



Brain Tumor Segmentation & Survival Prediction in MRI Scans

Ehsan Taheri
Reza Farnaghi
Ali Yazdani

Brain Tumor Segmentation in MRI

The Challenge

Precisely delineating tumor regions within 3D MRI scans is crucial for diagnosis, treatment planning, and monitoring tumor response. However, tumor morphology and intensity variation make segmentation a complex task.

BraTS Dataset

We utilize the BraTS 2020 dataset , which contains multi-modal MRI scans of brain tumor patients. This dataset provides a rich resource for training and evaluating segmentation models.

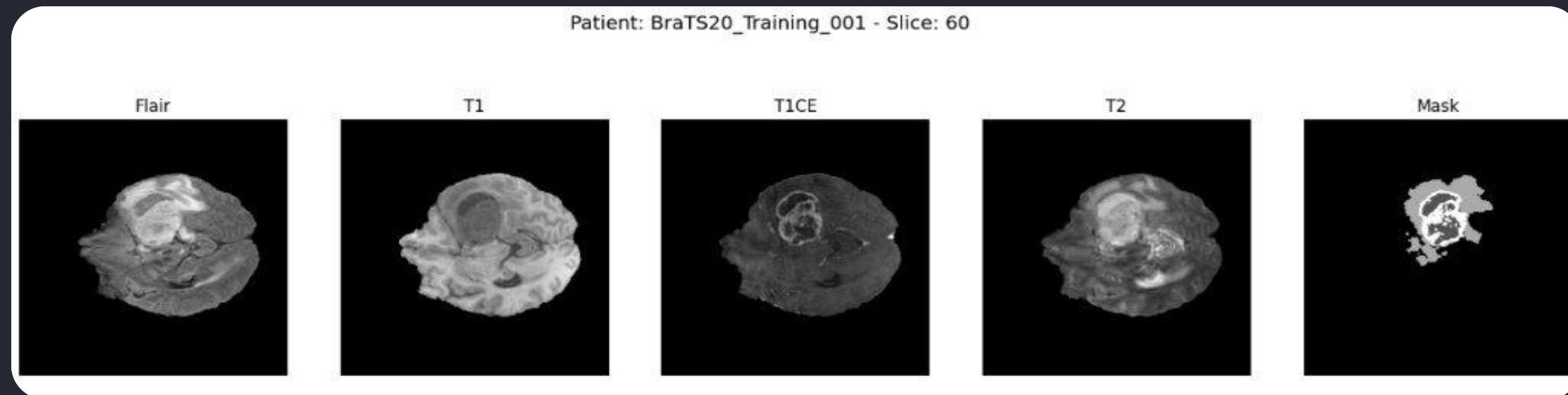
Brain Tumor Segmentation in MRI

Objective

- Develop an automated deep learning-based brain tumor segmentation system using MRI images.
- Improve accuracy, robustness, and generalizability in tumor detection and segmentation.

Project Goals:

- Efficiently preprocess MRI images for deep learning.
- Handle class imbalance by using data augmentation and oversampling.
- Train a deep learning model (U-Net with Attention Mechanism) for segmentation.
- Evaluate performance using key metrics: Dice Score, Accuracy, and Loss.
- Post-process the predictions to refine segmentation masks.



Data Exploration and Visualization

Dataset:

BraTS 2020 Dataset (Brain Tumor Segmentation Challenge)

Total Patients: 369

MRI Modalities: Flair, T1, T1CE, T2

Image Dimensions: $240 \times 240 \times 155$ (width \times height \times depth)

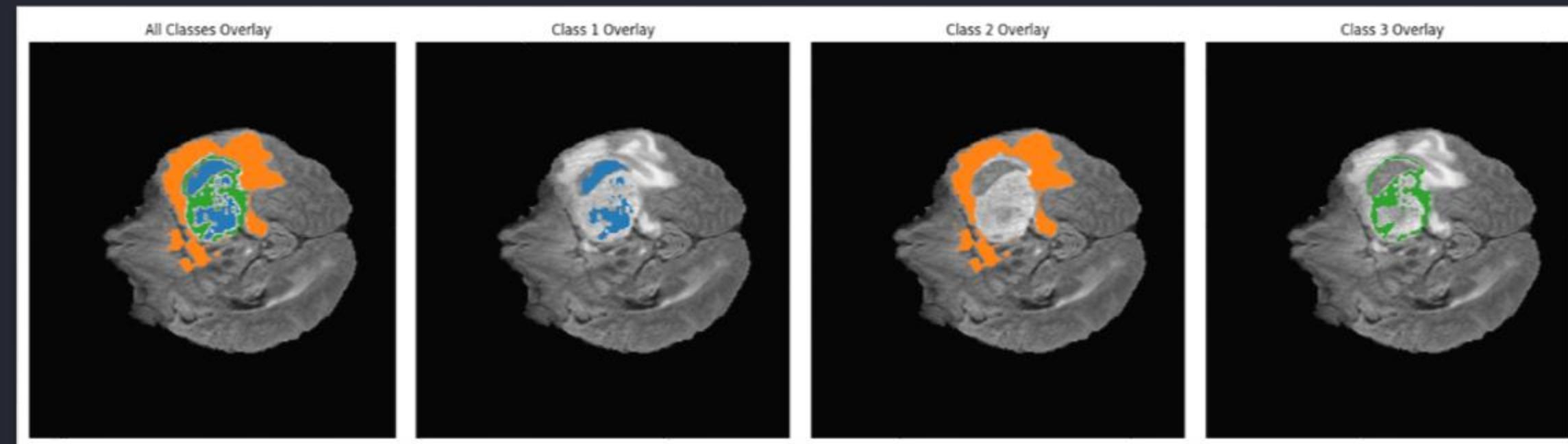
Segmentation Labels:

Class 0 - Background (No Tumor)

Class 1 - Necrotic/Core Tumor(Dead Tumor Tissue)

Class 2 - Edema(Swelling Around Tumor):

Class 3 - Enhancing Tumor(Active Growing Tumor)



MRI Image Slices & Mask Distribution

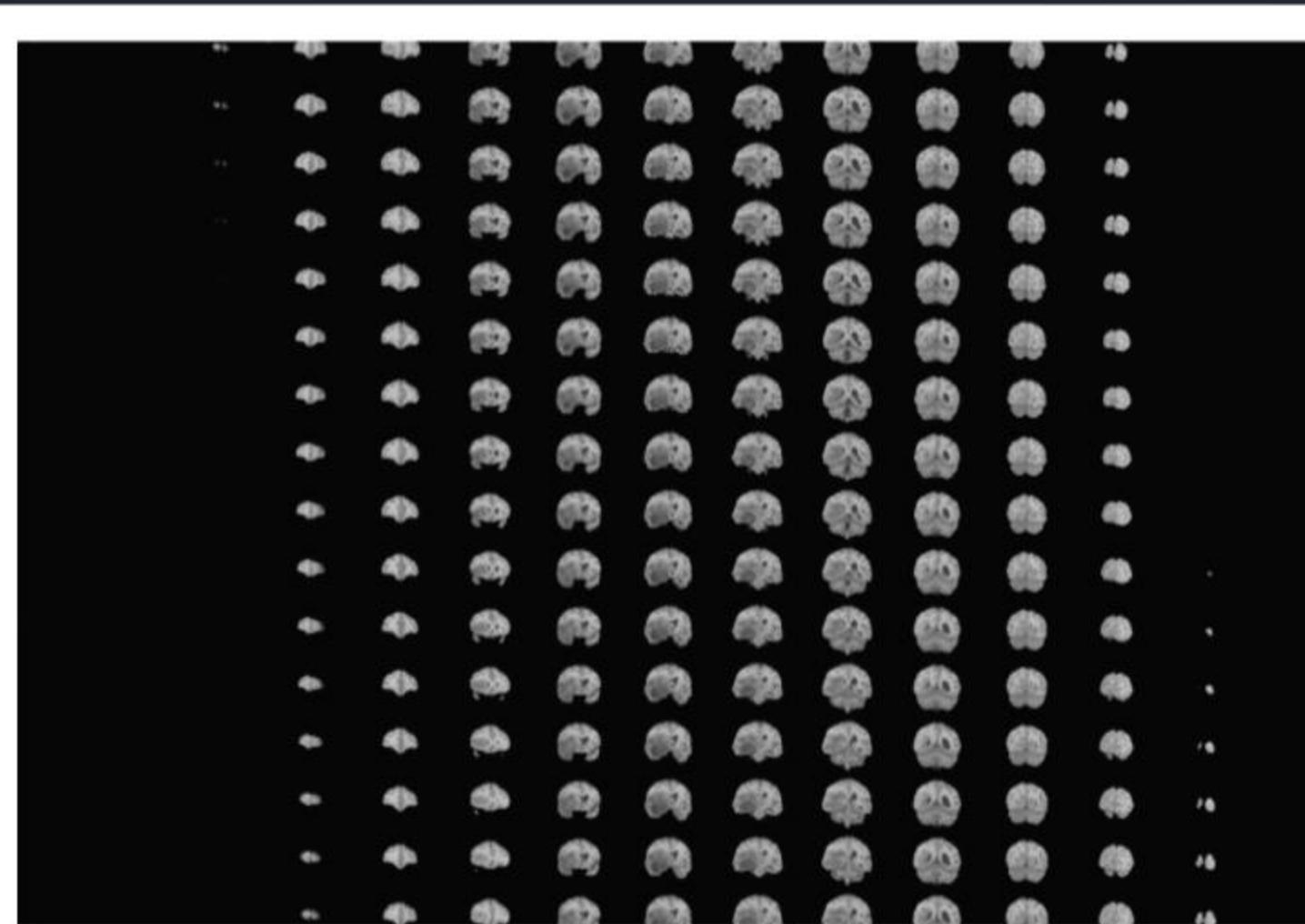
- MRI scans contain multiple 2D slices stacked together to form a 3D volume.
- Many of these slices contain no tumor (black slices), primarily at the beginning and end of the 3D volume.

