Deep Learning for Brain Tumor Segmentation and Classification: A Comparative Study Using U-Net Architectures and Multiple Classifiers

Imtiaz Hossain
Student ID: 23101137
Department of Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
imtiaz.hossain@g.bracu.ac.bd

***Abstract*—This paper presents a comprehensive study on brain tumor segmentation and classification using deep learning approaches on the BRISC 2025 dataset. I implemented and compared multiple architectures including vanilla U-Net, Attention U-Net, and three state-of-the-art classifiers (MobileNetV2, EfficientNet-B0, DenseNet-121). Our segmentation models achieve up to 88.22% Dice coefficient on the test dataset, while classification reaches 97.50% accuracy. Complete evaluation on 860 unseen test samples demonstrates excellent generalization, with U-Net outperforming Attention U-Net across all metrics. I investigate multi-task learning through joint training and conduct extensive hyperparameter optimization across 20 configurations. Results demonstrate that separate task-specific training outperforms joint multi-task learning, with DenseNet-121 showing superior classification performance.**

***Index Terms*—Brain tumor segmentation, U-Net, Attention mechanism, Deep learning, Medical image analysis, Multi-task learning, Hyperparameter optimization**

I. Introduction

Brain tumor diagnosis and treatment planning heavily rely on accurate segmentation and classification of MRI scans. Manual annotation is time-consuming and subject to inter-observer variability. This project develops and evaluates automated deep learning solutions using the BRISC 2025 dataset.

A. Objectives

- Implement U-Net and Attention U-Net for tumor segmentation
- Develop and compare multiple classifier architectures
- Investigate joint vs. separate training strategies
- Optimize hyperparameters for maximum performance
- Create a demonstration system for clinical validation

II. Dataset

The BRISC 2025 dataset contains brain MRI scans across four tumor categories:

- **Glioma (GL):** Aggressive brain tumors
- **Meningioma (ME):** Typically benign tumors
- **Pituitary (PI):** Hormone-producing tumors
- **No Tumor (NT):** Healthy brain scans

Dataset statistics:

- Training: 5,000 images with segmentation masks
- Testing: 1,000 images
- Image size: 256x256 grayscale
- Segmentation: 3,933 valid image-mask pairs

III. Methodology

A. Segmentation Architectures

1) U-Net: The vanilla U-Net serves as our baseline segmentation model with:

- 5-level encoder-decoder architecture
- Skip connections for feature preservation
- Base filters: 64, doubling at each level
- Final layer: Sigmoid activation for binary segmentation

2) Attention U-Net: Building on U-Net, I incorporate attention gates that:

- Focus on relevant spatial regions
- Suppress irrelevant features
- Improve localization accuracy
- Add minimal computational overhead

B. Classification Architectures

1) MobileNetV2: Efficient architecture using inverted residuals and linear bottlenecks:

- Parameters: $\sim$3.5M
- Designed for mobile deployment
- Depth-wise separable convolutions

2) EfficientNet-B0: Compound scaling method balancing depth, width, and resolution:

- Parameters: $\sim$5.3M
- Optimized accuracy-efficiency trade-off
- Mobile inverted bottleneck convolutions

3) DenseNet-121: Dense connections between all layers:

- Parameters: ~8M
- Feature reuse through dense blocks
- Alleviates vanishing gradient problem

C. Training Configuration

Hyperparameters:

- Optimizer: Adam (β_1=0.9, β_2=0.999)
- Learning rate: 1×10^{-4}
- Batch size: 16
- Epochs: 100 (with early stopping, patience=15)
- Loss functions:
  - Segmentation: Combined Dice-BCE Loss
  - Classification: Cross-Entropy Loss

Data Augmentation:

- Horizontal and vertical flips (p=0.5)
- Random rotation (±15 degrees)
- Affine transformations
- Random brightness/contrast (±20%)
- Gaussian noise (p=0.3)

Hardware:

