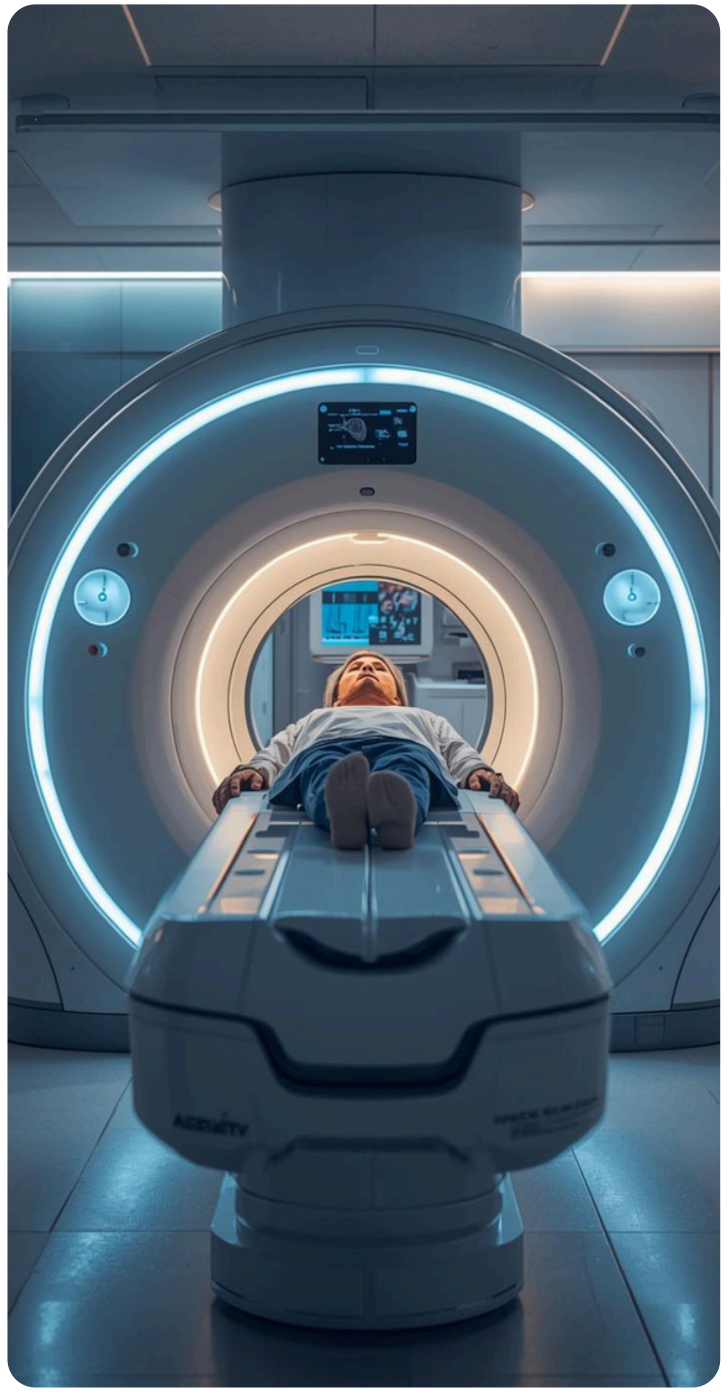
AI-Enhanced Virtual Biopsies for Brain Tumor Diagnosis in Low Resource Settings

Areeb Ehsan
Georgia State University

Why Brain Tumor Diagnosis is Difficult

- MRI is the primary imaging tool, but diagnosis is time-consuming and resource-dependent.
- Surgical biopsy is invasive, risky, and not always feasible (deep or sensitive tumor locations).
- Many hospitals, especially in low-resource settings, lack access to a proper infrastructure and experts.
- Modern 3D CNN models often require powerful GPUs that rural / underfunded hospitals don’t have.



MRI VARIABILITY

Variability affects diagnostic accuracy significantly.

