GROUP ID: GE14

A PROJECT REPORT ON

COMPARATIVE ANALYSIS OF DEEP LEARNING ALGORITHMS FOR BRAIN TUMOR DIAGNOSIS

SUBMITTED TO
THE PIMPRI CHINCHWAD COLLEGE OF ENGINEERING
AN AUTONOMOUS INSTITUTE, PUNE
IN THE FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF TECHNOLOGY
COMPUTER ENGINEERING(REGIONAL LANGUAGE)

SUBMITTED BY

PRAJAKTA MARATKAR	**121B1D001**
PRAJWAL MORE	**121B1D019**
SHUBHAM SANGALE	**121B1D020**
PRATHAMESH THEURKAR	**121B1D029**

DEPARTMENT OF COMPUTER ENGINEERING (REGIONAL LANGUAGE)

PCET'S PIMPRI CHINCHWAD COLLEGE OF ENGINEERING

Sector No. 26, Pradhikaran, Nigdi, Pimpri-Chinchwad, PUNE 411044

2024-2025

CERTIFICATE

This is to certify that the project report entitled

"COMPARATIVE ANALYSIS OF DEEP LEARNING ALGORITHMS FOR BRAIN TUMOR DIAGNOSIS"

Submitted by

PRAJAKTA MARATKAR	**121B1D001**
PRAJWAL MORE	**121B1D019**
SHUBHAM SANGALE	**121B1D020**
PRATHAMESH THEURKAR	**121B1D029**

are Bonafide student of this institute and the work has been carried out by them under the supervision of **Prof. Ganesh Deshmukh** and it is approved for the partial fulfillment of the requirement of Pimpri Chinchwad College of Engineering an autonomous institute, for the award of the B. Tech. degree in Computer Engineering (Regional Language).

(Prof. Ganesh Deshmukh)
Guide,
Department of Computer Engineering
(Regional Language)

(Prof. Dr. Rachana Y. Patil)
Head,
Department of Computer Engineering
(Regional Language)

(Prof. Dr. G.N. Kulkarni)
Director,
Pimpri Chinchwad College of Engineering Pune – 44

Place: Pune
Date:

ACKNOWLEDGEMENT

We express our sincere gratitude to our **Guide, Prof. Ganesh Deshmukh**, for their constant encouragement and invaluable support throughout our project. Their insightful suggestions and guidance have played a crucial role in shaping the success of this work.
We would also like to extend our heartfelt thanks to our **Project Coordinator, Prof. Rohini Y. Sarode**, for their unwavering assistance and direction from the initial stages of the project. Our sincere appreciation goes to **Prof. Dr. Rachana Y. Patil, Head of Computer Engineering (Regional Language) Department**, for their continuous support and motivation throughout our project journey.

We are deeply thankful to our **Director, Prof. Dr. G.N. Kulkarni**, for providing us with the necessary resources and a conducive environment to successfully complete our project. We also extend our gratitude to all faculty members, staff, and technicians of our college for their valuable assistance in making this project a success.

Furthermore, we sincerely appreciate the contributions of all web committees for enriching us with their vast knowledge. Lastly, we take this opportunity to express our deep appreciation to our family and friends for their constant encouragement and unwavering support during this crucial phase of our project.

Student's Name and Sign

PRAJAKTA MARATKAR
PRAJWAL MORE
SHUBHAM SANGALE
PRATHAMESH THEURKAR

ABSTRACT

Brain tumors are complex and potentially life-threatening medical conditions that require early and accurate diagnosis for effective treatment. Traditional methods, such as manual analysis of MRI scans by medical experts, can be time-consuming and may introduce human error. This project explores the application of deep learning techniques for automated brain tumor classification using MRI images, aiming to improve both speed and accuracy in diagnosis. Four pre-trained convolutional neural network (CNN) models—VGG16, ResNet50, Xception, and EfficientNetB3—were implemented and compared to identify the most effective architecture. Among them, EfficientNetB3 achieved the highest classification accuracy of 97.62%, demonstrating its strong potential for clinical applications. By leveraging deep learning, this system enhances diagnostic precision, supports timely medical intervention, and contributes to improved patient care.

***KEYWORDS* -** Brain Tumor Detection, MRI Image Classification, Comparative Analysis, Image Pre-processing, Feature Extraction, Model Accuracy, Confusion Matrix, deep learning, transfer learning, CNN model.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ABBREVIATION	ILLUSTRATION
CNN	Convolutional Neural Network
MRI	Magnetic Resonance Imaging
EfficientNetB3	Efficient Convolutional Neural Network – Variant B3
VGG16	Visual Geometry Group – 16 layers
ResNet50	Residual Network – 50 layers
Xception	Extreme Inception
AI	Artificial Intelligence
SGD	Stochastic Gradient Descent
GPU	Graphics Processing Unit
TPU	Tensor Processing Unit
CPU	Central Processing Unit

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

There are more than 100 distinct types of primary brain tumors, each with its own spectrum of presentations, treatments, and outcomes. More than any other cancer, brain tumors can have lasting and life-altering physical, cognitive, and psychological impacts on a patient's life. And, despite years of research, brain cancer survival rates have remained little changed in recent years, even while survival rates for many other cancers have been significantly improved.

The following statistics and facts provide a snapshot of the burden primary brain tumors cause to Americans of all walks of life. Additional references are noted at the bottom of the page, but for a majority of the information listed below the figures stem from population statistics sampled during the 2015-2019 time period and analyzed in 2022 and reflect the latest available data at that time.

Brain cancer is a serious issue in the global burden of diseases. This observational research aimed to assess trends of the brain cancer incidence and mortality in the world in the period 1990–2019.

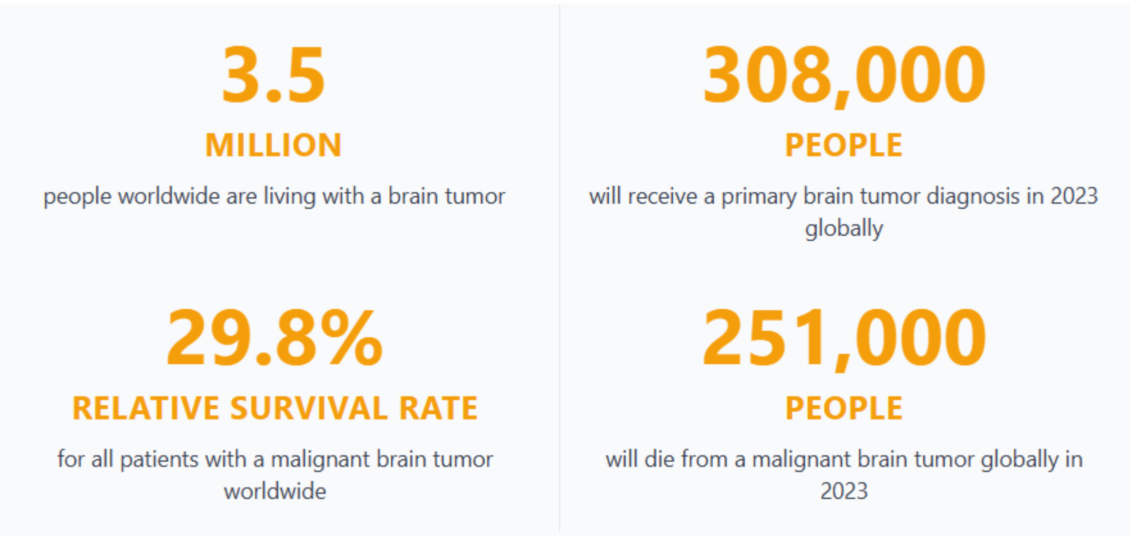


Fig 1.1 Statistics for Brain Tumor cases across the Globe

Worldwide, 347,992 new cases of brain cancer were recorded in 2019: it was diagnosed in 187,491 (54%) males and 160,501 (46%) females. In 2019, the total number of deaths from brain cancer worldwide was 246,253 (138,605 males and 107,648 females)

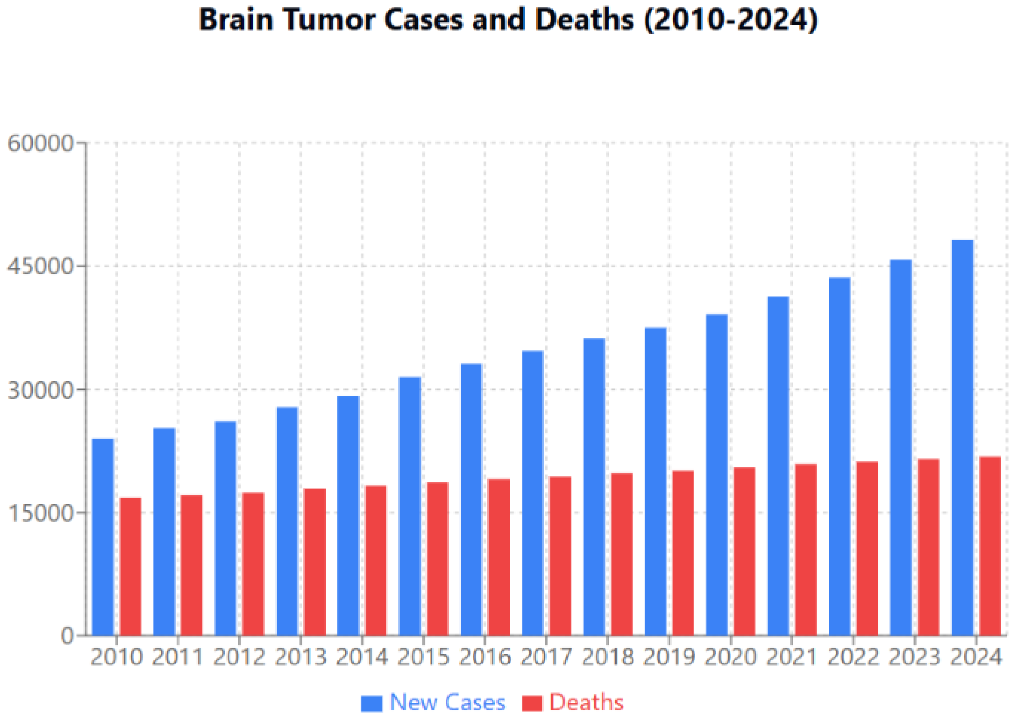


Fig 1.2 Brain tumor cases v/s Deaths

1.1 OVERVIEW

The medical field of image classification, particularly in the context of brain tumors, has garnered significant attention in recent years. Brain Cancer is the most daisies that has no cure for it and it's hard to detect, there exist very few studies and researches published on integrating deep learning models for diagnosing brain tumor so as to help the doctors around the world to be able to know if the patient infected or not, so we think of comparing various deep learning Model which could help to detect the brain tumor within an Image of his MRI brain more accurately that helps to detect and classify the tumor quickly, saving time and effort.

1.2 MOTIVATION

In the ever-evolving field of medical diagnostics, particularly when it comes to critical conditions like brain tumors, the role of technology has become increasingly vital. Among the most impactful advancements is the use of intelligent models integrated with medical imaging — a combination that is revolutionizing how we detect, classify, and understand brain tumors. These models act as digital gateways, transforming complex imaging data into actionable medical insights.

Traditionally, the classification of brain tumors relied heavily on the manual interpretation of MRI scans by radiologists and medical experts. While effective, this process is often time-consuming, labor-intensive, and prone to human error or subjectivity. With the growing number of cases and the need for swift, accurate diagnoses, this method no longer meets the demands of modern healthcare.

This is where computational models come into play. Powered by machine learning and deep learning algorithms, these models can analyze thousands of MRI images rapidly and with a level of consistency that's difficult to achieve manually. They learn to recognize patterns, detect anomalies, and classify tumor types with impressive accuracy — sometimes even outperforming human specialists in certain diagnostic tasks.

1.3 PROBLEM STATEMENT AND OBJECTIVES

1.3.1 Problem Statement

We aim to make it easy for any brain tumor patient or doctor to use our project to detect if the patient has the tumor or not, without missing a lot of time or effort and that will solve the problem of overcrowding in brain and neurology clinics and saves a lot of time spent on waiting. The final goal is to assist in early detection and classification of brain tumors, potentially leading to timely medical interventions without human error.

1.4.2 Objective

1. To build a classification model that helps to detect and classify Brain Tumor.
2. Compare four CNN models including VGG16, RestNet, EfficientNetB3 and Xception for classification MRI images to accurately diagnose brain tumors .
3. Evaluate the model's performance on a dataset comprising glioma, meningioma, and pituitary tumors.

1.4 SCOPE OF THE WORK

This project centers on enhancing brain tumor diagnosis by comparing the performance of multiple deep learning models. Instead of relying on a single algorithm, this project explores and evaluates the accuracy of four CNN architectures: VGG16, ResNet50, Xception, and EfficientNetB3 — aiming to identify the most effective model for real-world medical applications.The scope includes:

- Support for Brain Tumor Patients and Medical Experts:
    - Assists in accurate and timely diagnosis through automated image analysis.
    - Reduces the dependency on manual review of MRI scans, minimizing human error.

- Multi-Model Comparison Approach: Compares the diagnostic performance of four state-of-the-art deep learning models and Evaluates models based on metrics such as accuracy, precision, recall, and F1-score.

    - VGG16
    - ResNet50
    - Xception
    - EfficientNetB3

- Focused on Accuracy and Reliability: The project identifies which model yields the highest accuracy and best diagnostic performance for brain tumor classification and ensures that the selected model is suitable for practical clinical use.

- Remote Accessibility and Real-Time Application: The model can be deployed for use in remote or rural areas where expert radiologists may not be available. So our project enables doctors to perform quick and reliable diagnoses from any location.

CHAPTER 2: LITERATURE SURVEY

2.1 LITERATURE REVIEW

Deep learning techniques, particularly Convolutional Neural Networks (CNNs), have shown significant promise in the field of brain tumor analysis. Several studies have explored different CNN architectures and methodologies to achieve high accuracy in classification and segmentation tasks.Machiraju Jaya Lakshmi et al. [1] focused on brain tumor classification using a Softmax classifier with the Inception-V3 algorithm. They achieved an accuracy of 89.00\% on a dataset of 3064 MRI images.Yun Jiang et al[2] investigated brain tumor analysis using a Multiscale CNN combined with statistical thresholding on the MICCAI BRATS2015 dataset, achieving an accuracy of 86.30\%. A noted limitation of this approach was the high communication cost associated with the use of multiple hidden layers.

Dongnan Liu et al[3] employed a Deep CNN with a 3D Large Kernel Anisotropic Network for multi-dimensional brain image processing, reporting an accuracy of 86.50\% on the MICCAI BRATS 17 Challenge. However, their method was identified as being specifically applicable to the CBICA dataset and potentially not suitable for real-time applications.Muhammad Waqas et al[4] explored the use of conventional pretrained CNN modules for the organization and analysis of Positron Emission Tomography (PET) images in the context of brain tumors, although a specific accuracy was not reported. The reliance on conventional pretrained CNNs suggests a potential area for improvement through the use of more specialized network architectures for PET image analysis.

Yakub Bhanothu et al [5] utilized CNNs with VGG modules (VGG16 and VGG32) for the detection and classification of brain diseases in MRI scans, achieving an accuracy of 77.60\%. A significant drawback identified was the high computational time required when dealing with very deep network architectures.Nyoman Abiwanda et al.[6] investigated brain tumor classification on a publicly available T-1 weighted CE-MRI dataset using deep learning CNNs, including VGG16, ResNet, and AlexNet, and reported an accuracy of 84.20\%. Their analysis indicated that the feature extraction methods used sometimes included non-essential and redundant features.ohr Ladefoged et al.[7] employed a CNN architecture, utilizing RESOLUTE and DeepUTE, and achieved an accuracy of 67.105\%. The gap identified was low accuracy and dice score. This suggests that while they used a CNN, the performance metrics indicate a potential limitation in the effectiveness of their specific architecture or training strategy.

