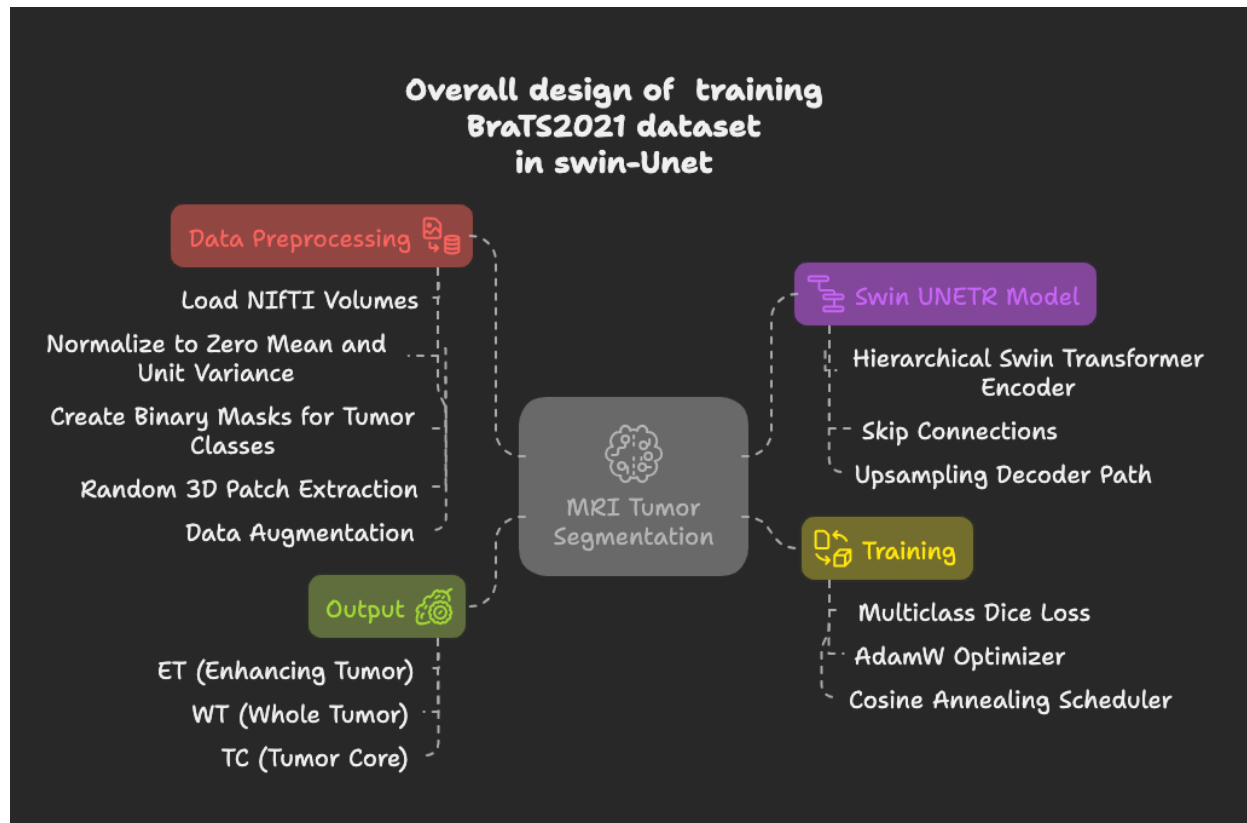# 1.Overall design of the end-to-end training pipeline
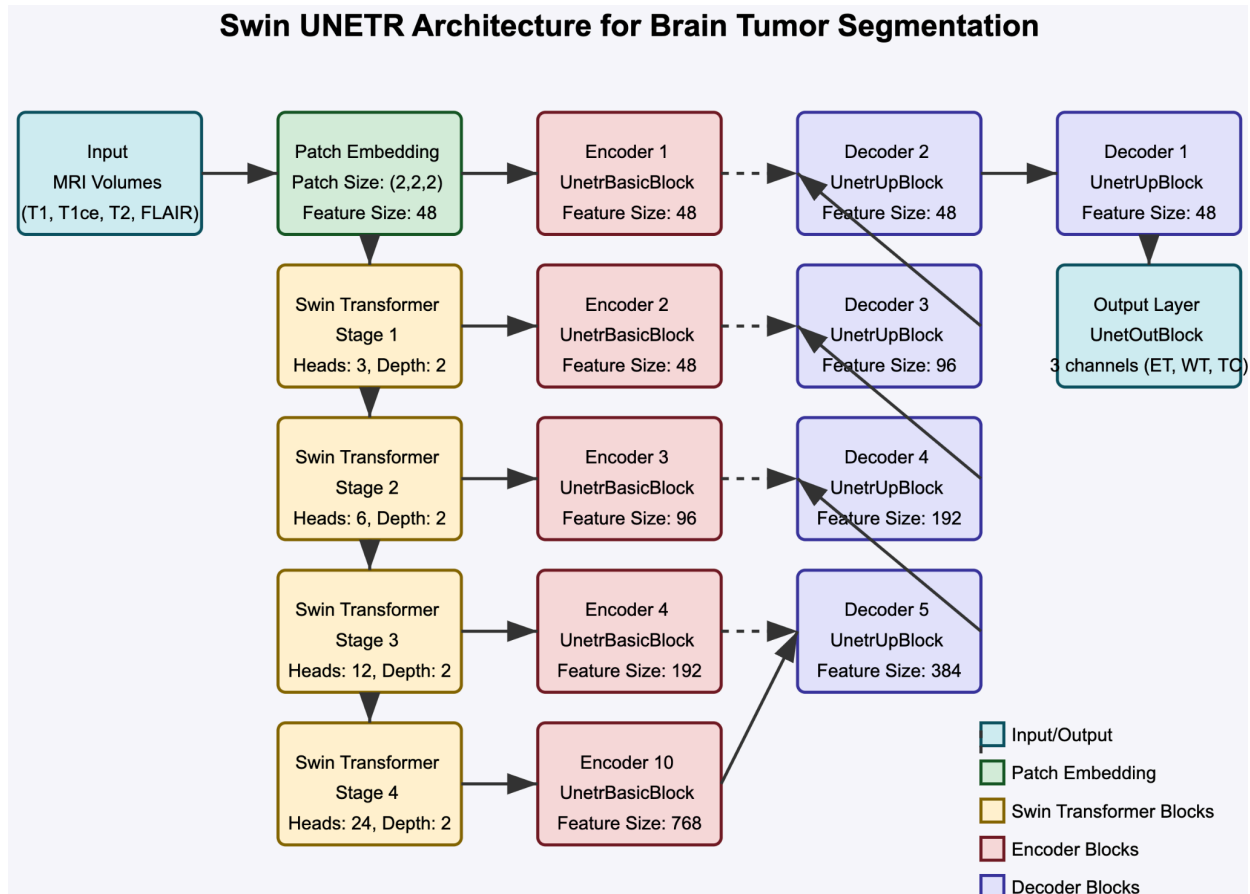

