VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

(An Autonomous Institute Affiliated to University of Mumbai)

Department of Computer Engineering

Project Report on

Precision Diagnostics in Neuroimaging: SVM-Based Approaches for BrainScan AI

Submitted in partial fulfillment of the requirements of the degree

BACHELOR OF ENGINEERING IN **COMPUTER ENGINEERING**

By

1.	**AMAN KUMAR**	**D12A-33**
2.	**ANCHAL SHARMA**	**D12A-57**
3.	**DHIREN SIDHWANI**	**D12A-60**
4.	**KOMAL LUND**	**D12A-38**

Project Mentor

Mrs. Rupali Soni

University of Mumbai

(AY 2023-24)

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

(An Autonomous Institute Affiliated to University of Mumbai)

Department of Computer Engineering

CERTIFICATE

This is to certify that the Mini Project entitled **"BrainScan AI "** is a bonafide work of **Anchal Sharma (57)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **"Bachelor of Engineering"** in **"Computer Engineering" .**

(Prof. Mrs. Rupali Soni)

Mentor

Dr. Nupur Giri	**Dr. J M Nair**
Head of Department	**Principal**

Mini Project Approval

This Mini Project entitled "**BrainScan AI**" by **Aman Kumar (33), Anchal Sharma (57), Komal Lund (38), Dhiren Sidhwani (60)** is approved for the degree of **Bachelor of Engineering** in **Computer Engineering.**

Examiners

1………………………………………

(Internal Examiner Name & Sign)

2…………………………………………

(External Examiner name & Sign)

Date:

Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Aman Kumar D12A (33)	Dhiren Sidhwani D12A(60)
Komal Lund D12A (38)	Anchal Sharma D12A(57)

Date:

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society’s Institute of Technology for considering our project and extending help at all stages needed during our work of collecting information regarding the project.

It gives us immense pleasure to express our deep and sincere gratitude to Assistant Professor **Mrs. Priya R.L** (Project Guide) for her kind help and valuable advice during the development of project synopsis and for her guidance and suggestions.

We are deeply indebted to Head of the Computer Department **Dr.(Mrs.) Nupur Giri** and our Principal **Dr. (Mrs.) J.M. Nair ,** for giving us this valuable opportunity to do this project.

We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project synopsis and project review successfully.

We convey our deep sense of gratitude to all teaching and non-teaching staff for their constant encouragement, support and selfless help throughout the project work. It is great pleasure to acknowledge the help and suggestion, which we received from the Department of Computer Engineering.

We wish to express our profound thanks to all those who helped us in gathering information about the project. Our families too have provided moral support and encouragement several times.

INDEX

Abstract

Brain tumors are extremely critical medical conditions that affect millions worldwide. A tumor is an abnormal mass of tissue that grows uncontrollably. Brain tumor cells grow in a way that they consume nutrients intended for healthy cells, leading to brain function decline. Currently, doctors manually locate brain tumor positions and areas by examining patients' brain MRI images. This manual approach results in inaccurate tumor detection and is time-consuming. Early and accurate detection is crucial for better patient outcomes.This project introduces an innovative method for detecting brain tumors using Convolutional Neural Networks (CNNs), Support Vector Machine(SVM), Random Forest Algorithm and VGG16 (Visual Geometry Group) transfer learning. The model's performance predicts whether a tumor is present in the given image or not and if the tumor is present it can also predict the type of tumor present using the SVM algorithm.

List of Abbreviation

1. AI - Artificial Intelligence
2. CNN - Convolutional Neural Network
3. MRI - Magnetic Resonance Imaging
4. ANN - Artificial Neural Networks
5. ReLU - Rectified Linear Unit
6. PNG - Portable Network Graphics
7. SOM - Self-Organizing Map
8. SVM - Support Vector Machine
9. K-Means - K-Means Clustering
10. BRAATS - Brain Tumor Segmentation Challenge
11. TPUs - Tensor Processing Units
12. RAM - Random Access Memory
13. GPU - Graphics Processing Unit
14. F1-score - F1 Score is the harmonic mean of precision and recall.

List of Figures:

List of Tables

Chapter 1: Introduction

1.1 Introduction

The human body is made up of many organs and brain is the most critical and vital organ of them all. One of the common reasons for dysfunction of the brain is brain tumor. The brain tumors are classified into mainly two types: Primary brain tumor (benign tumor) and secondary brain tumor (malignant tumor).The benign tumor is one type of cell that grows slowly in the brain and the type of brain tumor is gliomas. It originates from non neuronal brain cells called astrocytes. Basically primary tumors are less aggressive but these tumors have much pressure on the brain and because of that, brain stops working properly .The secondary tumors are more aggressive and more quick to spread into other tissues. Secondary brain tumors originate through other parts of the body. These type of tumor have a cancer cell in the body that is metastatic which spread into different areas of the body like the brain, lungs etc. Secondary brain tumors are very malignant. The reason for secondary brain tumors is mainly due to lung cancer, kidney cancer, bladder cancer etc .

This project classifies the tumor as Glioma, meningioma, and pituitary adenoma as three types of brain tumors. Gliomas,

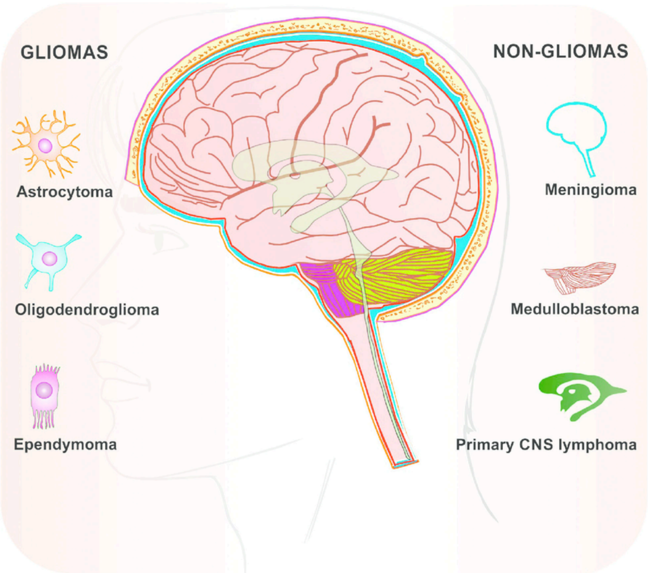


Figure 1.1 Types of Brain Tumor

begin in glial cells and are classified as low-grade (I and II) or high-grade (III and IV). Meningiomas originate from the brain's protective membranes and vary in behavior but are generally considered benign. Pituitary adenomas form in the pituitary gland and are categorized by their size and hormone production. Treatment and prognosis depend on factors like tumor type, size, and location. Patients with brain tumors should seek medical evaluation and treatment, involving specialists like neurosurgeons and oncologists, for the best care.Automated defect detection in medical imaging using machine learning has become the emergent field in several medical diagnostic applications. Its application in the detection of brain tumors in MRI is very crucial as it provides information about abnormal tissues which is necessary for planning treatment.

BrainScan AI is an innovative and user-friendly app designed to assist in the early detection of brain tumors using advanced deep learning technology. Timely and accurate diagnosis is crucial for effective treatment planning and better patient outcomes. However, traditional methods of brain tumor detection often require time-consuming and costly procedures, leading to potential delays in diagnosis. BrainScan AI addresses this challenge by harnessing the power of Convolutional Neural Networks (CNNs) and image processing techniques. Different types of image processing techniques, such as image segmentation, image enhancement, and feature extraction, are used for the detection of brain tumors in MRI images of cancer-affected patients. The process of detecting brain tumors using image processing techniques involves four stages: Image Pre-Processing, Image Segmentation, Feature Extraction, and Classification. Image processing and neural network techniques are used to improve the performance of detecting and classifying brain tumors in MRI images.