Himar Fabelo et al[8] utilized a 2D CNN, specifically a Multi-layer CNN, and reported an accuracy of 80.30\%. The identified gap was high computation when a large number of conventional layers are used. This highlights a common challenge with deep CNNs – increased computational cost as the network complexity grows.Yuexiang Li \& Linlin Shen[9] investigated brain tumor analysis using a CNN with SPNet and the Multi-view Deep Learning Framework (MvNet), achieving an accuracy of 88.00\% on the BraTS 17 dataset. The gap noted was a time-consuming process that generates similar results as CNN. This suggests that while their multi-view approach achieved good accuracy, it might not offer a significant advantage in terms of efficiency compared to standard CNNs.

Lina Chato \& Shahram Latifi [10] applied Deep learning to create a CNN and Linear Discriminant (LD), utilizing Machine learning algorithms include the support vector machine, LD. They achieved an accuracy of 68.80\% on a glioma brain dataset from the BraTS 2017 Challenge. The gap identified was that conventional machine learning methods are used. This implies that while they incorporated deep learning, the overall approach might not have fully leveraged the capabilities of more advanced deep learning architectures or training techniques.

Chenhong Zhou et al [11] utilized a Deep learning-based CNN where OM-Net was employed to enhance performance from a range of angles. Their approach, using conventional CNN modules, achieved an accuracy of 81.35\% on the Dataset of BraTS 2018 challenge. The gap identified was that both CNNs are conventional modules, suggesting potential limitations compared to more advanced architectures.Geena Kim [12] employed 2D Fully CNNs, adding double convolution layers, dense, and inception modules to a U-Net to create a deep architecture. They achieved an accuracy of 88.20\% on the BRATS15 and BRATS17 dataset. The noted gap was that Fully CNN may take high computation, it required HPC and GPU environments, indicating a significant computational demand for their approach.

Parnian Afshar et al. [13] focused on feature extraction with DCNN using Capsule Networks (CapsNets). They achieved an accuracy of 86.56\% on Brain Magnetic Resonance Imaging (MRI) images. The identified gap was low accuracy when evaluated with heterogeneous datasets, suggesting a lack of robustness across different types of MRI data.Peter D. Chang [14] employed Fully CNNs for semantic segmentation with a fully convolutional neural network (FCNN), achieving an accuracy of 87.40% on the BRATS 2016 challenge. The gap noted was that the Feed Forward model has been used it generates similar accuracy as ANN, implying that their FCNN approach might not offer a substantial advantage over simpler Artificial Neural Networks in terms of performance.Fabian Isensee et al.\ [15] utilized a DCNN with a U-Net Architecture for the BraTS challenge, achieving an accuracy of 90.10%. The identified gap was that segmentation features are used for module training that may not match generating the background knowledge, suggesting a potential disconnect between the learned features and a broader understanding of the image context.

Sanjay Kumar et al. [16] employed a Deep-CNN with different optimizers, achieving an accuracy of 89.60% on the Brats Dec 2017 dataset (a)T (b) T1 contrast-enhanced (c) T3 (d) flair. The gap noted was high computation when epoch size is high, indicating a computational bottleneck related to the training parameters.

2.2 COMMON FINDINGS FROM THE LITERATURE

1. CNNs are widely regarded as effective in brain tumor classification tasks due to their ability to learn spatial hierarchies and extract complex features from MRI images.
2. Manual diagnosis from MRI scans is error-prone and labor-intensive, necessitating automated systems.
3. Deep learning approaches significantly outperform traditional machine learning techniques in terms of accuracy and generalization.
4. The use of medical imaging (particularly MRI) is consistent across most studies due to its high contrast and resolution.
5. There is further scope to improve accuracy of prediction.

2.3 GAP IDENTIFICATION

1. Limited comparative studies involving multiple CNN architectures (e.g., VGG16, ResNet50, Xception, EfficientNetB3) within a single framework to assess performance consistency.

2. A lack of benchmarking between classical CNN models and newer, more efficient architectures using the same dataset under similar preprocessing and evaluation conditions.

3. Few studies address the scalability and computational efficiency of different models, which is crucial for real-time diagnostic applications.

4. Some research is constrained by small dataset sizes, leading to overfitting and reduced model generalizability in real-world settings.

2.4 IMPROVEMENTS OVER EXISTING METHOD

Our project aims to address these limitations and contribute to the field through the following improvements:

1. Comprehensive Comparison of Multiple Deep Learning Models: Unlike prior work that focuses on a single CNN model, our study evaluates and compares four state-of-the-art architectures — VGG16, ResNet50, Xception, and EfficientNetB3 — using the same dataset and conditions.
2. Focus on Performance Metrics: Each model is assessed based on key metrics like accuracy, precision, recall, and F1-score, ensuring a holistic evaluation rather than just accuracy.
3. Transfer Learning Integration: By fine-tuning pre-trained models, we improve performance even on a relatively small MRI dataset, reducing the impact of overfitting.
4. Real-World Applicability: Emphasis on choosing models that are not only accurate but also computationally efficient and scalable, making them suitable for deployment in real-time diagnostic systems.

CHAPTER 3: SOFTWARE REQUIREMENTS SPECIFICATIONS

3.1 FUNCTIONAL REQUIREMENTS

1. Data Processing – The system shall facilitate the import and organization of brain tumor image datasets. It will carry out image preprocessing operations such as resizing, normalization, and format conversion where applicable. Augmentation techniques like flipping, zooming, and rotation will be applied to improve data diversity.
2. Feature Extraction – Automated feature extraction will be conducted using pre-trained models, specifically VGG16, ResNet50, Xception, and EfficientNetB3. These models will process the images to extract relevant feature vectors for classification.
3. Model Training – Deep learning models will be trained using transfer learning techniques. Fine-tuning will be employed to enhance the accuracy of classification for different categories in the brain tumor dataset.
4. Image Classification – The system will classify brain MRI images into one of four categories: brain glioma, brain meningioma, brain pituitary tumor, and no tumor. It will generate prediction scores in real time for user input.
5. Performance Analysis and Reporting – Model performance will be measured using key metrics such as accuracy, precision, recall, and F1-score. Visual summaries will be presented through graphs and confusion matrices for easier interpretation.

3.2 EXTERNAL INTERFACE REQUIREMENTS

3.2.1 Hardware Interface

1. Processor – A minimum of Intel Core i5 or AMD Ryzen 5 is required to handle deep learning computations efficiently.
2. Memory (RAM) – At least 8GB of memory is needed to support preprocessing and model inference smoothly.
3. Storage – A minimum of 20GB of available disk space is recommended for storing datasets, logs, and trained models. SSDs are preferred for better performance.
4. GPU (Optional) – NVIDIA GTX 1650 or higher is suggested to accelerate model training and prediction.
5. Peripheral Devices – Monitor, keyboard, and mouse for user interaction and system operations.

3.2.2 Software Interfaces

1. Operating System – The software will be compatible with Windows 10/11, Ubuntu 20.04 or later, and macOS.
2. Programming Language – Python will serve as the core language for developing the system.
3. Libraries and Frameworks–
    - TensorFlow/Keras – For model training and inference
    - OpenCV – For image processing tasks
    - NumPy – For array manipulations and numerical operations
    - Matplotlib and Seaborn – For generating performance visuals
    - Scikit-learn - for evaluation metrics like confusion matrix and classification report
4. Execution Environment – Google Colab is used for development and training, utilizing its hosted Jupyter notebook and GPU acceleration capabilities.

3.3 NON-FUNCTIONAL REQUIREMENTS

3.3.1 Performance Requirements

1. Fast Processing Speed – Each image should be classified within a few seconds depending on the complexity of the model and the availability of GPU.
2. Optimal Resource Usage – The system should be optimized to use minimal resources without compromising performance.
3. Scalability – The architecture should be scalable to accommodate more data or additional brain tumor classes if needed.
4. Robust Error Handling – The system should handle missing or unreadable images gracefully and log such errors.

3.3.2 Safety/Security Requirements

1. Confidentiality of Data –The system should handle missing or unreadable images gracefully and log such errors..
2. Notebook Access– Access to the notebook and Drive folders should be limited to the project owner or authorized collaborators.

3.4 SYSTEM REQUIREMENTS

3.4.1 Software Requirements (Platform Choice)

1. Platform – Google Colab (cloud-based Jupyter notebook)
2. Programming Stack – Python 3.x with essential libraries
3. Frameworks & Libraries –
    - Deep Learning: TensorFlow/Keras for implementing and training VGG16, ResNet50, Xception and EfficientNetB3 models.
    - Data Processing: OpenCV and NumPy for image manipulation and preprocessing.
4. Visualization: Matplotlib and Seaborn for creating performance and analysis reports.
5. Scikit-learn – for classification evaluation, confusion matrix, and classification report

3.4.2 Hardware Requirements

1. Processor – Minimum Intel Core i5 (10th Gen or higher) or AMD Ryzen 5 or or equivalent cloud-based processor for efficient processing of machine learning models.
2. Memory (RAM) – At least 8GB RAM to handle image processing, feature extraction, and classification tasks smoothly.
3. Storage – Minimum 20GB free disk space, preferably SSD, for storing datasets, extracted features, and trained models.
4. Graphics Processing Unit (GPU) – NVIDIA GTX 1650 or higher recommended if deep learning models are used for classification. Cloud-based GPU provided by Google Colab (recommended for training deep learning models)
5. Peripheral Devices – Monitor, keyboard, and mouse for system operation. Basic display and input devices for accessing Colab in a web browser

CHAPTER 4: PROPOSED METHODOLOGY

4.1 PROPOSED SYSTEM ARCHITECTURE / BLOCK DIAGRAM

The proposed system is designed to classify brain MRI images into four categories—glioma, meningioma, pituitary tumor, and no tumor—by utilizing a deep learning approach based on transfer learning. A custom dataset of 20,000 brain MRI images collected from various online sources was curated and used for model training and evaluation. The models used include pre-trained convolutional neural networks (CNNs) such as VGG16, ResNet50, EfficientNetB3, and Xception. The block diagram below presents the high-level system architecture:

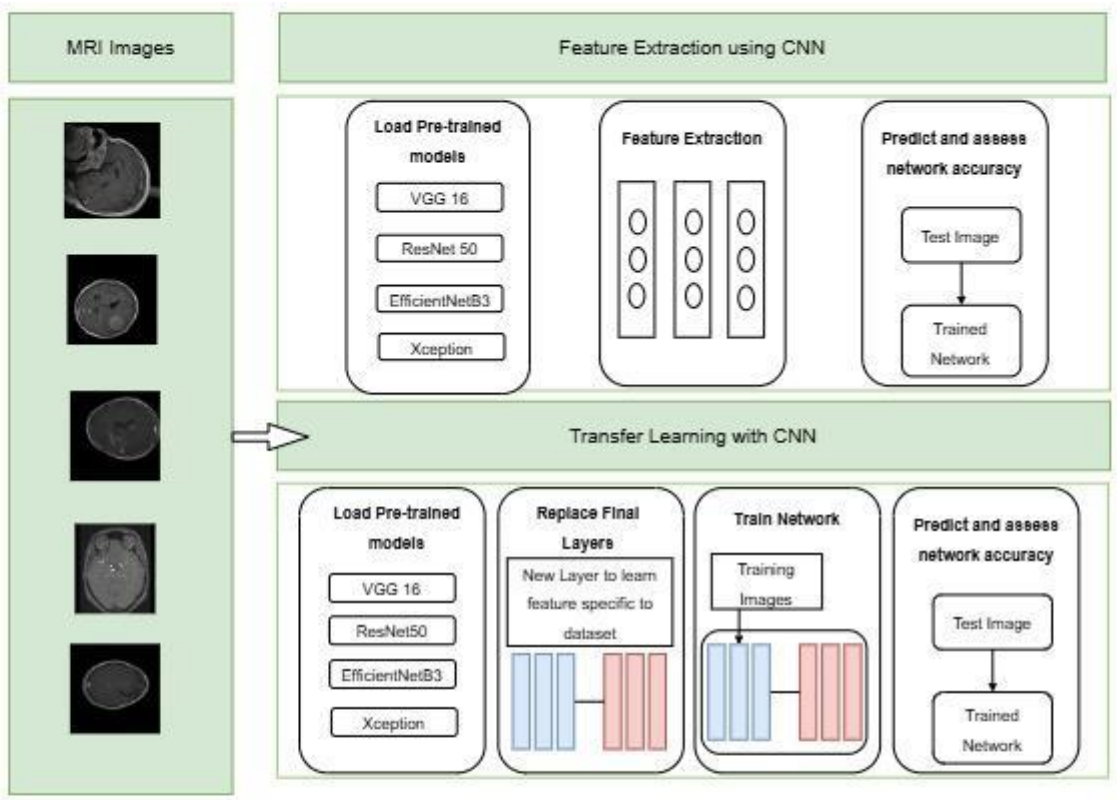


Fig 4.1: Feature Extraction and Transfer Learning Block Diagram

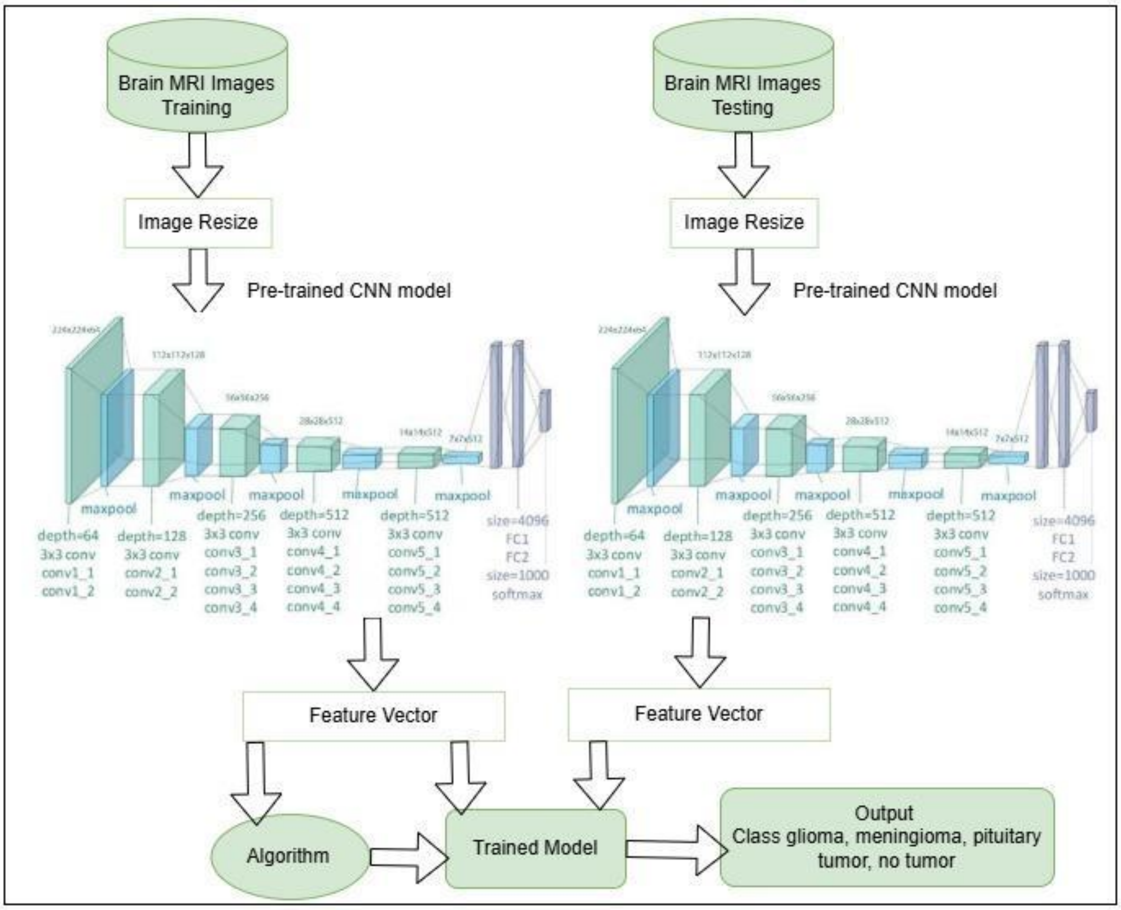


Fig 4.2: System Architecture Block Diagram

4.1.1 Model Optimization and Performance Improvement

During the course of model development for brain tumor classification, several strategic modifications were made to enhance the model's performance, generalization, and accuracy. These changes involved architectural tuning, hyperparameter adjustments, and training enhancements, which collectively led to a noticeable improvement in results.

a. Experimentation with Pre-trained Models

Multiple well-established pre-trained models were evaluated to identify the most suitable backbone for the task. These included:

- EfficientNetB3
- VGG16
- ResNet50
- Xception

Each model was fine-tuned on the custom dataset. EfficientNetB3 outperformed others in terms of validation accuracy due to its balanced compound scaling of depth, width, and resolution.

b. Redesigning the Classification Head

To better adapt the pre-trained models to the specific requirements of brain MRI classification:

- A custom classification head was added on top of the base models.
- Fully connected layers were deepened, with ReLU activations to capture complex non-linear patterns.
- Dropout layers were inserted between dense layers to reduce overfitting.

