

Project Summary: Alzheimer's and Brain Tumor Detection Using Deep Learning.

Define the Problem:

"How can we automatically detect brain tumors and stages of Alzheimer's disease from MRI scans to assist doctors in making faster and more accurate diagnoses?"

Break It Down

☐ Real-World Scenario:

- Doctors examine **MRI scans** manually to look for:
 - Brain tumors (like glioma, meningioma, pituitary)
 - Early or late stages of Alzheimer's (mild to severe memory loss)
- Manual review is:
 - Time-consuming
 - Prone to human error
 - Requires high expertise

Project Objective:

We aim to build an automated image recognition system using deep learning (CNN) that can classify brain tumors and Alzheimer's stages from MRI scans. The model will be trained using publicly available labeled datasets and integrated into a Flask-based web application, allowing doctors or users to upload scans and receive instant predictions. This system is designed to reduce diagnostic delays, improve accuracy, and assist healthcare professionals, especially in resource-constrained settings.

Goal of Your Project:

Build a **deep learning model** that can:

- **Take an MRI image** as input
- **Predict** if there is a tumor or Alzheimer's stage
- **Display the result** on a web interface for doctors

Datasets Used

- **Brain Tumor Dataset** (from Kaggle): MRI images labeled as Tumor/No Tumor and tumor types.
- **Alzheimer's Dataset** (from Kaggle/ADNI): Labeled images for stages like Very Mild, Mild, Moderate, No Dementia.

Preprocess the Data: Preprocessing is essential for preparing MRI images so that your deep learning model can learn meaningful patterns.

Prepare your brain tumor and Alzheimer's MRI images for training by:

- Resizing
- Normalizing

- Augmenting

Build and Train the CNN Model (In a Practical Way)

Now that your MRI images are preprocessed, it's time to **build a Convolutional Neural Network (CNN)** and train it to detect **Brain Tumors** or **Alzheimer's Disease Stages**. Use **Keras + TensorFlow** to define a CNN model

Web App Using Flask: Upload an MRI scan, predict the result using your trained CNN, and display it on a web page.

Tools You'll Use:

- **Flask** – Web framework (Python)
- **HTML/CSS** – Frontend
- **TensorFlow/Keras** – Load trained model (brain_model.h5)
- **PIL / OpenCV** – For image handling

Tool	Purpose
OpenCV	To access webcam and capture
Flask	Backend routing
Keras	Load and use saved model

Integrate Webcam :

Want real-time MRI image capture via webcam? This step adds camera integration for live image prediction.

Allow users to:

- **Capture an MRI scan** (from printed reports or screens) using a **webcam**
- Predict the disease using the trained CNN model
- Display results in real time

Result :

Accuracy achieved:

- Brain Tumor: ~92%
- Alzheimer's Stages: ~91%