AI-Powered Brain MRI Tumor Segmentation

A DEEP LEARNING APPROACH FOR FAST AND ACCURATE DIAGNOSTICS

AHMAD NAYFEH / JUNE 2025

Outline

- Background
- Dataset
- Dataset Preprocessing
- AI Models
- Evaluation & Results
- Demo & Deployment
- Conclusion

Background

The Clinical Challenge: Why This Matters

- Primary brain tumors represent a significant clinical challenge; Glioblastoma (GBM) is the most aggressive form, with a 5-year survival rate of only 6.9%.
- Accurate tumor segmentation from MRI scans is fundamental for diagnosis, treatment planning, and monitoring disease progression.
- The manual process done by radiologists is extremely time-consuming, taking anywhere from **30 minutes to several hours** for a single tumor.

Outlining brain tumors by hand is slow and inconsistent.

The Problem: Human Inconsistency

- Manual segmentations suffer from significant **inter-observer variability**, meaning two different experts will produce different results for the same scan.
- Consistency scores (known as Dice scores) between two human experts can range from just **0.73 to 0.85.**
- This inconsistency can impact the precision of treatment and how a tumor's response to therapy is measured.

Even experts don't always agree, which can affect patient care.

The Solution: An AI-Powered Approach

- **Speed:** AI models can perform a complex segmentation task in **under 5 minutes**, freeing up valuable clinician time.
- **Consistency:** AI models achieve a high degree of repeatability, with Dice scores often **exceeding 0.90**—better than human-to-human consistency.
- **Performance:** State-of-the-art models like the **U-Net** can match or even exceed human expert performance in this task.

We can leverage deep learning to build a tool that is faster, more consistent, and performs at an expert level.

Dataset

The Data: Fueling Our AI Model

We are using a real-world, public dataset of multi-modal brain MRI scans.

- Dataset Name: BraTS-Africa
- Contents:
  - 146 Patients with brain tumors (primarily Glioma).
  - For each patient, we have 4 types of MRI scans and 1 expert-annotated segmentation mask.

The Inputs: Four Views of the Brain

- We give our AI four different “views” of the tumor to give it the most complete information possible.
  - We use four MRI sequences: **T1, T2, T1-contrast, and FLAIR**.
  - Think of them as different photo filters.
  - The **FLAIR** scan (far right) is often the most informative because it makes the tumor swelling "pop" with high contrast.

Our Target: The "Answer Key" Map

- This is a **segmentation task**
- For each patient, we have a ground-truth map that shows the exact location and different parts of the tumor.
- The different colors represent different tumor sub-regions identified by an expert radiologist:
  - **Edema** (swelling)
  - **Necrotic Core** (dead tissue)
  - **Enhancing Tumor** (active/aggressive part)