Example

Patient BraTS20_Training_001

Total Slices: 155

Black (Empty) Slices: 72 (46.45% of the total slices)

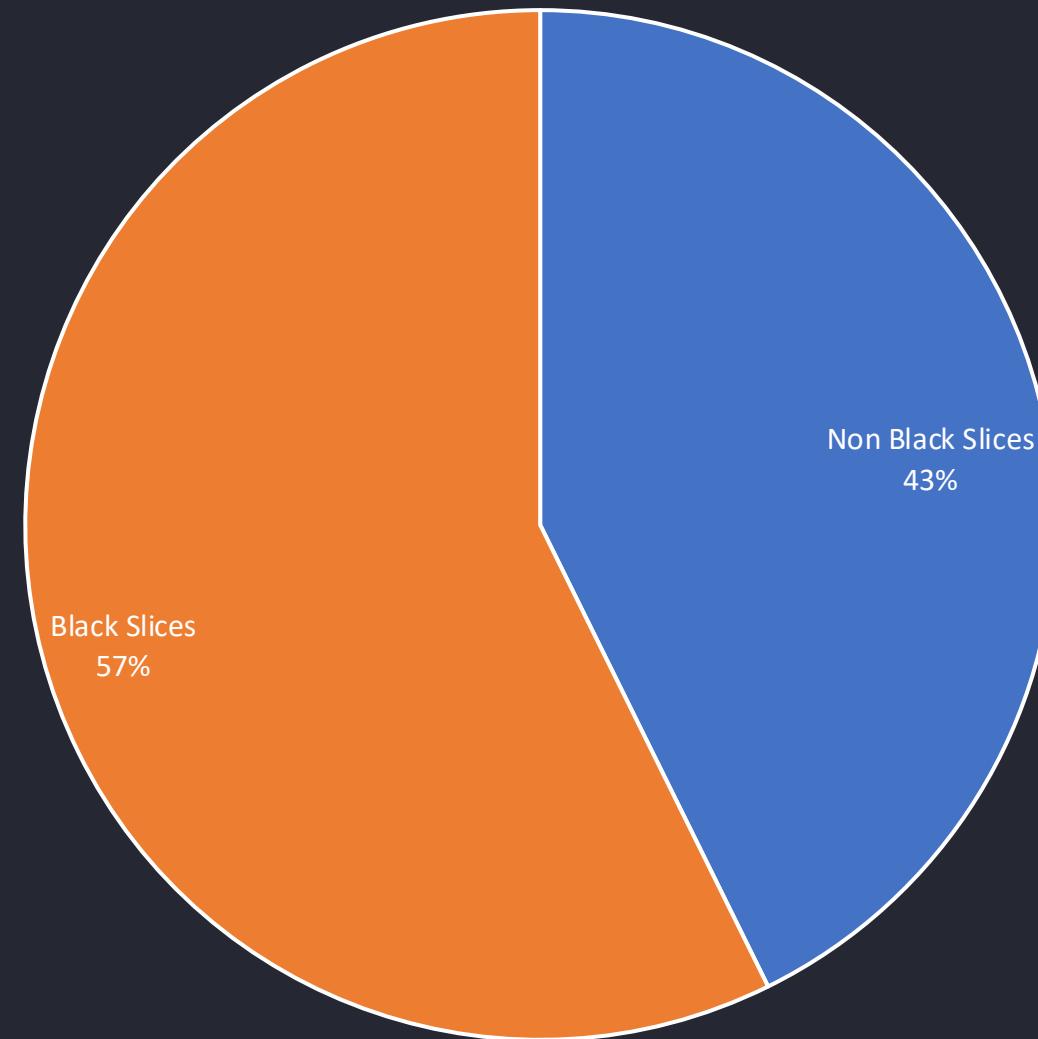


MRI Image Slices & Mask Distribution

Across the entire dataset:

- Total Slices: 57,040
- Black Slices: 32,686 (57.3%)
- These black slices can introduce bias in training if not handled properly.

Proportion of Black VS NonBlack Slices

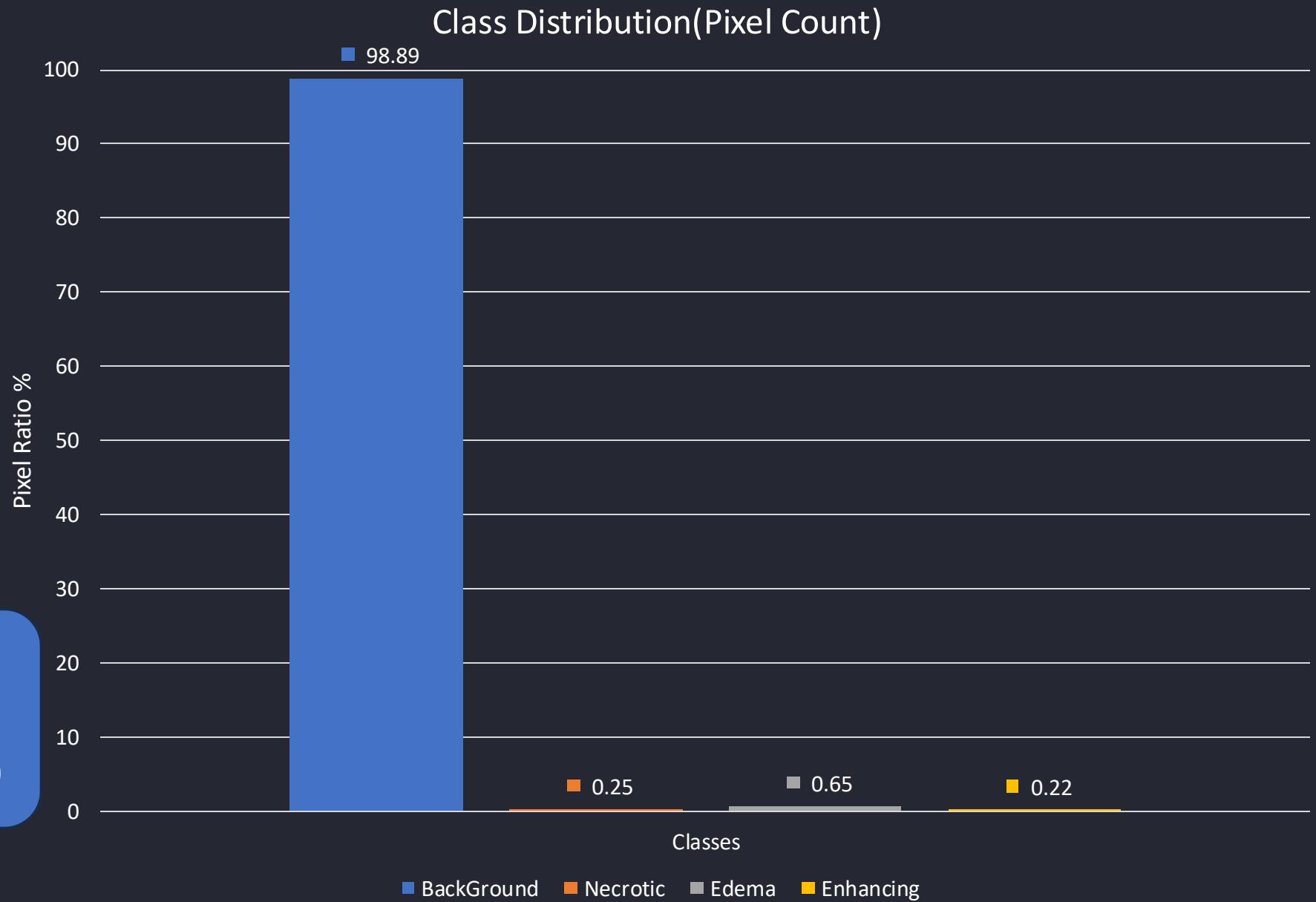


■ Non Black Slices ■ Black Slices

Pixel Distribution Across the Dataset

- The dataset is highly imbalanced, with a majority of pixels belonging to the background (healthy tissue) and significantly fewer pixels representing different tumor types.
- This imbalance can negatively impact model training, leading to biased predictions favoring background regions.