BIOPSY RISKS

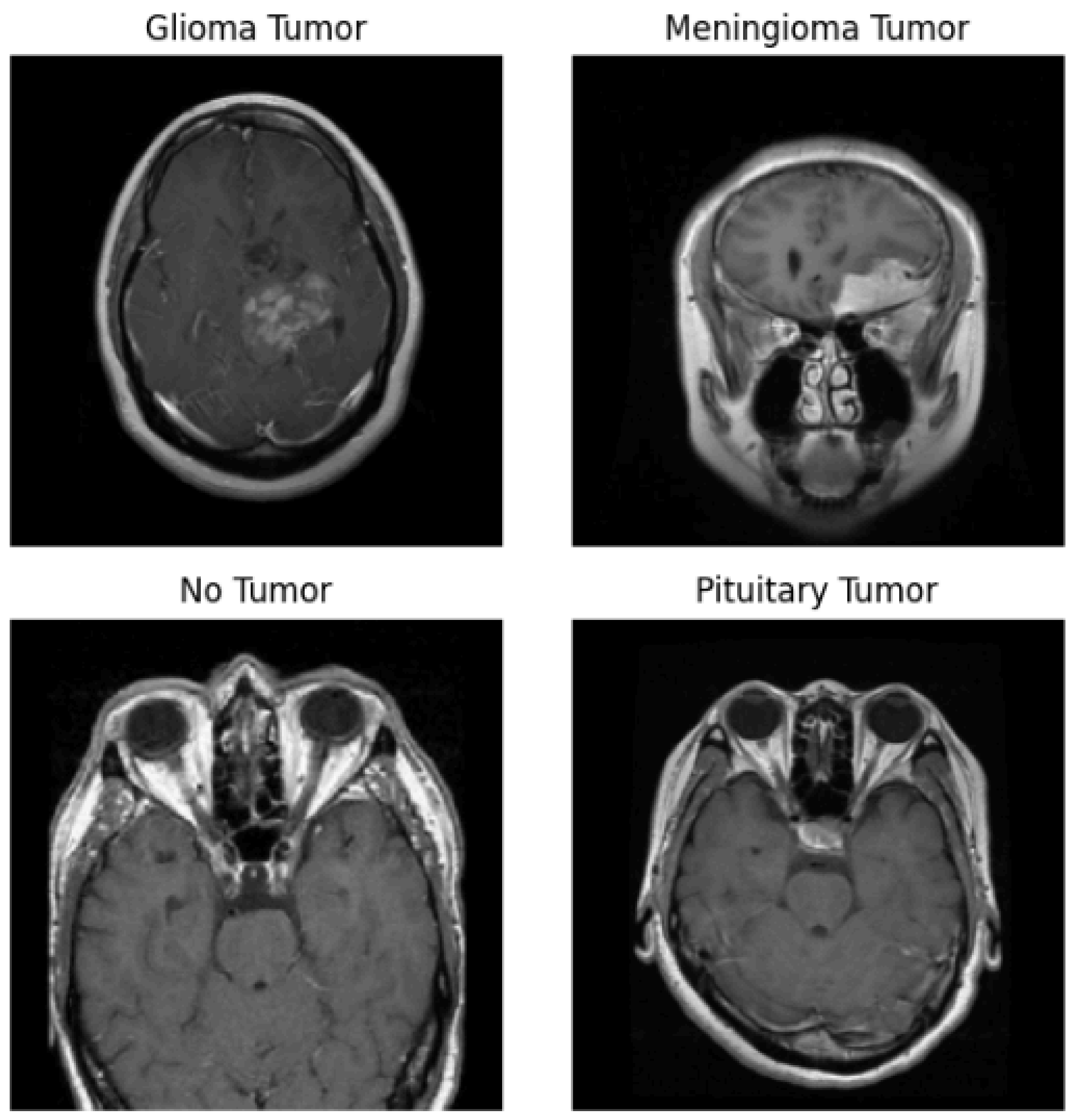
Invasive procedures pose health risks for patients.

Research Objective Overview

- Develop a non-invasive virtual biopsy system using MRI scans only.
- Combine radiomics features (shape, intensity, texture) with CNN deep features (spatial patterns).
- Make predictions explainable using:
  - Grad-CAM heatmaps
  - Radiomics-based feature summaries
- Evaluate model robustness under low-resource conditions:
  - Lower resolution images
  - Noisy scans
- Aim for a system that is accurate, interpretable, and deployable on standard hospital hardware

Dataset Overview

- **4 classes:**
  - Glioma tumor
  - Meningioma tumor
  - Pituitary tumor
  - No tumor
- **Train/val split:**
  - Training set: 2,870 images (approx., after split)
  - Validation set: 20% of training (~574 images in my setup)
- **Test set:**
  - 394 images

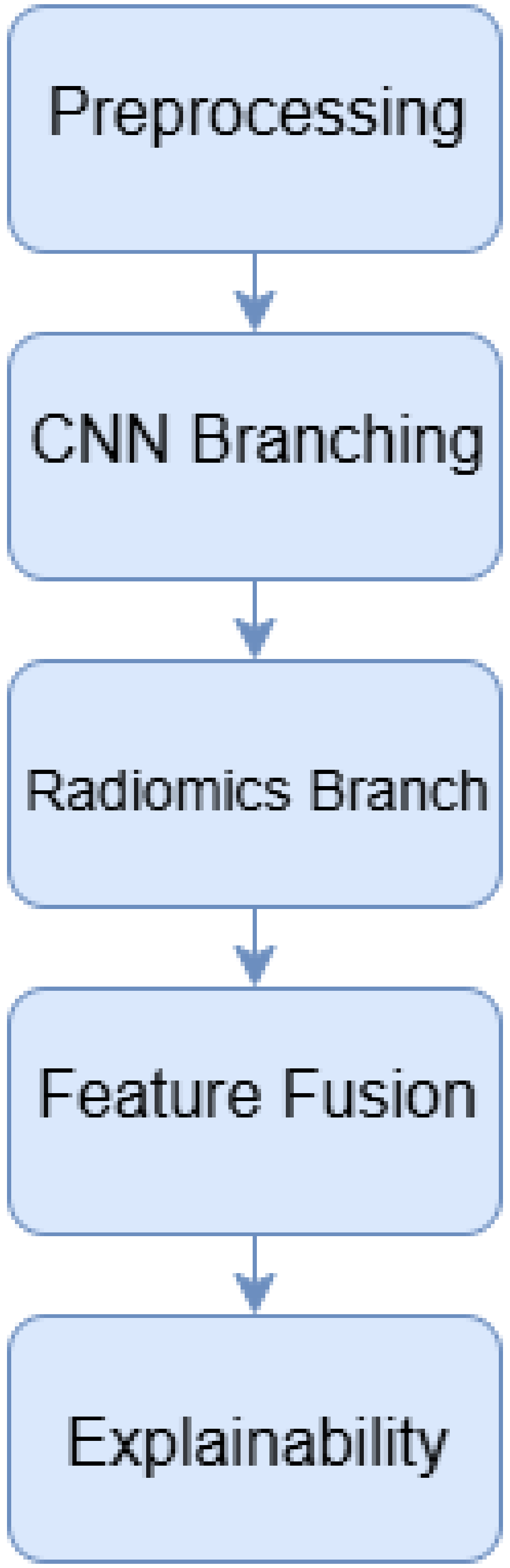


Dataset: Brain Tumor Classification (MRI) by Sartaj Bhuvaji (Kaggle).