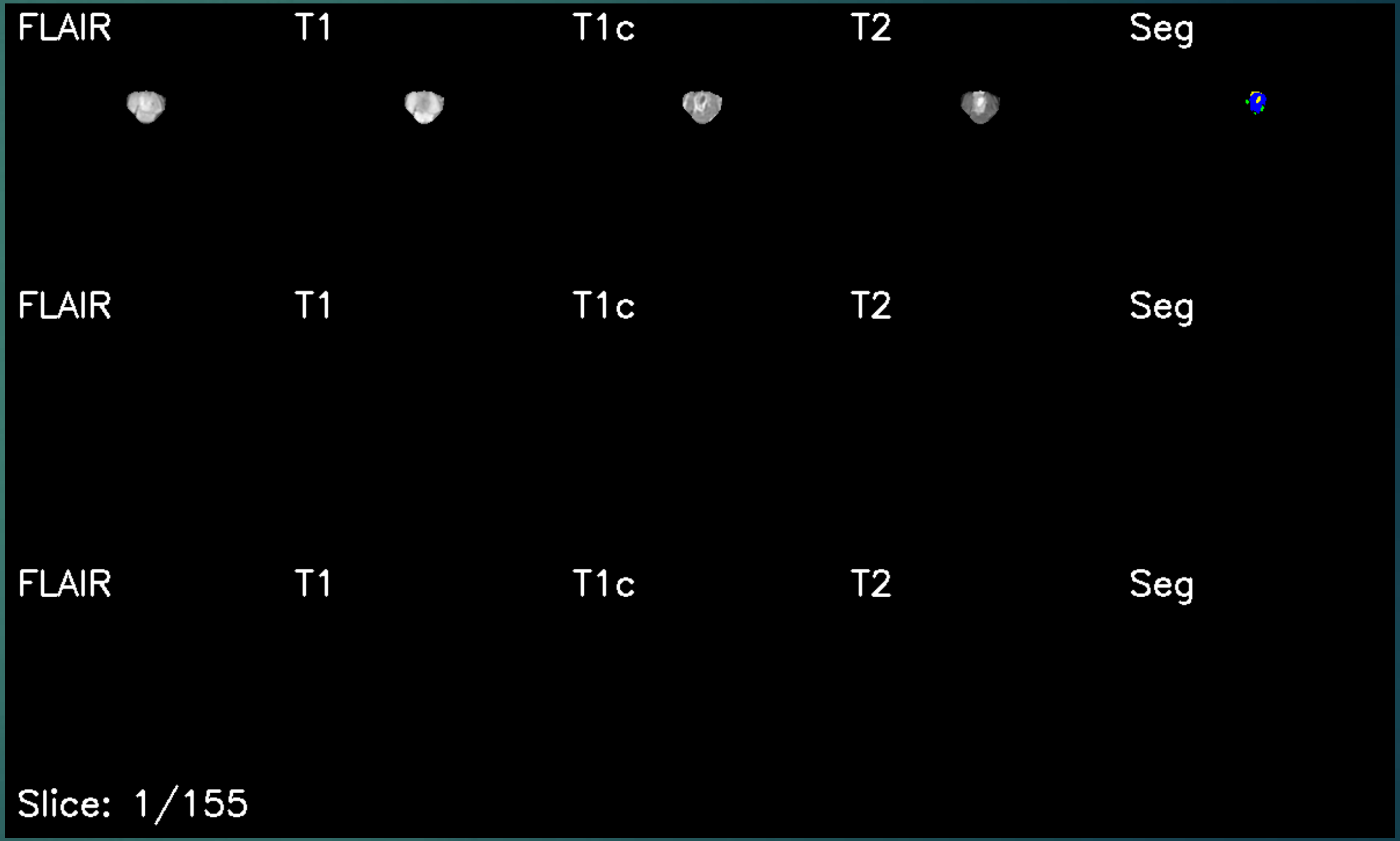
Goal: learn to automatically create this colored map just by looking at the four input MRI scans

Dataset at a Glance: Key Statistics

- **Total Patients:** 146
- **Data per Patient:** 5 Files
  - 4 MRI Scans (T1, T1c, T2, FLAIR) + 1 Segmentation Mask
  - **File Dimensions:** (240, 240, 155)
    - Each 3D volume contains **155 slices**.
    - The dimension of each individual 2D slice is **(240, 240)**
- **Total Files:** 730
- **Data Consistency:**
  - File Count: Every patient has exactly 5 files.
  - Dimensions: All files have the same 3D dimensions