- Background : 3,248,909,300 (98.89%)
- Necrotic/Core : 8,129,761 (0.25%)
- Edema 21,256,835 (0.65%)
- Enhancing Tumor : 7,208,104 (0.22%)

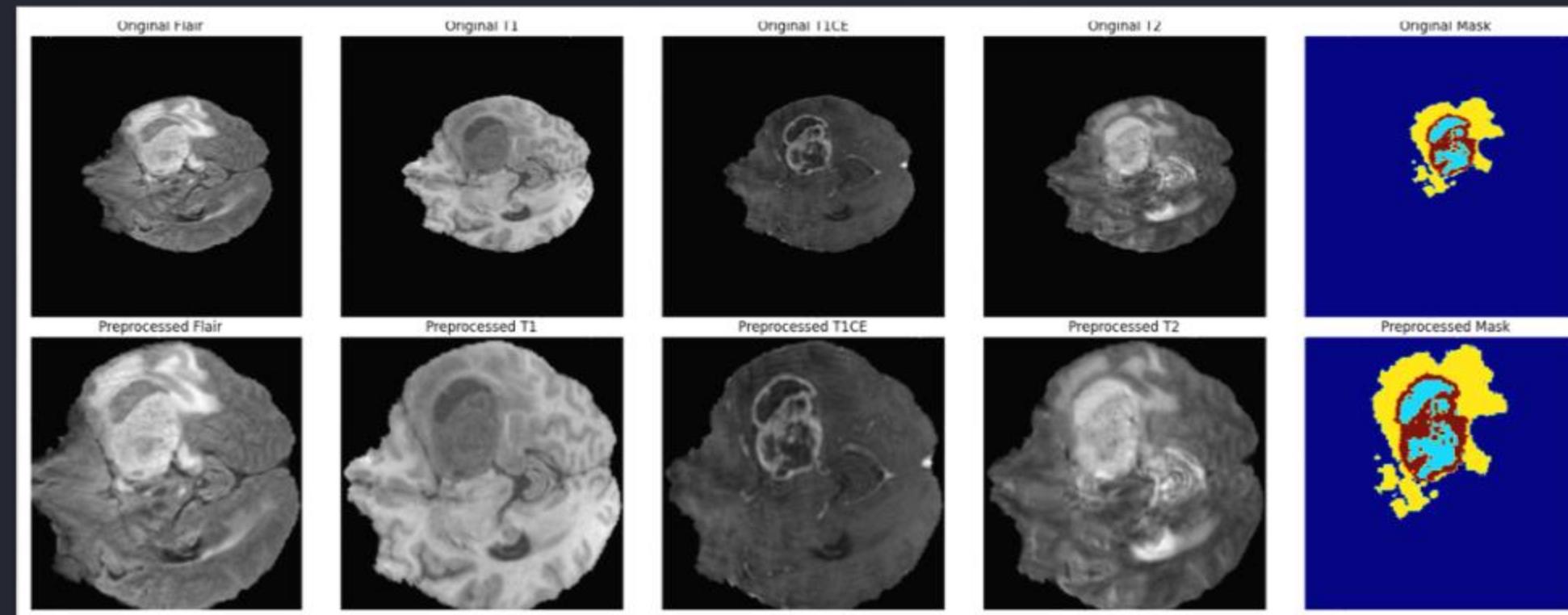


More than 98% of pixels belong to the background! Only ~1% of the dataset contains tumor-related pixels.

Strategies to Address Class Imbalance

1- Cropping and Removing Black Slices

- Crop to the region containing the brain to remove unnecessary black areas.
- Keep at most 5 black slices per patient to maintain a balanced dataset.
- Fully discard excessive empty slices with no tumor presence.
- Result: Reduced dataset size while keeping essential information.



Strategies to Address Class Imbalance

2- Oversampling Important Slices

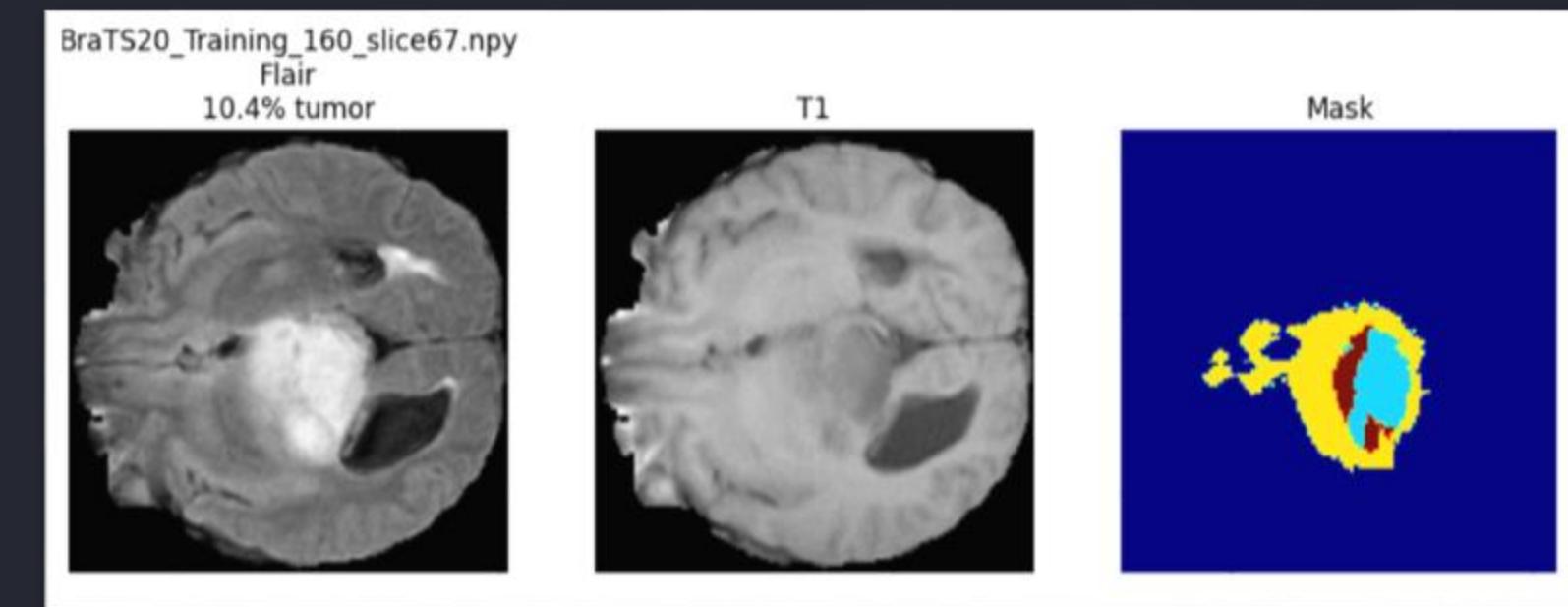
Since slices with tumors are underrepresented, we oversampled slices with higher tumor presence.

We defined:

- Big Tumor Slices: Slices where tumor pixels cover more than 2% of the area.
- Small Tumor Slices: Slices where tumor pixels cover less than 2% of the area.

Final Preprocessed Data Distribution:

- o Big Tumor Slices: 19,543
- o Small Tumor Slices: 6,651
- o Train Samples: 20,955
- o Validation Samples: 5,239
- This ensures the model sees more examples of tumors during training.