kaggle

System Architecture Overview

- **Preprocessing & Normalization**
  - Resize to 224×224, normalize channels, standard transforms.
- **CNN Branch (MobileNetV2)**
  - Pretrained on ImageNet
  - Outputs a 1280-dimensional latent feature vector per image.
- **Radiomics Branch**
  - Extracts 8 interpretable features:
    - Shape: area, eccentricity, solidity
    - Intensity: mean, standard deviation
    - Texture (GLCM): contrast, homogeneity, entropy
- **Feature Fusion**
  - Concatenate: [CNN features | Radiomics features]
  - Train a RandomForest classifier on fused features.
- **Explainability**
  - Grad-CAM heatmaps highlight influential tumor regions.
  - Radiomics feature importance gives numerical justification.



Data Loading & Preprocessing

What Was Implemented

- Imported libraries: PyTorch, Skimage, OpenCV, sklearn
- Downloaded dataset using KaggleHub
- Preprocessing:
  - Resize → 224×224
  - Normalize
  - Convert to Tensor
- Train/val split: 80/20
- Created PyTorch DataLoaders

CNN Model

MobileNetV2 Classifier

- Pretrained on ImageNet
- Replaced final layer → 4 tumor classes
- 3 training epochs on CPU (efficient)
- Evaluation:
  - Validation accuracy = 88-95%
  - Test accuracy: 74.6%
  - Macro-F1: ~0.71
- Issues: test set domain shift reduces accuracy

```python
import torch.optim as optim # import optimizer module

# Loss and optimizer
criterion = nn.CrossEntropyLoss() # define loss function
optimizer = optim.Adam(model.parameters(), lr=1e-4) # Adam optimizer

num_epochs = 3  # starting small but we can increase later

for epoch in range(1, num_epochs + 1): # epochs
    model.train() # set to training mode
    running_loss = 0.0

    for images, labels in train_loader: # iterating over batches
        images = images.to(device)
        labels = labels.to(device)

        optimizer.zero_grad() # zero the gradients
        outputs = model(images)
        loss = criterion(outputs, labels)
        loss.backward()
        optimizer.step()

        running_loss += loss.item() * images.size(0) # accumulate loss

    # Average training loss for this epoch
    train_loss = running_loss / len(train_loader.dataset)

    # Evaluating on the validation split
    val_acc, val_f1 = evaluate_model(model, val_loader, device)

    print(f"Epoch {epoch}/{num_epochs} "
          f"- Train loss: {train_loss:.4f} "
          f"- Val Acc: {val_acc:.4f} "
          f"- Val Macro-F1: {val_f1:.4f}")
```

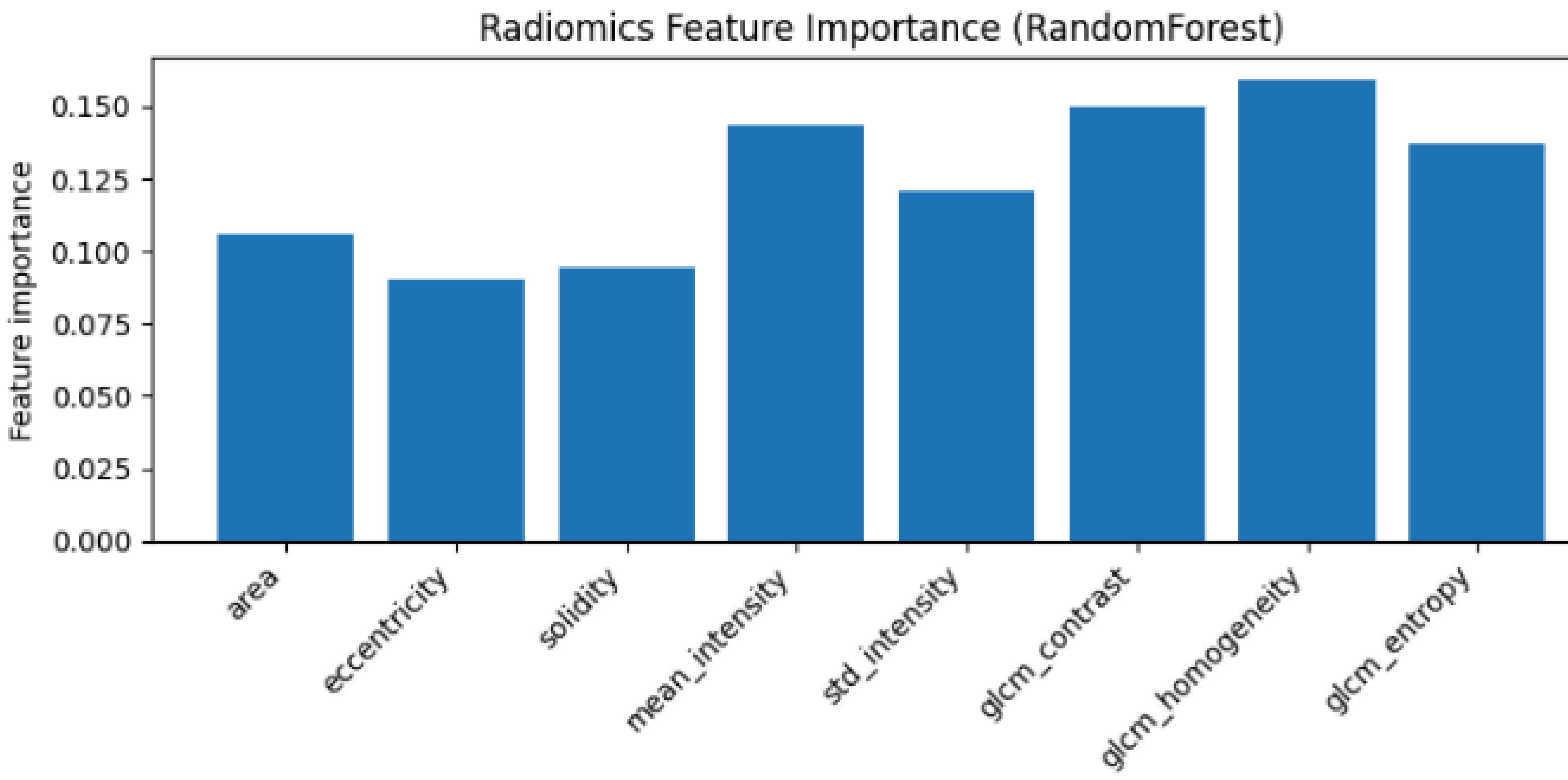
✓ 3m 48.7s

```
Epoch 1/3 - Train loss: 0.7347 - Val Acc: 0.8780 - Val Macro-F1: 0.8805
Epoch 2/3 - Train loss: 0.2680 - Val Acc: 0.9164 - Val Macro-F1: 0.9185
Epoch 3/3 - Train loss: 0.1313 - Val Acc: 0.9512 - Val Macro-F1: 0.9539
```

