A logo with a design on it

Description automatically generated

**National University Of Modern Languages Islamabad**

**Group Members:**

**Mueez Ahmad**

**Faizan Ahmer**

**Muhammad Zubair Khan**

**Muhammad Ali Hassan Khan**

**Date: 31/05/2025**

**MRI Brain Tumor Detection Using Deep Learning**

**1. Introduction**

Brain tumors are among the most critical health issues that require early and accurate diagnosis. Magnetic Resonance Imaging (MRI) is a widely used imaging technique for detecting brain tumors. In this project, we build a deep learning model using PyTorch to classify MRI images into two categories:

* **Tumor**
* **No Tumor (Healthy)**

The goal is to automate the tumor detection process and assist radiologists by providing quick and reliable diagnoses.

**2. Dataset Overview**

The dataset used in this project is sourced from Kaggle:  
🔗 Brain MRI Images for Brain Tumor Detection

* The dataset consists of MRI images classified into two folders:
  + yes/ — Images that contain a brain tumor.
  + no/ — Images that do not contain a tumor.
* Images are resized to 128x128 pixels for consistency and processed using OpenCV.

**3. Libraries and Tools Used**

* PyTorch — Deep learning framework.
* OpenCV — Image processing.
* NumPy — Numerical operations.
* Matplotlib — Data visualization.
* sklearn — Metrics for evaluating model performance.

**4. Image Preprocessing**

* All images are resized to 128x128.
* Images are converted from BGR to RGB.
* Data is separated into two lists: tumor and healthy.
* Both categories are combined into a single dataset for training.

**5. Custom Dataset Class**

A custom PyTorch Dataset class is created to handle:

* Image loading
* Tensor conversion
* Label assignment (0 = healthy, 1 = tumor)

Data is split into:

* **Training Set**
* **Test Set**

**6. Model Architecture**

The model is a custom Convolutional Neural Network (CNN) with the following layers:

* **Convolutional Layers**: Extract features from the MRI images.
* **MaxPooling Layers**: Reduce spatial dimensions.
* **Fully Connected Layers**: Perform final classification.

Activation function used: ReLU  
Final layer activation: Sigmoid for binary classification.

**7. Training the Model**

* Loss function: Binary Cross Entropy (BCE)
* Optimizer: Adam
* Accuracy is calculated after every epoch.
* Training is performed over multiple epochs using mini-batch gradient descent.

**8. Evaluation**

* The model's performance is evaluated using:
  + Accuracy Score
  + Confusion Matrix

These metrics help in understanding the true positives, true negatives, false positives, and false negatives.

**9. Results and Visualization**

* Sample MRI images with predicted labels are visualized.
* Loss curve is plotted to show how the model improves over epochs.
* The confusion matrix shows a good separation between healthy and tumor cases.

**10. Conclusion**

This project successfully demonstrates the application of deep learning for brain tumor detection in MRI images. With further tuning and a larger dataset, the model can be enhanced for even better accuracy and clinical use.