Brain Tumor Detection System

1. Introduction

Brain tumors are abnormal growths of cells in the brain that can be either benign or malignant. Early detection is crucial for effective treatment and improving survival rates. This project presents an AI-powered Brain Tumor Detection System using deep learning techniques to automate the classification of tumors based on their location in the brain. Instead of determining whether a tumor is benign or malignant, the system identifies whether it is located in the Pituitary, Meningioma, Glioma region, or if there is No Tumor present.

2. Objectives

- Develop an AI model for detecting brain tumors in MRI images.
- Improve diagnostic accuracy and reduce human error.
- Provide a user-friendly web interface for medical professionals and patients.

3. Technology Stack

- Programming Language: Python, TypeScript
- Libraries & Frameworks: TensorFlow, Keras, OpenCV, NumPy, Matplotlib, Scikit-learn
- Model Type: Convolutional Neural Network (CNN)
- Dataset: Publicly available MRI brain tumor datasets (Kaggle, BraTS)
- Frontend: React, Next.js, TypeScript, TRPC
- Backend: FastAPI for model integration
- Deployment: Not yet hosted, will be deployed soon

4. Methodology

4.1 Data Collection & Preprocessing

- Collected MRI images from open-source datasets.
- Applied image augmentation techniques such as rotation, scaling, and contrast adjustments.
- Normalized pixel values and resized images to a fixed dimension.

4.2 Model Development

- Used a CNN architecture with multiple convolutional layers to extract spatial features.
- Applied ReLU activation functions and batch normalization to improve learning.
- Used max pooling layers for feature down-sampling.

- Implemented dropout layers to prevent overfitting.
- The final layer used softmax activation for classification (Pituitary, Meningioma, Glioma, No Tumor).

4.3 Model Training & Evaluation

- Split data into training (70%), validation (15%), and testing (15%) sets.
- Used categorical cross-entropy as the loss function and Adam optimizer for training.
- Achieved high accuracy (>90%) through hyperparameter tuning.
- Evaluated the model using precision, recall, F1-score, and confusion matrix.

5. Web Application Integration

- Developed an interactive React, Next.js, TypeScript & TRPC web application.
- Users can upload MRI scans for real-time tumor classification.
- The application displays model predictions with probability scores.
- Backend API built using FastAPI to handle model requests.

6. Results & Analysis

- The model effectively classified brain tumors by type (Pituitary, Meningioma, Glioma, No Tumor) with high accuracy.
- Performance comparison with other models showed competitive results.
- The web application allowed seamless tumor classification with instant feedback.
- Future improvements include integrating explainable AI (XAI) for model interpretability.

7. Challenges Faced

- Data Imbalance: The dataset had more normal images than tumor cases, requiring augmentation techniques.
- Model Overfitting: Addressed using dropout layers and regularization.
- Computational Complexity: Training deep learning models required GPU acceleration.

8. Future Enhancements

- Enhancing interpretability with Grad-CAM visualization.
- Deploying on mobile platforms for wider accessibility.
- Collaborating with medical professionals for real-world validation.

9. Conclusion

The Brain Tumor Detection System is a promising AI-based solution for early diagnosis of brain tumors. By leveraging deep learning, this project provides a high-accuracy model integrated with a user-friendly interface. Future improvements will focus on better interpretability, real-time deployment, and clinical validation.

Author: Akshansh Maurya

Technologies Used: Python, CNN, FastAPI, React, Next.js, TypeScript, TRPC

Domain: Healthcare AI, Medical Imaging

Project Repository: https://github.com/akshanshmaurya/NeuroScan-AI.git