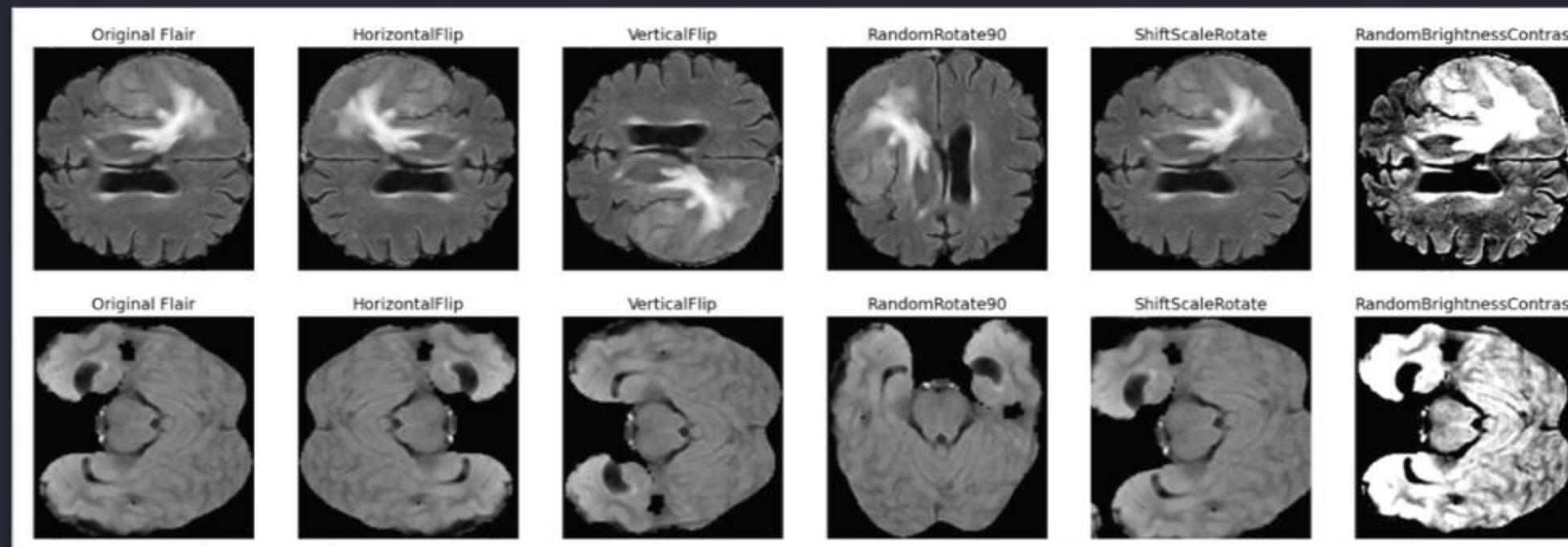
Strategies to Address Class Imbalance

3- Data Augmentation

Since tumors have diverse shapes and appearances, we applied realistic augmentations to make the model more generalizable.

Augmentation techniques used (50% probability each):

- Rotation (up to 90°)
- Horizontal Flips
- Vertical Flips
- Random Brightness & Contrast Adjustments
- ShiftScaleRotate



Strategies to Address Class Imbalance

4- Focal Tversky Loss:

Standard loss functions (like Cross-Entropy) struggle with imbalanced data.

We used Focal Tversky Loss (FTL):

Tversky Index Formula:

$$TI = \frac{TP}{TP + \alpha \cdot FP + \beta \cdot FN}$$

- α (False Positives weight) = 0.7, β (False Negatives weight) = 0.3
- Higher weight on False Negatives ensures the model does not ignore tumors.

Focal Tversky Loss Formula:

$$L = (1 - TI)^\gamma$$

- $\gamma = 4/3$ focuses on misclassified tumor regions, improving detection.

→ Effect: Balances the importance of small tumor classes while preventing overconfidence in background predictions.