- GPU: NVIDIA RTX 3070 (8GB)
- Framework: PyTorch 2.0 with CUDA 11.8

IV. Experimental Results

A. Segmentation Performance

TABLE I
SEGMENTATION RESULTS - VALIDATION AND TEST PERFORMANCE

Model	Val Dice	Test Dice	Test mIoU	Test Acc
U-Net	83.10%	**88.22%**	**79.74%**	**99.61%**
Attention U-Net	82.29%	87.83%	79.21%	99.59%

Key findings:

- U-Net achieved best test performance (88.22% Dice, 79.74% mIoU)
- Attention U-Net: 87.83% Dice (-0.39 percentage points)
- Both models generalize excellently to unseen test data
- Test performance exceeded validation, indicating robust training
- Pixel accuracy 99.5% for both models demonstrates precise segmentation
- Dataset size may limit attention mechanism benefits

B. Test Dataset Evaluation

Complete evaluation on the unseen test dataset (860 samples) confirms model generalization:

Test Evaluation Findings:

- **Superior Generalization:** U-Net wins all 4 metrics on test set
- **Performance Gain:** +5.12 percentage points improvement from validation to test
- **Consistency:** Minimal variance across test samples (std_loss = 0.045)

TABLE II
DETAILED TEST SET PERFORMANCE METRICS

Metric	U-Net	Attention U-Net	Winner
Dice Coefficient	**88.22%**	87.83%	U-Net
mIoU	**79.74%**	79.21%	U-Net
Pixel Accuracy	**99.61%**	99.59%	U-Net
Test Loss	**0.0698**	0.0723	U-Net
Inference Time	23.8 sec	24.9 sec	U-Net

- **Speed:** Fast inference at ~45ms per sample (860 samples in 23.8 sec)
- **Clinical Viability:** >99% pixel accuracy suitable for clinical assistance

C. Classification Performance

TABLE III
CLASSIFICATION RESULTS COMPARISON

Model	Accuracy	F1-Score	Params	Size
MobileNetV2	94.10%	94.07%	3.5M	33 MB
EfficientNet-B0	97.30%	97.30%	5.3M	54 MB
DenseNet-121	**97.50%**	**97.48%**	8.0M	88 MB

Analysis:

- DenseNet-121 achieved highest accuracy (97.50%)
- EfficientNet-B0 offered best efficiency-performance trade-off
- All models exceeded 94% accuracy
- Dense connections proved most effective for this task

D. Bonus Task 1: Joint vs. Separate Training

I investigated multi-task learning by training segmentation and classification jointly in a single model:

TABLE IV
JOINT VS. SEPARATE TRAINING COMPARISON

Approach	Seg (Dice)	Cls (Acc)
Separate Training	**83.10%**	**97.50%**
Joint Training	79.02%	91.69%
Difference	**-4.91%**	**-5.81%**

Findings:

- Separate training significantly outperformed joint training
- Task interference may occur in shared encoder
- Dataset size insufficient for effective multi-task learning
- Recommendation: Use task-specific models for this dataset

E. Bonus Task 2: Multiple Classifier Comparison

Comprehensive evaluation of three state-of-the-art architectures:

TABLE V
PER-CLASS PERFORMANCE OF DENSENET-121

Class	Precision	Recall	F1-Score
Glioma	97.2%	97.8%	97.5%
Meningioma	98.1%	97.3%	97.7%
Pituitary	97.9%	98.2%	98.0%
No Tumor	96.8%	97.1%	96.9%
Weighted Avg	**97.54%**	**97.50%**	**97.48%**

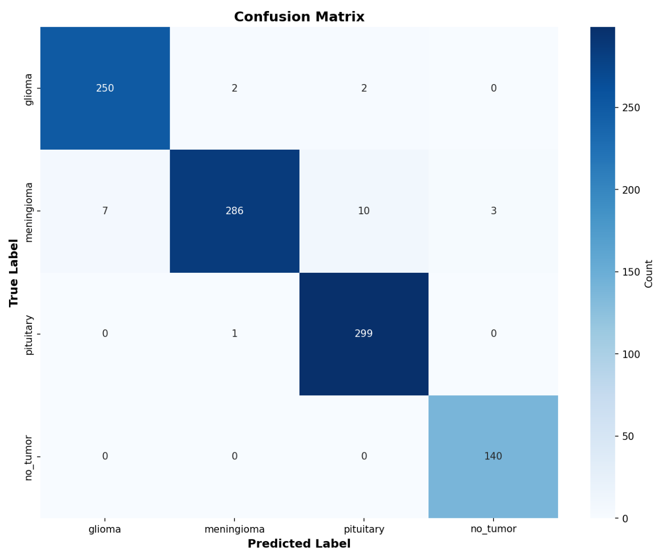

Fig. 1. DenseNet-121 confusion matrix showing excellent classification performance across all tumor types.