This architectural change resulted in improved learning of tumor-specific features and boosted classification performance.

c. Adjustment of Training Parameters

Several training parameters were fine-tuned to achieve stable learning and faster convergence:

- The number of training epochs was increased to allow the model more time to learn deep features.
- The batch size was optimized to ensure balanced training with efficient memory usage
- The optimizer and loss function were carefully selected for multi-class classification tasks..

These hyperparameter updates helped the model reach higher accuracy with better loss minimization.

d. Integration of Early Stopping

To avoid overfitting and ensure optimal weight retention, early stopping was implemented during training. This mechanism halted the training once the validation loss stopped improving, and restored the best-performing model weights. It allowed for efficient model training without overtraining.

e. Enhanced Image Preprocessing

Preprocessing methods such as resizing, normalization, and data shuffling were applied to ensure uniforminput data and improve generalization. These steps helped the model adapt better to variations in MRIimage characteristics.

4.1.2 System Architecture Workflow

1. MRI Image Acquisition – Brain MRI images are collected from publicly available datasets and combined to form a comprehensive dataset containing 20,000 images, ensuring a balanced representation of all four tumor classes.

2. Preprocessing – All images are resized to a standard dimension to match the input size required by the pre-trained models. Further preprocessing such as normalization and augmentation is applied to enhance model performance and prevent overfitting.

3. Feature Extraction Using CNN – Pre-trained CNN models (VGG16, ResNet50, EfficientNetB3, and Xception) are used to extract deep features from the MRI images. These models are chosen for their proven effectiveness in image classification tasks.

4. Transfer Learning – The last fully connected layers of the pre-trained models are replaced with new layers suitable for the four-class classification problem. The models are fine-tuned on the new dataset, allowing them to adapt to brain tumor-specific features.

5. Model Training – The training dataset is fed into the modified CNNs. The models learn discriminative features through backpropagation and optimization algorithms like Adam or SGD.

6. Model Evaluation – The trained models are tested on unseen MRI images. Metrics such as accuracy, precision, recall, F1-score, and confusion matrix are used to evaluate performance.

7. Classification Output – The system predicts the class label of the input MRI image as either glioma, meningioma, pituitary tumor, or no tumor. The results are analyzed to determine the most effective model for the classification task.

4.2 DATASET DESIGN

The design of the dataset plays a crucial role in determining the effectiveness and robustness of any deep learning model. In this project, a custom dataset was curated specifically for brain tumor classification by collecting brain MRI scans from multiple publicly available sources across the internet.

1. Dataset Collection

   To ensure diversity and generalization, brain MRI images were gathered from various research publications, open-access repositories, and healthcare imaging databases. The collected images were carefully selected to represent different tumor types and healthy cases, resulting in a comprehensive and representative dataset.

2. Dataset Composition

   The finalized dataset contains approximately **20,000 MRI images**, categorized into four distinct classes:

   - Glioma Tumor
   - Meningioma Tumor
   - Pituitary Tumor
   - No Tumor

Each class contains thousands of samples captured under varying conditions, which adds robustness to the training process and improves model generalization.

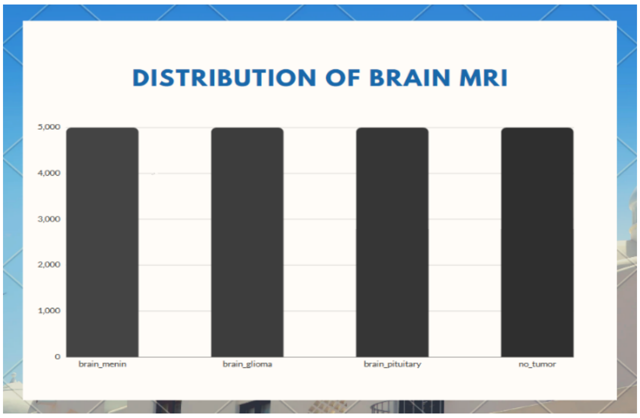


Fig 4.3. Distribution of Brain MRI bar graph

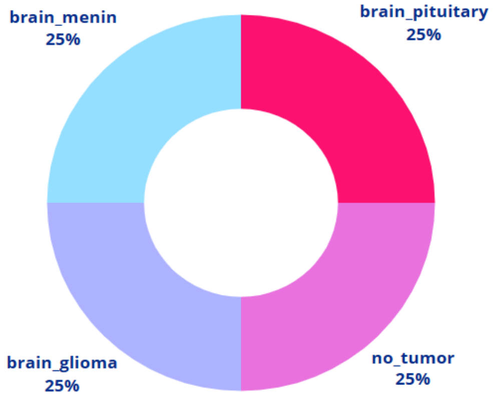


Fig 4.4. Distribution of Brain MRI Pie Chart

3. Data Organization

The images were structured into a directory-based format for efficient preprocessing and model training:

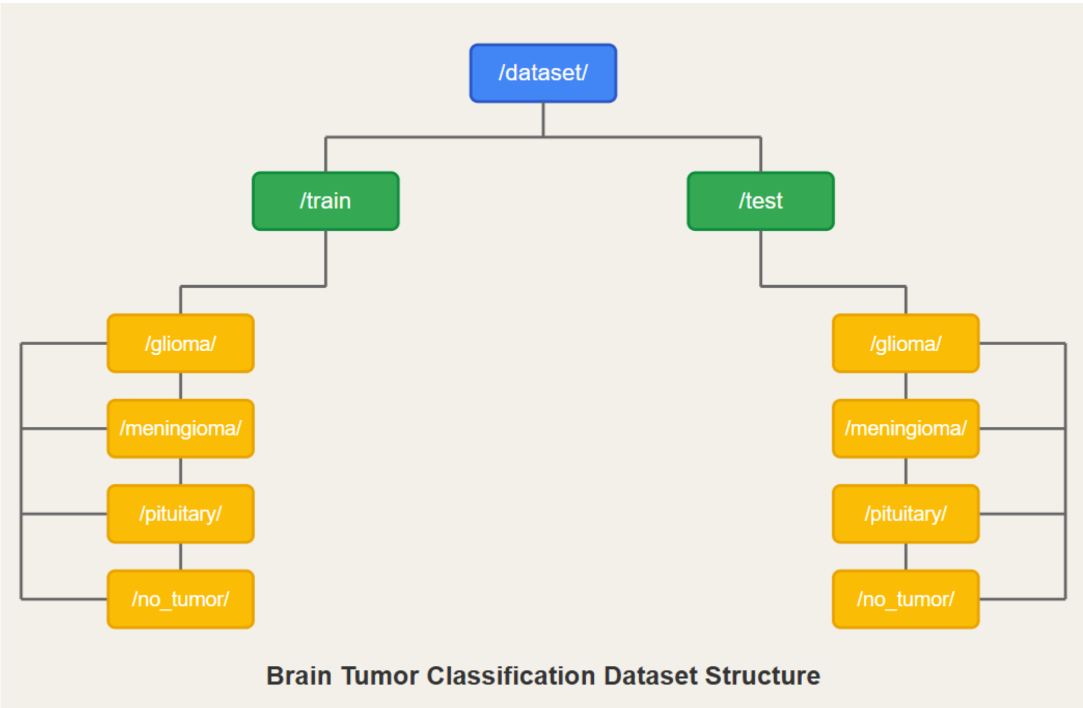


Fig 4.5. Dataset Structure Diagram

This structured hierarchy allows seamless integration with deep learning libraries like TensorFlow and PyTorch, facilitating easy batch loading and label assignment.

4. Data Split

The entire dataset was split into training and testing sets using an 80:20 ratio:

- Training Set: 16,000 images used for model learning.
- Testing Set: 4,000 images used for model evaluation on unseen data.

This separation ensures a fair assessment of the model's performance and helps prevent overfitting.

5. Dataset Properties

Each image in the dataset:

- Was resized to 512×512 pixels for uniform input dimensions.
- Maintains grayscale or RGB format based on the model's requirement.
- Includes sufficient variation in contrast, orientation, and intensity, ensuring the model learns meaningful features.

6. Dataset Integrity and Quality Checks

Manual inspection and validation were performed to:

- Remove duplicate or corrupted images.
- Ensure accurate labeling.
- Balance class distributions to avoid bias during model training.

7. Visual Analysis

Bar plots and pie charts were used to visualize the class-wise distribution of the dataset. This helped verify that the dataset was balanced and no class was underrepresented.

4.3 OVERVIEW OF PROJECT MODULES

4.3.1 IMAGE ACQUISITION MODULE

The Image Acquisition Module plays a critical role in gathering and managing brain MRI images for classification purposes. It is responsible for sourcing, verifying, and organizing the MRI images used in both training and testing phases of the brain tumor classification system.

4.3.1.1 Functions of the Image Acquisition Module

1.
2. Dataset Loading – Loads brain MRI images from various online medical image repositories, such as Kaggle, public hospital datasets, or open-access research archives. A custom dataset of 20,000 images was compiled to ensure diversity and robustness in training.
3. Image Upload Interface (Optional) – Provides the option for users or practitioners to upload new MRI images in standard formats such as JPEG, PNG, or DICOM for real-time classification or testing purposes.
4. Image Validation – Ensures that each uploaded or sourced image meets the required quality, resolution, and format standards before moving on to preprocessing. This step helps maintain consistency and prevents erroneous data from affecting model performance.
5. Image Storage and Organization – Stores all images in a structured directory system categorized by tumor types (glioma, meningioma, pituitary tumor, and no tumor). This structured approach facilitates easier data handling during training and evaluation.
6. Real-Time Image Capture (Optional) – If integrated with hospital imaging systems or MRI machines, the module can be extended to capture real-time brain MRI images for immediate classification and feedback.

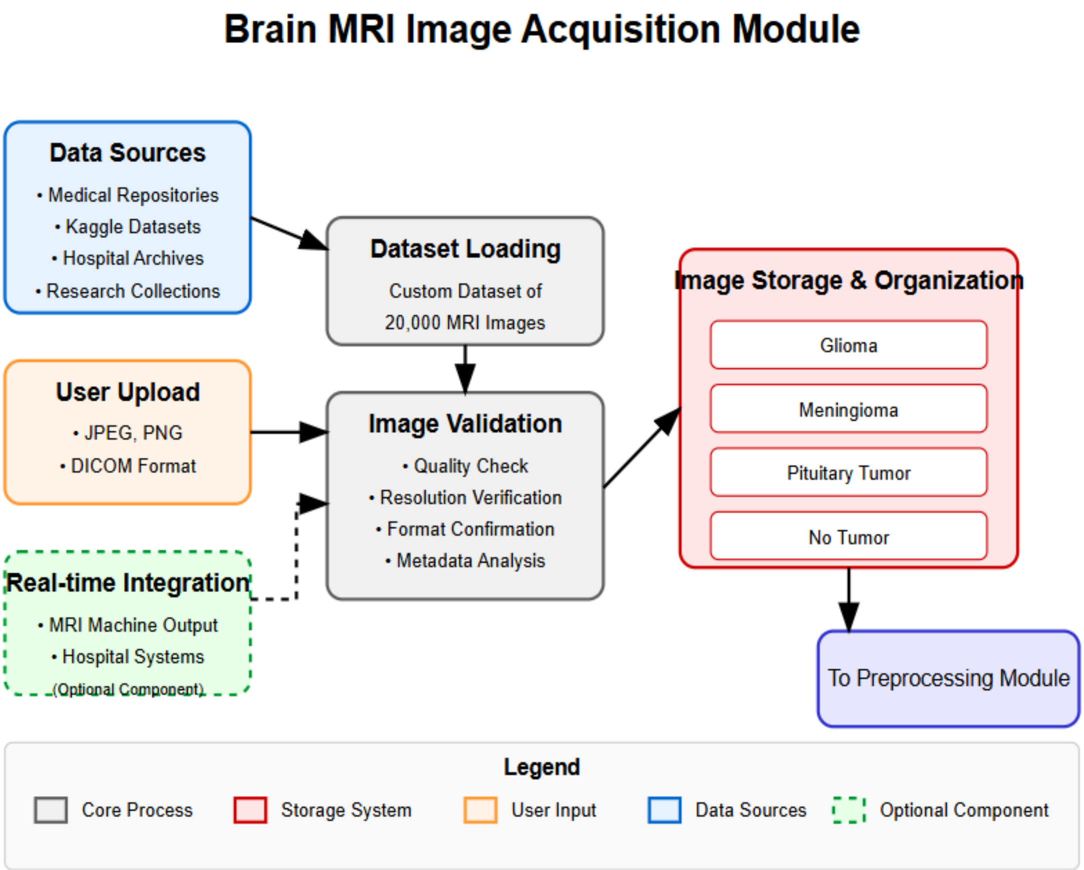


Fig 4.6. Image Acquisition Module Block Diagram

4.3.2 PREPROCESSING MODULE

The Preprocessing Module is responsible for preparing brain MRI images before feeding them into the deep learning models. These steps ensure consistent formatting and compatibility with the pre-trained CNN architectures used in the project.

4.3.2.1 Functions of the Preprocessing Module

1. Image Resizing – All MRI images are resized to 128×128 pixels to match the input requirements of models like EfficientNetB3, VGG16, ResNet50, and Xception.
2. Data Structuring – The dataset is split into training and testing sets using an 80:20 ratio, helping to streamline the training pipeline and maintain separation for evaluation.
3. Class Visualization – The distribution of samples across the four categories—glioma, meningioma, pituitary tumor, and no tumor—is visualized using bar plots to confirm dataset balance.
4. Array Conversion – Image data and corresponding labels are converted into NumPy arrays, making them suitable for processing by TensorFlow and Keras-based deep learning models.
5. Model Input Preparation – All preprocessing steps ensure that the input format aligns with the expectations of the transfer learning models used in the classification pipeline.