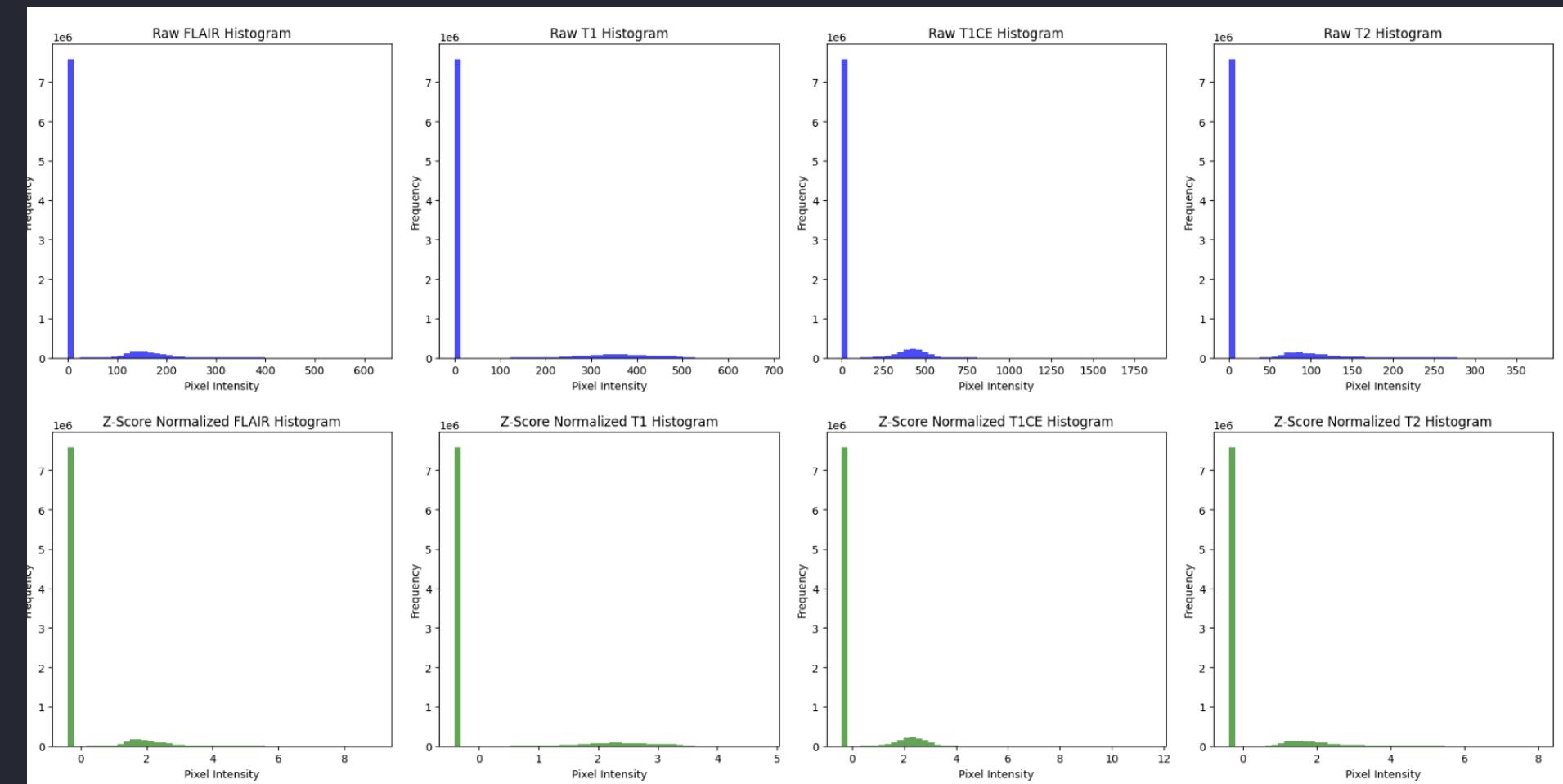
Normalize Image Intensities

Z-Score Normalization

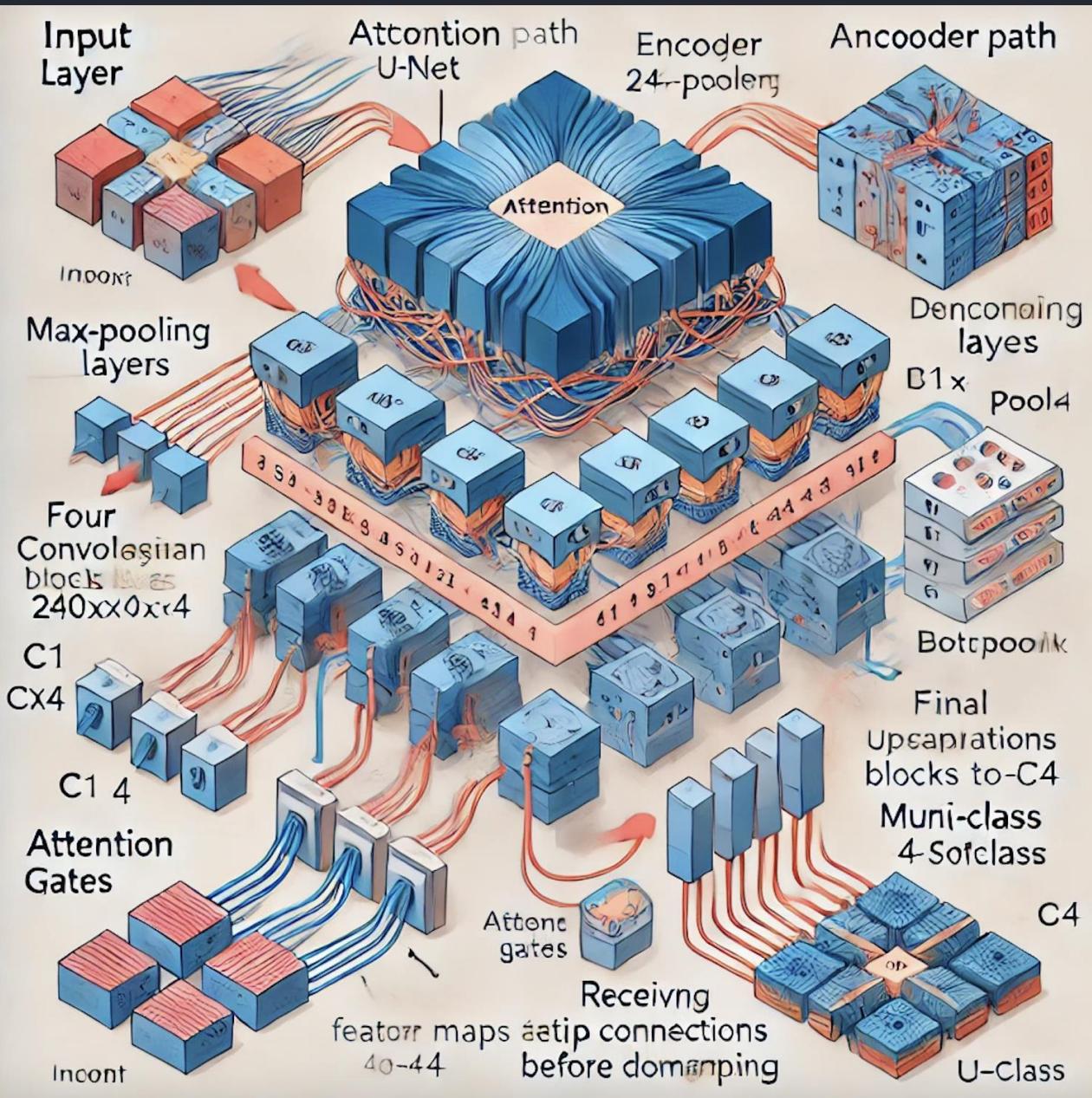
We perform Z-score normalization on each slice to standardize intensity values across different scans. This helps the model learn robust features independent of intensity variations.

$$X_{\text{norm}} = \frac{X - \mu}{\sigma}$$

μ = Mean intensity value of the non-zero pixels in the image.
 σ = Standard deviation of non-zero pixels.



Model Architecture: U-Net with Attention



U-Net

The U-Net architecture, known for its encoder-decoder structure with skip connections, is a powerful tool for medical image segmentation. It allows the model to learn both global and local features, leading to precise segmentation.



Attention Gates

Attention gates are added to the decoder part of the U-Net. They help the decoder focus on the most relevant regions of the input, enhancing the segmentation accuracy. Attention gates effectively guide the model to prioritize key tumor areas.

Model Architecture Breakdown



① Encoder (Contracting Path - Feature Extraction)

Series of convolutional blocks with:

- 3×3 Convolution layers (ReLU activation)
- Batch Normalization (stabilizes training)
- Max Pooling (2×2) (reduces spatial dimensions)

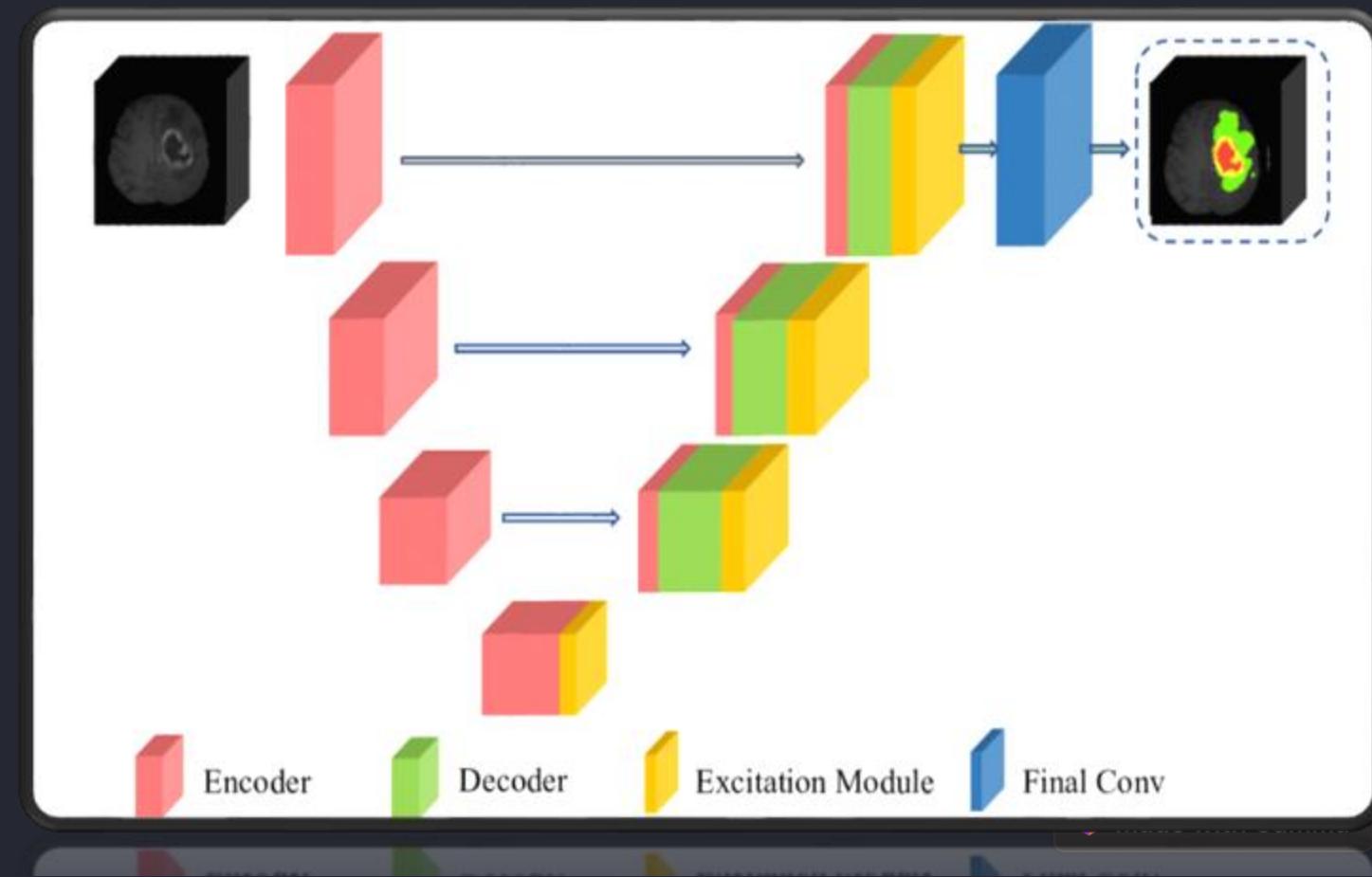
Extracts hierarchical features, reducing spatial resolution but increasing depth.

② Bottleneck Layer

- Lowest-resolution layer connecting the encoder and decoder.
- Contains deep convolutional layers to learn high-level abstract features.

③ Decoder (Expanding Path - Reconstruction)

- Upsampling layers to restore original image size.
- Skip connections from encoder help retain spatial information lost during downsampling.
- Final convolution (1×1) layer maps output to 4 segmentation classes.



