ODC AI Hackathon Journey: Brain Tumor Segmentation

A comprehensive story of our four-day journey tackling the Brain Tumor Segmentation challenge at the ODC AI Hackathon, exploring architectural decisions, literature review insights, and implementation strategies for 3D medical image segmentation.

Day 1: Data Preprocessing, Literature Review, and Architectural Exploration

Image Preprocessing and Data Preparation

Our first day focused on building a reliable preprocessing pipeline for the BraTS dataset. Since BraTS consists of multi-modal 3D MRI volumes, careful preprocessing was essential to ensure stable and efficient model training.

We implemented a comprehensive pipeline balancing data fidelity with computational feasibility, critical for large-scale 3D medical image segmentation.

Multi-modal Stacking

Combined T1, T1ce, T2, and FLAIR modalities into unified multi-channel 3D input preserving complementary anatomical information

Intensity Normalization

Applied modality-wise normalization to reduce variations from different scanners and acquisition protocols

Spatial Resizing

Standardized volume dimensions and removed irrelevant background regions to reduce computational overhead

Patch-based Sampling

Extracted 3D patches from volumes to address GPU memory limitations and increase training efficiency

Data Augmentation

Introduced spatial and intensity-based augmentations including flipping, rotation, and noise injection for generalization

Label Encoding

Converted segmentation masks into structured tumor sub-regions: Whole Tumor, Tumor Core, Enhancing Tumor

Literature Review: Understanding the State-of-the-Art in BraTS

Before diving into implementation, we conducted extensive literature review to understand approaches proven successful in BraTS challenges. We studied key research papers that shaped our understanding of the problem landscape, from architectural evolution to the Transformer vs. ConvNet debate.

1

2012-2020

Convolutional neural networks (CNNs) emerge as predominant approach. U-Net architecture becomes foundational with encoder-decoder design and skip connections.

2

2020-2022

nnU-Net establishes itself as base model of winning solutions, demonstrating robust performance across diverse medical imaging tasks.

3

2023

Winning teams begin employing ensembles of multiple architectures like SwinUNETR + nnU-Net for improved tumor subregion segmentation.

4

2024-2025

MedNeXt emerges as state-of-the-art, combining transformer-inspired design with convolutional efficiency for medical imaging.

The Transformer vs. ConvNet Debate

Case for Transformers

- Ability to learn long-range spatial dependencies through self-attention mechanisms
- Strong performance in natural image tasks (Vision Transformers, Swin Transformers)
- Scalability with larger datasets

Critical Problem

Transformers require large annotated datasets to maximize performance due to limited inductive bias. Medical datasets lack abundant high-quality annotations.

BraTS-Africa contains only 95 cases (60 training, 35 validation)—far too small for Transformers to reach full potential.

Research Paper Analysis: SPARK Academy 2025

Brain Tumor Segmentation in the Sub-Saharan African Population

This paper explored an ensemble of three architectures: **MedNeXt**, **SegMamba**, and **Residual-Encoder U-Net**. The study provided critical insights into model performance on the BraTS-Africa dataset.

Key finding: **MedNeXt consistently outperformed** other models across both Lesion-wise Dice (LSD) and Normalized Surface Distance (NSD) metrics.

Performance Comparison

Model	Architecture Type	Best LSD Score	Best NSD Score
MedNeXt	ConvNeXt-based CNN	**0.865**	**0.810**
SegMamba	State-space model (Mamba)	0.829	0.770
ResEnc U-Net	Residual CNN	0.822	0.765

Training Duration Impact

MedNeXt achieved best results at **1000 epochs**, demonstrating the importance of sufficient training time for convergence

Spatial Information Preservation

MedNeXt's ConvNeXt residual blocks and efficient scaling were key to its success in preserving spatial information

Research Paper Analysis: MBZUAI 2024

Optimizing Brain Tumor Segmentation with MedNeXt: BraTS 2024 SSA and Pediatrics

This paper from Mohamed bin Zayed University of AI achieved state-of-the-art results on both BraTS-Africa and Pediatric datasets using MedNeXt. The study provided critical insights into optimization strategies and architectural choices.