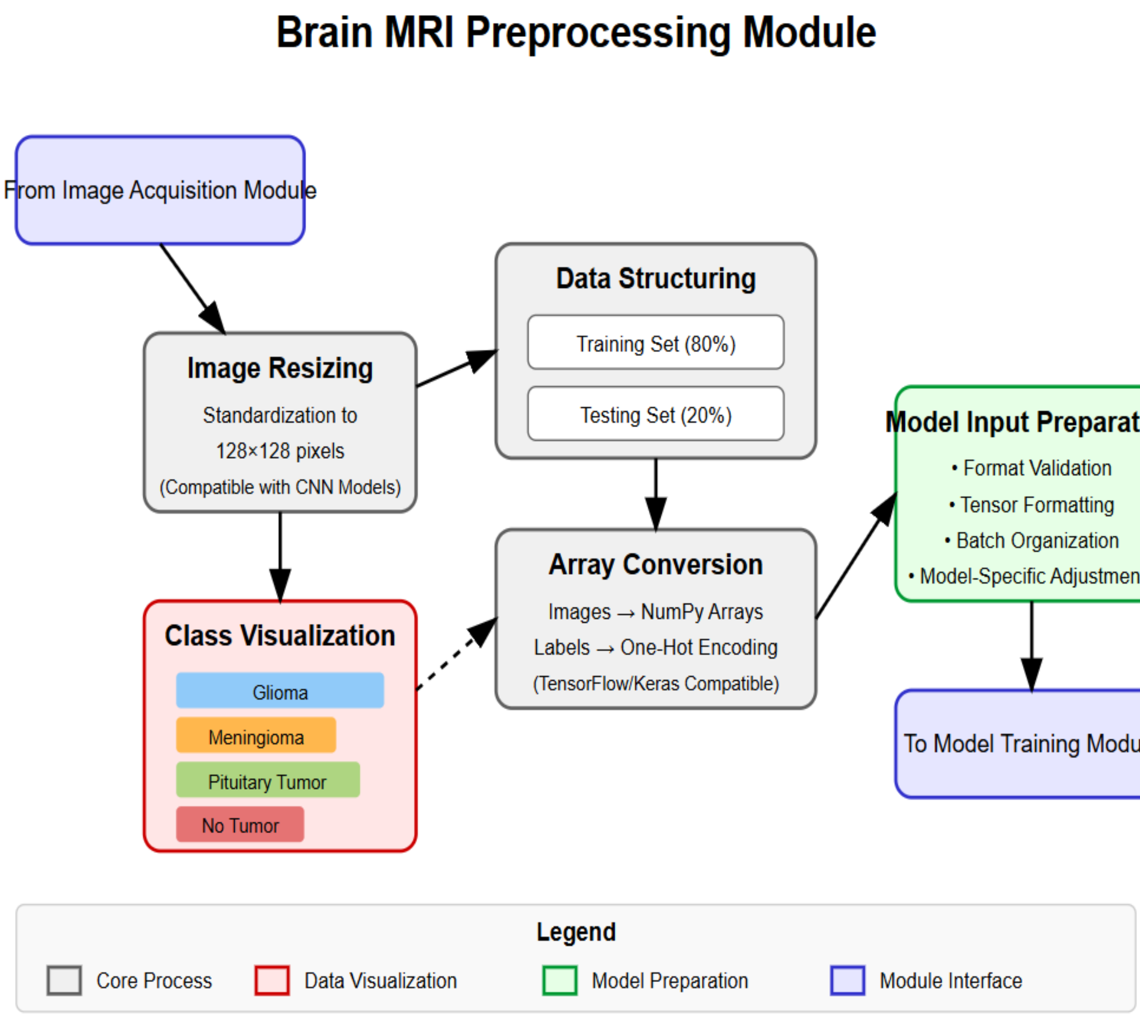


Fig 4.7. Preprocessing Module Block Diagram

4.3.2 FEATURE EXTRACTION USING CNN MODULE

The Feature Extraction Module is responsible for leveraging pre-trained Convolutional Neural Networks (CNNs) to extract high-level features from brain MRI images. These features are passed to dense layers for classification. The use of transfer learning with models trained on large-scale datasets ensures efficient learning and enhanced accuracy, even with medical images.

4.3.2.1 Functions of the Feature Extraction Module

1. Transfer Learning Integration – Pre-trained CNN models (EfficientNetB3, VGG16, ResNet50, Xception) are imported with their ImageNet weights and used without their top classification layers. This enables deep feature reuse while reducing training time.
2. Convolutional Base Usage – Only the convolutional layers of each model are used to extract spatial and semantic patterns from input images. These patterns include edges, textures, tumor shapes, and tissue structures.
3. Layer Freezing – All layers in the pre-trained models are frozen during the initial training phase to retain their learned feature representations and avoid overfitting.
4. Flattening and Dense Layer Addition – The output feature maps from the CNN base are flattened and passed through custom dense layers. These layers fine-tune the learned representations for the specific classification task.

5. Model Compilation and Training – The full model (CNN base + dense layers) is compiled using the Adam optimizer and sparse categorical crossentropy. It is then trained on the preprocessed MRI data to classify images into four categories: brain glioma, meningioma, pituitary tumor, and no tumor.

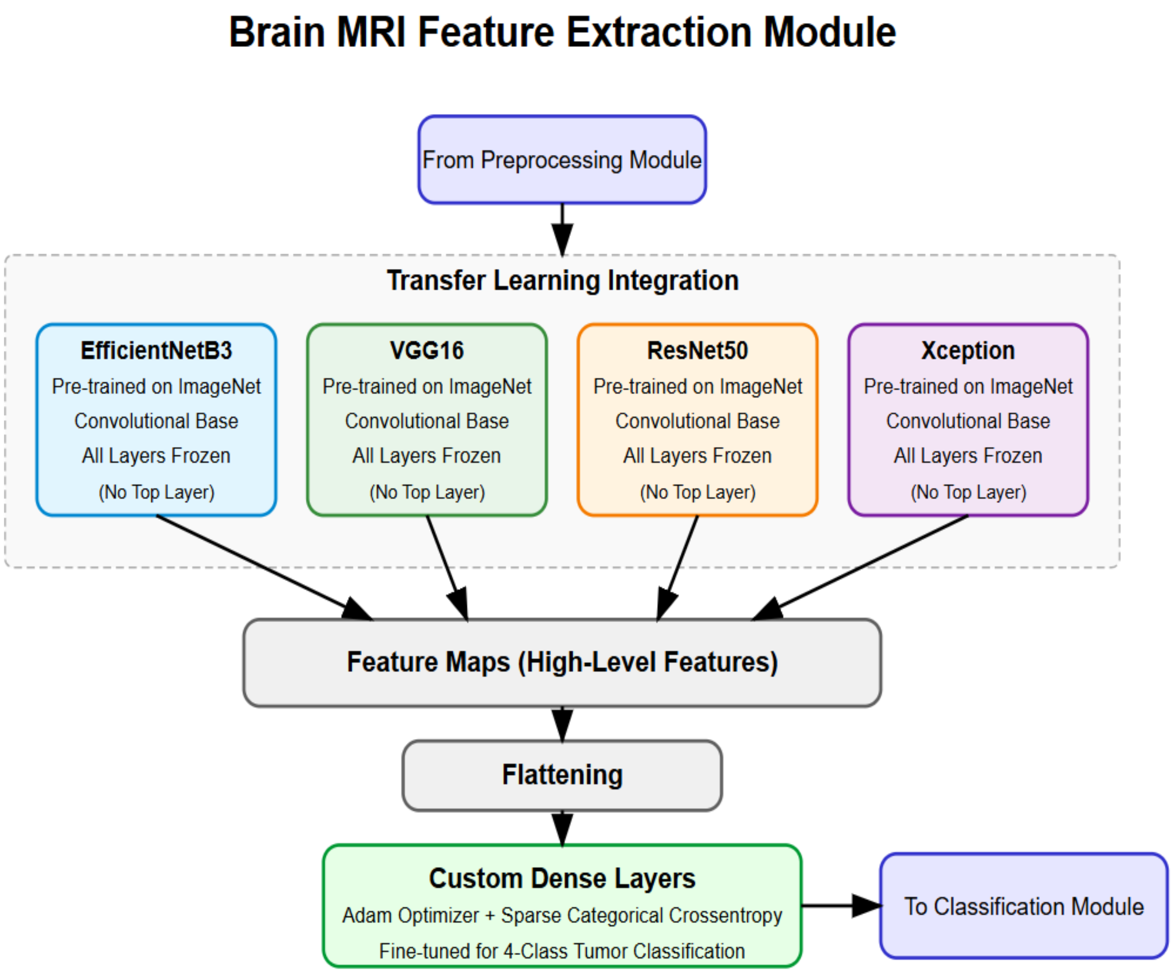


Fig 4.8. Feature Extraction Module Block Diagram

4.3.4 MODEL TRAINING MODULE

The Model Training Module is responsible for fitting the customized CNN architectures to the preprocessed brain MRI dataset. During this phase, the models adjust their internal parameters through iterative learning to accurately classify tumors based on the input features.

4.3.4.1 Functions of the model training module

1. Training Input Feeding – The preprocessed training dataset, containing labeled MRI images, is fed into the CNN models in batches. Each image is passed through the convolutional layers and custom classification head.
2. Backpropagation Execution – The models compute the loss between predicted and actual labels. Using backpropagation, gradients are calculated and propagated backward through the network to update the model weights.
3. Optimization Algorithms – Optimizers like Adam are used to minimize the loss function efficiently. These algorithms adjust the learning process by adapting the learning rate for each parameter, ensuring faster and more stable convergence.
4. Validation Monitoring – The model's performance is continuously evaluated on the test dataset during training to track generalization ability and detect overfitting early.
5. Early Stopping Integration – An early stopping callback is implemented to halt training once the validation loss stops improving for a defined number of epochs. This prevents overfitting and reduces unnecessary computation.
6. Model Performance Logging – Throughout training, metrics such as **accuracy** and **loss** for both training and validation sets are recorded and visualized using plots to provide insights into model behavior.

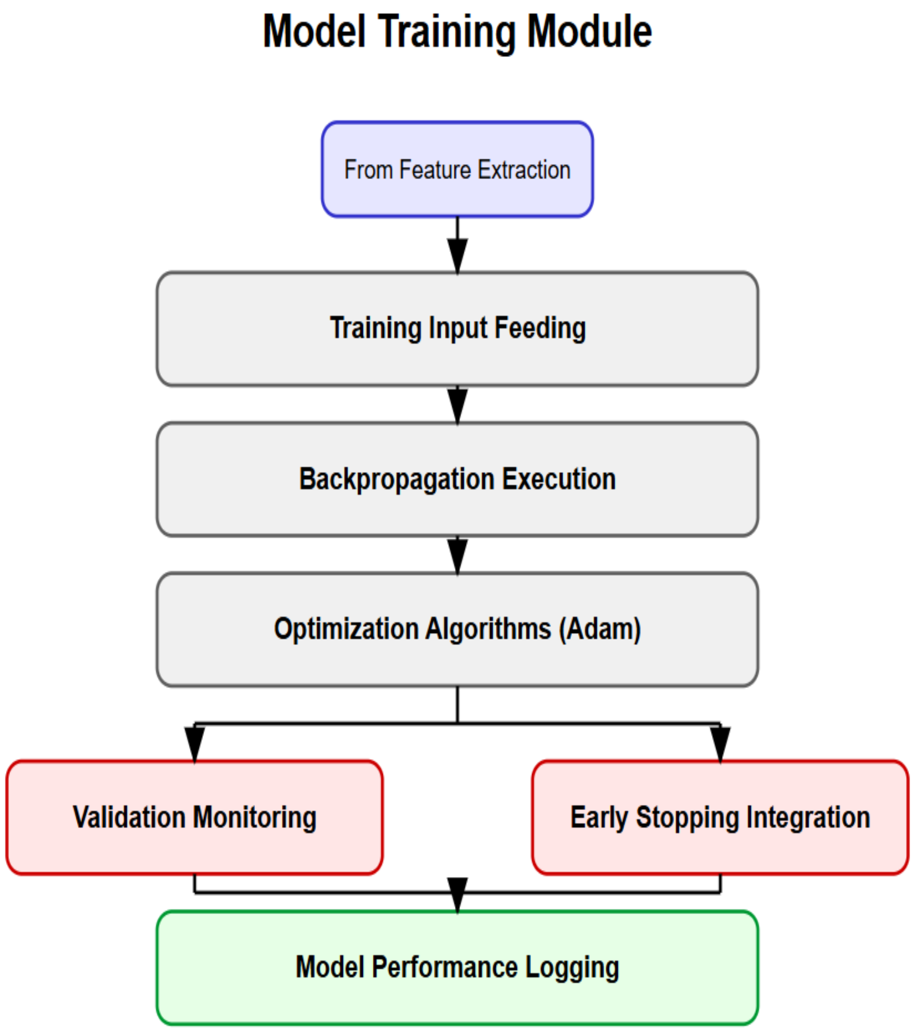


Fig 4.9. Model Training Module Block Diagram

4.3.5 MODEL EVALUATION MODULE

The Model Evaluation Module assesses the performance of the trained CNN models using a set of standardized metrics. It ensures that the models generalize well to unseen MRI images and provides a comprehensive analysis of classification effectiveness across all four tumor categories.

4.3.5.1 Functions of the model evaluation module

1. Testing on Unseen Data – After training, the models are evaluated using the 20% of the dataset reserved for testing. These images were not exposed to the models during training, ensuring a fair assessment of generalization.
2. Accuracy Calculation – The overall accuracy is computed to represent the percentage of correctly classified MRI images out of the total test set.
3. Precision and Recall – Precision measures the correctness of positive predictions, while recall indicates the model’s ability to identify all relevant cases for each class. These metrics are especially critical in medical diagnostics.
4. F1-Score – The harmonic mean of precision and recall, the F1-score provides a balanced metric that considers both false positives and false negatives, particularly valuable in imbalanced class scenarios.
5. Confusion Matrix Analysis – A confusion matrix is generated to visualize the distribution of correct and incorrect predictions across all classes. This matrix helps identify any misclassifications between similar tumor types.
6. Visual Evaluation – Bar plots and heatmaps are used to present the model's performance visually, aiding in quick interpretation of strengths and categories.

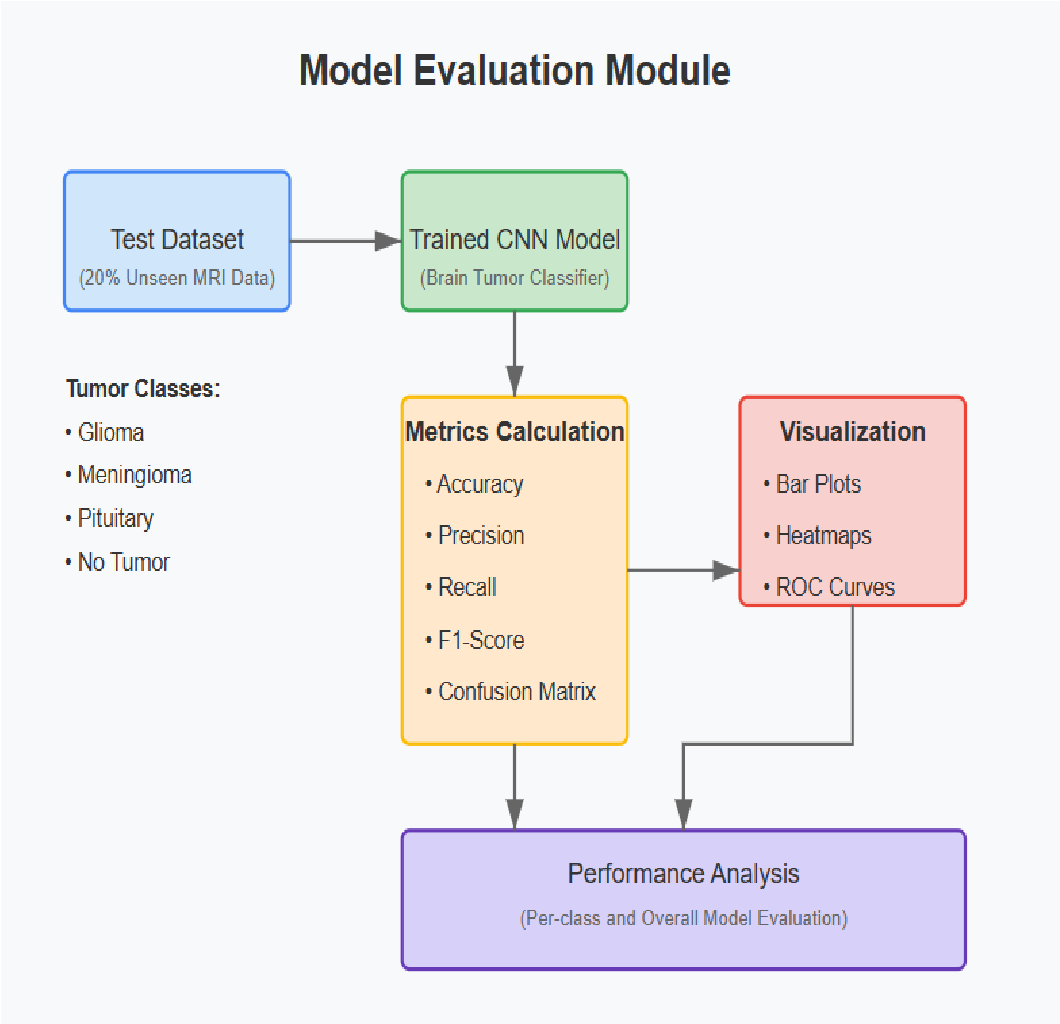


Fig 4.10. Model Evaluation Module Block Diagram

4.4 TOOLS AND TECHNOLOGIES USED

1. Programming Languages

- Python – Primary language used for image preprocessing, deep learning model implementation, training, and evaluation. Python's ecosystem offers extensive support for scientific computing and machine learning workflows.