Data Preprocessing Details

Key preprocessing steps:

1. **Loading MRI volumes**: All four modalities are loaded along with segmentation masks
2. **Normalization**: Each modality is normalized independently to have zero mean and unit variance (calculated only for non-zero voxels)
3. **Tumor class extraction**: Three binary tumor region masks are created from the segmentation mask:
   - Enhancing Tumor (ET): Regions labeled as 4
   - Whole Tumor (WT): Regions labeled as 1, 2, or 4
   - Tumor Core (TC): Regions labeled as 1 or 4
4. **3D patch extraction**: Random 128×128×128 patches are extracted for training
5. **Data augmentation**: Applied during training to improve model generalization:
   - Random flipping along different dimensions
   - Random intensity shifts
   - Random intensity scaling

2. Model architecture

Swin UNETR Architecture for Brain Tumor Segmentation

Specific Implementation Details:

- Input patch embedding layer converts 3D patches to embedded tokens
- Four stages of Swin Transformer blocks with different feature dimensions
- Window-based attention computes self-attention within local windows
- Shifted window mechanism allows for information exchange between windows
- Patch merging layers reduce spatial resolution and increase channel dimension
- UnetrUpBlock modules in the decoder path for upsampling and feature fusion
- Final output layer produces segmentation masks for three tumor regions

Conclusion:
The architecture follows a U-shaped structure typical of U-Net designs but replaces standard convolutional layers with Swin Transformer blocks for improved feature extraction that is why it

is Swin-Unet. The hierarchical design with varying head dimensions (3→6→12→24) enables capturing both local and global contextual information at different scales.