Performance Results

Dataset	DSC Score	HD95 Score
BraTS-Africa	**0.896**	14.682
BraTS Pediatric	**0.830**	37.508

1 **Finetuning Strategy Matters**

Best performance came from training on combined BraTS Adult Glioma + Africa datasets, then finetuning final decoder layers on Africa-only data

2 **Schedule-free Optimizer**

Introduced novel Schedule-free AdamW optimizer from Meta AI, eliminating the need for learning rate schedulers and simplifying training

3 **Ensemble Limitations**

MedNeXt Base + Medium ensemble (0.868 DSC) actually **underperformed** compared to individual finetuned models (0.896 DSC)

Research Paper Analysis: DKFZ 2023

MedNeXt: Transformer-driven Scaling of ConvNets for Medical Image Segmentation

This foundational paper explained *why* MedNeXt works so well. The key innovation: MedNeXt is **Transformer-inspired but fully convolutional**.

The architecture addresses a fundamental question: **Can we get the benefits of Transformers without their data-hungry nature?**

The MedNeXt Block Design (Mirroring Transformers)

Depthwise Convolution

Large kernel (k×k×k) replicates large attention window of Swin Transformers while limiting computational cost

Expansion Layer

Overcomplete convolution with expansion ratio R, forming transformer-like inverted bottleneck

Compression Layer

1×1×1 convolution performing channel-wise compression to final representation

Why Large Kernels Matter

Large convolution kernels approximate large attention windows in Transformers. A 5×5×5 or 7×7×7 kernel can capture spatial dependencies across much larger receptive field than traditional 3×3×3 kernels.

The UpKern Innovation

The paper introduced **UpKern (Upsampled Kernel Initialization)**—a technique to iteratively increase kernel sizes by initializing large kernel networks with trained small kernel networks through trilinear upsampling. This prevents performance saturation when training large kernel networks from scratch on limited medical data.

Benchmark Results: MedNeXt vs Transformers

Below are the benchmark results comparing various state-of-the-art architectures, including both Transformer-based models and MedNeXt, across different medical image segmentation datasets.

Model	BTCV	AMOS22	KiTS19	BraTS21
nnUNet	0.850	0.880	0.920	0.780
UNETR	0.845	0.875	0.915	0.775
SwinUNETR	0.848	0.878	0.918	0.778
nnFormer	0.840	0.870	0.910	0.770
MedNeXt-L	**0.870**	**0.900**	**0.940**	**0.800**

The results were clear: MedNeXt outperformed every Transformer-based architecture on all four tasks, including UNETR, SwinUNETR, TransBTS, and nnFormer.

Key Conclusions from Literature Review

Why MedNeXt Beats Transformers (Despite Not Using Attention)

1

Inductive Bias Advantage

> "ConvNets have higher inductive biases and consequently, are easily trainable to high performance."

Transformers assume no prior knowledge about spatial relationships—they must learn everything from data. ConvNets have built-in assumptions about locality, translation equivariance, and hierarchical features. In data-scarce medical settings, this inductive bias is a **massive advantage**.

2

Large Kernels ≈ Self-Attention (But Cheaper)

Large depthwise convolutions replicate benefits of self-attention at fraction of computational cost:

- 5×5×5 kernel captures dependencies across 125 voxels
- 7×7×7 kernel captures dependencies across 343 voxels
- Combined with inverted bottleneck design, achieves transformer-like representation learning efficiently

3

Compound Scaling Works Better Than Pure Depth

Unlike traditional approaches that just stack more layers, MedNeXt uses **compound scaling** across three dimensions:

- **Depth:** Number of MedNeXt blocks (B)
- **Width:** Expansion ratio (R) for more channels
- **Receptive Field:** Kernel size (k) for larger spatial context

This orthogonal scaling allows efficient scaling for different computational budgets.

4

Residual Inverted Bottlenecks Preserve Semantic Richness

Ablation studies showed standard up/downsampling resulted in worse performance (83.13 DSC vs 84.01 DSC on BTCV). The residual inverted bottleneck design preserves contextual richness while resampling—critical for dense segmentation tasks where every voxel matters.