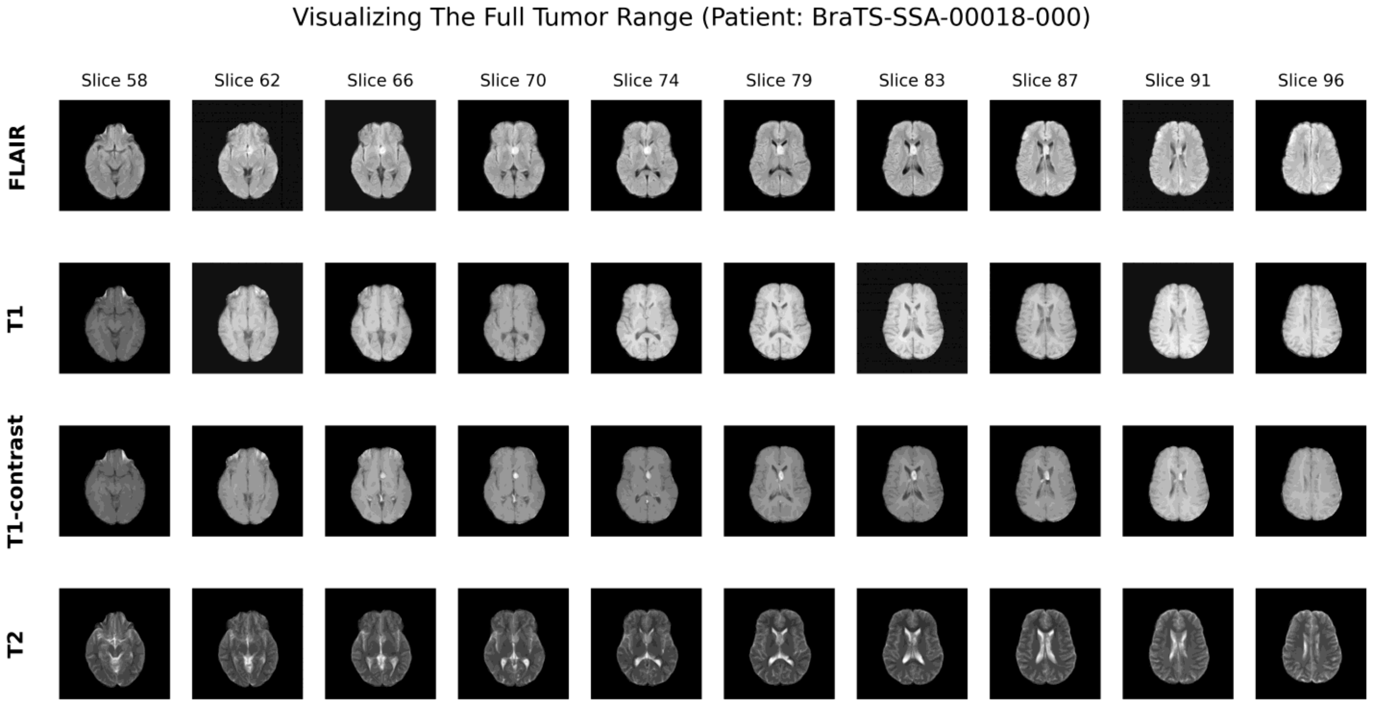
Exploring the 3D Data

This animation demonstrates the 3D structure of our data by scrolling through all 155 slices for multiple random patients across different MRI modalities (FLAIR, T1, etc.) simultaneously.



Visualizing the 2D Training Data

After understanding the 3D volumes, we extract and visualize the individual 2D slices. This is the actual data our AI model will learn from, and this figure provides a survey of the dataset's diversity.



Dataset Preprocessing

Step 1: Stacking MRI Modalities

- The four distinct MRI modalities (T1, T1c, T2, FLAIR) are stacked into a single, 4-channel 3D volume, providing a comprehensive view for every point in the scan.
    - combined image volume with a shape of (240, 240, 155, 4)
    - The segmentation mask will remain a separate, single-channel volume.
- Why?
    - By stacking the modalities, every single 2D slice we extract will contain information from all four MRI types
    - This provides the AI with a much richer set of information for every pixel, allowing it to learn the subtle differences and patterns across modalities that are key to identifying the tumor.

Step 2: Crop Empty "Air" Slices

- As we saw in slide 12, a significant portion of the 155 slices is just black space (air) at the top and bottom of the head.
- We will find the **first** and **last** slices that contain any brain tissue or tumor in the segmentation mask.
  - Discard other slices that don't belong to the range [**first** slice, **last** slice]
- Why?
  - **Efficiency:** It drastically reduces the amount of data we need to process and store
  - **Model Performance:** It prevents the model from wasting its learning capacity on thousands of blank, meaningless images.

