17	94.32	88.67	87.76	99.95	99.87	99.87	5.63	6.25	7.34
Proposed	90.63	87.16	84.31	94.88	90.86	89.66	99.94	99.89	99.97	3.37	5.98	4.52

Table 3: Comparison of Model Efficiency in Terms of FLOPs and Parameters on BraTS Benchmark Dataset

Model	Parameters (M)	FLOPs (G)
3D U-Net [3]	16.21	1670
VcaNet [7]	79.32	1140.67
Swin UNETR [6]	61.98	394.84
TransBTS [4]	32.99	333
DResU-Net [8]	30.47	374.04
HNF-Netv2 [9]	17.91	449.79
DMFNet [10]	3.88	27.04
LATUP-Net [11]	3.07	15.79
Proposed	3.57	21.26

The proposed model achieves significant improvements over existing methods on both BraTS 2021 and BraTS 2020 validation datasets. On BraTS 2021, it attained the highest Mean Dice scores and best boundary delineation with an HD95 of 2.58 mm for WT. Similar trends are observed for TC and ET, where the model consistently outperformed others in both segmentation accuracy and boundary precision.

On BraTS 2020, the model achieved strong performance, with HD95 values of 3.37 mm (WT) and 5.98 mm (TC), indicating excellent generalization. Compared to lighter architectures like LATUP-Net and DMFNet, the proposed model strikes a better balance between segmentation accuracy, inference speed, and computational efficiency.

Its lightweight architecture leverages pseudo-3D convolutions, adaptive DMF units, ECA modules, and structured pruning with QAT. This results in a compact design (3.57M parameters, 21.26 GFLOPs) and fast inference (0.016 seconds/volume on an NVIDIA RTX 3090), making it suitable for real-time clinical deployment.

5.2. Qualitative Analysis

To provide an intuitive comparison of segmentation performance, we present visualizations from competing methods alongside our proposed model, using cases from the BraTS 2021 and BraTS 2020 datasets. Figure 6 shows 2D overlaid segmentation outputs, while Figure 7 displays 3D renderings of segmentation results from the proposed model.

The outputs from the proposed model demonstrate clear improvements in delineating the tumor subregions — Whole Tumor (WT), Tumor Core (TC), and Enhancing Tumor (ET). In Figure 6, TransUNet displays minor estimation errors and some incorrect prediction regions in the final rows. These errors may stem from the model’s limited ability to effectively leverage cross-modality information.

In contrast, the proposed model produces more accurate segmentations, particularly for WT and TC regions, while maintaining competitive accuracy for ET. This performance is primarily due to two innovations: (1) replacing traditional 3D convolutions with pseudo-3D convolutions to reduce computational load while preserving volumetric context, and (2) using a hybrid loss function combining Dice Loss and Boundary Loss to enhance precision at the boundaries and detect small lesions effectively.

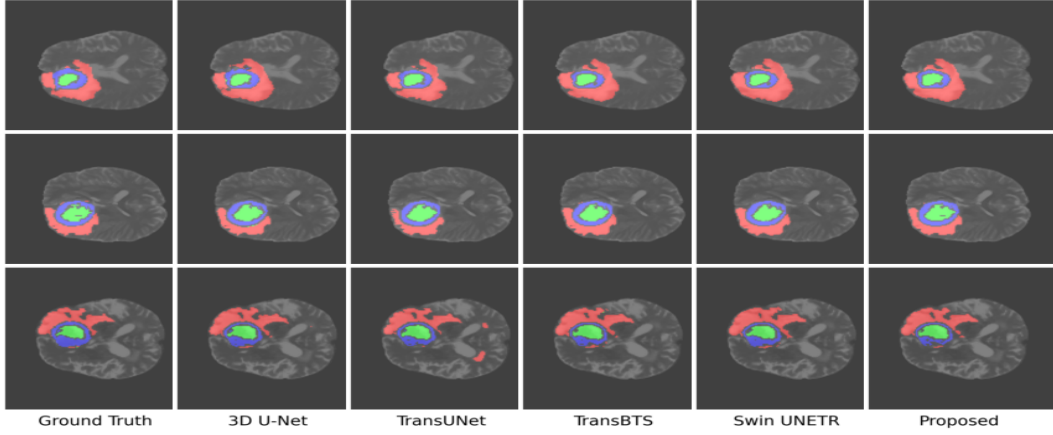


Figure 6: Visualization and Comparison of Segmentation Results from Various Methods on the BraTS 2021 Dataset. Overlay colors: Red = WT, Green = TC, Blue = ET.

Figure 7 shows 3D segmentation results for several sample volumes. The first two rows contain samples from BraTS 2021 cases (BraTS2021_00216, 00266, 00336, 00789, 00816, and 01324). The third row includes examples from BraTS 2020 cases (BraTS20_Training_047, 053, and 338).

These results illustrate the model’s ability to accurately segment tumor subregions in 3D, demonstrating strong performance in both BraTS 2021 and 2020 datasets by preserving anatomical structures and capturing subtle tumor boundaries.