Traditional methods, while valuable, now confront challenges in keeping pace with the accelerating demand for accurate and timely diagnosis. Recognizing this, the focus of this project is to introduce "BrainScan AI" application, integrating advanced deep learning techniques. This initiative is directed towards enhancing the precision and efficiency of brain tumor detection, thereby amplifying the potential for improved patient outcomes and treatment planning

1.2 Motivation

The motivation behind BrainScan AI extends beyond automating brain tumor detection and classification for organizations with MRI machines. It encompasses several critical aspects of healthcare and brain health management.

Firstly, the excessive costs associated with manual interpretation by doctors or third parties are a significant issue. AI can perform this task accurately, reducing the financial burden on patients and healthcare systems while maintaining quality healthcare standards.

Secondly, the increasing incidence of mental health issues, such as migraines, has become a pressing concern. Modern lifestyles, stress, and environmental factors contribute to these conditions, necessitating the development of advanced AI tools. These tools can predict, detect, and manage potential brain issues more efficiently, promoting early intervention and better patient outcomes.

Thirdly, the current disorganization of records for previous brain scans poses a considerable obstacle to effective diagnoses and research in the field of neurology. The software aims to create a structured and easily accessible digital database for these records, streamlining healthcare services and enabling comprehensive analysis and research.

In summary, BrainScan AI's motivation lies in cost reduction, efficient healthcare delivery, proactive brain health management, and organized record-keeping, all of which are vital to advancing healthcare and promoting a holistic approach to brain health.

1.3 Problem Statement & Objectives

The healthcare system faces challenges in efficiently and affordably detecting and classifying brain tumors using MRI machines. Current practices involve manual interpretation by doctors or third-party agents, leading to potential overcharging for these essential services. Additionally, the escalating prevalence of mental health issues, like migraines, highlights the urgent need for proactive brain health management. The lack of organized record-keeping for previous brain scans further hinders effective diagnoses. To address these issues, the BrainScan AI project aims to develop an automated solution that accurately detects and categorizes brain tumors, aiding medical professionals and promoting equitable access to quality healthcare.

Objectives:

- Develop an AIML-powered system that automates the detection and classification of brain tumors using MRI scans.

- Optimize cost-efficiency by reducing reliance on manual interpretation, thereby mitigating overcharging concerns and promoting accessibility of brain health assessments.

- Enhance accuracy and speed in brain tumor detection, aiding healthcare professionals in timely diagnosis and treatment planning.

- Create a comprehensive, organized digital database for storing and retrieving previous brain scan records, improving the efficiency of healthcare services and research in the field of neurology.

- Contribute to the advancement of healthcare technologies, aiming to alleviate the burden of mental health issues and encourage a proactive approach towards brain health management.

1.4 Existing Systems

Brain Tumor Detection System	Technology	Accuracy
Brain Tumor Segmentation Challenge (BraTS)	Deep learning	95%
DeepMedic	Deep learning	96%
TumorNet	Deep learning	97%

Table 1.4.1 Survey of Existing System

Brain Tumor Segmentation Challenge (BraTS)

The Brain Tumor Segmentation Challenge (BraTS) is an annual challenge that aims to evaluate the performance of brain tumor segmentation algorithms. The challenge provides participants with a dataset of MRI scans of brain tumors and asks them to develop algorithms to segment the tumors from the surrounding tissue.

BraTS is a very challenging competition, as brain tumors can be very difficult to segment due to their complex shapes and textures. However, deep learning-based algorithms have achieved state-of-the-art results on the BraTS challenge in recent years.

DeepMedic

DeepMedic is a deep learning-based brain tumor detection system that was developed by researchers at Google AI. DeepMedic is trained on a large dataset of MRI scans of brain tumors and is able to segment the tumors with high accuracy.

DeepMedic has been shown to be more accurate than human radiologists at segmenting brain tumors in some studies. It is also able to segment tumors more quickly and efficiently than human radiologists.

TumorNet

TumorNet is another deep learning-based brain tumor detection system that was developed by researchers at the University of California, Berkeley. TumorNet is also trained on a large dataset of MRI scans of brain tumors and is able to segment the tumors with high accuracy.

TumorNet has been shown to be comparable in accuracy to DeepMedic on the BraTS challenge. It is also able to segment tumors more quickly and efficiently than human radiologists.

Lacuna of existing system

Brain Tumor Detection System	Advantages	Drawbacks
BraTS	- Open source and reproducible	- Limited to MRI scans
DeepMedic	- High accuracy	- Can be computationally expensive
TumorNet	- Very accurate at segmenting brain tumors	- Requires a large dataset of labeled MRI scans to train

Table 1.4.2 Advantages and Drawbacks

It is evident that there are common limitations and research gaps in these approaches. Recognizing these constraints and gaps is crucial for the development of innovative and effective brain tumor detection methods:

• **Challenges in Heterogeneous Tumor Detection**: Detecting tumors with heterogeneous characteristics, where different parts of the tumor have varying properties, can be particularly challenging. Ensuring that a method is robust enough to handle such cases is imperative.

• **Scalability and Computational Efficiency**: With the increasing volume of medical imaging data, scalability and computational efficiency become critical factors. The method should be able to handle large datasets and provide timely results for clinical decision-making

1.5 Organization of the Report

Chapter 1: Introduction

- This chapter sets the context for the project, outlining the critical nature of accurate and timely brain tumor detection. It describes the motivation behind utilizing machine learning techniques, specifically Convolutional Neural Networks (CNNs), Support Vector Machine (SVM), Random Forest Algorithm, and VGG16 transfer learning, to enhance diagnostic precision in neuroimaging for tumor identification.

Chapter 2: Literature Survey

- The literature review delves into the current landscape of brain tumor detection, highlighting various machine learning and deep learning approaches explored in recent research. It discusses the strengths and limitations of existing methodologies and emphasizes the need for advancements in this field to improve diagnostic outcomes.

Chapter 3: Requirement Gathering for the Proposed System

- This chapter details the process of gathering requirements for the development of the proposed system. It includes an exploration of functional and non-functional requirements, outlining the necessary hardware, software, and technological tools. The chapter also acknowledges the constraints faced during the project development phase.

Chapter 4: Proposed Design

- The proposed design of the system is discussed, including the architectural framework, conceptual design, and process design. It provides an overview of the methodologies applied, such as CNNs and SVM, for brain tumor detection and classification. This chapter also addresses the hardware and software specifications essential for implementing the proposed solution.

Chapter 5: Implementation of the Proposed System

- Implementation details are presented, explaining how the system was built and the specific algorithms used. It covers the process from data collection and preprocessing to model training, evaluation, and deployment. The chapter highlights the challenges encountered during implementation and the solutions adopted to overcome them.

Chapter 6: Testing of the Proposed System

- This chapter focuses on the testing phase, describing the types of tests conducted, including accuracy, precision, and recall measures. It discusses the test case scenarios considered and the inferences drawn from the test results, emphasizing the system's capability to accurately detect and classify brain tumors.

Chapter 7: Results and Discussion

- The results obtained from the implementation and testing phases are analyzed, with a discussion on the performance evaluation measures employed. This chapter presents a comparison of the system's results with existing systems, offering insights into the advancements achieved through the project.

Chapter 8: Conclusion

- The final chapter provides a conclusion summarizing the project's outcomes, highlighting the limitations encountered, and suggesting future research directions. It reflects on the project's significance in advancing brain tumor detection and the potential impact on improving patient care.