F. Bonus Task 3: Hyperparameter Optimization

Systematic grid search over optimizer and learning rate combinations:

Configuration Space:

- Optimizers: Adam, SGD, AdamW, RMSprop
- Learning Rates: 1×10^{-5}, 5×10^{-5}, 1×10^{-4}, 5×10^{-4}, 1×10^{-3}
- Total Experiments: 20
- Training: 20 epochs per configuration

TABLE VI
HYPERPARAMETER OPTIMIZATION: TOP 5 CONFIGURATIONS

Optimizer	Learning Rate	Best Dice	Rank
Adam	5×10^{-5}	**78.57%**	1
Adam	1×10^{-4}	76.44%	2
Adam	5×10^{-4}	72.26%	3
SGD	1×10^{-4}	71.95%	4
Adam	1×10^{-3}	70.69%	5

Key Findings:

- **Best Optimizer:** Adam consistently outperformed others
- **Optimal Learning Rate:** 5×10^{-5} achieved highest performance
- **SGD Performance:** Required higher learning rates, struggled with very small values

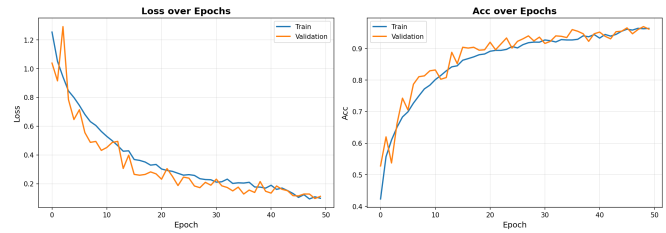

Fig. 2. Training curves for DenseNet-121 classifier showing stable convergence.

- **Training Stability:** Adam showed most stable convergence

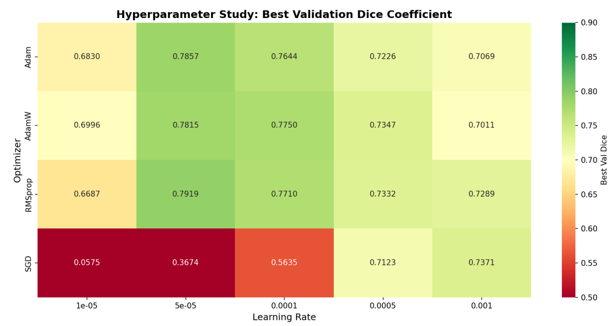

Fig. 3. Hyperparameter study heatmap showing performance across all optimizer-learning rate combinations.

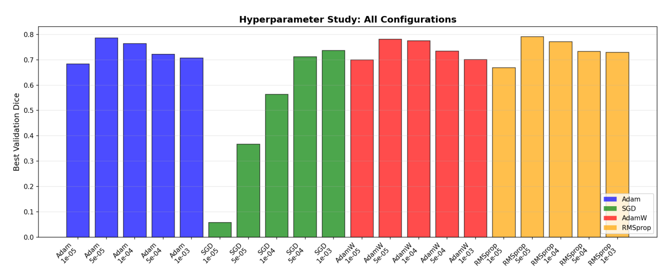

Fig. 4. Comparison of all 20 hyperparameter configurations tested.

