BRAIN TUMOR DETECTION

Introduction

Title: *Brain Tumor Classification Using Deep Learning and Web Deployment*

Content:

- A web-based platform to classify brain MRI images into four tumor types.
- Upload image → Model processes it → Displays predicted tumor class.
- Developed using a custom CNN model and deployed via Flask.

Background of the Project

Problem:

- Early and accurate diagnosis of brain tumors is critical.
- Manual review of MRIs is time-consuming and error-prone.

Data & Labels:

- Dataset with MRI scans labeled as:
  - `glioma_tumor`, `meningioma_tumor`, `no_tumor`, `pituitary_tumor`

Model Architecture:

- Deep Convolutional Neural Network (CNN) with:
  - Conv2D, MaxPooling2D, Dropout layers.
  - Final layer: Dense(4, softmax) for multi-class prediction.

Training Process

Model Training Details:

- Input shape: 150×150×3
- Normalized image inputs (pixel values scaled to 0–1).
- Used dropout layers (0.3) to reduce overfitting.
- Final model trained and saved as `braintumor.h5`.

Model Summary:

- Total layers: 20+
- Loss Function: Categorical Crossentropy
- Optimizer: Adam
- Accuracy Achieved: ~95% (example)

Model Deployment & Findings

System Workflow:

1. User uploads MRI image via `upload.html`.
2. Flask routes image to `predictor.py`.
3. Model predicts tumor type and returns result.

`4.` `complete.html` shows uploaded image + result.

Findings:

- Model performs well on clearly segmented MRIs.
- High confidence when tumor features are distinct.
- `no_tumor` class predicted confidently when no anomaly is found.

Conclusion & Acknowledgement

Conclusion:

- Efficient and user-friendly tool for tumor classification.
- Reduces workload on radiologists.
- Can be integrated into hospital systems or mobile apps.

Acknowledgement:

- Thanks to mentors, dataset providers, and TensorFlow/Keras open-source community.