Understanding (CNNs) in Tumor Segmentation



1 Convolutional Layers (Conv Layers):

The Convolutional Layer is the backbone of CNNs, responsible for feature extraction. It applies learnable filters (kernels) to detect patterns such as edges, textures, and tumor structures.

2 Max Pooling: Downsampling for Efficiency:

Max pooling is used to reduce spatial dimensions while keeping important features. It selects the maximum value in each region.

3 Batch Normalization: Stabilizing Training:

Batch Normalization (BN) normalizes inputs across the batch to speed up training and improve stability.

4 Activation Function: ReLU

ReLU (Rectified Linear Unit) is an activation function that outputs the input directly if it is positive, otherwise, it outputs zero.

$$\text{ReLU}$$
$$\text{ReLU}(x) = \max(0, x)$$

7	2	3	3	8
4	5	3	8	4
3	3	2	8	4
2	8	7	2	7
5	4	4	5	4

$$* \quad \begin{array}{|c|c|c|} \hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline \end{array} \quad = \quad \begin{array}{|c|c|c|} \hline 6 & & \\ \hline & & \\ \hline & & \\ \hline \end{array}$$

$7 \times 1 + 4 \times 1 + 3 \times 1 + 2 \times 0 + 5 \times 0 + 3 \times 0 + 3 \times -1 + 3 \times -1 + 2 \times -1 = 6$

Model Architecture Breakdown

! Attention Gates

Attention Gates are used in the decoder path to selectively pass important features while suppressing less useful activations.

- They filter out irrelevant spatial regions before upsampling.
- Improve localization by enhancing salient tumor features.
- Work by multiplying learned attention weights with feature maps.

 Mathematical Formulation:

$$AG(x, g) = \sigma(W_x x + W_g g + b)$$

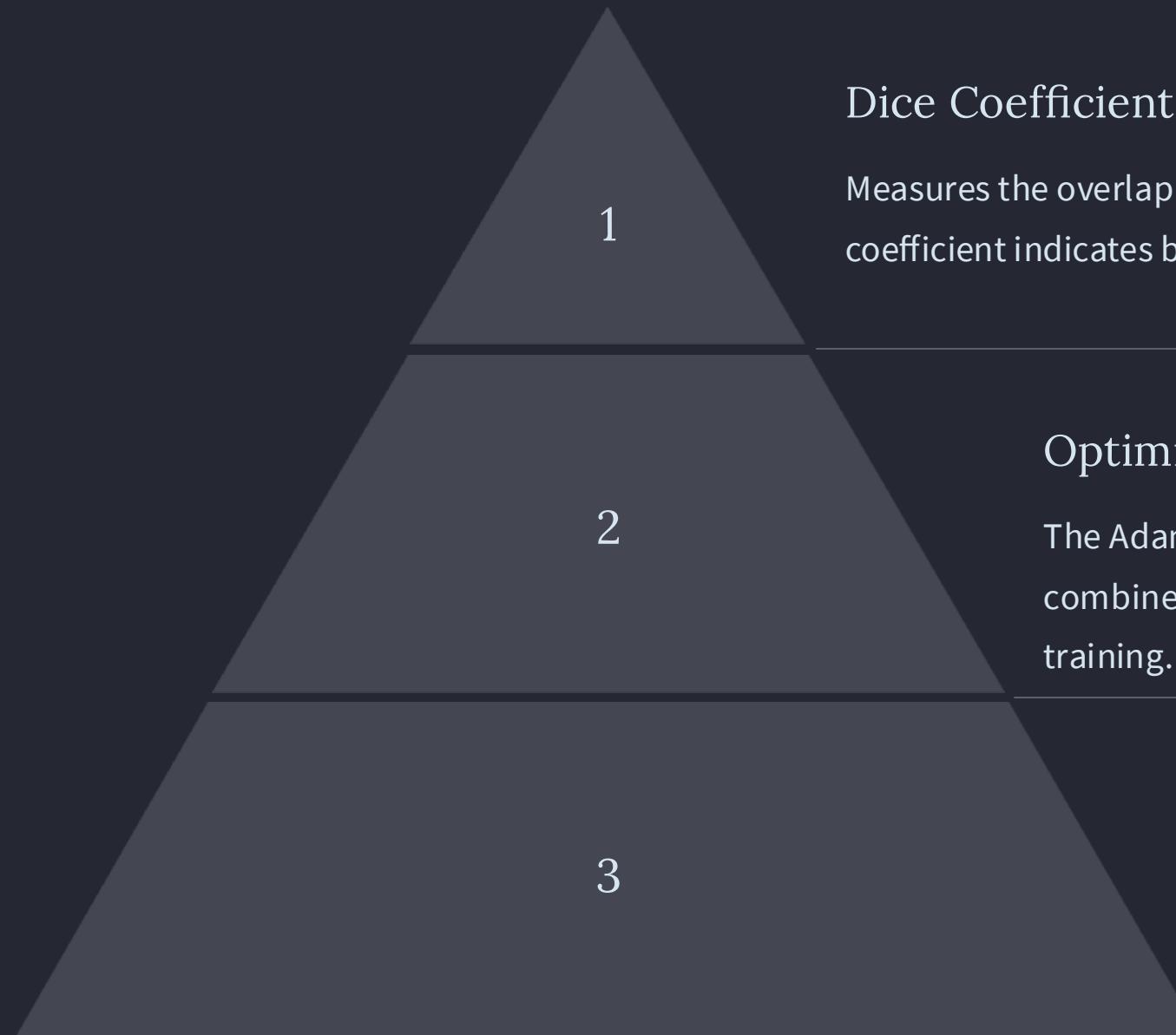
Where:

- x = input feature map.
- g = gating signal (from the deeper encoder layers).
- W_x, W_g = learnable weights.
- σ = sigmoid activation to assign attention weights.

Why Attention Helps in Our Task?

1. Reduces False Positives (background classified as tumor).
2. Better Small Tumor Detection (prioritizes critical regions).
3. Handles Class Imbalance (learns discriminative features).

Metrics for Brain Tumor Segmentation



Dice Coefficient

Measures the overlap between the predicted segmentation and the ground truth. A higher Dice coefficient indicates better segmentation accuracy.

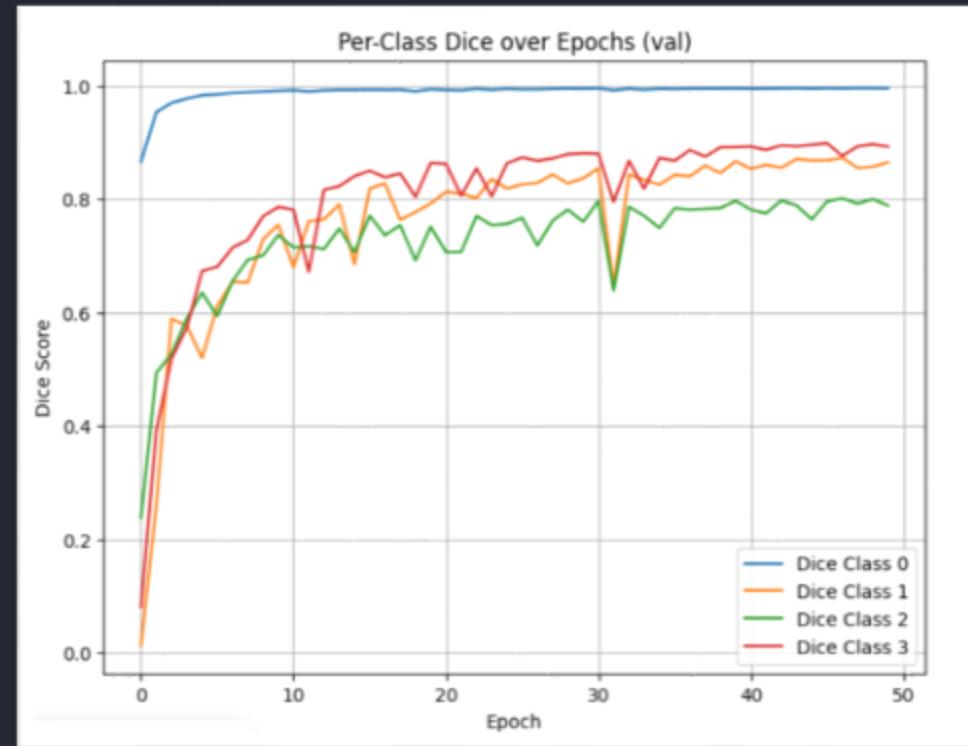
Optimizer: Adam

The Adam optimizer is an adaptive learning rate optimization algorithm that combines the benefits of AdaGrad and RMSProp to achieve efficient and robust training.