2. Deep Learning Libraries

- TensorFlow/Keras – Used for building and training transfer learning models such as VGG16, ResNet50, EfficientNetB3, and Xception. These libraries offer high-level APIs and GPU acceleration for deep learning tasks.

3. Image Processing and Data Handling

- OpenCV – Used for reading, resizing, and preprocessing brain MRI images. It facilitates noise handling, image normalization, and format standardization.
- NumPy – Assists in managing image arrays and performing numerical operations.
- Pandas – Used for structuring datasets and handling class labels and metadata.

4. Visualization and Performance Analysis

- Matplotlib – Plots training curves (accuracy/loss) and prediction outputs for interpretability.
- Seaborn – Generates heatmaps and bar plots for visualizing class distribution and confusion matrices.
- Scikit-learn – Supports evaluation metrics such as precision, recall, F1-score, and also facilitates creation of confusion matrices and classification reports.

5. Dataset and Storage

- Google Drive – Utilized to store the custom dataset of 20,000 MRI images collected from various online sources. It also manages saved models and training logs within the Colab environment.

- Google Colab – Provides a cloud-based GPU platform for training deep learning models efficiently.

6. Development Environment

- Google Colab – Used as the primary IDE for executing the entire pipeline, from data loading and preprocessing to model training and evaluation, with built-in GPU support.
- Jupyter Notebook – Alternative interactive environment for prototyping and debugging, particularly for testing model segments or analyzing results.

4.5 MATHEMATICAL MODEL

The mathematical model for the Brain Tumor Classification System defines how the features of MRI images are extracted, processed, and classified into tumor categories (glioma, meningioma, pituitary tumor, and no tumor) using transfer learning with pre-trained Convolutional Neural Networks (CNNs).

1. Problem Definition (Set Theory Representation)

Let:

- S be the overall system
- I be the set of input MRI images
- P be the preprocessing function
- F be the feature extraction function (using CNNs)
- C be the classification function
- O be the output (tumor category prediction)

The system can be represented as:

$$S = \{I, P, F, C, O\}$$

Where:

- $P : I \rightarrow I'$ (Preprocessed image)
- $F : I' \rightarrow V$ (Feature vector extraction)
- $C : V \rightarrow O$ (Classification using ML models)

2. Convolution Operation

The convolution layer is the core of CNNs and is directly responsible for feature extraction. It slides a small matrix (called a kernel or filter) over the input image and calculates the dot product between the kernel and the overlapping region.

Given an input image *I* and a filter/kernel *K*, the convolution output O(*i*, *j*) at position (*i*, *j*) is:

$$O(i,j) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} K(m,n) \,.\, I(i+m, j+n)$$

Where:

- $M \times N$ is the size of the kernel
- i, j iterate over the spatial dimensions of the output

3. Activation Function (ReLU)

After convolution, a non-linear function such as ReLU (Rectified Linear Unit) is applied to add non-linearity to the model.

$$f(x) = \max(0, x)$$

This introduces non-linearity into the model.

1. Pooling Operation (Max Pooling)

Pooling layers reduce the spatial dimensions of the feature maps, keeping the most prominent features intact while reducing computational complexity.

Mathematical Equation:

$$P(i, j) = \max\ O(i+m, j+n)$$

Where:

- P(i, j): Pooled output
- The pooling operation (usually max) selects the strongest activation in a neighbourhood

2. Fully Connected Layer (Dense Layer)

These layers act as **classifiers**, interpreting the high-level features extracted earlier.

Mathematical Equation:

$$z = \sum_{i=1}^{n} w_i x_i + b$$

- w_i : Weight for input x_i
- *b*: Bias term
- *z*: Linear combination of features

3. Softmax Activation

Used at the final output layer for multi-class classification.

Mathematical Equation:

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

- z_j : Score for class j
- K: Number of classes (here, 4)
- $\sigma(z)_j$: Probability of class j

4. Loss Function (Categorical Cross-Entropy)

   Used to measure how well the model's predictions align with actual labels during training.

   Mathematical Equation:

$$L = \sum_{i=1}^{C} y_i \ \log \overline{y_i}$$

- y_i : True class label (one-hot encoded)
- $y\hat{}_i$: Predicted probability
- C: Number of classes

5. Performance Metrics

   - Accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP+TN+FP+FN}$$

   - Precision

$$\text{Precision} = \frac{TP}{TP + FP}$$

   - Recall

$$\text{Recall} = \frac{TP}{TP + FN}$$

   - F1-Score

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.6 ALGORITHM DETAILS

4.6.1 Algorithm 1: Feature extraction via convolutional neural networks (cnn)

Feature extraction via Convolutional Neural Networks (CNNs) is a key process in image analysis, where the network automatically learns important patterns from input images. It begins with convolutional layers that apply filters to detect low-level features such as edges, corners, and textures. As the network deepens, these features become more complex, capturing shapes and object parts. Activation functions like ReLU introduce non-linearity, enabling the model to learn intricate patterns. Pooling layers follow, reducing the spatial dimensions while retaining essential information, which improves computational efficiency and helps prevent overfitting. This hierarchical extraction of features—from simple to complex—makes CNNs highly effective for tasks such as image classification, object detection, and medical image analysis.

1. Start
2. Acquire brain MRI images from the constructed dataset (20,000 images collected from online sources).
3. Preprocess each image:

- Resize to 128×128 pixels.
- Normalize pixel values for standardization.

4. Input the image into a pre-trained CNN model (EfficientNetB3, VGG16, ResNet50, or Xception).
5. Apply convolutional layers to extract hierarchical spatial features such as edges, textures, and region patterns.
6. Use max / average pooling to reduce dimensionality while preserving critical spatial information.
7. Forward the output to deeper convolutional layers for more abstract feature learning.
8. Extract the final set of deep features from the last convolutional block.
9. Flatten the features into a one-dimensional vector, creating a feature representation of the image.
10. End

4.6.2 Algorithm 2: Classification using cnn (transfer learning approach)

Classification using CNNs with the transfer learning approach involves leveraging a pre-trained CNN model, originally trained on a large dataset like ImageNet, and adapting it to a new but related task. Instead of training a CNN from scratch, which requires extensive data and computational power, transfer learning allows the reuse of learned features such as edges, shapes, and patterns. The lower layers of the pre-trained model are typically frozen, as they capture generic features, while the higher layers are fine-tuned or replaced with new layers specific to the target classification task. This approach significantly reduces training time, improves accuracy, and is especially effective when working with limited data, making it a popular choice in applications like medical image diagnosis and custom object classification.

1. Start
2. Replace the final fully connected layers of each pre-trained model with a new classification head tailored for four categories:

- Glioma
- Meningioma
- Pituitary Tumor
- No Tumor

3. Fine-tune the model using backpropagation with the Adam optimizer.
4. Train on the processed dataset (80% for training, 20% for testing).
5. Predict tumor classes for unseen test images.
6. Evaluate the model using:
    - Accuracy
    - Precision
    - Recall
    - F1-score
    - Confusion Matrix
7. End

4.7 COMPLEXITY OF PROJECT

1. Data Processing Complexity

    1. Large medical image dataset (20,000 images) requires consistent resizing, normalization, and color standardization.
    2. Preprocessing is GPU-intensive due to high-resolution and batch processing.

2. Computational Complexity

    1. CNN Convolutional Operations: $O(K \times F \times F \times D)$, where:
        - K = number of filters
        - F = filter size
        - D = depth of input
    2. Transfer Learning Fine-Tuning: Relies on gradient descent, requiring multiple forward-backward passes.

3. Algorithmic Complexity

    1. Deep feature extraction using CNN involves stacking multiple convolutional and pooling layers.
    2. Fine-tuning pre-trained models increases the parameter space, especially in dense layers.

4. Implementation Complexity

    1. Requires integration of OpenCV (image handling), TensorFlow/Keras (CNNs), and Scikit-learn (metrics).
    2. Managing class imbalance, early stopping, and learning rate scheduling demands experience with training optimization.

5. Performance Complexity
    1. Achieving high accuracy (>97% in EfficientNetB3) involved careful choice of model architecture and preprocessing.
    2. Real-time inference on new MRI scans may require model compression or acceleration for deployment.

6. Overall Complexity Level: Moderate to High
    1. Due to the combined use of pre-trained models, statistical feature engineering, and deep learning pipelines.
    2. System is optimized for both performance and accuracy, making it suitable for clinical decision support.

CHAPTER 5: PROJECT PLAN

5.1 RISK MANAGEMENT

5.1.1 RISK ANALYSIS

Risk management is crucial for ensuring the successful execution of the malocclusion detection system using fingerprints. Since this project involves medical image processing and deep learning models, various risks can affect data quality, image augmentation, classification accuracy, and system scalability. Identifying these risks early allows for implementing effective strategies to minimize their impact, ensuring a reliable and efficient system. This section highlights potential risks, their severity, and the corresponding mitigation strategies to maintain the project's success.

5.1.2 OVERVIEW OF RISK MITIGATION, MONITORING AND MANAGEMENT

In our projects i.e. brain tumor diagnosis using MRI images, risk management is not just about ensuring technical correctness; it is also about protecting patient safety, ensuring model reliability, and building trust among clinical stakeholders. Our project compares the performance of four advanced CNN architectures—VGG16, ResNet50, Xception, and EfficientNetB3—on classifying MRI scans of brain tumors. Given the medical significance of the task, it becomes crucial to identify and mitigate potential risks early in the development cycle to ensure our results are both scientifically valid and clinically meaningful. Below is a detailed overview of the risks we identified and the strategies employed to manage them.

1. Data-Related Risks

- Low-Quality or Noisy MRI Images
  MRI scans may suffer from motion blur, low resolution, or imaging artifacts, which can hinder accurate feature extraction and degrade classification performance.

Variations in machine calibration and scanning protocols across hospitals can also introduce inconsistencies in the data.

- Imbalanced Dataset Distribution
  Brain tumor datasets often contain significantly more samples of certain tumor types or normal cases compared to others. This imbalance may cause the model to become biased toward the majority class, leading to poor performance on minority classes.
- Limited Dataset Size
  Due to privacy concerns and the cost of acquiring labeled medical data, the available dataset may not be large enough to support deep learning models, especially the deeper ones like EfficientNetB3, without overfitting.

Mitigation:

To address this, we used data augmentation techniques such as flipping, rotation, zooming, and contrast adjustments to artificially expand the training dataset. Additionally, we sourced publicly available, pre-labeled datasets like BraTS and Figshare, which are widely accepted in academic research. We also implemented cross-validation to ensure that the performance comparison across models is statistically reliable and not affected by a particular train-test split.

2. Algorithmic and Model Risks

- Overfitting of Deep Models
  High-capacity networks like Xception and EfficientNetB3 are more prone to memorizing training data instead of generalizing, particularly when the training set is small or lacks diversity.
- Model Interpretability Challenges
  Deep neural networks are often considered "black boxes." Without proper explainability tools, their predictions might be difficult to interpret for medical professionals, affecting trust and adoption.

Mitigation:
We incorporated regularization techniques such as dropout layers, L2 weight regularization, and early stopping to halt training when the model’s performance on the validation set

plateaued. We also applied data augmentation to expose the model to more diverse examples and reduce the risk of overfitting on the limited data available.

3. Computational and Performance Risks

- High Training Time and Resource Requirements: Deep CNN models, especially EfficientNetB3, require significant computational power and memory, which may be challenging to manage within academic or low-resource environments.
- Latency in Real-Time Scenarios: If this system is eventually integrated into a clinical workflow, the inference speed could become a bottleneck, especially for high-resolution MRI scans.

Mitigation:

We used Google Colab Pro with TPU v2-8 acceleration to speed up training without sacrificing performance. Training was conducted in mini-batches to optimize memory usage. We also monitored resource usage and dynamically adjusted batch size and image resolution to strike a balance between performance and feasibility.

4. User and Adoption Risks

- Lack of Clinical Trust in AI Predictions:Clinicians may hesitate to rely on AI predictions unless the model demonstrates consistently high accuracy and provides transparent justifications for itsdecisions.
- Resistance to New Technologies: Hospitals and radiology departments may have established workflows, and integrating AI-based diagnostic tools may face resistance due to cost, retraining, or regulatory approval processes.

Mitigation:

The visual explanations were added alongside performance metrics such as accuracy, precision, recall, and F1-score. This combination helps build clinician trust by offering both statistical evidence and human-understandable insights into the model’s decision-making process.

CHAPTER 6: MODEL TESTING

Although our project does not involve building a fully functional software application, testing still played a key role in ensuring the reliability and effectiveness of our deep learning models for brain tumor classification. Instead of testing a software system in the traditional sense, we focused on validating our models through various evaluation techniques to guarantee meaningful and trustworthy comparisons.

6.1 TYPE OF TESTING

1. Unit Testing
   - Description: This involved testing individual parts of the workflow such as the MRI image preprocessing pipeline, data augmentation methods, and evaluation metric functions.
   - Purpose: To verify that each component—like image normalization or F1-score calculation—worked correctly in isolation and did not introduce any logical errors into the model training process.

2. Integration Testing
   - Description: Each of the selected deep learning models (VGG16, ResNet50, Xception, and EfficientNetB3) was trained and validated on the same dataset split to ensure uniform testing conditions.
   - Purpose: To check how well each model performs on unseen MRI images, which is critical for evaluating its reliability in real-world scenarios.

3. Cross-Validation Testing
   - Description: In order to reduce the risk of biased outcomes from a single data split, cross-validation techniques were used where applicable.
   - Purpose: To ensure the consistency and robustness of each model's performance by training and testing on different parts of the dataset

4. Performance Testing
    - Description: This testing focused on assessing the computational aspects, such as training time, inference speed, and resource consumption (CPU/GPU usage) for all models.
    - Purpose: To confirm the models' efficiency and practicality, especially in environments where quick and accurate medical diagnoses are essential.

5. Comparative Testing
    - Description: A side-by-side comparison of all four models was conducted using key performance indicators—accuracy, precision, recall, and F1-score.
    - Purpose: To determine which model delivers the most accurate and balanced results for brain tumor classification and is most suitable for medical deployment.

6. Misclassification Analysis
    - Description: MRI images that were incorrectly predicted by the models were examined to identify common patterns and challenging tumor cases.
    - Purpose: To understand the limitations of the models and explore possible reasons behind incorrect predictions, which can inform future improvements.