Step 3: Normalize Pixel Intensities

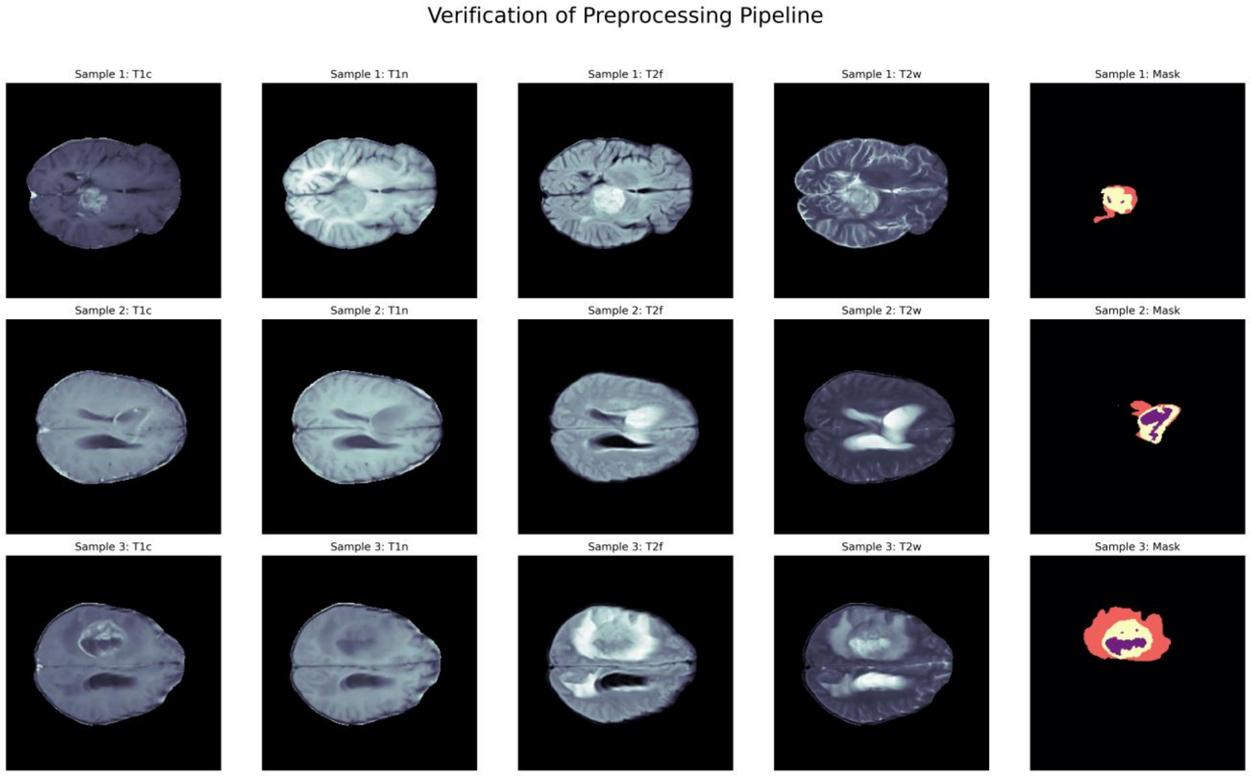
- We will scale the intensity values of the pixels in the 4-channel image volume.
  - A simple and effective method is **Min-Max Normalization**, where we scale the values in each of the four channels to be within a consistent range, typically [0, 1]
- Why?
  - The raw MRI values can be large and vary widely between different scans.
  - Normalization makes the training process much more stable and faster, ensuring that no single modality with unusually large intensity values dominates the learning process

Step 4: Convert the 3D Volume into 2D Slices

- We will slice our cropped, normalized 4-channel 3D volume along the axial (top-down) axis.
  - Each patient volume has a shape of (240, 240, 120, 4), this step will produce 120 individual 2D image slices, each with a shape of (240, 240, 4)
  - The same thing will be done for the segmentation mask, creating 120 corresponding mask slices of shape
- Why?
  - This conversion perfectly formats the data for our 2D U-Net models and massively expands the dataset size, turning 146 scans into thousands of training examples.