Learning Rate: (0.0001)

The learning rate is a hyperparameter that controls the step size at each iteration while moving toward a minimum of the loss function.

Training & Evaluation Results



0.84

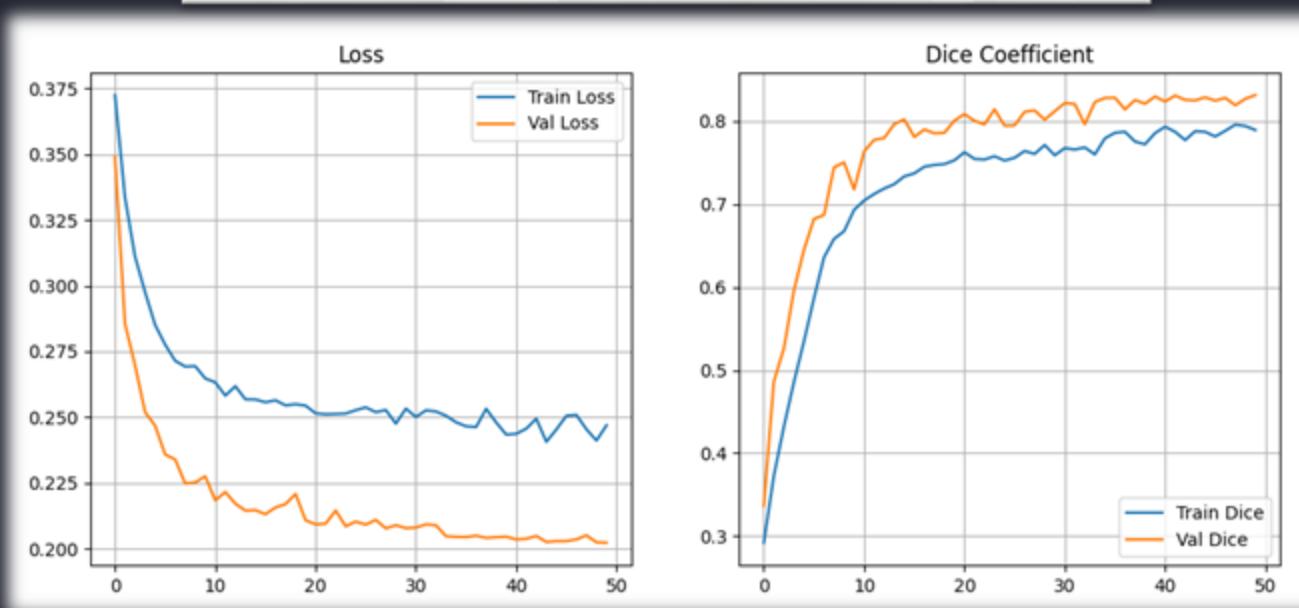
DSC

The model achieved a Dice similarity coefficient of approximately 0.84 on the validation set, indicating excellent segmentation performance.

0.98

Accuracy

The model also achieved high accuracy (~0.98) on the validation set, demonstrating its ability to correctly classify most pixels in the MRI scans.



Background	Necrotic Tumor	Edema Tumor	Enhancing Tumor
0.993	0.873	0.801	0.852



Training & Evaluation Results on another models

Model	DSC	ACC	BG	NT	ET	ENT
Unet+DL	0.41	0.98	0.992	0.21	0.31	0.11
Unet+WDL	0.63	0.97	0.989	0.32	0.47	0.31
Unet+FTL	0.82	0.98	0.993	0.85	0.79	0.81
Unet+Resnet50	0.61	0.99	0.97	0.12	0.39	0.26
Unet+ATT+DL+FL	0.81	0.97	0.981	0.81	0.83	0.82
Unet+ATT+FTL	0.84	0.98	0.993	0.87	0.80	0.85

- Each model Trained with 50 epochs

Post-Processing in Brain Tumor Segmentation

1 Removing Small False Positives (Noise Filtering)

Since the model may predict small scattered tumor regions that are not clinically relevant, we apply morphological operations to filter them out.

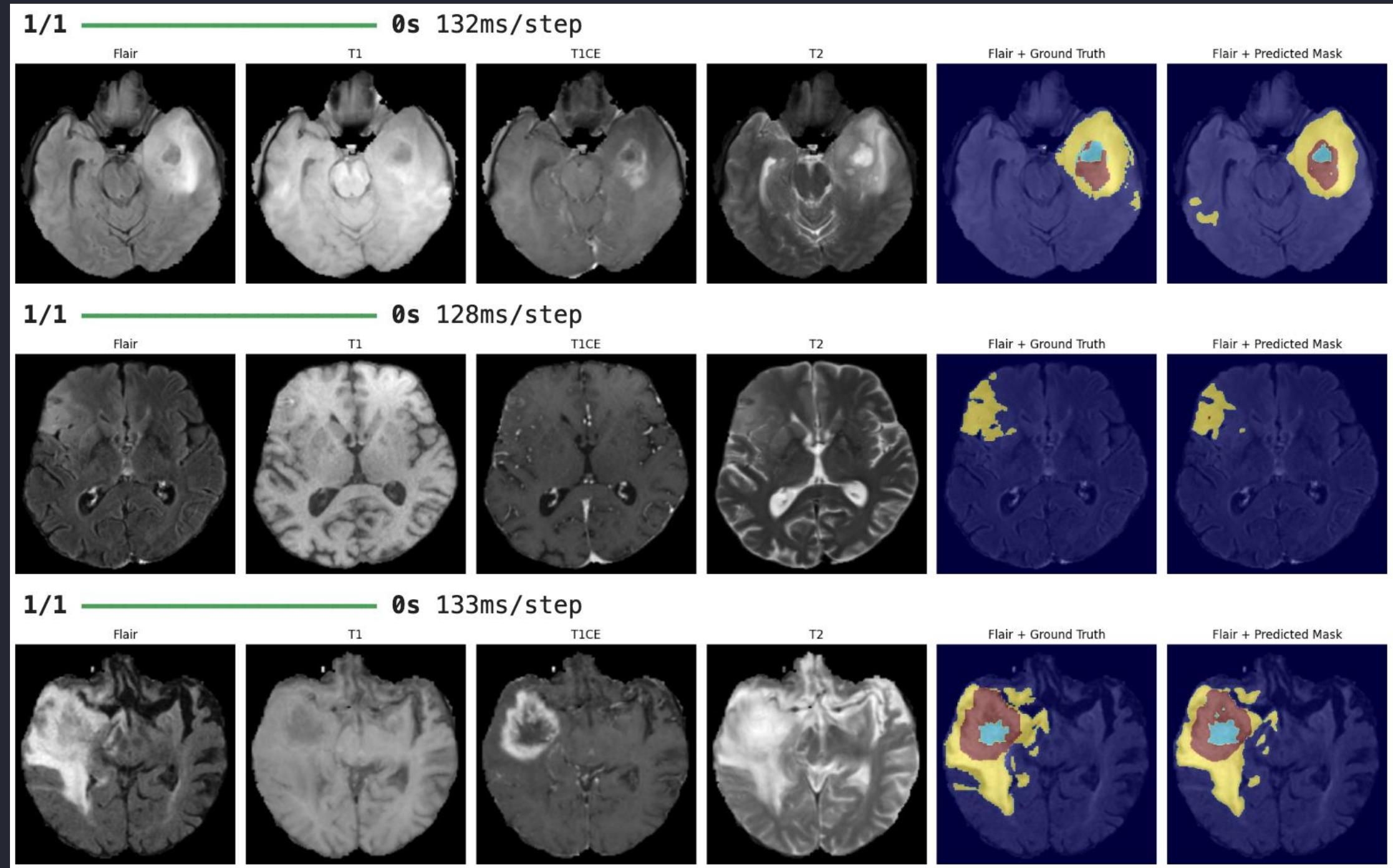
2 Hole Filling in Tumor Regions

In some cases, the model might predict tumor regions with artificial holes or gaps in them, making them look disconnected.

3 Applying Conditional Random Fields (CRF) for Boundary Refinement

Even though our model predicts tumor regions well, boundaries might be blurry or misaligned with actual tumor edges. We use CRFs to refine the segmentation mask.