3.Hyperparameters

Parameter	Value
Batch Size	2
Patch Size	(128, 128, 128)
Feature Size	48
Transformer Depths	(2, 2, 2, 2)
Number of Heads	(3, 6, 12, 24)
Learning Rate	0.001
Optimizer	AdamW
Scheduler	CosineAnnealing
Number of Epochs	10
Train/Val Split	0.8/0.2
Model Parameters	84,842,997

4. Evaluation

Loss Function

The model was trained using a differentiable Dice loss function that preserves gradients for backpropagation. The loss is calculated as:

1. Apply sigmoid activation to predicted outputs
2. Calculate soft Dice coefficient between predictions and ground truth
3. Convert to loss by taking (1 - Dice)
4. Handle empty ground truth cases appropriately

Evaluation Metrics

The primary evaluation metric is the Dice coefficient, which measures spatial overlap between predicted and ground truth segmentations. The Dice coefficient was calculated separately for each tumor subregion:

1. **ET (Enhancing Tumor)**: Measures segmentation accuracy for active tumor regions
2. **WT (Whole Tumor)**: Measures segmentation accuracy for entire tumor, including edema
3. **TC (Tumor Core)**: Measures segmentation accuracy for tumor core (necrosis and enhancing regions)

Additionally, an overall average Dice score was calculated across all three regions to assess general performance.
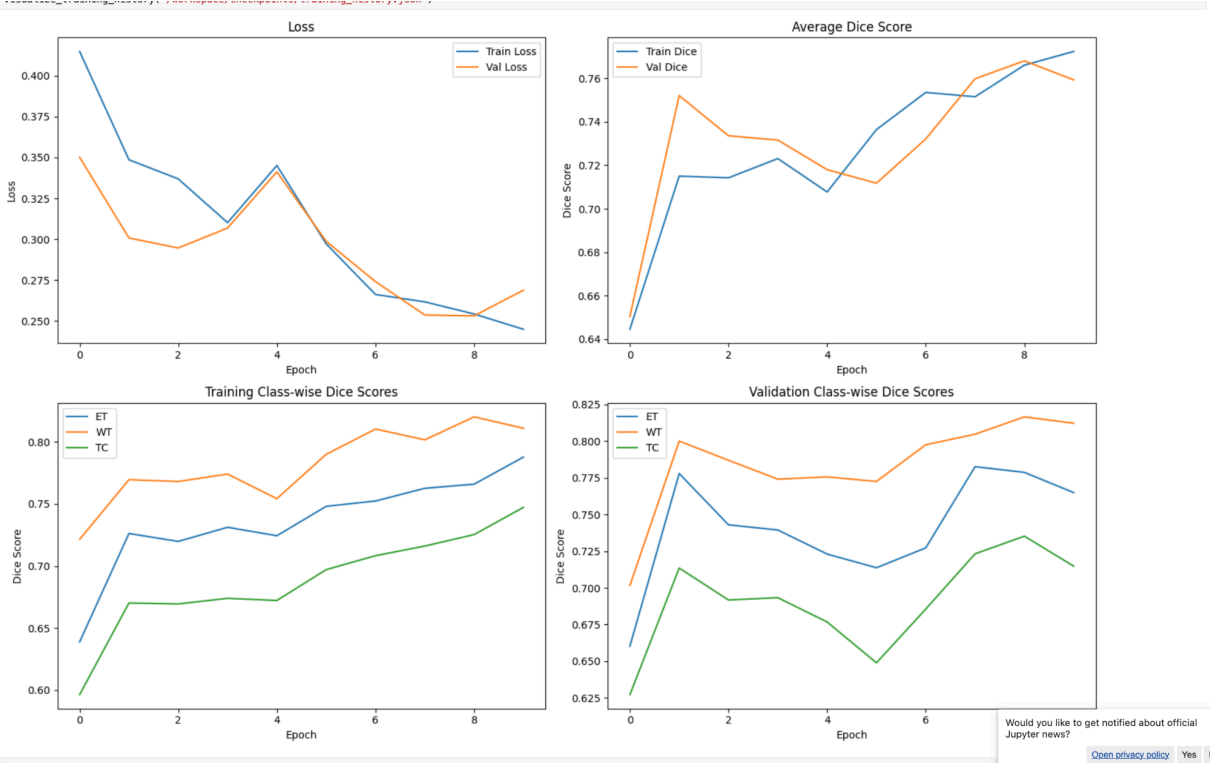
Training Process

The model was trained using a rigorous process that included:

- Training/validation split of 80%/20%
- Regular evaluation on validation data
- Saving checkpoints based on best validation performance
- Tracking of class-wise Dice scores
- Learning rate scheduling using cosine annealing

The training employed multiple checkpoint strategies:

- Saving the best model based on validation Dice score
- Saving the best model based on validation loss
- Regular checkpoint saving every 5 epochs

Results:

visualize_training_history(/workspace/checkpoints/training_history.json)

Epoch 9/10, Train Loss: 0.2543, Train Dice: 0.7660, Train ET: 0.7659, Train WT: 0.8201, Train TC: 0.7253, Val Loss: 0.2531, Val Dice: 0.7680, Val ET: 0.7787, Val WT: 0.8166, Val TC: 0.7353, Train Time: 3922.85s, Val Time: 1106.12s
Saved best model with validation Dice score: 0.7680
Saved best model with validation loss: 0.2531