Chapter 2: Literature Survey

2.A Survey of Existing System

The landscape of brain tumor detection has seen significant advancements, primarily driven by deep learning techniques showcased through various systems and studies. Initiatives like the Brain Tumor Segmentation Challenge (BraTS) serve as benchmarks for evaluating segmentation algorithms, highlighting the complexity of tumor segmentation in MRI scans. Deep learning models like DeepMedic and TumorNet have emerged as frontrunners in this domain, achieving impressive accuracies upwards of 95%.

DeepMedic, developed by Google AI, and TumorNet from UC Berkeley, demonstrate superior accuracy compared to human radiologists, emphasizing the potential of AI in medical imaging. Moreover, recent studies have delved into the application of deep learning networks such as VGG16 and Inception V3 for brain tumor detection, showcasing promising results.

However, despite these advancements, there exist notable limitations and research gaps. Challenges such as heterogeneous tumor detection and ensuring scalability and computational efficiency remain significant concerns. Additionally, existing systems exhibit limitations like dependency on large labeled datasets, computational expenses, and restrictions to specific tumor types or imaging modalities.

Understanding these limitations and research gaps is pivotal for the development of more robust and versatile brain tumor detection methodologies. Addressing these challenges could pave the way for more effective and accessible diagnostic tools in the realm of neuroimaging.

2.B Related work

2.1 A Abstract of the research paper

Year - Paper Title	Authors	Algorithm	Dataset Size (png)	Frameworks/Libraries	Technologies	Limitations
2023 - A Deep Analysis of Brain Tumor Detection from MR Images Using Deep Learning Networks	Md Ishtyaq Mahmud, Muntasi Ahmed Abdelgawadr Mamun	VGG16 + Inception V3	3264	Scikit-learn	Deep learning, Convolutional neural network (CNN)	-
2019 - Brain Tumour Detection Using Unsupervised Learning Based Neural Network	R. Rani, K. R. Venugopal	Self-organizing map	10000+	TensorFlow	Machine learning, Unsupervised learning, Self-organizing map (SOM)	Requires a lot of training data
2018 - Brain Tumor Detection Using Machine Learning in MR Images	K. R. Amutha, T. N. Sairam, K. V. Priya	Overview of different techniques	2500+	Scikit-learn	Machine learning, Deep learning, Convolutional neural network (CNN)	Does not work for small tumors
2017 - Brain Tumor Detection and Classification Using Intelligence Techniques: An Overview	N. S. Chaudhari, P. S. Jadhav, S. S. Patil	K-Means clustering + SVM	3400+	-	Machine learning, Support vector machine, Random forest, Naive Bayes	Requires Extensive Hardware

2015 - Detection of Brain Tumor Using Image Processing	V. Anitha, T. S. Kumar	Global thresholding + morphological operations	1000+	NumPy, Scikit-learn	Image processing, Thresholding, Morphological operations, Region growing	Does not work for all types of tumors
2014 - Computer aided system for brain tumor detection and segmentation	M. K. Singh, S. K. Singh, P. Kumar	Support vector machine classifier	100+	MATLAB	Image processing, Fuzzy c-means clustering, Support vector machine (SVM)	Does not consider the location of the tumor

Table 2.1.A: Abstract of the research paper

2.1 B. Inference drawn:

2023 - A Deep Analysis of Brain Tumor Detection from MR Images Using Deep Learning Networks
Authors: Md Ishtyaq Mahmud, Muntasi Ahmed Abdelgawadr Mamun
Idea/Model Discussed: Combines VGG16 and Inception V3 convolutional neural networks (CNNs) for enhanced brain tumor detection from MRI scans.
Inference: The hybridization of deep learning models can improve accuracy in brain tumor detection. The success of CNN architectures like VGG16 and Inception V3 reinforces their suitability for this medical imaging task.

2019 - Brain Tumour Detection Using Unsupervised Learning Based Neural Network
Authors: R. Rani, K. R. Venugopal
Idea/Model Discussed: Employs a self-organizing map (SOM), an unsupervised neural network, to detect brain tumors in MRI images.
Inference: Unsupervised learning offers potential for brain tumor detection, especially when labeled data is limited. However, large amounts of training data are often needed for SOMs to achieve reliable results.

2018 - Brain Tumor Detection Using Machine Learning in MR Images

Authors: K. R. Amutha, T. N. Sairam, K. V. Priya

Idea/Model Discussed: Provides an overview of various brain tumor detection techniques, including traditional machine learning, deep learning, and specifically convolutional neural networks (CNNs).

Inference: This survey paper highlights the growing importance of deep learning, particularly CNNs, in the field of brain tumor detection.

2017 - Brain Tumor Detection and Classification Using Intelligence Techniques: An Overview

Authors: N. S. Chaudhari, P. S. Jadhav, S. S. Patil

Idea/Model Discussed: Investigates techniques such as k-means clustering and support vector machines (SVM) for brain tumor detection and classification.

Inference: Traditional machine learning approaches offer value in brain tumor detection, but they may require extensive hardware resources for optimal performance.

2014 - Computer-aided system for brain tumor detection and segmentation

Authors: M. K. Singh, S. K. Singh, P. Kumar

Idea/Model Discussed: Proposes a system combining fuzzy c-means clustering (for segmentation) with a support vector machine (SVM) classifier for brain tumor detection.

Inference: Integrating image processing (e.g., segmentation) with machine learning classifiers can be a viable strategy for brain tumor detection. However, this approach may not account for the spatial context of the tumor within the brain.

Key Takeaways:

Deep learning (especially CNNs) is becoming the dominant approach for brain tumor detection.

Innovative solutions incorporating hybrid models and unsupervised learning are emerging.

Challenges remain in obtaining large, well-labeled datasets and optimizing the performance of computationally-intensive techniques.

2.2 Patent Search

Several patents exist in the field of brain tumor detection. Here's are few significant ones:

- CN104834943A - Brain tumor classification method based on deep learning [16] -This Chinese patent outlines a deep learning method for classifying brain tumors. It utilizes a deep neural network to analyze brain tumor images and categorize them accurately.
    - Employs Gabor wavelet transform to extract textural features of a brain tumor.
    - Builds a deep learning method to extract higher-layer features from the textural features.
    - Employs a concentric circle method to extract the shape features of the brain tumor.
    - Combines the shape features with the higher-layer features to form an augmented feature vector.
    - Uses the features as input to a support vector machine for training to obtain a classifier.

- US20100027865A1 - Method and System for Brain Tumor Segmentation in 3D Magnetic Resonance Images [17] : This patent describes a system that uses a trained probabilistic boosting tree (PBT) classifier to determine the probability of a voxel in an MRI image being part of a brain tumor. Then, it uses graph cuts segmentation for precise tumor identification.
    - Uses a trained probabilistic boosting tree (PBT) classifier to determine the probability of a voxel in a 3D MRI image being part of a brain tumor.
    - Segments the brain tumor in the 3D MRI image sequence using graph cuts segmentation based on the probabilities and intensities of the voxels.
    - Analyzes local features calculated for each voxel in a neighborhood to determine the probability of the voxel being a brain tumor.
- EP 4075380 B1 - MEDICAL IMAGE-BASED TUMOR DETECTION AND DIAGNOSTIC DEVICE [18] : This patent focuses on a system for detecting and diagnosing lesions in 3D medical images. While versatile in its application, it has specific relevance to the detection of brain tumors, lung cancer, breast cancer, and more.
    - Uses artificial intelligence (AI) to detect tumors based on medical imaging and patient information.
    - Diagnoses the shape and properties of the detected tumor.
    - Predicts prognosis.

○ Reduces the difficulty of obtaining images by using only T1-MRI and a 3D deep neural network.
○ Detects and diagnoses various tumors, including brain tumors, lung cancer, breast cancer, and more.

