

Department: Computer Science and Engineering

PROJECT

# PROJECT TITLE: BRAIN TUMOR CLASSIFICATION USING CNN

**Presented by**

K. Suhitha 22B01A0576

M. Chaturya Sri Vasavi 22B01A05B5

N. Hima Rupa Sri 23B05A0513

K. Sai Rishitha 22B01A0587

# ABSTRACT:

This project focuses on the classification of brain tumors using Magnetic Resonance Imaging (MRI) scans with the help of Convolutional Neural Networks (CNNs), a powerful deep learning technique. Brain tumors are life-threatening conditions, and early detection is crucial for effective treatment. Traditional manual analysis of MRI images is time-consuming and prone to human error, making automated solutions highly desirable. Our model is trained on a publicly available Kaggle dataset containing MRI images classified into four categories: glioma, meningioma, pituitary tumor, and no tumor. The images are preprocessed through normalization, resizing, and augmentation techniques to improve the model's generalization and performance.

A deep CNN architecture was developed to learn spatial features from the input images and accurately classify the tumor type. After training and validation, the model achieved high accuracy and robustness in recognizing different tumor categories. To make the model accessible to non-technical users, a user-friendly web application was built using Flask. This interface allows users to upload an MRI image and receive real-time predictions regarding the type of brain tumor. This project demonstrates the potential of deep learning in the medical domain by offering an efficient, automated, and scalable solution for brain tumor classification.

# INTRODUCTION:

Brain tumors are abnormal growths of tissue in the brain that can severely disrupt neurological functions and, if undetected, lead to life-threatening conditions. Accurate and timely diagnosis is essential for effective treatment, but traditional diagnostic methods such as manual examination of MRI scans are often time-consuming, subjective, and require significant medical expertise. These limitations can lead to inconsistencies in diagnosis and delayed treatments, which may affect patient outcomes. With the increasing volume of medical imaging data, there is a growing need for automated and reliable diagnostic tools that can support radiologists and clinicians.

Recent advancements in artificial intelligence, especially in the field of deep learning, have shown promising results in medical image analysis. Convolutional Neural Networks (CNNs), a type of deep learning architecture specifically designed for image data, have proven highly effective for tasks such as classification, detection, and segmentation. This project utilizes CNNs to classify brain tumors from MRI images into four categories: glioma, meningioma, pituitary tumor, and no tumor. By combining deep learning techniques with a user-friendly web interface, the system provides a fast, accessible, and accurate method to assist in the early detection and classification of brain tumors.

# RELATED WORK:

## Challenges

 **Limited and Imbalanced Data**: The dataset contained a limited number of labeled MRI images for certain tumor types, which posed a challenge for training a well-generalized model without overfitting.

 **Variability in MRI Image Quality**: Differences in image resolution, orientation, and noise levels across the dataset made it difficult for the model to consistently extract meaningful features, requiring extensive preprocessing and augmentation.

## Existing Solutions

1. **Sajjad et al. (2019)** developed a deep CNN model for brain tumor classification but encountered issues with overfitting due to a limited dataset, highlighting the need for data augmentation and regularization techniques.
2. **Mohsen et al. (2018)** proposed a hybrid model combining CNN and SVM, which improved classification accuracy but was computationally expensive and less practical for real-time applications.
3. **Chakrabarty et al. (2020)** used a U-Net architecture for segmenting tumor regions before classification. While effective, it required detailed annotations that are often unavailable in public datasets.
4. **Islam et al. (2020)** implemented transfer learning using VGG16 for classifying MRI images. Although it achieved good accuracy, it demanded extensive fine-tuning due to the grayscale nature of medical images.

# PROBLEM STATEMENT:

Manual diagnosis of brain tumors using MRI scans is time-consuming, prone to human error, and requires specialized expertise. In many cases, early detection is delayed due to inconsistencies in interpretation, leading to ineffective treatment and poor patient outcomes. There is a pressing need for an automated, accurate, and scalable solution that can classify brain tumors efficiently using medical imaging data. This project aims to address this gap by developing a Convolutional Neural Network (CNN)-based system that can accurately classify brain tumors into categories such as glioma, meningioma, pituitary tumor, and no tumor using MRI images. The solution also includes a web application to make the tool easily accessible for real-world use.

# METHODOLOGY:

### 1. Requirement Analysis

In this phase, we gathered both functional and non-functional requirements necessary for developing an automated brain tumor classification system. The primary stakeholders include medical professionals, patients, and system administrators. Key requirements identified include:

* A simple interface for users to upload MRI brain scan images.
* Secure backend that handles image processing and classification.
* Deployment of a trained CNN model to predict tumor type (glioma, meningioma, pituitary tumor, or no tumor).
* Real-time display of classification results to the user.
* System for logging and managing uploaded images and predictions.
* Responsive web interface accessible across devices.
* Backend support for efficient storage and retrieval of image data and results.