Last Format of the preprocessed dataset

- Each row represents one entire sample that the AI model will take
- The first 4 columns represent the 4 channels in the input to the model
- The last column is the ground truth label that the model will try to generate
- Total number of the 2D slices we have: **10,692**



Data Splits

- The entire dataset of processed slices was rigorously split into three sets: **Training (70%)**, **Validation (15%)**, and **Test (15%)**
- Total Set: 10692 slices
- Training Set: 7484 slices
- Validation Set: 1604 slices
- Test Set: 1604 slices

AI Models

Benchmarking Three Distinct Architectures

- Baseline U-Net: The foundational model, built from scratch.
  - It is the gold standard in medical imaging due to its elegant architecture and powerful "skip connections" that preserve fine-grained detail.
- ResNetUNet: An advanced architecture that utilizes transfer learning.
  - It replaces the standard U-Net encoder with a powerful, pre-trained ResNet34 model, giving it a significant head-start in understanding complex visual features.
- TransUNet (Simplified): A cutting-edge hybrid model that represents the latest research trends.
  - It aims to combine the local feature extraction strengths of a CNN with the global context understanding of a Vision Transformer.

Important Note: TransUNet

- The **TransUNet** implementation represents our initial CNN-based foundation, designed as a stepping stone toward incorporating full Vision Transformer components.

Training Setup: Hyperparameters

	BaselineUNet	ResNetUNet	TransUNet
Loss Function	A combined `BCEWithLogitsLoss` + `DiceLoss`	A combined `BCEWithLogitsLoss` + `DiceLoss`	A combined `BCEWithLogitsLoss` + `DiceLoss`
Optimizer	Adam	Adam	Adam
Learning Rate	1e-4	1e-4	1e-5
Batch Size	16	16	8
Training Epochs	100	100	100

Why a Combined Loss?

- Binary Cross-Entropy (BCE) Loss:
  - This is our "pixel-level" loss. It pushes the prediction for every single pixel to be either 0 (background) or 1 (tumor), ensuring fine-grained accuracy. It provides smooth gradients for stable training.
- Dice Loss:
  - This is our "structural" loss. It directly optimizes for the Dice score (our main evaluation metric), forcing the model to get the overall shape and overlap of the tumor correct. It excels at handling the severe class imbalance present in our images