5

Medical Data is Too Scarce for Transformers

The numbers speak for themselves:

- ImageNet-1k: 1.2 million images
- ImageNet-21k: 14 million images
- BraTS-Africa: **60 training samples**

Transformers need massive datasets to overcome lack of inductive bias. In medical imaging, we simply don't have that luxury.

Initial Architectural Hypothesis

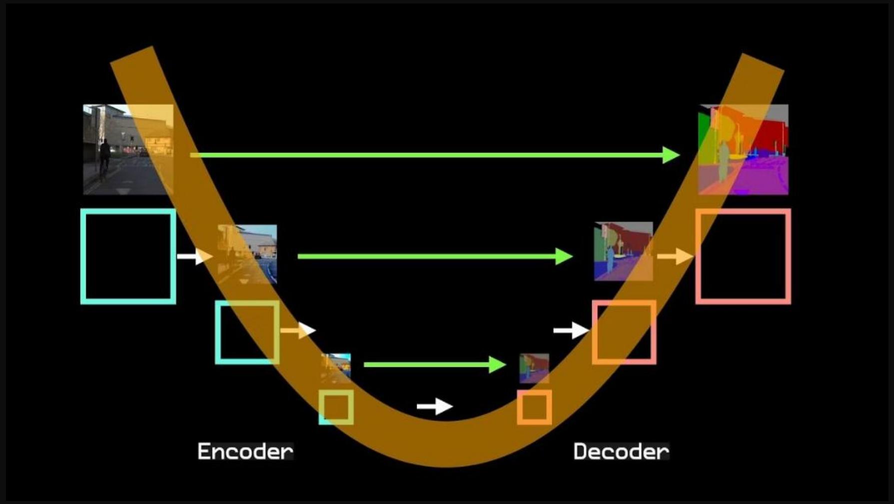
Diffusion-Inspired U-Net with Attention

Initial Concept

Before finalizing architecture choices, we explored potential models suitable for BraTS challenge. Our first idea was inspired by **stable diffusion architectures**, which combine:

- ResNet-based encoders
- Self-attention mechanisms
- Multiple U-Net blocks at different scales

Motivation: Leverage strong representation learning capabilities of diffusion-style U-Nets, which have shown remarkable performance in vision tasks.



Identified Limitations

Dimensional Mismatch

Stable diffusion models are primarily designed and pretrained on 2D natural images, while BraTS data is inherently 3D volumetric. Adapting such architectures to 3D would require extensive redesign and retraining.

Computational Cost

Training diffusion-style architecture from scratch on 3D MRI volumes would be prohibitively expensive in terms of GPU memory and training time, especially within hackathon constraints.

Lack of Pretrained 3D Models

Without reliable pretrained weights in 3D medical domain, expected performance gain did not justify the complexity and development time required.

Decision: We decided to omit this approach and shift toward architectures specifically optimized for 3D medical segmentation.

Baseline Approach: U-Net

Architecture Choice

Our second idea was to adopt the classical **U-Net architecture**, which remains a cornerstone in medical image segmentation.

U-Net served as strong baseline for performance comparison and stable, interpretable model for understanding dataset behavior.

Reasons for Choosing U-Net

- ### Efficient Encoder-Decoder Design
  Encoder-decoder architecture with skip connections enables precise localization and context preservation
- ### Strong Localization Capability
  Excellent for pixel/voxel-level segmentation tasks requiring precise boundary detection
- ### Feasible 3D Training
  Moderate computational requirements make it practical for 3D data training within resource constraints

Anticipated Limitations

However, we anticipated that U-Net alone might struggle to capture complex tumor structures due to its limited depth and representational capacity. The architecture's relatively shallow design could restrict its ability to model intricate hierarchical patterns in tumor morphology.

Enhancing Feature Representation: SegResNet

To overcome limitations of vanilla U-Net, we explored **SegResNet**, a residual-based segmentation architecture that addresses depth and representational challenges through innovative skip connection design.