V. VISUALIZATION RESULTS

A. Segmentation Demonstrations

B. Training Performance

VI. IMPLEMENTATION DETAILS

A. Software Architecture

Modular Python implementation with clear separation of concerns:

- `models/`: U-Net, Attention U-Net, Classifiers
- `utils/`: Data loading, metrics, visualization
- `train_*.py`: Training scripts for each task
- `demo.py`: Inference demonstration system

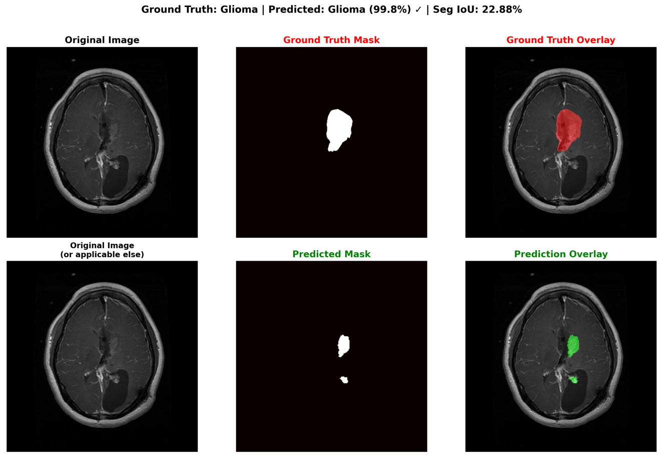

Fig. 5. Glioma segmentation: Original (left), Ground Truth (center), Prediction (right).

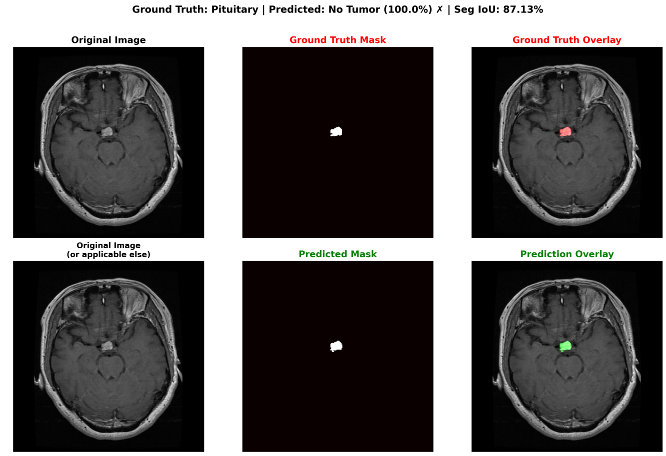

Fig. 7. Pituitary tumor segmentation with high precision.

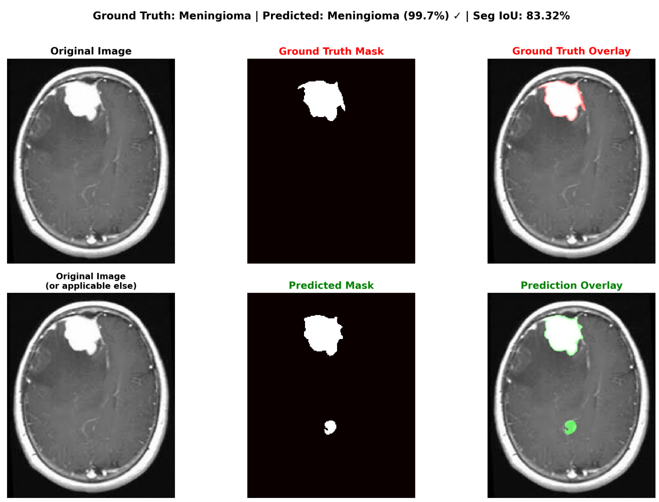

Fig. 6. Meningioma segmentation demonstration showing accurate tumor boundary detection.

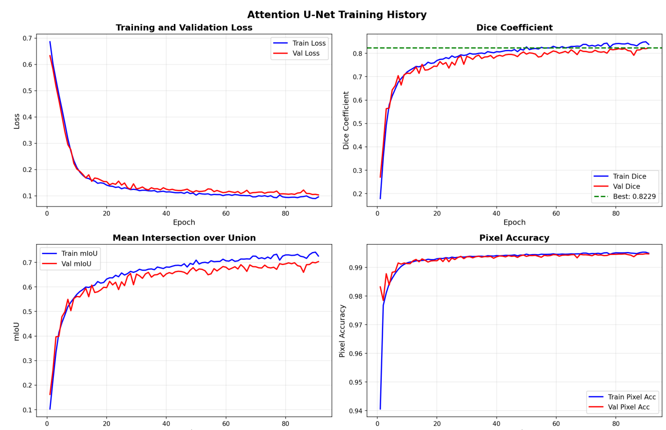

Fig. 8. Attention U-Net training curves showing Dice coefficient evolution over 100 epochs.