Radiomics Feature Extraction

8 Radiomics Features Extracted

- Shape: Area, Eccentricity, Solidity
- Intensity: Mean, Standard Deviation
- Texture (GLCM): Contrast, Homogeneity
- Simple Otsu-based tumor mask
- Classical ML model: RandomForest



Fusion Model

CNN + Radiomics Hybrid Classifier

Fusion vector:
→ [CNN 1280 features | Radiomics 8 features]
→ Total = 1288 dimensions

Results:

- Validation:
  - Accuracy: 95.12%
  - Macro-F1: 95.12%
- Test:
  - Accuracy: 75.38%
  - Macro-F1: 72.39%

1 Model Performance Summary

Model	Val. Accuracy	Val. Macro F1	Test Accuracy	Test Macro F1
CNN Only	0.85	0.84	0.746	0.74
Radiomics Only	0.7422	0.7463	0.6701	0.6307
Fusion (CNN + Radiomics)	**0.9512**	**0.9512**	**0.7538**	**0.7239**

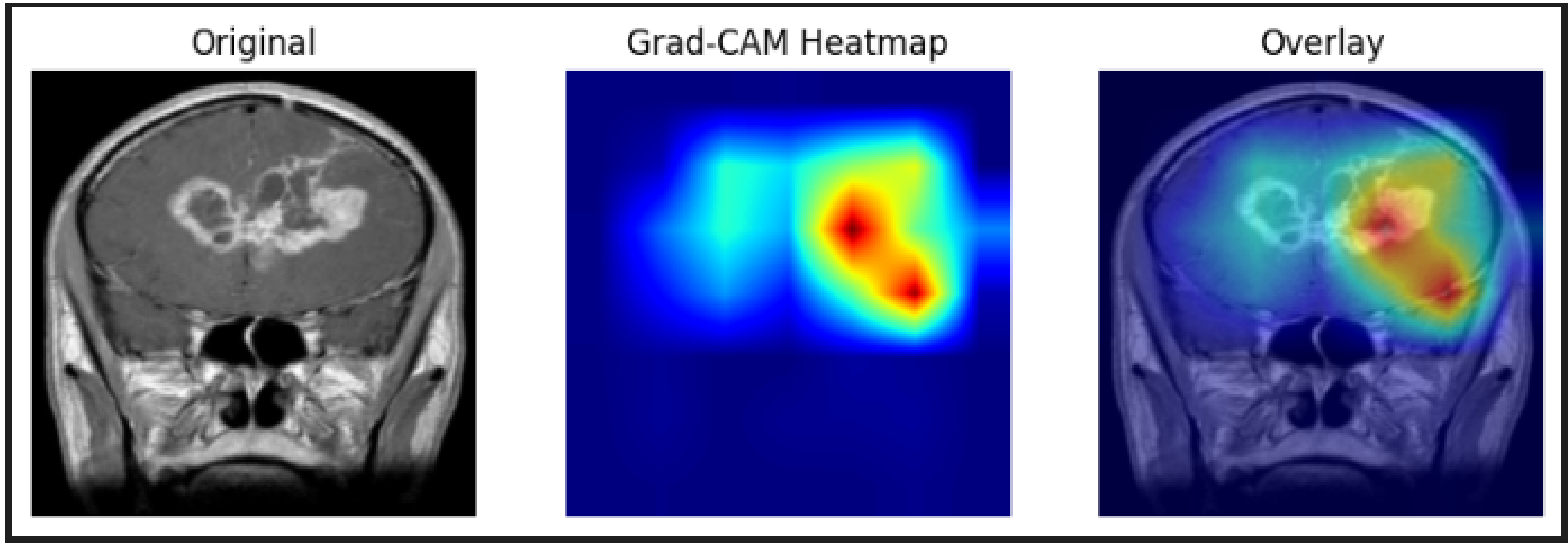
Interpretation:

- Fusion improves validation performance
- Robust to class imbalance
- Test gap due to domain shift

Grad-Cam Explainability

Explainability Module

- Implemented hook-based Grad-CAM
- Extracted:
  - Final convolutional layer gradients
  - Activation maps
  - Weighted heatmap
- Produced:
  - Original MRI
  - Heatmap
  - Overlay
- Shows where the CNN focuses