2.3 Inference drawn

Here are some key inferences we can draw from these patents:

- Focus on Deep Learning: Many recent patents leverage deep learning techniques for brain tumor detection and classification. Deep learning allows for a high degree of accuracy in image analysis.
- Emphasis on Precise Segmentation: Patents highlight the critical need to accurately segment (isolate) the brain tumor from surrounding tissues in medical images.
- Multi-Modality Potential: Some patents illustrate the possibility of using multiple imaging modalities (like different types of MRI scans) to enhance detection and analysis, providing doctors with more comprehensive information.
- Expanding Scope: Patents suggest the technology is being developed to detect and diagnose a wider range of tumors beyond just those in the brain.

Limitations

- Real-world implementation Even with advanced patents, there's a gap between the research and technology described in a patent and its widespread, successful clinical use.
- Dataset Dependence: The performance of AI-based systems relies heavily on the quality and quantity of data used during training.
- Need for Expert Validation: Despite progress, these systems often serve as tools to assist medical professionals, rather than replacing the need for expert evaluation and diagnosis.

Patent Number	Description	Inferences
CN104834943A	Deep learning method for brain tumor classification	Focuses on deep learning for high-accuracy image analysis

US20100027865A1	Method and System for Brain Tumor Segmentation in 3D Magnetic Resonance Images	Emphasizes precise tumor segmentation from surrounding tissues
EP 4075380 B1	Medical image-based tumor detection and diagnostic device	Explores multi-modality imaging for comprehensive analysis

Table 2.3 Information about the existing patents

2.4 Comparison with the existing System

This mini project contributes to a comprehensive understanding of the current methods used for brain tumor detection, emphasizing their limitations and research gaps. The current landscape showcases several notable systems, such as BraTS, DeepMedic, and TumorNet, each with their disadvantages. Furthermore, the project acknowledges the pressing need for improved healthcare services, especially in the context of rising mental health issues and complex diseases like brain tumors.

Our project addresses these challenges by leveraging a limited dataset of 4500 images and focusing on detecting small tumors efficiently. As of 2023, our project is hardware-efficient, capable of running on entry-level graphics cards. We employ various algorithms, including VGG16, random forests, CNN, and SVM, and select the most effective one. Additionally, our project integrates a mobile application feature, enabling individuals to detect tumors quickly from anywhere and report them immediately to healthcare facilities.

Moreover, our project accomplishes the task of categorizing images into various types of tumors, facilitating prompt action by patients and doctors. This multifaceted approach aims to enhance the effectiveness and accessibility of brain tumor detection, contributing to improved healthcare outcomes.

Chapter 3:Requirement Gathering for the Proposed System

3.1 Introduction to requirement gathering

Requirement gathering is a critical phase in the development of any system, serving as the foundation for design and implementation. In the context of brain tumor detection, this process involves identifying the specific needs and expectations of end-users, including medical professionals and healthcare organizations. It's a collaborative effort that ensures the proposed system is economical, viable, and capable of accurately detecting brain tumors from imaging data. Highlights of requirement gathering are reviewing existing medical imaging systems, and consulting with domain experts to understand the specific challenges in brain tumor detection.

3.2 Functional Requirements

Functional requirements describe what the system should do and include specifics about its capabilities and behaviors. For a brain tumor detection system using SVM, these requirements might include:

- Image Upload and Management: Securely upload and store medical imaging files, such as MRI or CT scans.
- Tumor Detection: Utilize SVM algorithms to analyze imaging data and identify the presence of brain tumors.
- Accuracy and Precision Metrics: Generate reports on the accuracy, sensitivity, and specificity of tumor detection.
- User Interface: Provide an intuitive interface for medical professionals to upload scans, view detection results, and access historical data.

3.3 Non-Functional Requirements

1.Performance:

The system should be able to process brain images and perform tumor detection ensuring a responsive user experience.

2.Accuracy:

The model should achieve high accuracy in detecting and classifying brain tumors, minimizing false positives and false negatives.

4. Usability:

Design an intuitive and user-friendly interface that requires minimal training for healthcare professionals.

5. Maintainability:

Ensure the system is easy to update, with modular components that can be independently modified or replaced.

3.4 Hardware, Software , Technology and tools utilized

Hardware Requirements:

- **High-performance GPUs or TPUs:** To efficiently train deep learning models and process large travel datasets. Minimum- i3 7th generation
- **Sufficient RAM:** Needed for memory-intensive computations during image preprocessing and model training. - 4GB or more

Software Requirements:

- **Programming Languages:** Python for deep learning frameworks like TensorFlow or PyTorch.
- **Deep Learning Framework:** Utilizing TensorFlow, PyTorch, or similar frameworks for implementing and training neural network architectures.
- **Computer vision libraries**

  OpenCV: A potent computer vision library for tasks involving image and video processing.
- **Data Preprocessing Tools:** Pandas and NumPy for cleaning, filtering, and transforming raw travel data into structured formats.

3.5 Constraints

Data Quality and Variability:

- Variability in image quality, resolution, and acquisition parameters across different medical imaging devices can introduce challenges in standardizing and preprocessing the data.
- Poor-quality images or artifacts may hinder the accuracy of tumor detection and classification.

Complexity and Heterogeneity of Tumors:

- Brain tumors exhibit diverse morphological, histological, and molecular characteristics, leading to challenges in accurately detecting and classifying different tumor types.

Ethical and Legal Considerations:

- Ethical constraints related to patient consent, data privacy, and responsible use of AI algorithms in healthcare decision-making may influence the design and operation of the application.

Technical Constraints:

- Lack of high-tech computer systems with necessary computational power for high precision.

Chapter 4:Proposed Design

4.1 Block diagram of the system

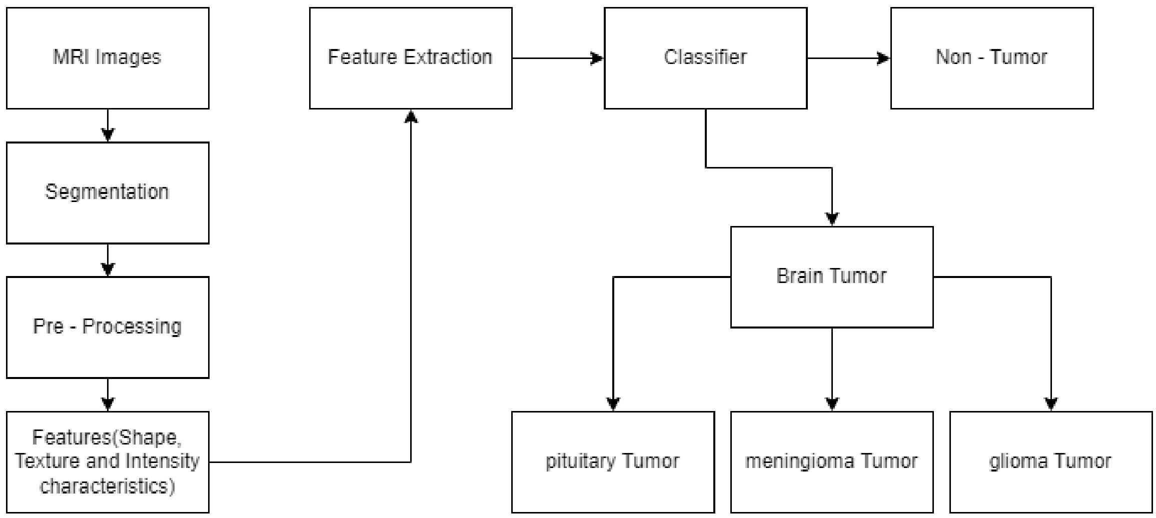


fig 4.1 Block diagram

Block diagram explanation:

The diagram shows the process of brain tumor classification using MRI images. The process consists of the following steps:

- Pre-processing: The MRI images are pre-processed to remove noise and artifacts, and to standardize the image intensity.
- Feature extraction: Features are extracted from the pre-processed images. These features can be based on the shape, texture, and intensity characteristics of the tumor.
- Classification: A classifier is used to classify the tumor into one of the different types of brain tumors.