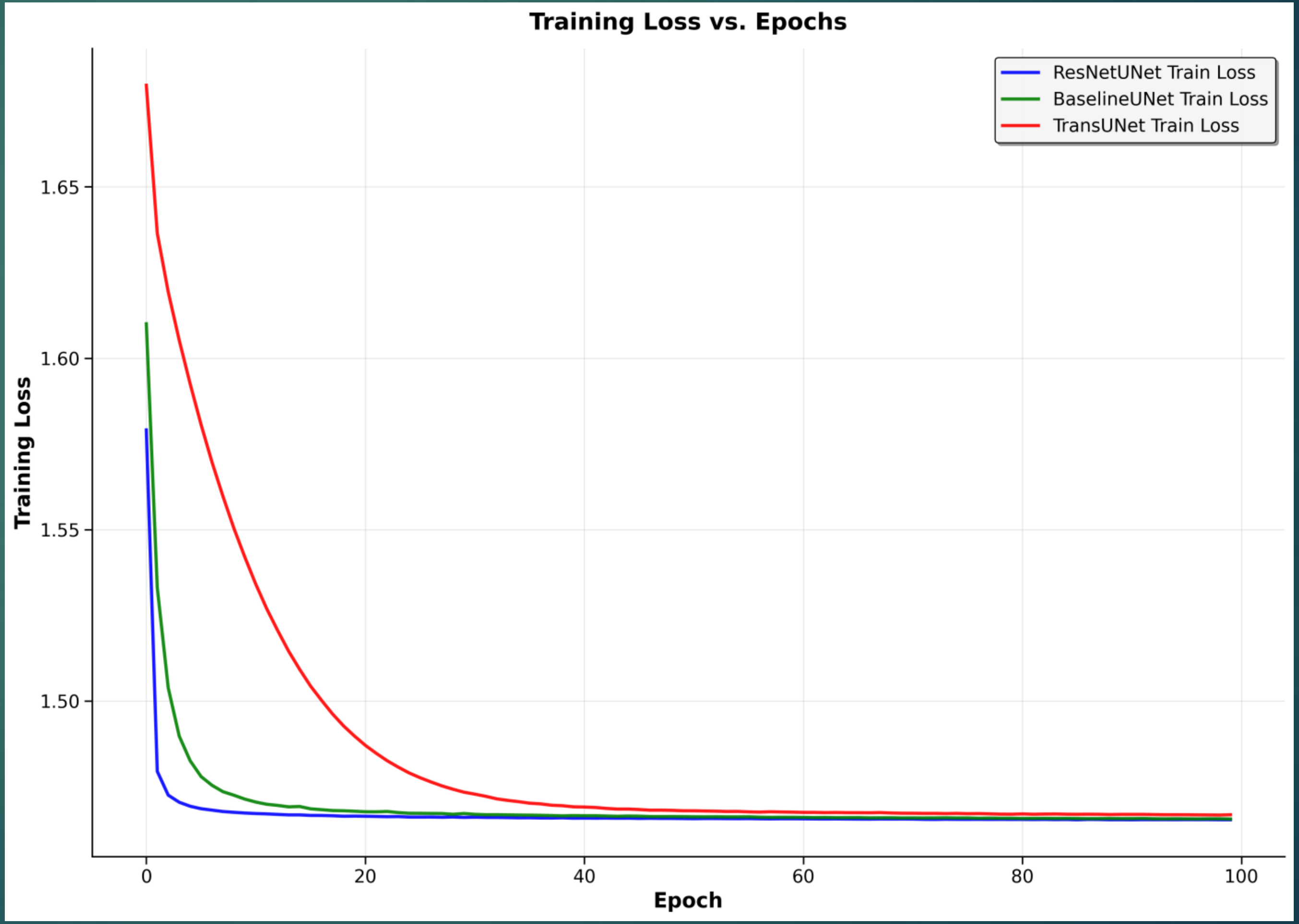
Evaluation & Results

Evaluation Metric

- The Dice Similarity Coefficient (DSC) is the primary metric used to evaluate segmentation performance.
  - It measures the overlap between the model's prediction and the ground truth
- It is calculated as: $DSC = \frac{2\times|A\cap B|}{|A|+|B|}$, where A is the predicted mask and B is the ground truth mask.
- A score of 1.0 indicates a perfect segmentation, while a score of 0 indicates no overlap. It is particularly effective for imbalanced classes, like a small tumor in a large image.

Model Convergence: Training Loss vs. Epochs

This plot shows the training loss for each model over 100 epochs.