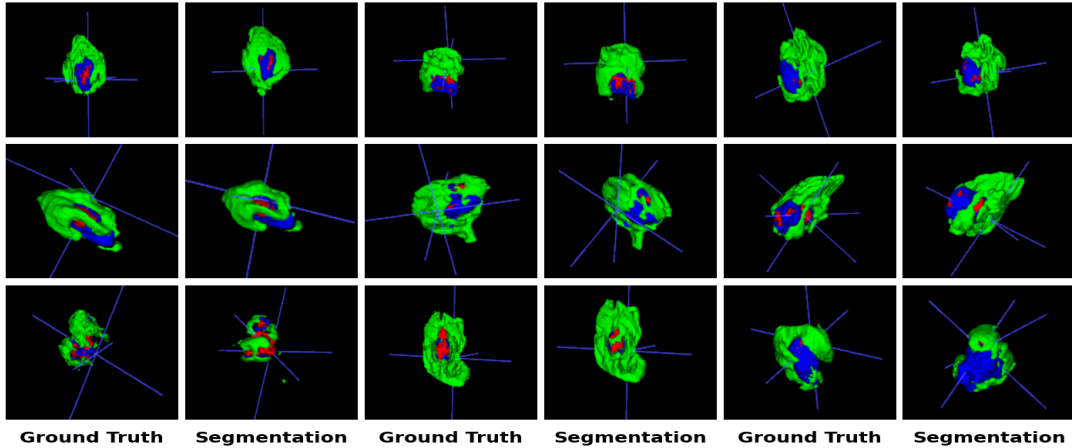


Figure 7: 3D Segmentation Results of Samples from the BraTS 2021 and BraTS 2020 Datasets. Overlay colors: Red = WT, Green = TC, Blue = ET.

6. Conclusion and Future Directions

The research findings demonstrate that the designed framework delivers superior brain tumor segmentation results with high efficiency across the BraTS 2021 and 2020 datasets. The in-

tegration of pseudo-3D convolutions with Adaptive DMF units, MF units, ECA modules, and structured pruning contributes to enhanced multi-scale representation, reduced computational complexity, and faster inference—achieving a runtime of only 0.016 seconds per 3D volume.

The lightweight system architecture, with just 3.57M parameters and 21.26 GFLOPs, maintains high segmentation performance. Quantization-Aware Training (QAT) further improves inference efficiency, making the model suitable for real-time clinical deployment without compromising segmentation accuracy.

However, the use of BraTS domain-specific data restricts the generalizability of this framework to other medical imaging modalities. The model’s practical deployment will require addressing variability due to MRI scanner inconsistencies and broader imaging contexts.

Future work will involve:

- Evaluating the framework on diverse datasets such as liver, knee, and cardiac imaging.
- Applying domain adaptation and self-supervised learning techniques to improve robustness and generalization.
- Exploring practical clinical deployment strategies in applications such as preoperative planning, radiotherapy, and automated diagnosis.

With its efficiency and precision, the proposed segmentation framework holds great promise for real-world medical imaging applications.

References

- [1] Menze, B.H., Jakab, A., Bauer, S., et al. The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). *IEEE Transactions on Medical Imaging*, vol. 34, no. 10, 2014, pp. 1993–2024. DOI: 10.1109/TMI.2014.2377694
- [2] Baid, U., Ghodasara, S., Mohan, S., et al. The RSNA-ASNR-MICCAI BraTS 2021 Benchmark on Brain Tumor Segmentation and Radiogenomic Classification. *arXiv preprint arXiv:2107.02314*, 2021. DOI: 10.48550/arXiv.2107.02314
- [3] Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., et al. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, Lecture Notes in Computer Science, vol. 9901, Springer, Cham, 2016, pp. 424–432. DOI: 10.1007/978-3-319-46723-8_49
- [4] Wenxuan, W., Chen, C., Meng, D., et al. TransBTS: Multimodal Brain Tumor Segmentation Using Transformer. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention – MICCAI 2021*, Springer, Cham, 2021, pp. 109–119. DOI: 10.1007/978-3-030-87199-4_11
- [5] Chen, J., Lu, Y., Yu, Q., et al. TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation. *arXiv preprint arXiv:2102.04306*, 2021. DOI: 10.48550/arXiv.2102.04306
- [6] Hatamizadeh, A., Nath, V., Tang, Y., et al. Swin UNETR: Swin Transformers for Semantic Segmentation of Brain Tumors in MRI Images. In: *BrainLes 2021: Brain Lesion Workshop*, Lecture Notes in Computer Science, vol. 12962, Springer, Cham, 2022, pp. 272–284. DOI: 10.1007/978-3-031-08999-2_22

- [7] Pan, D., Shen, J., Al-Huda, Z., et al. VcaNet: Vision Transformer with Fusion Channel and Spatial Attention Module for 3D Brain Tumor Segmentation. *Computers in Biology and Medicine*, vol. 186, March 2025. DOI: 10.1016/j.compbimed.2025.109662
- [8] Raza, R., Bajwa, U.I., Mehmood, Y., et al. dResU-Net: 3D Deep Residual U-Net Based Brain Tumor Segmentation from Multimodal MRI. *Biomedical Signal Processing and Control*, vol. 79, 2023, p. 103861. DOI: 10.1016/j.bspc.2022.103861
- [9] Jia, H., Bai, C., Cai, W., et al. HNF-Netv2 for Brain Tumor Segmentation Using Multimodal MR Imaging. In: *International MICCAI Brainlesion Workshop*, Lecture Notes in Computer Science, vol. 12963, Springer, Cham, 2021, pp. 106–115. DOI: 10.1007/978-3-031-09002-8_10
- [10] Chen, C., Liu, X., Ding, M., et al. 3D Dilated Multi-Fiber Network for Real-Time Brain Tumor Segmentation in MRI. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2019*, Lecture Notes in Computer Science, vol. 11766, Springer, Shenzhen, 2019, pp. 314–322. DOI: 10.1007/978-3-030-32248-9_21
- [11] Alwadee, E.J., Sun, X., Qin, Y., et al. LATUP-Net: A Lightweight 3D Attention U-Net with Parallel Convolutions for Brain Tumor Segmentation. *Computers in Biology and Medicine*, vol. 184, January 2025, article 109353. DOI: 10.1016/j.compbimed.2024.109353