The classifier is typically a machine learning model, such as a convolutional neural network (CNN). CNNs have been shown to be very effective in brain tumor classification tasks.

The diagram also shows the different types of brain tumors that can be classified using this process. These include:

- Pituitary tumor: A tumor that arises in the pituitary gland.
- Meningioma tumor: A tumor that arises from the meninges, the membranes that surround the brain and spinal cord.
- Glioma tumor: A tumor that arises from the glial cells
- Non-tumor images are also classified using this process. This helps to ensure that the classifier is able to distinguish between tumors and normal brain tissue.

4.2 Modular design of the system

- Backend (Analysis):
  - Prepares MRI images.
  - Uses a trained model to detect brain tumors.

- Frontend (User Interaction):
  - Allows users to upload MRI scans.
  - Sends images to the backend for analysis.
  - Displays the tumor detection results.

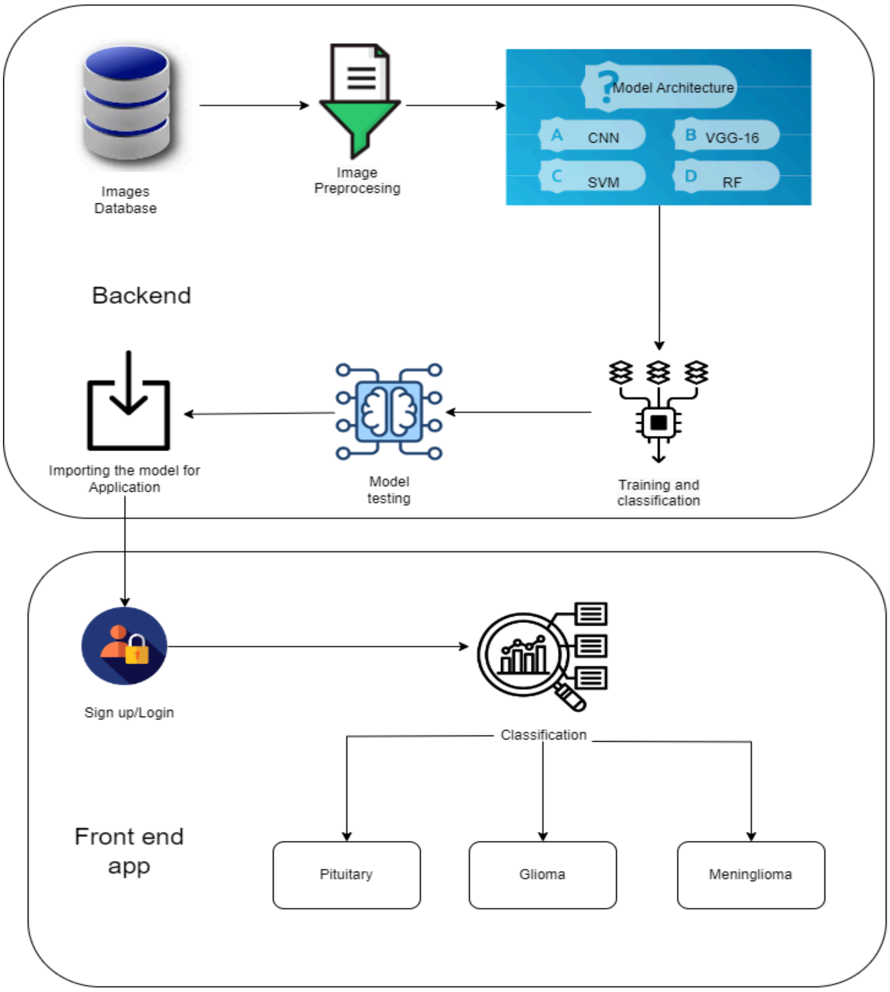


Fig. 4.2.1 Modular diagram of the system

Chapter 5. Implementation of the Proposed System

5.1 Methodology employed for development

BrainScan AI is a system designed to assist detection of brain tumors using advanced deep learning CNN algorithm. Timely and accurate diagnosis is crucial for effective treatment planning and better patient outcomes. However, traditional methods of brain tumor detection often require time-consuming and costly procedures, leading to potential delays in diagnosis. BrainScan AI addresses this challenge by Convolutional Neural Networks (CNNs) and image processing Different types of image processing techniques, such as image segmentation, image enhancement, and feature extraction, are used for the detection of brain tumors in MRI images of cancer-affected patients. The process of detecting brain tumors using image processing techniques involves four stages: Image Pre-Processing, Image Segmentation, Feature Extraction, and Classification. Image processing and neural network techniques are used to improve the performance of detecting and classifying brain tumors in MRI images.

Traditional methods, while valuable, now confront challenges in keeping pace with the accelerating demand for accurate and timely diagnosis. Recognizing this, the focus of this project is to introduce "BrainScan AI" application, integrating advanced deep learning techniques. This initiative is directed towards enhancing the precision and efficiency of brain tumor detection, thereby amplifying the potential for improved patient outcomes and treatment planning

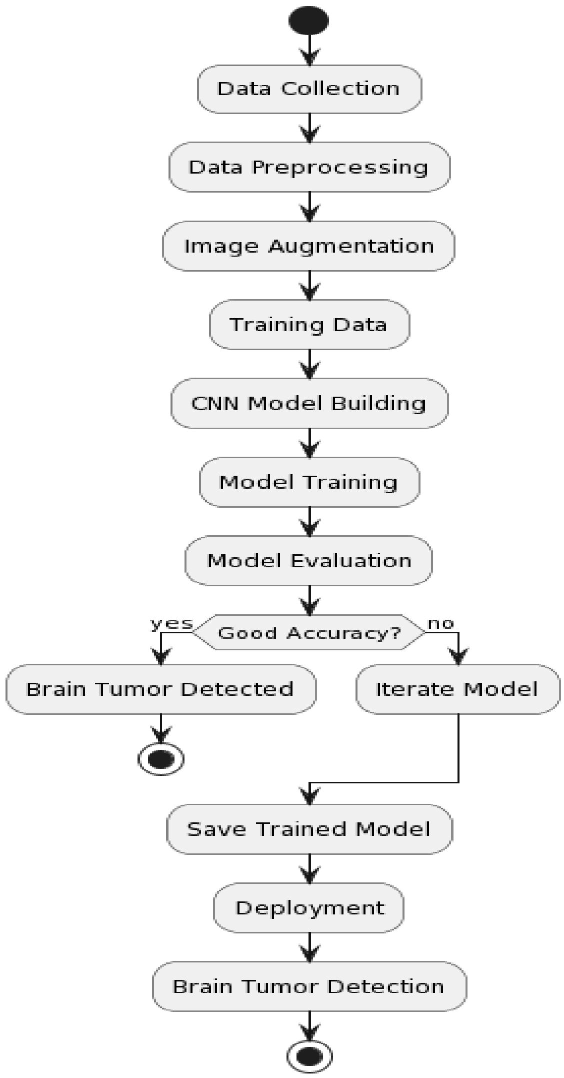


Figure 5.1 Flow Diagram

The diagram shows the steps of a brain tumor detection system. The system takes MRI images as input and outputs a binary classification result: brain tumor or no brain tumor.

