Brain Tumor Segmentation – Open CV

**Project Overview**

The goal is to develop an automated system for segmenting brain tumors from MRI images. This involves:

* **Data Preprocessing**: Preparing MRI images and corresponding masks.
* **Model Development**: Implementing segmentation algorithms.
* **Evaluation**: Assessing model performance.
* **Deployment**: Creating a user-friendly interface for end-users.

**📁 1. Data Collection and Preparation**

**Dataset**: Utilize the provided dataset containing MRI images and their corresponding binary masks, where:

* **White pixels (255)** represent tumor regions.
* **Black pixels (0)** represent healthy tissue.

**Actions**:

* **Normalization**: Scale pixel intensity values to a range of 0 to 1.
* **Resizing**: Ensure all images and masks are resized to a consistent resolution (e.g., 256x256 pixels).

**🧪 2. Exploratory Data Analysis (EDA)**

**Objectives**:

* **Visualization**: Display sample MRI images alongside their masks to verify alignment and annotation quality.
* **Intensity Analysis**: Plot histograms to understand the distribution of pixel intensities in tumor vs. non-tumor regions.
* **Tumor Characteristics**: Analyze tumor sizes and shapes to identify variability and potential segmentation challenges.
* **Class Distribution**: Assess the ratio of tumor to non-tumor pixels to detect class imbalance.
* **Data Splitting**: Divide the dataset into training, validation, and test sets, ensuring no data leakage.
* **Augmentation Effects**: Apply transformations (e.g., rotation, flipping) to evaluate their impact on data diversity and model robustness.

**🧰 3. Segmentation Algorithm Development**

**Approach**:

* **Traditional Methods**:
  + **Thresholding**: Apply binary thresholding to separate tumor regions based on intensity values.
  + **Edge Detection**: Use algorithms like Canny edge detection to identify tumor boundaries.
  + **Contour Detection**: Extract contours to delineate tumor regions.
  + **Morphological Operations**: Use dilation and erosion to refine segmentation results.
* **Deep Learning Methods**:
  + **Model Architecture**: Implement a U-Net model, known for its effectiveness in biomedical image segmentation.
  + **Training**: Train the model using the preprocessed MRI images and masks.
  + **Validation**: Evaluate model performance on a separate validation set.

**📊 4. Model Evaluation**

**Metrics**:

* **Dice Coefficient**: Measures the overlap between the predicted and ground truth masks.
* **Intersection over Union (IoU)**: Evaluates the accuracy of the predicted segmentation.
* **Precision and Recall**: Assess the model's ability to correctly identify tumor regions.
* **Inference Time**: Measure the time taken to process a single MRI image.
* **Model Size**: Determine the storage requirements of the trained model.
* **Generalization**: Test the model on unseen data to evaluate its robustness.

**🌐 5. Deployment with Streamlit**

**Features**:

* **Image Upload**: Allow users to upload MRI images for segmentation.
* **Visualization**: Display the original image, ground truth mask, and the model's segmentation output.
* **Download Option**: Enable users to download the segmentation results for further analysis.[GitHub](https://github.com/Shravan103/Brain-Tumor-Detection?utm_source=chatgpt.com)

**📄 6. Documentation and Deliverables**

**Components**:

* **Trained Model**: Provide the final trained model file (e.g., .h5 or .pt).
* **Streamlit Application**: Deliver the complete Streamlit app code for deployment.
* **Jupyter Notebook**: Include a notebook detailing the entire workflow, from data preprocessing to model evaluation.
* **README File**: Offer comprehensive instructions on setting up and running the project.

**🔗 Additional Resources**

For further guidance and reference, consider exploring the following resources:

* **LearnOpenCV's 3D U-Net Tutorial**: Provides an in-depth guide on training a 3D U-Net model for brain tumor segmentation. [LearnOpenCV](https://learnopencv.com/3d-u-net-brats/?utm_source=chatgpt.com)[GitHub+4LearnOpenCV+4LearnOpenCV+4](https://learnopencv.com/tag/brain-tumor-segmentation-with-3d-u-net/?utm_source=chatgpt.com)
* **GitHub Repository - Brain Tumor Detection**: Offers code examples and methodologies for brain tumor detection using OpenCV and deep learning. [GitHub](https://github.com/harveyphm/brain-tumor-detection?utm_source=chatgpt.com)[LearnOpenCV+4GitHub+4GitHub+4](https://github.com/Shravan103/Brain-Tumor-Detection?utm_source=chatgpt.com)
* **GitHub Repository - Tumor Detection Using Image Segmentation**: Demonstrates the application of U-Net architecture for tumor detection in MRI images. [GitHub](https://github.com/pijush2022/Tumor-Detection-Using-Image-Segmentation?utm_source=chatgpt.com)

Bottom of Form