### 2. System Design

System design was divided into three main components: frontend, backend, and model integration.

* **Frontend Design**: Built using HTML, CSS, and JavaScript to create a clean and interactive interface where users can upload MRI images and view results.
* **Backend Design**: Implemented using Python and Flask. The backend handles image validation, preprocessing, and calls the trained CNN model to perform inference. It also returns the classification result to the user interface.
* **Architecture**: The application follows a modular structure:
  + **Presentation Layer**: HTML/CSS/JS for user interaction.
  + **Business Logic Layer**: Flask for handling requests and integrating the trained CNN model.
  + **Data Layer**: Images and results stored using file system or lightweight database (SQLite/PostgreSQL) if extended.

### 3. Model and Data Design

The model and data design follow a structured pipeline that ensures robust training and accurate predictions.

* **Dataset**: The model was trained on the "Brain Tumor MRI Dataset" from Kaggle, containing labeled MRI images of four classes: glioma, meningioma, pituitary tumor, and no tumor.
* **Preprocessing**: Images were resized (128x128), normalized, and augmented using techniques like rotation, zoom, and flips to improve generalization.
* **Model Design**: A CNN architecture consisting of convolutional layers, ReLU activations, max pooling, dropout, and dense layers ending in a softmax classifier.
* **Training and Storage**: Model was trained using Keras and TensorFlow, with the final model saved for inference in the deployed Flask app.

### 4. Data Flow and Integration

The interaction between the frontend, backend, and model follows this flow:

* The user uploads an MRI image through the web interface.
* The backend receives the image via a POST request and validates the input.
* The image is preprocessed and passed into the trained CNN model for prediction.
* The predicted tumor class is returned and displayed on the frontend.
* Each request is logged for optional analysis or record-keeping.

**Example Flow:**

1. User opens the app → uploads MRI image → backend processes the image.
2. Model returns result (e.g., “Glioma Tumor”) → displayed to user in the browser.

### 5. Security and Validation

Security and validation are essential to ensure the robustness and reliability of the system:

* **Client-side Validation**: Checks for file format (e.g., only .jpg, .png allowed).
* **Server-side Validation**: Ensures images are processed only if they meet size and format standards.
* **Model Safety**: Image input is normalized and reshaped securely before feeding into the model.
* **Session Management**: If login-based extension is used, Flask session management can handle user roles.
* **Error Handling**: Custom messages are returned for incorrect file types, missing input, or server errors.
* **Protection**: Routes and backend API are secured to prevent injection or misuse.

### 6. Results and Module-wise Functionality

**Module 1: Home Page**

* **Purpose**: Acts as the entry point of the web application.
* **Features**:
  + User-friendly design
  + Introduction to the platform’s purpose (Brain Tumor Detection)
  + Direct navigation to the image upload module

**Module 2: Image Upload**

* **Purpose**: Allows users to upload MRI brain images for classification.
* **Features**:
  + Upload form accepting JPEG/PNG formats
  + Instructions for optimal image quality
  + Immediate redirection to result page upon submission

**Module 3: Prediction Results**

* **Purpose**: Displays the tumor classification predicted by the model.
* **Features**:
  + Result (e.g., “Meningioma Tumor”)
  + Option to upload another image
  + Basic explanation or link about the identified tumor type

**CONCLUSION:**

This project successfully demonstrates the power of Convolutional Neural Networks (CNNs) in accurately classifying brain tumors from MRI images. By automating the diagnostic process, the system reduces the reliance on manual interpretation by radiologists, which can often be time-consuming and inconsistent. The use of image preprocessing and data augmentation techniques has significantly improved the model’s ability to generalize and perform well on unseen images, making it a reliable tool for medical image analysis.

The integration of a web-based application further enhances the usability and accessibility of the system. Users can easily upload an MRI image through a simple interface and receive real-time predictions about the tumor type. This feature makes the tool practical for clinical settings, educational purposes, or remote diagnosis, especially in areas with limited access to expert radiologists. The system is designed with a focus on accuracy, performance, and user experience.

Looking forward, the project can be extended to include features like tumor segmentation, severity grading, and integration with electronic health records. Incorporating more diverse and larger datasets can also improve accuracy and robustness. With further improvements and deployment in real-world environments, this system has the potential to become a valuable decision-support tool in the early detection and treatment of brain tumors.

Link : https://github.com/HimaRupasri/Brain-Tumor-Detection-and-Classification