The system consists of the following steps:

- **Data collection:** A large dataset of MRI images is collected, including images of both brain tumors and normal brains.
- **Data preprocessing**: The MRI images are preprocessed to remove noise and artifacts, and to standardize the image intensity.

- **Image augmentation:** The MRI images are augmented to increase the size and diversity of the dataset. This helps to improve the performance of the machine learning model.
- **Training data:** The preprocessed and augmented MRI images are split into training and test sets. The training set is used to train the machine learning model, and the test set is used to evaluate the performance of the trained model.
- **CNN model building:** A convolutional neural network (CNN) is built to classify the MRI images as brain tumor or no brain tumor. CNNs are a type of machine learning model that are particularly well-suited for image classification tasks.
- **Model training:** The CNN model is trained on the training dataset. This involves feeding the model the preprocessed MRI images and their corresponding labels (brain tumor or no brain tumor). The model learns to identify the features of the images that are most indicative of brain tumors.
- **Model evaluation:** The trained CNN model is evaluated on the test dataset. This involves feeding the model the preprocessed MRI images in the test set and comparing the model's predictions to the known labels. The model's accuracy and other performance metrics are calculated.
- **Deployment:** If the model achieves a satisfactory performance on the test dataset, it can be deployed to production. This means that the model can be used to classify new MRI images and detect brain tumors.

The diagram also shows a feedback loop between the model evaluation and model building steps. This loop is used to improve the performance of the model. If the model's performance on the test dataset is not satisfactory, the model can be retrained with a different set of hyperparameters or with a different CNN architecture.

Brain tumor detection systems are used to help medical professionals diagnose brain tumors. Early diagnosis of brain tumors is important for improving the patient's prognosis.

5.2 Algorithms and flowcharts for the respective modules developed :

Convolutional Neural Network (CNN) is the extended version of artificial neural networks (ANN) which is predominantly used to extract the feature from the grid-like matrix dataset. For example visual datasets like images or videos where data patterns play an extensive role.

CNN Architecture

Convolutional Neural Network consists of multiple layers like the input layer, Convolutional layer, Pooling layer, and fully connected layers.

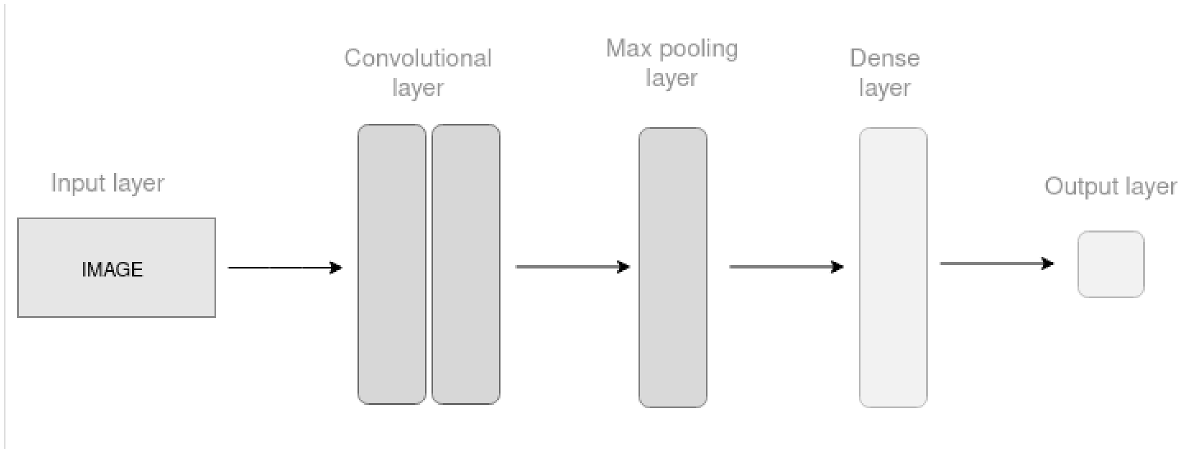


Figure 5.2.1 CNN layers

The Convolutional layer applies filters to the input image to extract features, the Pooling layer downsamples the image to reduce computation, and the fully connected layer makes the final prediction. The network learns the optimal filters through backpropagation and gradient descent.

How Convolutional Layers works

Convolution Neural Networks are neural networks that share their parameters. Imagine you have an image. It can be represented as a cuboid having its length, width (dimension of the image), and height (i.e the channel as images generally have red, green, and blue channels). Now imagine taking a small patch of this image and running a small neural network, called a filter or kernel on it, with say, K outputs and representing them vertically. Now slide that neural network across the whole image, as a result, we will get another image with different widths, heights, and depths. Instead of just R, G, and B channels now we have more channels but lesser width and height. This operation is called Convolution. If the patch size is the same as that of the image it will be a regular neural network. Because of this small patch, we have fewer weights

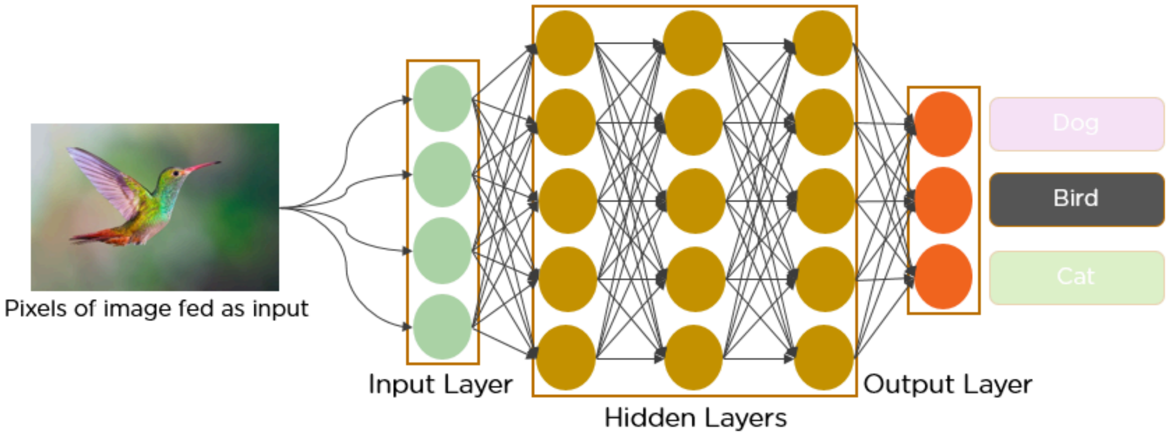


Figure 5.2.2 Convolution Neural Networks working

Flatten the Feature Maps:

- After extracting features using the CNN, the feature maps need to be flattened into a 1D vector. This is typically done before feeding the features to the SVM classifier.
- Flattening the feature maps converts the spatial information into a format that can be input to the SVM classifier. Each element of the flattened vector corresponds to a specific feature extracted by the CNN.