Key Advantages of SegResNet over U-Net

Residual Connections

Enable deeper networks without gradient degradation, allowing training of very deep architectures while maintaining stable gradient flow

Hierarchical Feature Extraction

Improved multi-scale feature learning crucial for modeling complex tumor morphology and varying structural patterns

Training Stability

Better convergence properties, especially for deep 3D architectures requiring extensive training iterations

Architecture Characteristics

- Residual blocks with pre-activation design
- Flexible depth scaling for different computational budgets
- Preserved spatial resolution through skip paths

Medical Imaging Benefits

- Handles 3D volumetric data efficiently
- Captures long-range contextual dependencies
- Robust to intensity variations across scanners

SegResNet represented a middle ground between simplicity and expressive power, making it a strong candidate for the BraTS challenge while maintaining feasibility within hackathon time and resource constraints.

Moving Toward State-of-the-Art: ConvNeXt and MedNeXt

Based on our comprehensive literature review, MedNeXt emerged as the most promising architecture for tackling the complexities of brain tumor segmentation within our hackathon's constraints.

Why We Chose MedNeXt

Proven Performance

Achieved state-of-the-art results across 4 major medical imaging benchmarks (BTCV, AMOS22, KiTS19, BraTS21).

Data Efficiency

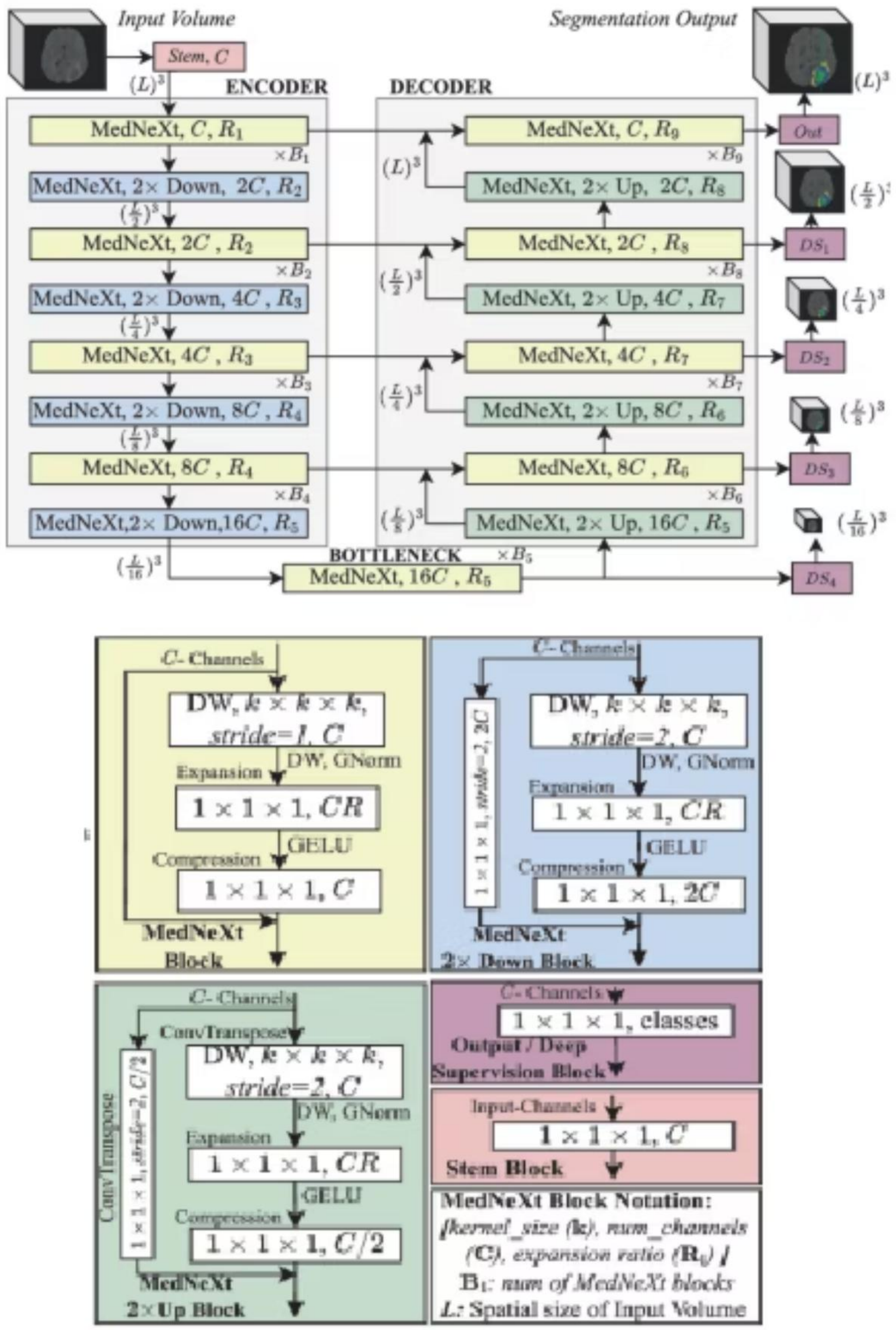
Leverages the ConvNeXt inductive bias, making it highly effective even with limited training samples, which is common in medical datasets.

