# **Brain Tumor Segmentation and Classification**

This repository contains three deep learning models implemented in Jupyter notebooks for segmenting brain tumors from MRI scans and classifying tumor grade. Each model includes built-in explainability using SHAP, LIME, Grad-CAM, and other techniques.

## **Models**

### **1. Standard U‑Net Segmentation (Mazen\_model1.ipynb)**

* **Purpose**: Pixel‑wise segmentation of tumor vs. healthy tissue on full MRI slices.
* **Core Features**:
  + Encoder–decoder with skip connections
  + Dice loss optimization
  + Contrast enhancement (optional)
* **Explainability**:
  + **Grad-CAM**: Visualize activation importance across regions.
  + **Vanilla Gradients**: Sensitivity maps showing which pixels most influence output.
  + **SHAP**: Pixel‑level contribution values for each segmentation decision.
  + **LIME**: Local perturbation‑based explanations for segmented regions.

### **2. Patch‑based Two‑Path CNN (Mazen\_model\_2.ipynb)**

* **Purpose**: Classify each patch’s center pixel as tumor or non‑tumor using local context.
* **Core Features**:
  + Dual convolutional streams merged in dense layers
  + On‑the‑fly patch extraction and balanced sampling
* **Explainability**:
  + **DeepDream**: Render patterns that maximize class activations.
  + **Integrated Gradients**: Attribute prediction scores back to input pixels.
  + **SHAP**: Quantify each patch pixel’s impact on the classification score.
  + **LIME**: Provide human‑readable explanations for individual patch decisions.

### **3. Glioma Grading CNNs (Mazen\_model3.ipynb)**

* **Purpose**: Separate CNNs for High‑Grade (HGG) and Low‑Grade (LGG) glioma patch classification.
* **Core Features**:
  + Multi‑layer CNN with two dense layers before softmax
  + Patch size: 33×33 pixels
* **Explainability**:
  + **Grad‑CAM**: Highlight patch regions critical to grade prediction.
  + **Saliency Maps**: Vanilla gradients showing key pixels.
  + **SHAP**: Feature contribution analysis for each patch input.
  + **LIME**: Local interpretable explanations of grading predictions.

## **Requirements**

* Python 3.8+
* TensorFlow 2.x
* NumPy, Pandas, Matplotlib
* OpenCV or Pillow

Install with:

pip install tensorflow numpy pandas matplotlib opencv-python pillow shap lime

## **Usage**

1. Clone this repository.
2. Organize your MRI images and masks:

/data/images/  
/data/masks/

1. Open the desired notebook in Jupyter or Colab.
2. Update file paths at the top of each notebook.
3. Run cells to preprocess data, train the model, and view explainability outputs.
   1. Visualization cells generate heatmaps and SHAP/LIME plots inline.

## **Results Summary**

|  |  |
| --- | --- |
| **Model** | **Metric** |
| U‑Net | Dice: ~0.80; IoU: ~0.75 |
| Patch‑based Two‑Path CNN | Accuracy: ~85% |
| HGG/LGG Classifiers | 88% / 82% Accuracy |

## **Directory Structure**

├── Mazen\_model1.ipynb # U‑Net segmentation  
├── Mazen\_model\_2.ipynb # Patch‑based CNN  
├── Mazen\_model3.ipynb # Glioma grading CNNs  
└── README.md # This file

## **Contact**

For questions or contributions, open an issue or contact the maintainer. Feel free to submit pull requests with improvements or additional explainability features.