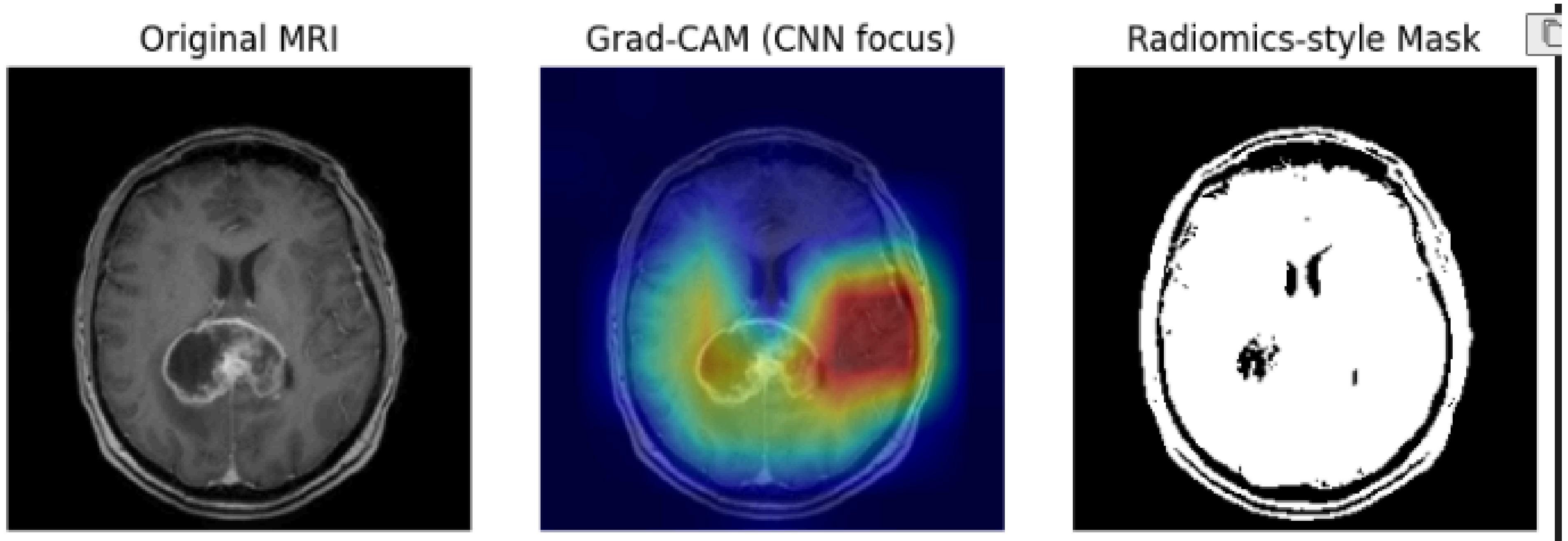
Case Study: Virtual Biopsy on Sample MRI #26

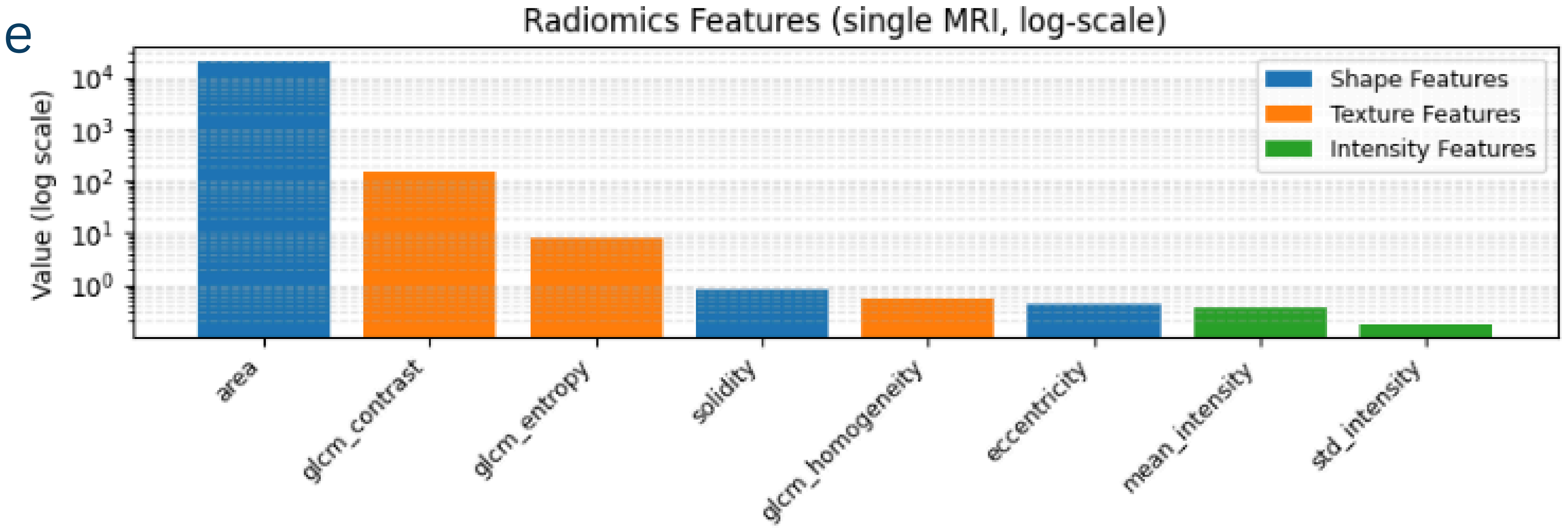
Patient MRI #26

- Ground truth: glioma tumor
- Fusion model prediction: glioma tumor (correct)