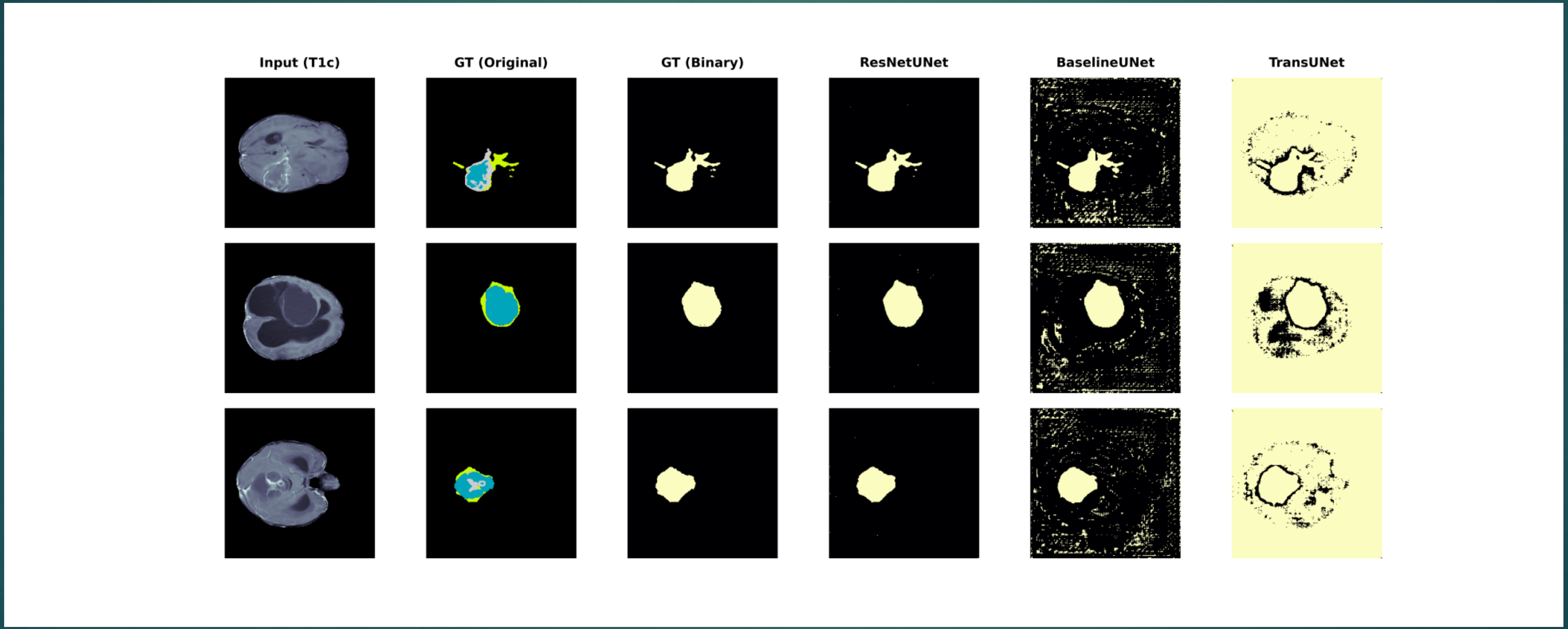


Quantitative Results: Final Test Scores

- **ResNetUNet:**
  - Achieved state-of-the-art performance through effective transfer learning
- **BaselineUNet:**
  - Moderate performance with successful but limited segmentation capability
- **TransUNet**
  - Failed to produce meaningful segmentations due to the simplified architecture used

Model Architecture	Test Dice Score
ResNetUNet	0.9634
BaselineUNet	0.5059
TransUNet	0.0686

Qualitative Results: Visual Prediction Analysis



Analysis of Model Failure: BaselineUNet

- **Observation:** The model correctly identified the main tumor mass but produced many small, scattered false positives across the image.
- **The Reason:** Weaker Generalization
  - Because this model was trained from scratch, it had to learn all visual features exclusively from our dataset.
  - It learned what tumors generally look like, but it isn't confident enough to perfectly distinguish between a real tumor and healthy tissue that just looks similar
  - This lower confidence results in a "noisier" prediction map, where it incorrectly flags small, uncertain regions.

Important: TransUNet

- The **TransUNet** implementation represents our initial CNN-based foundation, designed as a stepping stone toward incorporating full Vision Transformer components. While the current version achieves good training convergence, the simplified architecture (without self-attention mechanisms) explains the lower test performance - providing valuable insights for our planned evolution to a complete TransUNet implementation with proper transformer blocks.