7. User Acceptance Testing(UAT): [Proposed for Future Work]
    - Description: While a front-end interface was not part of the current project, a basic web-based tool for image upload and prediction could be developed and tested by medical professionals in future.
    - Purpose: To ensure that the system is usable, interpretable, and helpful to radiologists or doctors in a real-world diagnostic setting.

CHAPTER 7: RESULTS AND DISCUSSION

This section presents the outcomes of the Brain Tumor Classification System using deep learning, including a comparative analysis of four convolutional neural network models—VGG16, ResNet50, Xception, and EfficientNetB3. The analysis covers accuracy, precision, recall, F1-score, and training efficiency across varying dataset conditions, along with validation metrics and visual representations of model performance

7.1 OUTCOMES

The proposed deep learning-based system for brain tumor classification using MRI images was evaluated using standard performance metrics such as accuracy, precision, recall, and F1-score. The key outcomes of the project are summarized below:

- The use of transfer learning with pre-trained CNN architectures—VGG16, ResNet50, Xception, and EfficientNetB3—enabled efficient and accurate classification of brain MRI images into tumor and non-tumor categories.
- Among the models, EfficientNetB3 achieved the highest accuracy of 97.62%, followed closely by Xception (97.57%), VGG16 (97.15%), and ResNet50 (96.93%).
- The results demonstrated that deeper architectures with optimized parameter efficiency (such as EfficientNetB3) outperformed older CNN models, particularly in terms of generalization and precision.
- The models showed strong performance on the test set, validating the robustness of the training approach and the effectiveness of pre-processing steps such as normalization, resizing, and preprocessing.
- This comparative model evaluation lays the foundation for selecting the most suitable architecture for real-time diagnostic integration in medical imaging systems.

7.2 RESULT ANALYSIS AND VALIDATION

The performance of each model was evaluated using several key metrics, including accuracy, precision, recall, and F1-score. These metrics provide a comprehensive evaluation of the model's effectiveness, particularly its ability to correctly identify brain tumor images while minimizing misclassifications.

Model	Accuracy
VGG16	97.15%
EfficientNetB3	97.62%
ResNet50	96.93%
Xception	97.57%

Table 1: Accuracy across different Algorithms

The EfficientNetB3 model demonstrated the best performance in terms of both accuracy and training efficiency. It achieved higher accuracy than traditional models while requiring fewer resources, making it a promising candidate for deployment in real-time medical diagnostic systems. Training times were consistent across the models, with EfficientNetB3 being slightly faster in converging compared to VGG16 and ResNet50 due to its optimized parameter efficiency.

The following comparative analysis demonstrates that deep learning models, particularly EfficientNetB3, show promising results for brain tumor classification based on MRI images. The use of pre-trained CNN architectures, along with data preprocessing techniques, contributed to the system’s high accuracy, precision, recall, and F1-score. The results indicate that EfficientNetB3, in particular, provides the best trade-off between accuracy and computational efficiency, making it an ideal candidate for real-time integration into medical imaging systems for brain tumor detection.

Performance Metrices	Tumor Classes	VGG16	EfficientNetB3	RestNet50	Xception
Precision	Glioma	0.97	0.97	0.99	0.97
	Menin	0.97	0.97	0.93	0.98
	Pituitary	0.95	0.96	0.96	0.94
	No Tumor	1.00	1.00	1.00	1.00
Recall	Glioma	0.98	0.97	0.96	1.00
	Menin	0.91	0.94	0.95	0.91
	Pituitary	0.99	0.99	0.97	0.98
	No Tumor	1.00	1.00	1.00	1.00
F1-Score	Glioma	0.97	0.97	0.97	0.98
	Menin	0.94	0.95	0.94	0.95
	Pituitary	0.97	0.98	0.96	0.96
	No Tumor	1.00	1.00	1.00	1.00
Support	Glioma	1000	1000	1010	1010
	Menin	1000	1000	1010	1010
	Pituitary	1000	1000	1010	1010
	No Tumor	1000	1000	1010	1010

Table 2: Comparative performance Analysis

7.3 SCREENSHOTS / GRAPHS

7.3.1 Model Evaluation Module Block Diagram:

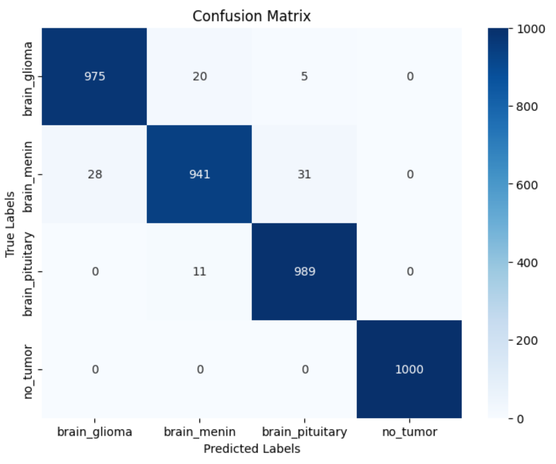


Fig 7.1 EfficientNetB3

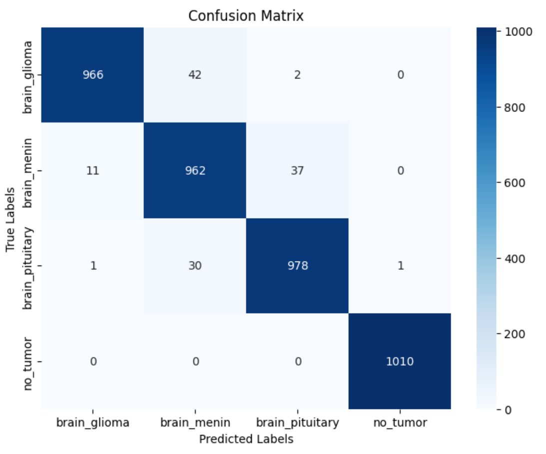


Fig 7.2 Resnet50

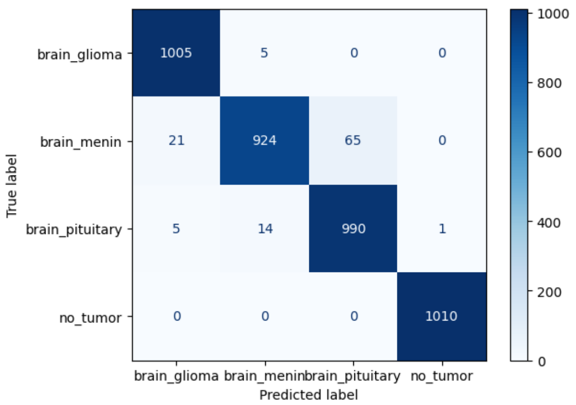


Fig 7.3 Xception

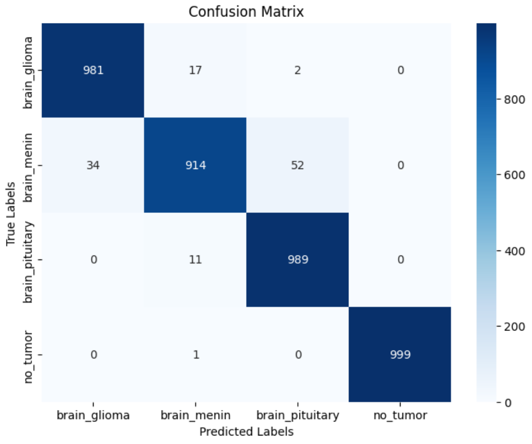


Fig 7.4 VGG16

7.3.2 Accuracy Table:

	precision	recall	f1-score	support
0	0.97	0.97	0.97	1000
1	0.97	0.94	0.95	1000
2	0.96	0.99	0.98	1000
3	1.00	1.00	1.00	1000
accuracy			0.98	4000
macro avg	0.98	0.98	0.98	4000
weighted avg	0.98	0.98	0.98	4000

Fig 7.5 EfficientNetB3

	precision	recall	f1-score	support
0	0.99	0.96	0.97	1010
1	0.93	0.95	0.94	1010
2	0.96	0.97	0.96	1010
3	1.00	1.00	1.00	1010
accuracy			0.97	4040
macro avg	0.97	0.97	0.97	4040
weighted avg	0.97	0.97	0.97	4040

Fig 7.6 Resnet50

	precision	recall	f1-score	support
0	0.97	0.98	0.97	1000
1	0.97	0.91	0.94	1000
2	0.95	0.99	0.97	1000
3	1.00	1.00	1.00	1000
accuracy			0.97	4000
macro avg	0.97	0.97	0.97	4000
weighted avg	0.97	0.97	0.97	4000

Fig 7.7 VGG16

Classification Report

	precision	recall	f1-score	support
brain_glioma	0.97	1.00	0.98	1010
brain_menin	0.98	0.91	0.95	1010
brain_pituitary	0.94	0.98	0.96	1010
no_tumor	1.00	1.00	1.00	1010
accuracy			0.97	4040
macro avg	0.97	0.97	0.97	4040
weighted avg	0.97	0.97	0.97	4040