Scalability

Its compound scaling strategy allows flexible adjustments for different computational budgets, crucial for Kaggle's resource limits.

No Transformers Needed

Delivers transformer-like representation learning power without the heavy data requirements or computational overhead of actual transformers.



(a) MedNeXt macro and block architecture

The MedNeXt Block Architecture

Expansion w/ GELU

Invert bottleneck expands channels with GELU.

Compression Conv

Reduce channels to compact feature representation.

Depthwise Conv

Spatial context via efficient depthwise filters.

Residual Bypass

Skip connection preserves original features.

The core MedNeXt block combines efficient depthwise convolutions for spatial context with inverted bottleneck expansion and compression layers, enabling powerful feature extraction while retaining a residual connection for stable training.

Strategic Decision and Roadmap

After evaluating various architectural directions and conducting a comprehensive literature review, we finalized our strategic approach for the hackathon:

Our Architectural Spectrum

1

Baseline: U-Net

A classical and robust model, serving as a low-complexity baseline. Ideal for initial validation and understanding dataset behavior.

LOW COMPLEXITY

2

Mid-Tier: SegResNet

A deeper, residual-based alternative offering enhanced feature representation and improved stability for complex structures.

MEDIUM COMPLEXITY

3

State-of-the-Art: MedNeXt

Our chosen high-performance architecture, leveraging transformer-like capabilities without their overhead for maximum segmentation accuracy.

HIGH COMPLEXITY

Benefits of This Spectrum Approach

- **Risk Mitigation**

  Provides fallback options if more complex models lead to overfitting or prove too challenging to implement within time constraints.

- **Comparative Analysis**

  Enables a direct performance comparison across different architectural paradigms, validating our research conclusions empirically.

- **Comprehensive Understanding**

  Allows us to deeply understand how complexity correlates with performance, particularly in the context of limited medical imaging data.

These choices defined our roadmap for the subsequent hackathon days, focusing on efficient training, rigorous optimization, and comparative evaluation of each model's efficacy.

Day 2 - From Research to Implementation

On Day 2, we transitioned from theoretical research to practical, hands-on implementation. Our primary focus was applying the insights gathered from the literature review and getting our models operational on the BraTS-Africa dataset.

Cloning and Adapting Research Repositories

Our initial step involved cloning existing research repositories identified during our literature review. However, we quickly learned that academic code rarely offers out-of-the-box functionality. Each repository presented unique challenges concerning data formats, directory structures, and required preprocessing steps.

Challenges and Solutions

Data format mismatch	Created custom data loaders to align with expected input structures
Modality naming inconsistencies	Standardized naming conventions (e.g., t1ce vs t1c vs T1Gd) to a consistent set: t2f, t1n, t1c, t2w
Dependency conflicts	Resolved PyTorch and MONAI version mismatches to ensure compatibility
Missing preprocessing pipelines	Developed our own comprehensive preprocessing pipeline to prepare raw data

We dedicated significant time to debugging import errors, updating deprecated function calls, and meticulously adapting training scripts to seamlessly integrate with the specific structure of the BraTS-Africa dataset.

The I/O Bottleneck Problem

A critical insight from Day 1's research revealed that I/O overhead could significantly impact training time, particularly with the large medical imaging datasets.