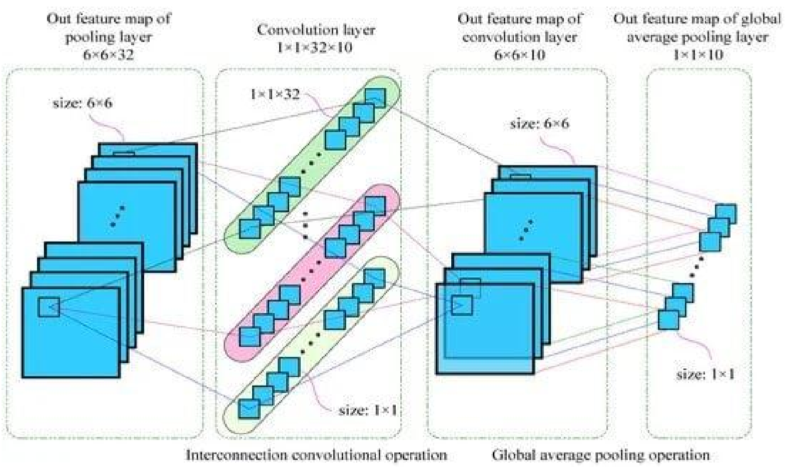
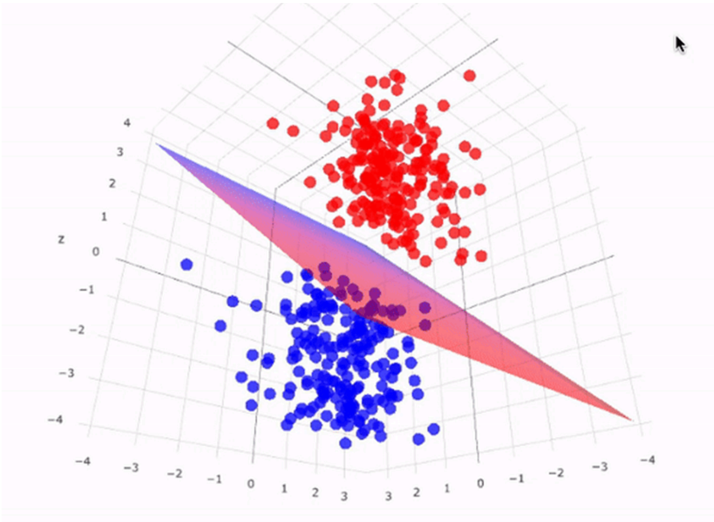


fig.5.2.3 Flatten Maps

SVM Classifier:

- Support Vector Machines (SVMs) are a type of supervised learning algorithm that can be used for classification tasks.
- SVMs work by finding the hyperplane that best separates the classes in the feature space. They maximize the margin between classes, which leads to better generalization.



- In the context of this architecture, the flattened feature vector serves as input to the SVM classifier. SVMs can efficiently handle high-dimensional feature vectors and are effective for both binary and multiclass classification tasks.
- The choice of the SVM kernel (e.g., linear, polynomial, radial basis function) can affect the performance of the classifier and may need to be tuned based on the dataset.

After extracting features using the CNN, the feature maps need to be flattened into a 1D vector. This is typically done before feeding the features to the SVM classifier.
Flattening the feature maps converts the spatial information into a format that can be input to the SVM classifier. Each element of the flattened vector corresponds to a specific feature extracted by the CNN.

Support Vector Machine (SVM)

Support Vector Machines (SVMs) are a strong alternative to Convolutional Neural Networks (CNNs) for detecting brain tumors from 2D MRI data. Unlike CNN's automatic feature learning capabilities, this methodology relies on manual feature extraction. Various properties, such as texture, intensity, shape, and statistical measurements, are methodically retrieved from MRI scans to ensure a complete depiction of each image.

Following feature extraction, a series of preprocessing processes are used to standardize and normalize the recovered features, ensuring that they are suitable for input into the SVM model. These preprocessing strategies improve the consistency of the feature vectors, easing the following classification procedure.

The selection strategy for SVM models is crucial in determining tumor detection efficacy. A variety of kernel functions are explored, including linear, polynomial, and radial basis function (RBF) kernels, with the decision determined by the dataset's features and the specific requirements of the problem at hand. The best kernel for optimizing the performance of the SVM model is determined through experimentation and careful evaluation.

The SVM model is trained using the preprocessed feature vectors and the relevant tumor labels. During this supervised learning process, the SVM learns to divide input feature vectors into two categories: tumor and non-tumor.

Finally, the efficacy of the trained SVM model is evaluated using unseen MRI images. Features are taken from these photos, preprocessed, and input into a trained SVM model for classification. Despite the lack of CNN's intrinsic spatial relationship capture skills, diligent feature engineering and kernel selection support the SVM's ability to effectively detect brain tumors, making it a viable alternative strategy in medical image processing.

Classification Using SVM:

The foundation of our process is to set up an SVM model with carefully chosen kernel functions and parameters. We investigate both linear and nonlinear kernels to determine which best represents the intricacies of tumor versus non-tumor classifications in medical imaging. The kernel used determines the model's capacity to accurately classify data.

To analyze the performance of our SVM model, we divide the dataset into training and testing sets. Initially, we utilized a 70:30 split, which yields 91.17% accuracy rate. This indicates the model's reliability in recognizing cancers. We then tweak the dataset divide to an 80:20 training/testing split, which improves the model's accuracy to 97.6%. This increase underlines the importance of having a large training dataset for the SVM to learn effectively.

The figure below shows our model's training and validation accuracy. Using varied epochs, we saw improved training and validation accuracy. After 100 epochs, the model achieved the highest accuracy for training and validation.

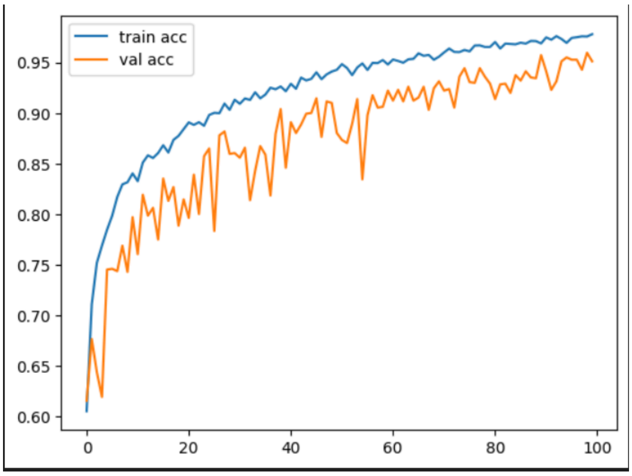


Fig. 5.2.4 Train and validation accuracy graph

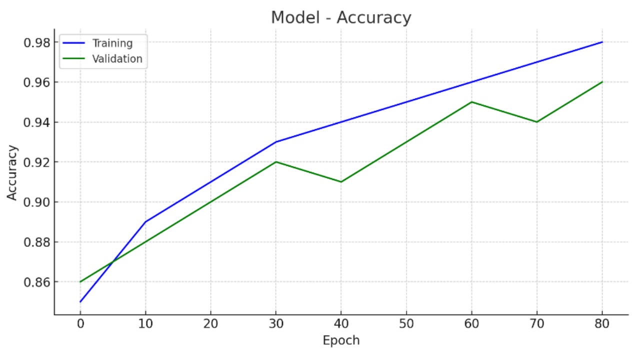


Fig. 5.2.5 Accuracy of the proposed SVM model

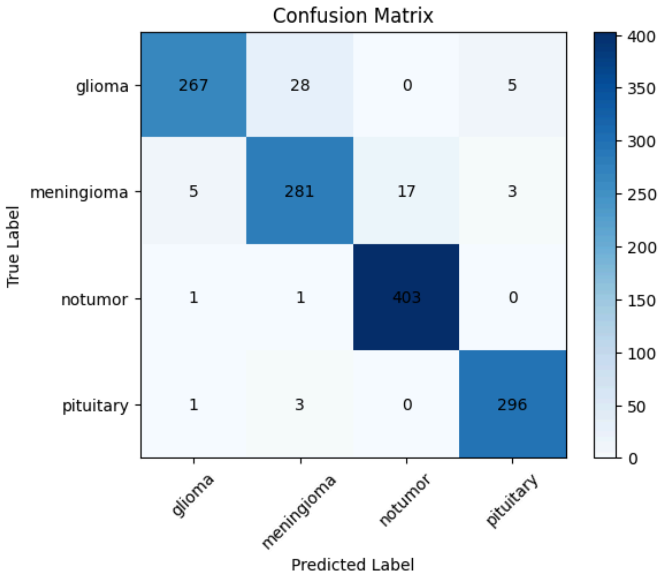


Fig 5.2.6 Confusion matrix graph(SVM)

Random Forest Algorithm

Random Forest is a popular ensemble learning technique used for both classification and regression tasks in machine learning. It operates by constructing multiple decision trees during training and outputs the mode of the classes (in classification) or the average prediction (in regression) of the individual trees.