Interpretability

- Grad-CAM highlights irregular tumor mass in the left hemisphere
- Radiomics mask isolates the lesion region automatically
- Radiomics profile shows:
  - High area (large lesion)
  - Elevated GLCM contrast/entropy (heterogeneity)
  - Moderate intensity variance





Robustness in Low-Resource Conditions

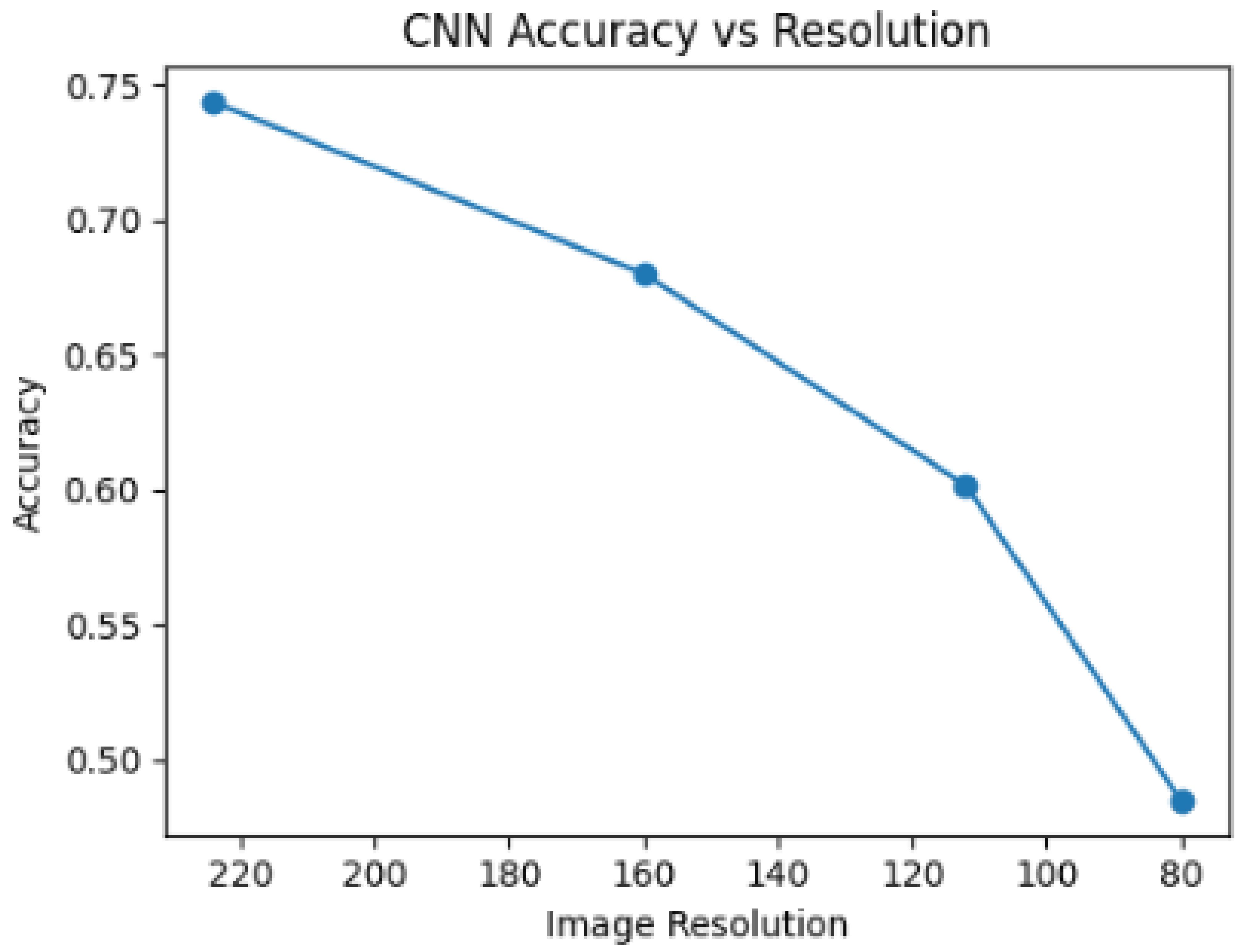
Low-Resource Simulation

Two corruption types:

1. Low-Resolution MRI
2. Gaussian Noise

Resolution Accuracy:

- 224×224 → 0.74
- 160×160 → 0.68
- 112×112 → 0.60
- 80×80 → 0.48



Robustness in Low-Resource Conditions

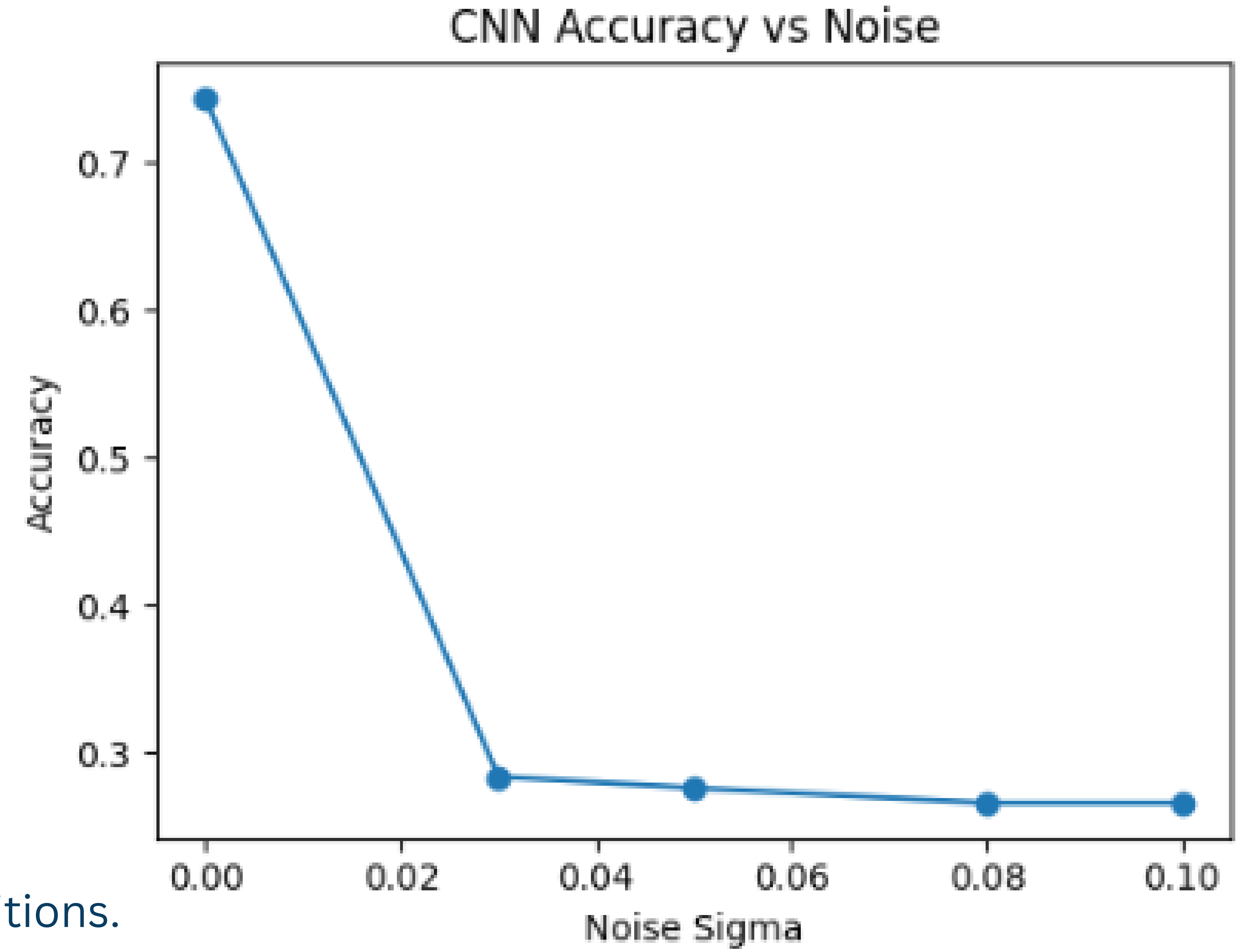
Gaussian Noise:

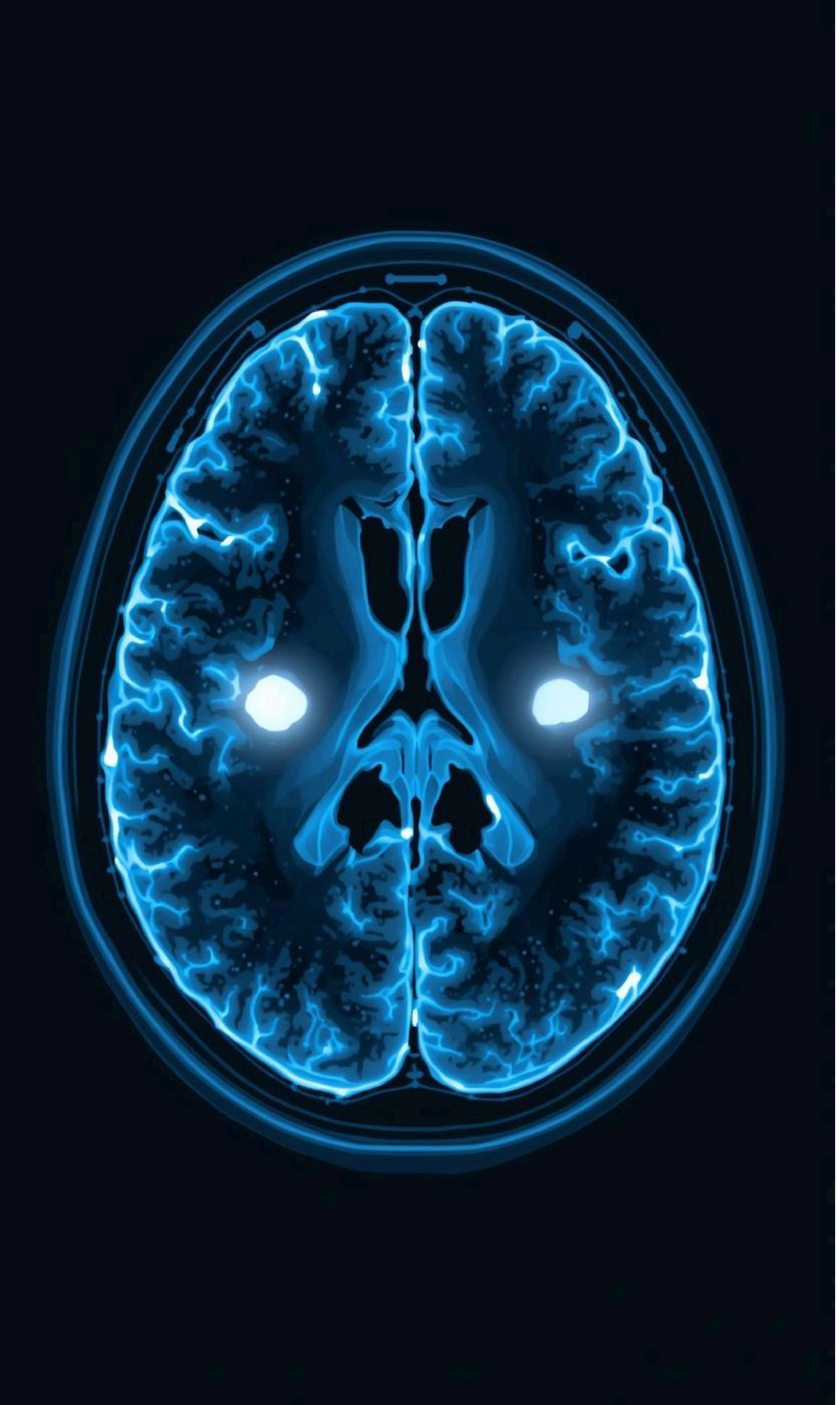
Noise Accuracy:

- σ=0.0 → 0.75
- σ=0.03 → 0.27
- σ=0.05 → 0.27
- σ=0.1 → 0.26

Conclusion:

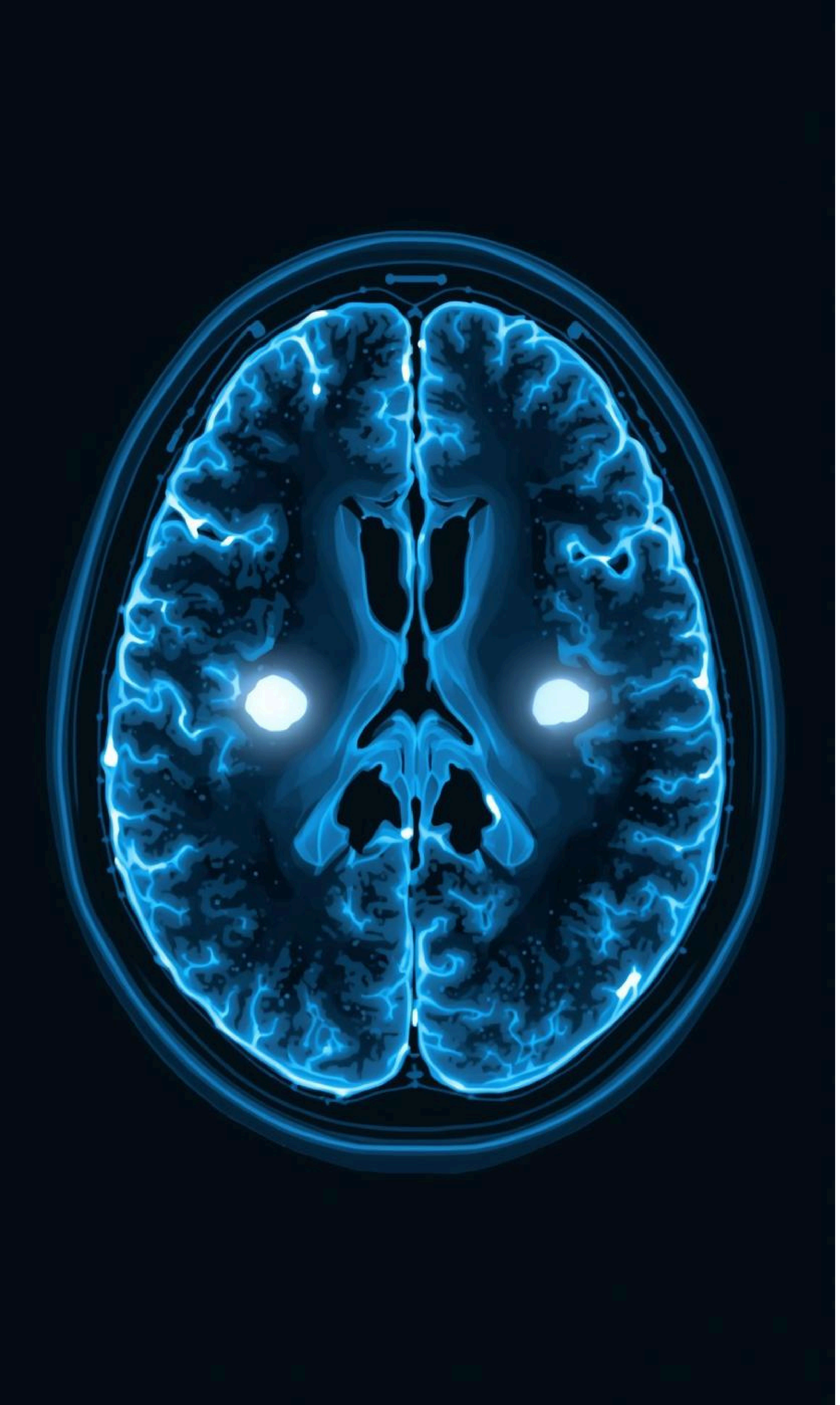
CNN is highly sensitive to low-resource conditions.





Current Progress

- Implemented CNN (MobileNetV2) baseline
- Extracted radiomics features (shape, intensity, texture)
- Built fusion classifier combining CNN + radiomics
- Added explainability via Grad-CAM
- Tested robustness under low-resource conditions



Next Steps

- Improve robustness under noise & low-resolution inputs
- Expand dataset variety (more scans, more variability)
- Add and refine more radiomics feature set
- Exploring model compression for CPU-friendly deployment
- Integrate everything into a cleaner virtual-biopsy pipeline

Thank You!