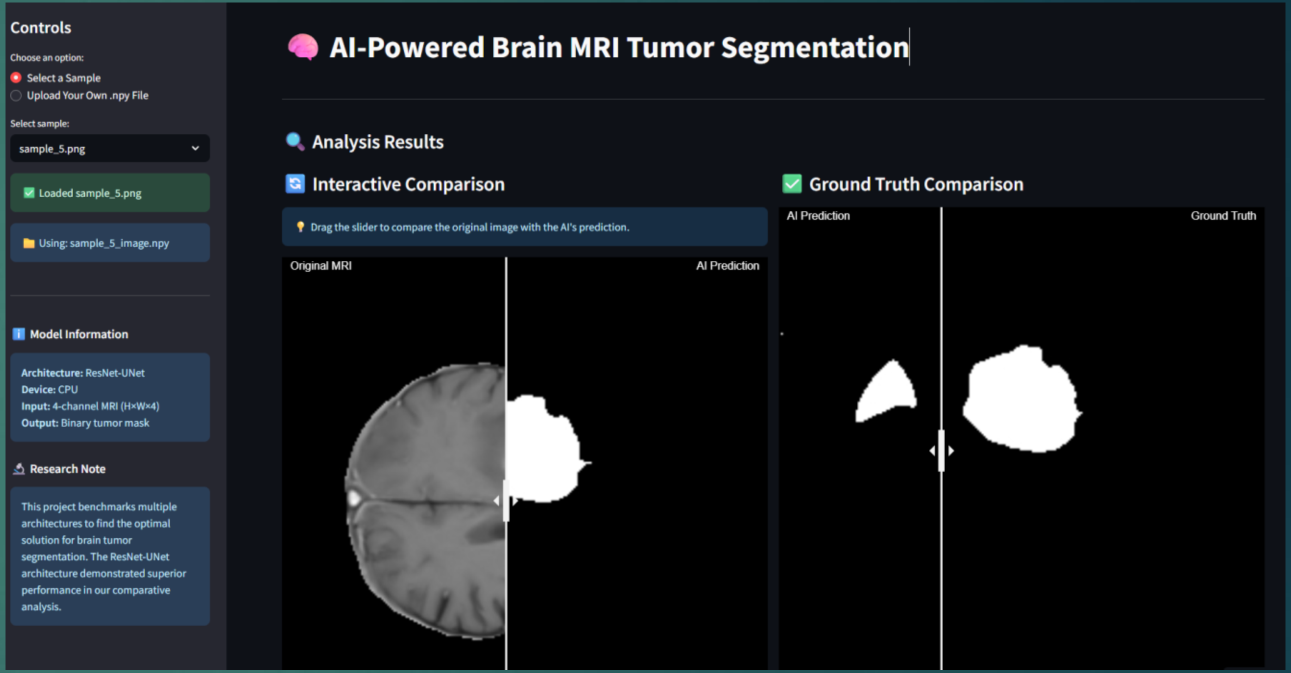
Analysis of Model Failure: TransUNet

- **Observation:** The model produced an "inverted" mask, highlighting the healthy brain and background *instead* of the tumor
- **The Reason:** Critical Architectural Flaw.
  - This is a classic case of **model non-convergence**.
    - Our simplified implementation of the TransUNet was missing the essential **skip connections** that are fundamental to U-Net-like architectures.
    - Without these connections, the decoder did not receive the high-resolution spatial details needed to reconstruct the precise shape of the tumor.
  - The model learned a simple, incorrect "shortcut" to minimize its loss: it segmented the majority class (the background), which was an easier pattern to learn than the complex tumor shape.

Demo & Deployment

Demo & Deployment

- Streamlit library was used for the deployment
- You can access the demo by clicking on this link:
  - https://mri-tumor-segmentation-nahmad.streamlit.app/



Controls

Choose an option:

● Select a Sample
○ Upload Your Own .npy File

Select sample:

sample_5.png

✓ Loaded sample_5.png

Using: sample_5_image.npy

Model Information

Architecture: ResNet-UNet
Device: CPU
Input: 4-channel MRI (H×W×4)
Output: Binary tumor mask

Research Note

This project benchmarks multiple architectures to find the optimal solution for brain tumor segmentation. The ResNet-UNet architecture demonstrated superior performance in our comparative analysis.

AI-Powered Brain MRI Tumor Segmentation

Analysis Results

Interactive Comparison

Drag the slider to compare the original image with the AI's prediction.

✓ Ground Truth Comparison

Conclusion

Conclusion

- **A Clear Winner:** The **ResNet-UNet** was the superior architecture, achieving a state-of-the-art **Test Dice Score of 0.9634**.
- **Transfer Learning is Key:** Leveraging a pre-trained backbone proved to be the most effective strategy for achieving high accuracy and clean, precise segmentations.
- **Critical Architectural Insights:** Model failures provided valuable lessons, confirming the necessity of **skip connections** and highlighting the generalization challenges of training from scratch.