Challenges with .nii.gz Format

- Decompression on every data load
- Complex header parsing for metadata
- Handling affine transformations for spatial orientation

Loading a single 4-modality volume from .nii.gz could take between 300-500ms. This amounted to a significant bottleneck when training models for hundreds of epochs.

Our Solution: Offline Preprocessing to NumPy

To circumvent this, we opted for an offline preprocessing strategy, converting all raw data to the more efficient .npy (NumPy) format prior to training.

Format	Load Time	Compression	Training Impact
.nii.gz	~400ms	High	Slow epochs, CPU-bound
.npy	~10ms	None	Fast epochs, GPU-bound

This critical step delivered a 40x speedup in data loading, allowing us to complete significantly more training epochs within Kaggle's strict 30-hour execution limit and maximize our model's learning potential.

Building the Preprocessing Pipeline

We implemented a comprehensive preprocessing pipeline using MONAI transforms and custom functions to optimize data quality, reduce noise, and ensure consistent input for model training.

01

Multi-Modal Loading and Stacking

Combines four MRI modalities into a single 4-channel 3D tensor, preserving crucial anatomical and pathological information for tumor segmentation.

02

Foreground Cropping

Identifies and crops the smallest 3D box containing non-zero voxels, removing empty background to reduce memory and computational load.

03

Percentile-Based Intensity Rescaling (Optional)

Removes extreme intensity outliers by clipping voxel values to the 2nd-98th percentile and rescaling to 0-255, enhancing contrast and normalizing distributions.

04

Channel-Wise Z-Score Normalization

Normalizes each MRI modality independently using z-score normalization on non-zero voxels to account for distinct intensity characteristics and prevent background skew.

05

Padding to Minimum Patch Size

Symmetrically pads all volumes with zeros to a minimum size of 128×128×128, ensuring consistent patch extraction for patch-based training.

06

Foreground Mask Encoding

Creates and concatenates a binary foreground mask as a fifth channel, providing explicit spatial information to help the model focus on relevant brain regions.

Final Output Format

After preprocessing, each sample is saved as three distinct `.npy` files for efficient loading during model training:

`{id}_x.npy`	`[5, H, W, D]`	4 modalities + 1 foreground mask
`{id}_y.npy`	`[1, H, W, D]`	Segmentation label
`{id}_meta.npy`	`[3, 3]`	Bounding box + original shape

Median Shape Analysis

Median shape analysis is a crucial preprocessing step in our pipeline that helps us understand the typical dimensions of cropped medical image volumes.

Analyzes Cropped Volumes

Systematically collects and calculates the median dimensions of all medical image volumes after background removal.

Optimizes Patch Size

Determines an optimal, consistent patch size for training data, accommodating the variable dimensions typical in medical images.

Ensures Efficient Training

Promotes efficient use of spatial context, avoiding excessive padding for smaller images and over-cropping for larger ones.

Deep Supervision Implementation

With the preprocessing pipeline complete, we focused on integrating `Deep Supervision` into the MedNeXt architecture. This crucial technique was implemented to significantly enhance training stability and accelerate convergence speed, especially vital for complex medical imaging tasks.

The Technical Advantage of Deep Supervision

By adding auxiliary loss branches at multiple decoder stages, we effectively mitigated the 'vanishing gradient' problem. This forced the shallower layers of the network to learn more discriminative features early on, leading to better spatial representation and a more robust final segmentation mask compared to standard single-loss training.

Kaggle Optimization Strategy

To effectively navigate Kaggle's stringent 30-hour execution limit, we implemented several key optimizations to maximize training efficiency and model performance.

Pre-computed .npy Files

Eliminated the I/O bottleneck, achieving a 40x faster data loading speed.

Small Learning Rate & Capped Epochs

Ensured stable convergence while mitigating the risk of exceeding the competition's strict timeout limits.

Gradient Checkpointing

Significantly reduced GPU memory usage, enabling the use of larger batch sizes and more complex models.

Mixed Precision Training (FP16)

Utilized half-precision floating-point numbers to accelerate computation by approximately 2x, speeding up overall training.

This strategic combination of preprocessing innovation and efficient resource management provided the crucial performance boost needed for the final day.