Fig 7.8 Xception

7.3.2 Training V/S Testing Graphs :

1. **EfficientNetB3**

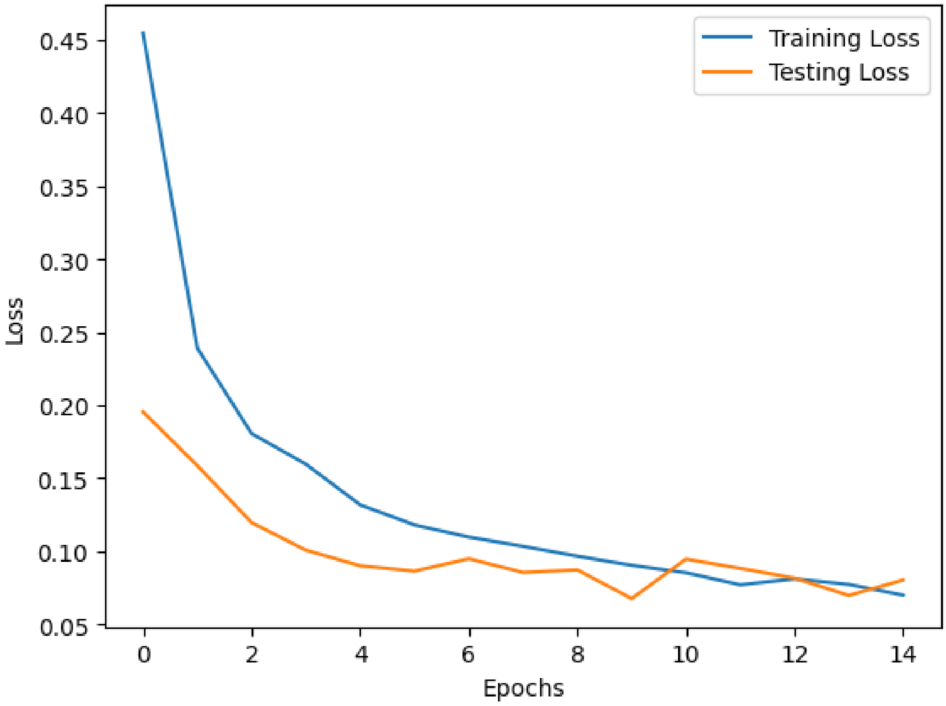


Fig 7.9 Training Loss V/S Testing Loss for EfficientNetB3

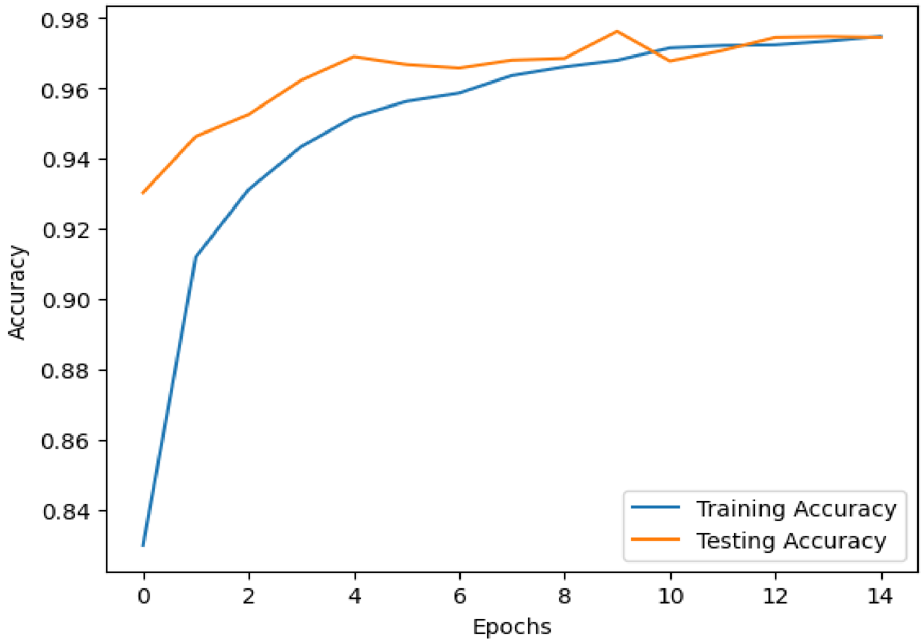


Fig 7.10 Training Accuracy V/S Testing Accuracy for EfficientNetB3

2. Resnet50

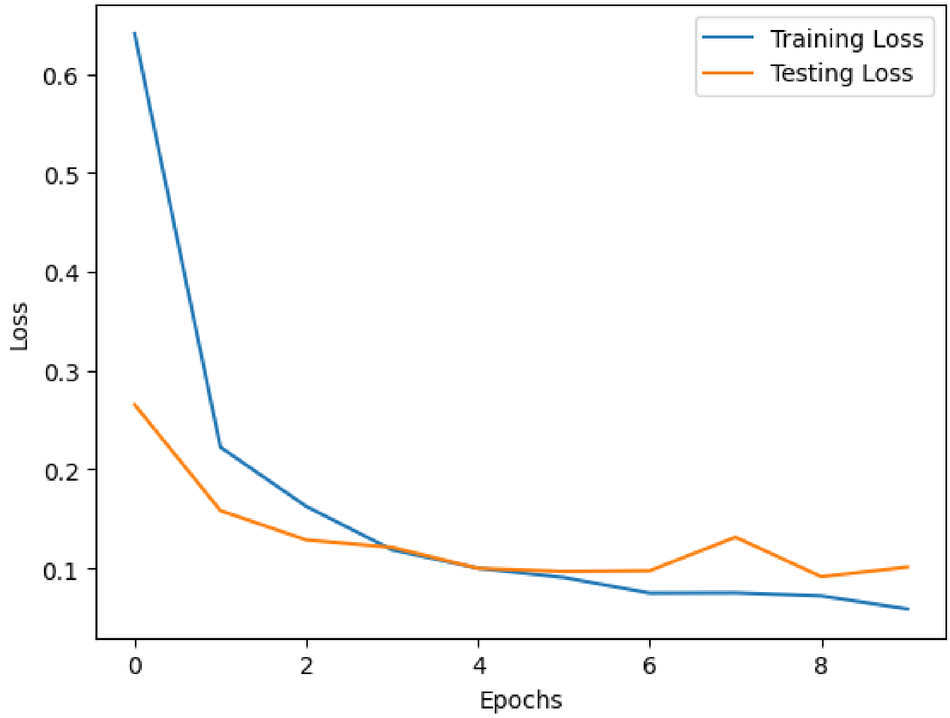


Fig 7.11 Training Loss V/S Testing Loss for Resnet50

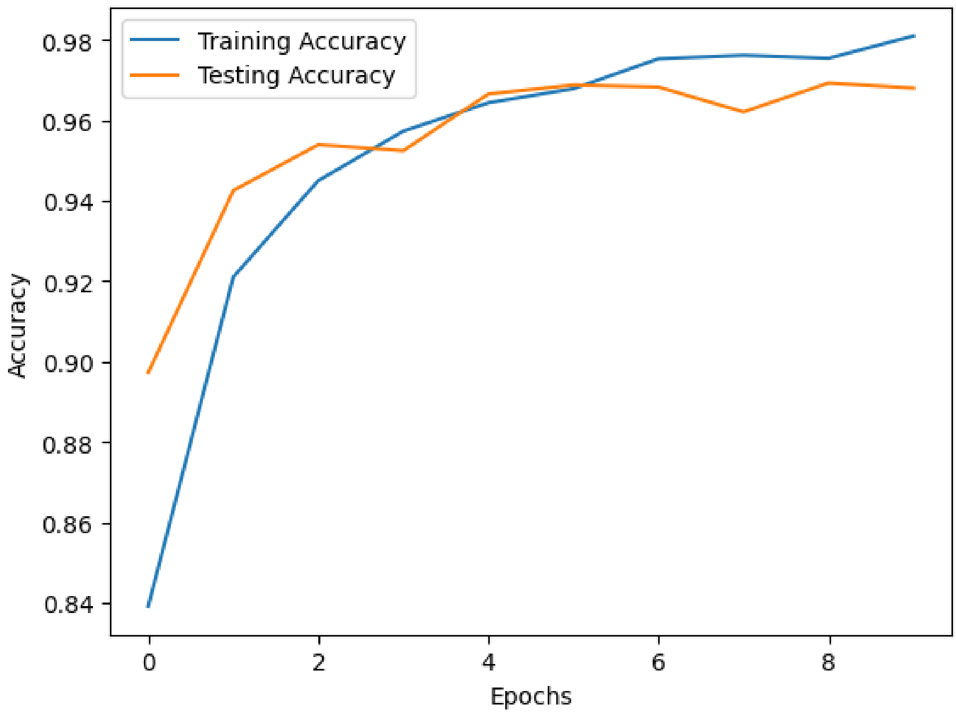


Fig 7.12 Training Accuracy V/S Testing Accuracy for Resnet50

3. VGG16

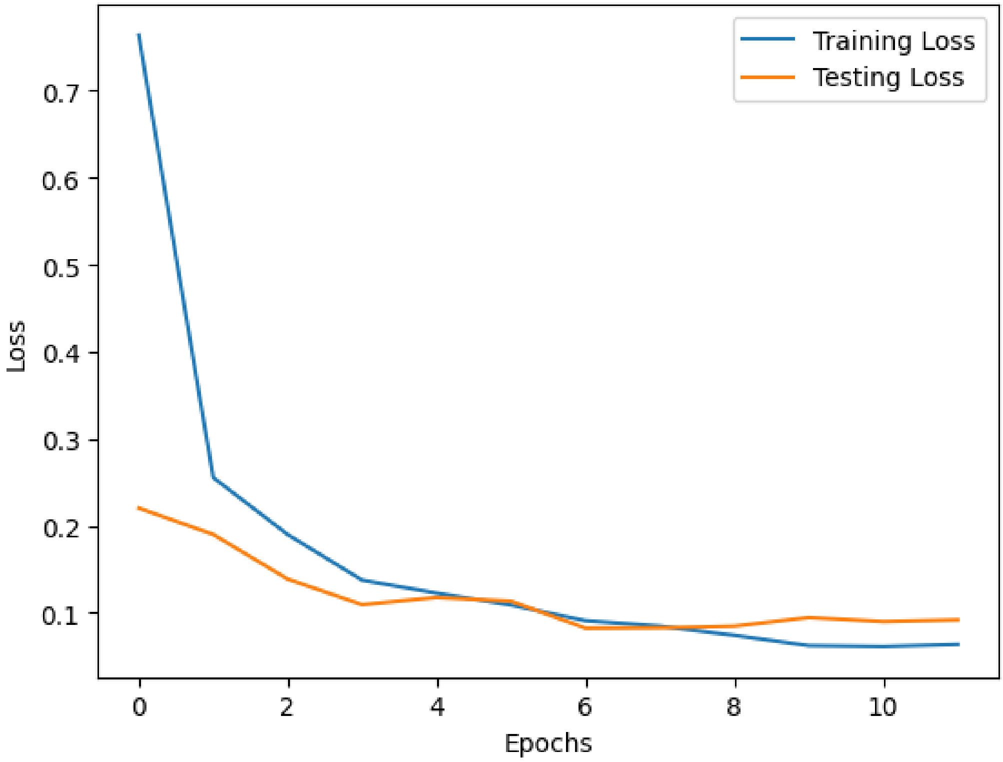


Fig 7.13 Training Loss V/S Testing Loss for VGG16

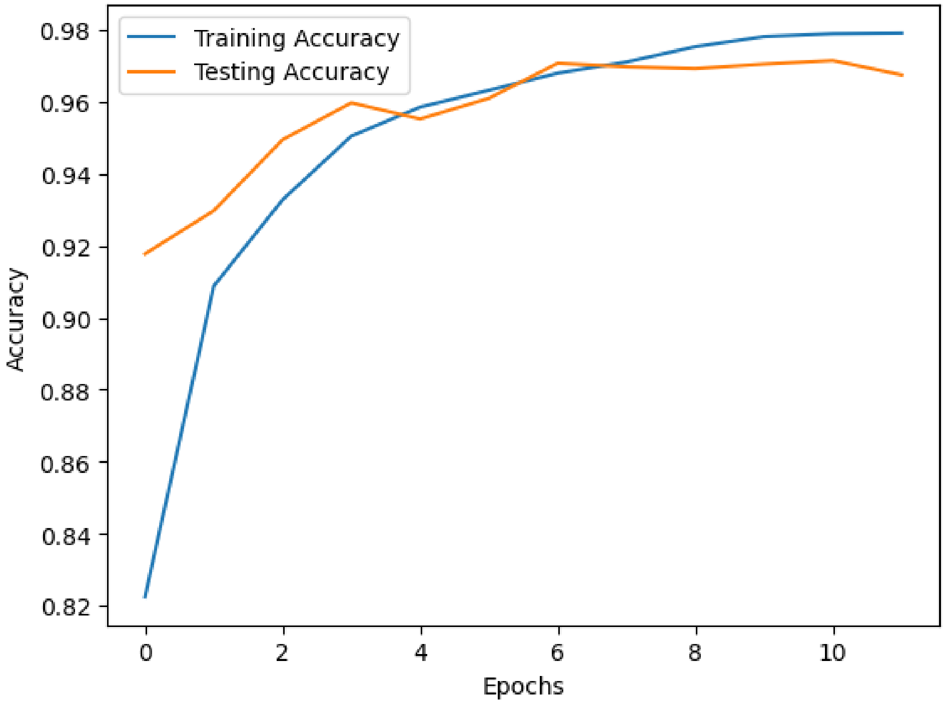


Fig 7.13 Training Accuracy V/S Testing Accuracy for VGG16

CHAPTER 8: CONTRIBUTION TO SUSTAINABLE DEVELOPMENT GOALS

8.1 INTRODUCTION TO SDGs

The United Nations' Sustainable Development Goals (SDGs) provide a shared global framework to end poverty, protect the planet, and ensure prosperity for all by 2030. In the context of this project, which focuses on brain tumor diagnosis using deep learning, the alignment with SDGs is significant. Early and accurate diagnosis of brain tumors can drastically improve treatment outcomes and save lives. With the increasing adoption of AI in healthcare, this project contributes to sustainable, scalable, and inclusive medical innovation, directly supporting the UN’s mission of equitable access to health services

8.2 MAPPING OF PROJECT TO RELEVANT SDG

SDG	GOAL TITLE	MAPPING OUR PROJECT
SDG 3	Good Health and Well-being	Our project enhances early detection of brain tumors through MRI image analysis using DL models like EfficientNetB3 (98% accuracy). This reduces the risk of late diagnosis and improves patient survival rates, especially in regions with limited access to specialists.

Table 3: Mapping Project to relevant SDG

8.3 PROJECT IMPACT ASSESMENT

The comparison of multiple deep learning models demonstrated the superior performance of EfficientNetB3, particularly in detecting the "brain glioma" and "no tumor" classes with precision and recall close to 1.00. This level of accuracy ensures trust in automated diagnosis. The project also explores underfitting issues in smaller datasets (observed in VGG16 and Xception), emphasizing the importance of data quality and model selection. Importantly, our work bridges the gap between AI innovation and healthcare impact, showcasing how a final-year engineering project can contribute to meaningful medical applications.

8.4 CHALLENGES AND FUTURE SCOPE

8.4.1 Challenges Faced

- Underfitting in VGG16 and Xception due to limited dataset size.
- Class imbalance slightly affecting performance for certain tumor types (e.g., brain meningioma).
- Dependency on high-quality annotated MRI images, which may not be available in all clinical settings.

8.4.2 Future Scope

- Enhancing model performance through data augmentation, transfer learning, and synthetic data generation.
- Integration with explainable AI techniques to ensure medical practitioners can interpret model decisions.
- Scaling the system as a cloud- or app-based diagnostic tool, targeting healthcare professionals in tier-2 and tier-3 cities.
- Collaborating with medical institutions for real-time deployment and clinical validation.

CHAPTER 9: CONCLUSION AND FUTURE SCOPE

9.1 CONCLUSION

The primary objective of this project was to compare the effectiveness of several pre-trained deep learning models—VGG16, ResNet50, EfficientNetB3, and Xception—for classifying brain MRI images into four categories: brain glioma, brain meningioma, brain pituitary tumor, and no tumor. Utilizing Google Colab and transfer learning techniques, we developed an image classification system capable of extracting meaningful features from medical images and providing accurate predictions.

After training and evaluating each model under identical conditions, the comparative results highlighted that EfficientNetB3 achieved the highest accuracy at **97.62%**, followed by VGG16 at **97.15%**, and ResNet50 at **96.93%**. These results demonstrate that EfficientNetB3, with its advanced architecture and parameter efficiency, is particularly well-suited for this task. The models were assessed using accuracy scores, classification reports, and confusion matrices, offering a well-rounded understanding of each model’s strengths.

The overall findings validate the applicability of deep learning in medical image analysis, showing its potential to aid healthcare professionals in diagnosing brain tumors more quickly and accurately.

9.2 FUTURE WORK

Although the models demonstrated strong performance, there are several areas that present opportunities for further improvement and research:

- Advanced Fine-Tuning: Unfreezing and retraining deeper layers of the pre-trained models could help boost their learning capabilities and improve classification performance.
- Dataset Expansion: Using larger and more diverse MRI datasets from different sources would allow the models to generalize better across varying imaging techniques and patient demographics.
- Cross-validation: Implementing k-fold cross-validation can provide more reliable performance estimates and reduce the risk of overfitting.
- Model Explainability: Introducing explainable AI tools such as Grad-CAM can help visualize decision-making regions in MRI scans, improving interpretability and clinical trust.
- Deployment Optimization: Techniques like model pruning, quantization, or conversion to lightweight formats can make deployment more feasible in low-resource or real-time environments.

This project not only highlights the comparative performance of widely used transfer learning models but also serves as a foundation for building more advanced and practical diagnostic systems for medical image classification.

9.3 APPLICATIONS

The brain tumor classification system based on deep learning and MRI imaging has wide-ranging applications in neurology, radiology, and AI-driven healthcare solutions. By leveraging advanced convolutional neural networks, this system enhances early tumor detection, diagnostic precision, and treatment planning, contributing to improved patient care and clinical outcomes.

Early Brain Tumor Detection & Screening[1]

- Facilitates early identification of brain tumors, enabling timely intervention and improved patient outcomes.
- Can be integrated into routine MRI screening workflows to flag suspicious cases for further review by radiologists.
- Supports preventive diagnostics in high-risk populations, including genetic predispositions to brain tumors.

AI-Assisted Radiology

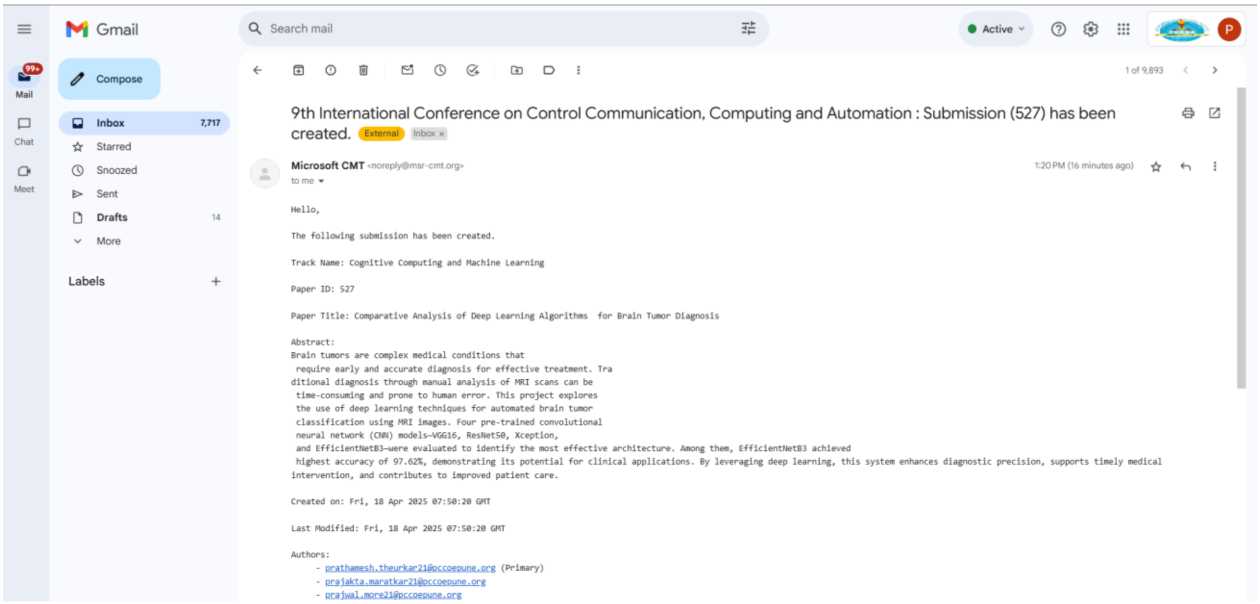
- Acts as a decision support system for radiologists by offering real-time predictions with high accuracy (up to 97.62% with EfficientNetB3).
- Reduces workload and minimizes diagnostic errors in busy clinical settings.
- Enhances objectivity in diagnosis by providing consistent and reproducible results across large datasets.

Clinical Integration & Hospital Use

- Can be deployed in hospital information systems or radiology PACS for automated tumor classification alongside MRI scan results.

- Enables triaging of critical cases, ensuring faster medical attention for patients with high-grade tumor.

APPENDIX A



Gmail
Compose
Inbox 7,717
Starred
Snoozed
Sent
Drafts 14
More
Labels
Mail
Chat
Meet
Search mail
Active
1 of 9,893
9th International Conference on Control Communication, Computing and Automation : Submission (527) has been created.
External
Inbox
Microsoft CMT <noreply@msr-cmt.org>
to me
1:20 PM (16 minutes ago)
Hello,
The following submission has been created.
Track Name: Cognitive Computing and Machine Learning
Paper ID: 527
Paper Title: Comparative Analysis of Deep Learning Algorithms for Brain Tumor Diagnosis
Abstract:
Brain tumors are complex medical conditions that
require early and accurate diagnosis for effective treatment. Tra
ditional diagnosis through manual analysis of MRI scans can be
time-consuming and prone to human error. This project explores
the use of deep learning techniques for automated brain tumor
classification using MRI images. Four pre-trained convolutional
neural network (CNN) models—VGG16, ResNet50, Xception,
and EfficientNetB3—were evaluated to identify the most effective architecture. Among them, EfficientNetB3 achieved
highest accuracy of 97.62%, demonstrating its potential for clinical applications. By leveraging deep learning, this system enhances diagnostic precision, supports timely medical
intervention, and contributes to improved patient care.
Created on: Fri, 18 Apr 2025 07:50:20 GMT
Last Modified: Fri, 18 Apr 2025 07:50:20 GMT
Authors:
- prathamesh.theurkar21@pccoepune.org (Primary)
- prajakta.maratkar21@pccoepune.org
- prajwal.more21@pccoepune.org

Submission Summary

Conference Name
9th International Conference on Control Communication, Computing and Automation

Track Name
Cognitive Computing and Machine Learning

Paper ID
527

Paper Title
Comparative Analysis of Deep Learning Algorithms for Brain Tumor Diagnosis

Abstract
Brain tumors are complex medical conditions that require early and accurate diagnosis for effective treatment. Tra ditional diagnosis through manual analysis of MRI scans can be time-consuming and prone to human error. This project explores the use of deep learning techniques for automated brain tumor classification using MRI images. Four pre-trained convolutional neural network (CNN) models—VGG16, ResNet50, Xception, and EfficientNetB3—were evaluated to identify the most effective architecture. Among them, EfficientNetB3 achieved highest accuracy of 97.62%, demonstrating its potential for clinical applications. By leveraging deep learning, this system enhances diagnostic precision, supports timely medical intervention, and contributes to improved patient care.