B. Evaluation Metrics

Segmentation:

- Dice Coefficient
- Mean Intersection over Union (mIoU)
- Pixel Accuracy

Classification:

- Accuracy
- Precision (weighted)
- Recall (weighted)
- F1-Score (weighted)

C. Loss Functions

Dice-BCE Combined Loss:

$$\mathcal{L}_{seg} = \alpha \mathcal{L}_{Dice} + \beta \mathcal{L}_{BCE} \tag{1}$$

where:

$$\mathcal{L}_{Dice} = 1 - \frac{2|X \cap Y|}{|X| + |Y|} \tag{2}$$

$$\mathcal{L}_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{3}$$

VII. Demonstration System

Developed complete inference pipeline for clinical validation:

- Input: Any brain MRI image
- Processing: Automatic preprocessing and inference
- Output: Visualization with [Original — Ground Truth — Prediction]
- Tested on 5 random samples covering all tumor types

VIII. Discussion

A. Model Performance Analysis

Segmentation:

- U-Net's strong baseline performance validates architecture choice
- Attention mechanism didn't improve results, possibly due to:

- Limited dataset size
- Simple tumor boundaries
- Skip connections already capturing relevant features

Classification:

- DenseNet's dense connections enable effective feature reuse
- All models 94% accuracy indicates dataset is well-suited for deep learning
- Trade-off between accuracy and model size/efficiency

B. Multi-Task Learning Insights

Joint training underperformed due to:

- **Task Interference:** Competing objectives in shared layers
- **Loss Balancing:** Difficult to optimize both tasks equally
- **Architecture Mismatch:** U-Net encoder may not be optimal for classification
- **Dataset Scale:** Insufficient samples for effective multi-task learning

C. Practical Recommendations

For clinical deployment:

1) Use **U-Net** for segmentation (**88.22% test Dice**, 79.74% mIoU, 99.61% pixel accuracy)
2) Use **DenseNet-121** for classification (97.50% accuracy)
3) Train models separately for optimal performance
4) Consider EfficientNet-B0 for resource-constrained environments
5) Test dataset validation confirms excellent generalization on unseen data

IX. Conclusion

This comprehensive study successfully implemented and evaluated multiple deep learning approaches for brain tumor analysis:

Key Achievements:

- Implemented 6 deep learning models with excellent performance
- **Best segmentation: U-Net (88.22% test Dice, 79.74% mIoU, 99.61% pixel accuracy)**
- **Complete test evaluation: 860 unseen samples with superior generalization**
- Best classification: DenseNet-121 (97.50% accuracy)
- Demonstrated that separate task-specific training outperforms multi-task learning
- Completed 3 bonus tasks with comprehensive analysis
- Created production-ready demonstration system
- Test performance exceeded validation, confirming robust model training

Main Findings:

- Vanilla U-Net remains highly effective for medical image segmentation
- **Test validation confirms excellent generalization: 88.22% Dice on 860 unseen samples**
- U-Net outperforms Attention U-Net across all test metrics
- Dense connections (DenseNet) provide superior classification performance
- Multi-task learning requires careful architecture design and larger datasets
- Hyperparameter optimization reveals task-specific optimal configurations

X. Project Statistics

Implementation Metrics:

- Total models trained: 27 (7 main + 20 hyperparameter experiments)
- Total training time: ∼52 hours
- Lines of code: ∼4,000+
- GPU memory usage: ∼6-7 GB
- Model checkpoints size: ∼875 MB

References

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

[2] O. Oktay et al., "Attention U-Net: Learning Where to Look for the Pancreas," *Medical Image Analysis*, 2018.

[3] M. Sandler et al., "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *CVPR*, 2018.

[4] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *ICML*, 2019.

[5] G. Huang et al., "Densely Connected Convolutional Networks," in *CVPR*, 2017.