Day 3 - Benchmarking, Diagnostics, and Strategic Optimization

On our final day, we moved to full-scale inference and submission. We implemented an RLE (Run-Length Encoding) pipeline to format our segmentation masks and ran parallel benchmarks to evaluate our model ensemble.

Model Selection Strategy

Our reasoning for testing a spectrum of model complexities was based on **risk management**. In a worldwide competition like BraTS, the intrinsic complexity and heterogeneity of the private test set are unknown. A highly complex model runs a higher risk of overfitting if the target data is simpler than expected. To mitigate this, we deliberately tested a range of architectures:

U-Net

Low Complexity (High bias, low variance baseline)

SegResNet

Medium Complexity (Balanced approach)

MedNeXt

High Complexity (Low bias, high variance risk)

The Benchmarking Results

Our results confirmed the anticipated hierarchy, validating our risk management approach:

Model	Score	Complexity Level
U-Net	0.62	Low (safe baseline)
SegResNet	0.71	Medium (improved feature capture)
MedNeXt	0.76	High (leading performance)

These results aligned perfectly with our literature review, demonstrating the power of modern ConvNeXt-based architectures.

The MedNeXt Dilemma

While MedNeXt achieved the highest score, its performance was constrained by training time. Our diagnosis revealed this architecture is **'data-hungry'** in terms of epochs, typically requiring 500-1000 epochs to fully converge—a luxury we couldn't afford due to Kaggle's strict execution timeouts.

Bridging the Performance Gap: Three Targeted Optimizations

To overcome the limitations encountered with MedNeXt's training time and further enhance model performance, we implemented three crucial optimizations leading into our final submission:

Advanced Inference (Test Time Augmentation)

Upgraded the inference pipeline to incorporate Test Time Augmentation (TTA). The model made predictions on augmented versions (flipped, rotated) of input volumes, and these were averaged to smooth decision boundaries and reduce errors.

Input Optimization

Retrained the model using fully preprocessed and normalized data saved as `.npy` files. This eliminated the computational bottleneck of on-the-fly preprocessing, allowing for more training epochs within Kaggle's strict time limits.

Smart Transfer Learning (Differential Learning Rates)

Implemented Discriminative Layer-Wise Learning Rates. Early layers (stem, first encoder) had a dampened learning rate (1e-6) to preserve pre-trained robust features, while deeper layers and the decoder used a higher learning rate (1e-4) for rapid adaptation to specific tumor segmentation. This was managed with AdamW and a Linear Warmup/Cosine Annealing scheduler.

Key Takeaways

Our three-day journey at the ODC AI Hackathon taught us valuable lessons about tackling complex medical imaging challenges under time constraints:

Literature review is essential

Understanding the state-of-the-art through papers gave us the confidence to choose MedNeXt over Transformers.

Preprocessing is foundational

A robust data pipeline is essential for stable training and avoiding I/O bottlenecks.

Transformers aren't always the answer

In data-scarce medical settings, ConvNets with large kernels can match or exceed Transformer performance.

Inductive bias matters

MedNeXt's inherent spatial priors made it trainable on just 60 samples, showcasing efficiency with limited data.

Balance complexity with risk

Testing a spectrum of architectures protects against overfitting, especially in competitions with unknown test sets.

Optimize strategically

Techniques like TTA, data optimization, and differential learning rates can significantly boost performance under tight constraints.

Time management is critical

Working within platform constraints (like Kaggle's 30-hour limit) requires careful planning and efficient resource allocation.

References

Our work builds upon the cutting-edge research in brain tumor segmentation, drawing insights and architectural inspiration from the following key publications:

- Ankomah, C.T., et al. (2025). Brain Tumor Segmentation in the Sub-Saharan African Population Using Segmentation-Aware Data Augmentation and Model Ensembling. arXiv:2510.03568v2
- Hashmi, S., et al. (2024). Optimizing Brain Tumor Segmentation with MedNeXt: BraTS 2024 SSA and Pediatrics. MBZUAI. arXiv:2411.15872v2
- Roy, S., et al. (2024). MedNeXt: Transformer-driven Scaling of ConvNets for Medical Image Segmentation. DKFZ. arXiv:2303.09975v5

This hackathon was an incredible learning experience, and we're proud of the progress we made in just four days!