Created
4/18/2025, 1:20:20 PM

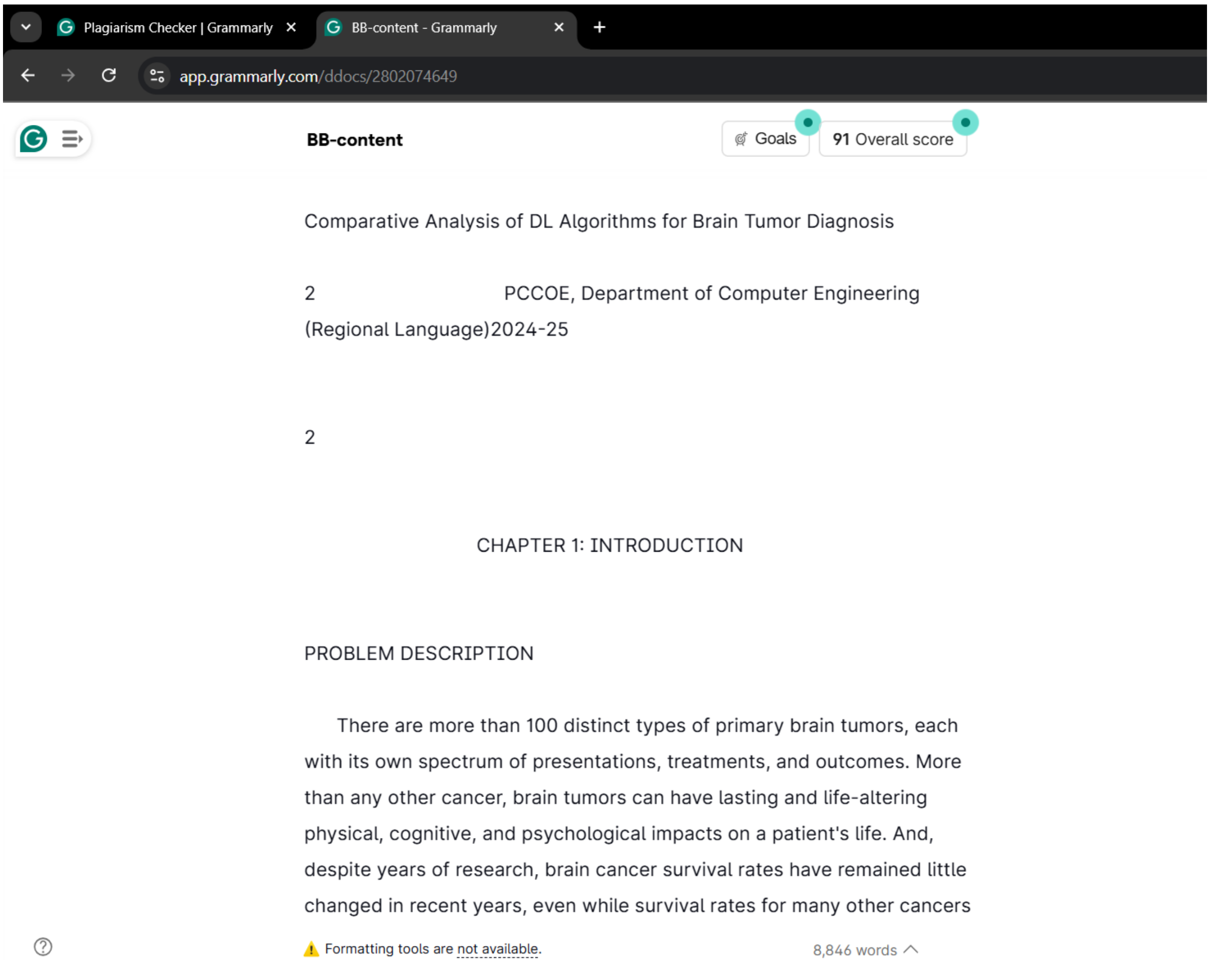
Last Modified
4/18/2025, 1:20:20 PM

Authors
Prathamesh Theurkar (Pimpri Chinchwad College of Engineering, Pune)
<prathamesh.theurkar21@pccoepune.org>
Prajakta Maratkar (Pimpri Chinchwad College of Engineering, Pune)
<prajakta.maratkar21@pccoepune.org>
Prajwal More (Pimpri Chinchwad College of Engineering, Pune)
<prajwal.more21@pccoepune.org>
Shubham Sangale (Pimpri Chinchwad College of Engineering, Pune)
<shubham.sangale21@pccoepune.org>

Ganesh Deshmukh (Pimpri Chinchwad College of Engineering, Pune)
<ganesh.deshmukh@pccoepune.org>

Submission Files
Brain Tumor Classification Research Paper.pdf (372.1 Kb, 4/18/2025, 1:19:57 PM)

APPENDIX B


Plagiarism Checker | Grammarly
BB-content - Grammarly
app.grammarly.com/ddocs/2802074649
BB-content
Goals
91 Overall score
Comparative Analysis of DL Algorithms for Brain Tumor Diagnosis
2 PCCOE, Department of Computer Engineering
(Regional Language)2024-25
2
CHAPTER 1: INTRODUCTION
PROBLEM DESCRIPTION
There are more than 100 distinct types of primary brain tumors, each with its own spectrum of presentations, treatments, and outcomes. More than any other cancer, brain tumors can have lasting and life-altering physical, cognitive, and psychological impacts on a patient's life. And, despite years of research, brain cancer survival rates have remained little changed in recent years, even while survival rates for many other cancers
Formatting tools are not available.
8,846 words

REFERENCES

[1] M. J. Lakshmi and S. N. Rao, “Brain tumor magnetic resonance image classification: A deep learning approach,” Soft Comput., vol. 26, no. 13, pp. 6245–6253, Jul. 2022, doi: 10.1007/s00500-022-07163-z.

[2] Y. Jiang, J. Hou, X. Xiao, and H. Deng, “A brain tumor segmentation new method based on statistical thresholding and multiscale CNN,”Intell. Comput. Methodologies, vol. 2, no. 3, pp. 235–245, 2019, doi:10.1007/978-3-319-95957-3 26.

[3] D. Liu, D. Zhang, Y. Song, F. Zhang, L. J. O’Donnell, and W. Cai,”3D large kernel anisotropic network for brain tumor segmentation” in Proc.Int. Conf. Neural Inf. Process. Cham, Switzerland: Springer, 2018 pp.444–454, doi: 10.1007/978-3-030-04239-4 40.

[4] M. W. Nadeem, M. A. A. Ghamdi, M. Hussain, M. A. Khan, K. M.Khan, S. H. Almotiri, and S. A. Butt, “Brain tumor analysis empowered with deep learning: A review, taxonomy, and future challenges,” Brain Sci., vol. 10, no. 2, pp. 118–151, 2020, doi: 10.3390/brainsci10020118.

[5] Y. Bhanothu, A. Kamalakannan, and G. Rajamanickam ”Detection and classification of brain tumor in MRI images using deep convolutional network” in Proc. 6th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS), Mar. 2020, pp. 248–252, doi: 10.1109/ICACCS48705.2020.9074375.

[6] N. Abiwinanda, M. Hanif, S. T. Hesaputra, A. Handayani, and T.R. Mengko, “Brain tumor classification using convolutional neural network,” in Proc. World Congr. Med. Phys. Biomed. Eng., vol. 68,2018, pp. 183–189, doi: 10.1007/978-981-10-9035-6 33.

[7] C. N. Ladefoged, L. Marner, A. Hindsholm, I. Law, L. Højgaard, and F.L. Andersen ”Deep learning-based attenuation correction of PET/MRI in pediatric brain tumor patients: Evaluation in a clinical setting” Frontiers Neurosci, vol. 2, p. 1005, Jan. 2018, doi: 10.3389/fnins.2018.01005.

[8] H. Fabelo, M. Halicek, S. Ortega, M. Shahedi, A. Szolna, J. Pi˜neiro,C. Sosa, A. O’Shanahan, S. Bisshopp, C. Espino, M. M´arquez, M. Hern´andez, D. Carrera, J. Morera, G. Callico, R. Sarmiento, andB. Fei ”Deep learning-based framework for in vivo identification of glioblastoma tumor using hyperspectral images of human brain” Sensors, vol. 19, no. 4, p. 920, Feb. 2019, doi: 10.3390/s19040920.

[9] Y. Li and L. Shen ”Deep learning based multimodal brain tumor diagnosis” in Proc. Int. MICCAI Brainlesion Workshop, vol. 10670, 2017, pp. 149–158, doi: 10.1007/978-3-319-75238-9 13.

[10] L. Chato and S. Latifi ”Machine learning and deep learning techniques to predict overall survival of brain tumor patients using MRI images” in Proc. IEEE 17th Int. Conf. Bioinf. Bioengineering (BIBE), Oct. 2017,pp.9–14, doi: 10.1109/BIBE.2017.00-86

[11] C. Zhou, S. Chen, C. Ding, and D. Tao, “Learning contextual and attentive information for brain tumor segmentation,” in Proc. Int. MICCAI Brainlesion Workshop, vol. 4798, 2018, pp. 497–507, doi: 10.1007/978-3-030-11726-9 44.

[12] G. Kim ”Brain tumor segmentation using deep fully convolutional neural networks” in Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries (Lecture Notes in Computer Science), vol. 10670, A. Crimi, S. Bakas, H. Kuijf, B. Menze, and M. Reyes, Eds. Cham, Switzerland: Springer, 2018, doi: 10.1007/978-3-319-75238-9 30

[13] P. Afshar, A. Mohammadi, and K. N. Plataniotis ”Brain tumor type classification via capsule networks” in Proc. 25th IEEE Int.Conf. Image Process. (ICIP), Oct. 2018, pp. 3129–3133, doi:10.1109/ICIP.2018.8451379.

[14] P. D. Chang ”Fully convolutional deep residual neural networks for brain tumor segmentation” in Proc. Int. Workshop Brainlesion, 2016, pp. 108–118, doi: 10.1007/978-3-319-55524-9 11.

[15] F. Isensee et al ”Brain tumor segmentation using large receptive field deep convolutional neural networks” in Bildverarbeitung f ̈ur die Medizin 2017 (Informatik aktuell), K. H. Maier-Hein, geb. Fritzsche, T. M. Deserno, geb. Lehmann, H. Handels, and T. Tolxdorff, Eds. Berlin, Germany: Springer Vieweg, 2017, doi: 10.1007/978-3-662-54345- 0 24.

[16] S. Kumar, A. Negi, and J. N. Singh ”Semantic segmentation using deep learning for brain tumor MRI via fully convolution neural networks”in Information and Communication Technology for Intelligent Systems.Singapore: Springer, 2019, pp. 11–19, doi: 10.1007/978-981-13-1742-2 2

[17] S. Sajid, S. Hussain, and A. Sarwar ”Brain tumor detection and segmentation in MR images using deep learning” Arabian J. Sci. Eng., vol. 44, no. 11, pp. 9249–9261, Nov. 2019, doi: 10.1007/s13369-019-03967-8.

[18] Y. Hu and Y. Xia ”3D deep neural network-based brain tumor segmentation using multimodality magnetic resonance sequences” in Proc. Int.MICCAI Brainlesion Workshop, 2017, pp. 423–434, doi: 10.1007/9783-319-75238-9 36.

[19] A. S. Akbar, C. Fatichah, and N. Suciati ”Single level UNet3D with multipath residual attention block for brain tumor segmentation” J. King Saud Univ. Comput. Inf. Sci., vol. 34, no. 6, pp. 3247–3258, Jun. 2022, doi: 10.1016/j.jksuci.2022.03.022.

[20] S. Shiraskar and D. Rizk, ”Advancements in Brain Tumor Detection:Utilizing Xception Enhanced Tumor Identifier Network,” 2024 2nd International Conference on Artificial Intelligence, Blockchain, and Internet of Things (AIBThings), Mt Pleasant, MI, USA, 2024, pp. 1-5, doi: 10.1109/AIBThings63359.2024.10863332.

[21] J. K. Periasamy, B. S and J. P, ”Comparison of VGG-19 and RESNET-50 Algorithms in Brain Tumor Detection,” 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), Lonavla, India,2023, pp. 1-5, doi: 10.1109/I2CT57861.2023.10126451.

[22] S. K. Chandra and M. Kumar Bajpai, ”EFFECTIVE ALGORITHM FOR BENIGN BRAIN TUMOR DETECTION USING FRACTIONAL CALCULUS,” TENCON 2018 - 2018 IEEE Region 10 Conference, Jeju, Korea (South), 2018, pp. 2408-2413, doi:10.1109/TEN-CON.2018.8650163.

[23] N. M. Dipu, S. A. Shohan and K. M. A. Salam, ”Deep Learning Based Brain Tumor Detection and Classification,” 2021 International Conference on Intelligent Technologies (CONIT), Hubli, India, 2021,pp. 1-6, doi: 10.1109/CONIT51480.2021.9498384.

[24] Srikanth B., Suryanarayana S.V. Multi-Class classification of brain tumor images using data augmentation with deep neural network. Mater. TodayProc. 2021

[25] ¨Oks¨uz C., Urhan O., G¨ull¨u M.K. Brain tumor classification using the fused features extracted from expanded tumor region. Biomed. Signal Process. Control. 2022

[26] Kadry S., Nam Y., Rauf H.T., Rajinikanth V., Lawal I.A. Automated Detection of Brain Abnormality using Deep-Learning-Scheme: A Study; Proceedings of the 2021 Seventh International conference on Bio Signals, Images, and Instrumentation (ICBSII); Chennai, India. 25–27,March 2021.

[27] Irmak E. Multi-Classification of Brain Tumor MRI Images Using Deep Convolutional Neural Network with Fully Optimized Framework. Iran.J. Sci. Technol. Trans. Electr. Eng. 2021

[28] Bjoern H. M et al. The Multi modal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE Transactions on Medical Imaging.

[29] Louis D.N., Perry A., Wesseling P., Brat D.J., Cree I.A., Branger D.F.,Hawkins C., Ng H.K., Pfister S.M., Reifenberger G., et al. The 2021WHO classification of tumors of the central nervous system: A summary.Neuro-Oncology. 2021

[30] Zaccagna F., Riemer F., Priest A.N., McLean M.A., Allinson K., Grist J.T., Dragos C., Matys T., Gillard J.H., Watts C., et al. Non-invasive assessment of glioma microstructure using VERDICT MRI: Correlation with histology. Eur. Radiol. 2019

[31] Mzoughi H., Njeh I., Wali A., Slima M.B., BenHamida A., Mhiri C.,Mahfoudhe K.B. Deep multi-scale 3D convolutional neural network (CNN) for MRI gliomas brain tumor classification. J. Digit. Imaging.