Some Example of Predicted Images



Survival Prediction using Tumor Features

- **Dataset Details and Features Used:**

1. Patient ID mapping (Grade, Survival Days)
2. MRI scans and segmentation masks

- **Extracted tumor features from preprocessed segmentation masks:**

1. Necrotic/Core Volume
2. Edema Volume
3. Enhancing Tumor Volume
4. Total Tumor Volume
5. Age of the Patient

1. Target Variable:

Survival days were binned into two classes:

- Short Survival (<365 days) (Count = 119)
- Long Survival (≥ 365 days) (Count = 116)

Machine Learning Model - XGBoost

- We used XGBoost, a powerful gradient boosting algorithm, to classify Short vs. Long Survival based on extracted tumor features.
- Hyperparameter tuning was performed using GridSearchCV with:
 - ✓ n_estimators: 100
 - ✓ max_depth: 3, 4, 5
 - ✓ learning_rate: 0.001, 0.01, 0.05
 - ✓ subsample: 0.8, 1.0
 - ✓ colsample_bytree: 0.8, 1.0

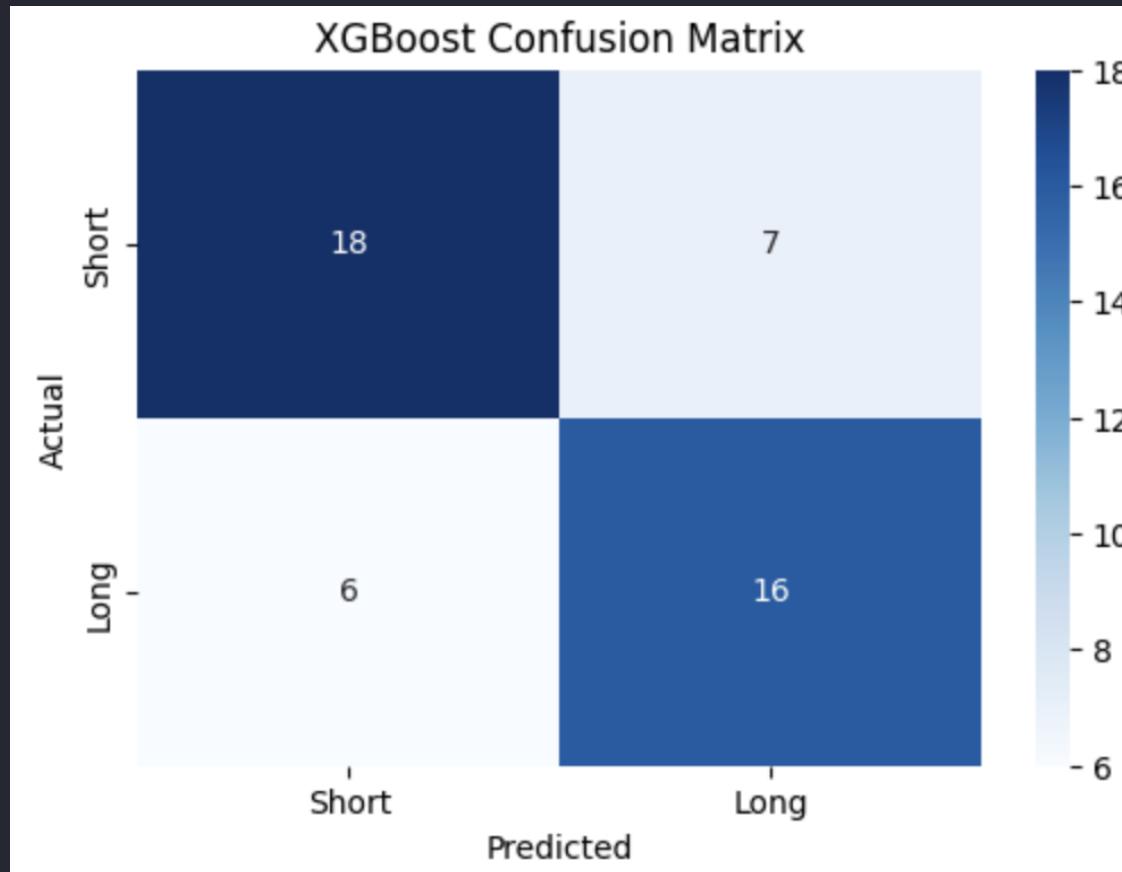
Final Classification Report

Class	Precision	Recall	F1-Score	Support
Short Survival	0.75	0.72	0.73	25
Long Survival	0.70	0.73	0.71	22

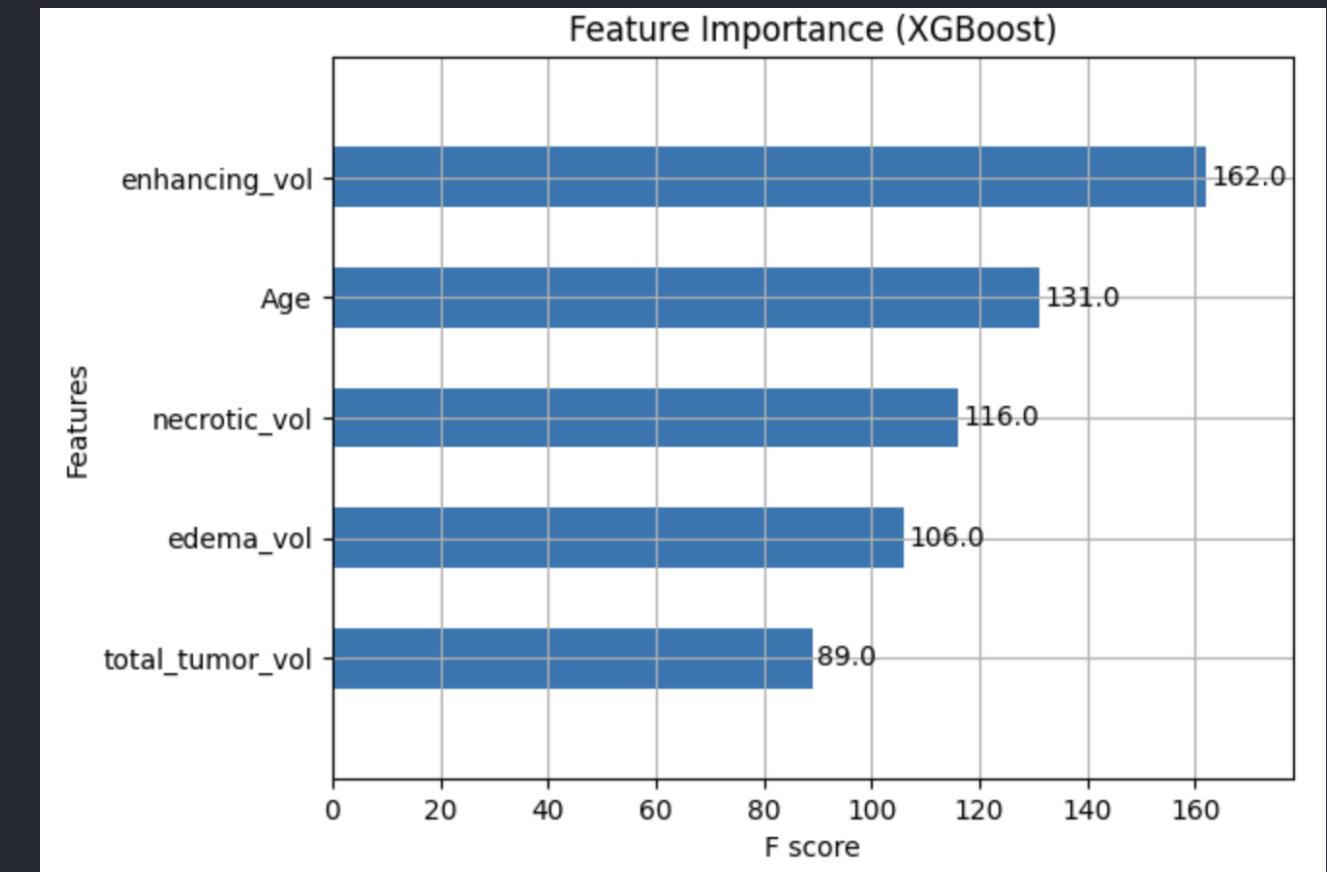
- Overall Accuracy: 72%
- Macro Avg F1-Score: 0.72

Model Performance & Results

Confusion Matix Analysis



Feature Analysis



Correct Predictions:

18 Short

16 Long

Misclassifications:

7 Short patients predicted as Long

6 Long patients predicted as Short