Random forest algorithm predicts whether the brain scan image has tumor or not and if yes then it also predicts it's type following are steps involved in implementation of this algorithm:

- To predict whether a brain tumor is present and its type, the Random Forest model takes a new, unseen brain scan as input.
- The model passes the input through each decision tree in the forest, and each tree casts a vote for the predicted class.
- In classification tasks, the final prediction is determined by taking the mode (most common) of the predicted classes across all trees.
- Once the model has made a prediction, it can output whether a tumor is present and, if so, which type it belongs to.

Model Evaluation:

The performance of the Random Forest model is evaluated using the testing set, where the true labels are known.

Metrics such as accuracy, precision, recall, and F1-score are calculated to assess how well the model predicts the presence or absence of tumors and their types.

The model can be fine-tuned by adjusting hyperparameters, such as the number of trees in the forest, maximum depth of trees, and the number of features considered at each split.

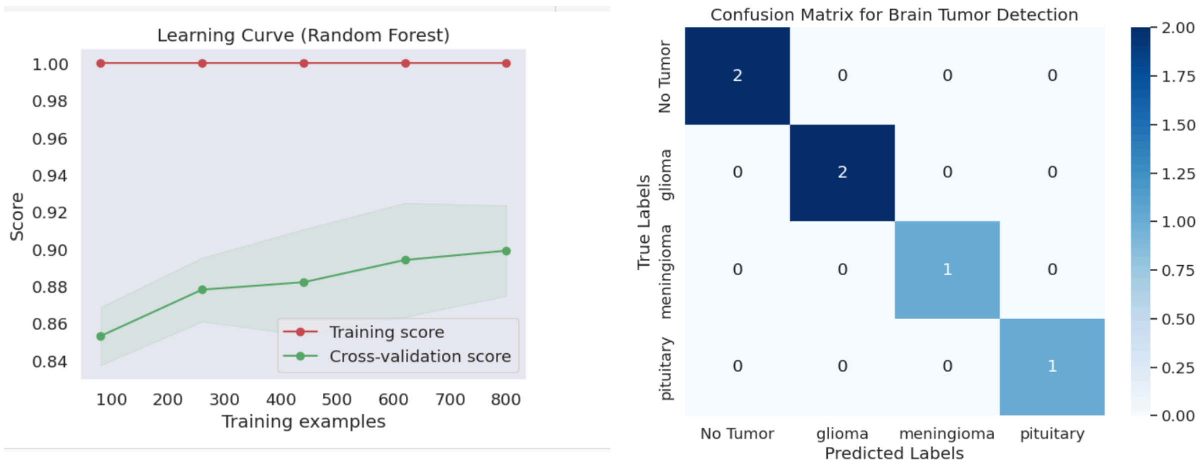
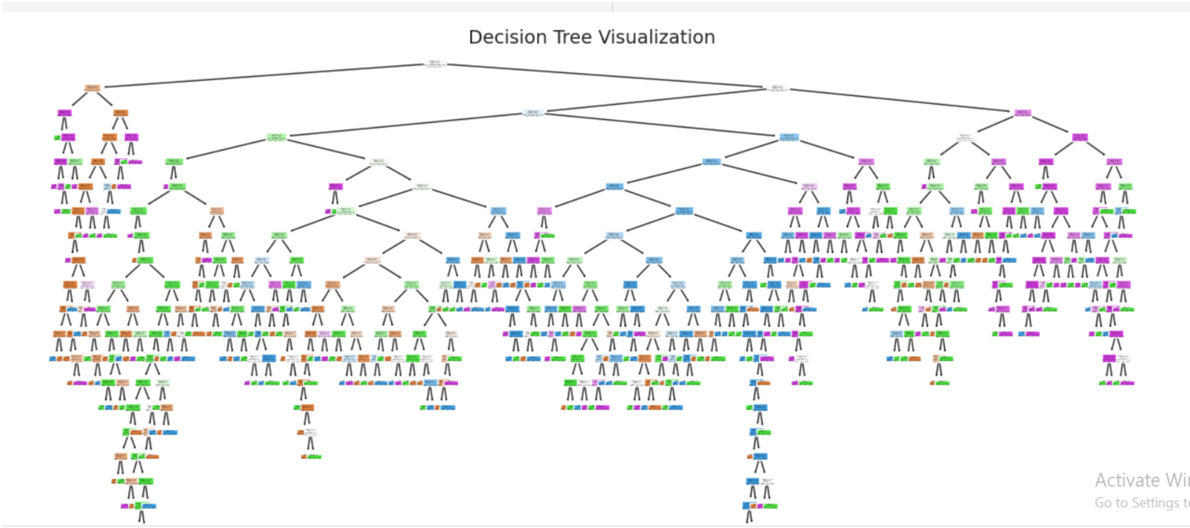


Fig 5.2.7 Learning curve graph of Random Forest and Confusion Matrix for Random Forest , accuracy

```
[35] accuracy = accuracy_score(y_test, y_pred)
     print("Accuracy:", accuracy)

     Accuracy: 0.9378827646544182
```



The above image shows the Decision Tree Visualization of random forest algorithm.

5.3 Datasets source and utilization

- **Data set:**

  This dataset contains 7023 images of human brain MRI images which are classified into 4 classes: **glioma** - **meningioma** - **no tumor** and **pituitary**. Acquiring datasets from hospitals can be challenging, so using publicly available datasets is a practical alternative.

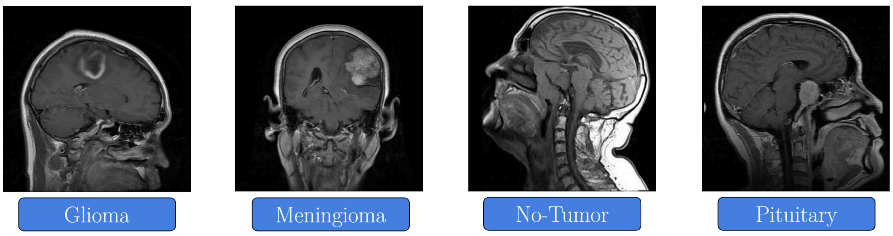


Figure 5.3.1 . MRI images of the brain.

- **Data Preprocessing:**

- Images are read and adjusted to a consistent size of 64x64 pixels.

- Libraries and frameworks used: OpenCV (cv2) for image processing and the Python Imaging Library (PIL) for image resizing.

- **Data Splitting:**

- The dataset is divided into two subsets: the training set and the testing set. This partition enables the evaluation of the model's performance on unseen data.
- Libraries and frameworks used: scikit-learn for the 'train_test_split' function.

- **Data Normalization:**

- The pixel values of the images are standardized using the 'normalize' function. This process enhances the model's ability to learn from the data by ensuring consistent scaling.
- Libraries and frameworks used: Keras for data normalization.

- **Model Architecture:**

- A Convolutional Neural Network (CNN) model is constructed using the Keras library. CNNs are adept at processing and classifying image data.
- The model's structure comprises convolutional layers for feature extraction, activation functions (ReLU) for introducing non-linearity, and max-pooling layers to capture salient features.
- The final layers consist of densely connected layers with dropout regularization to prevent overfitting. A softmax activation function is used for classification.
- Libraries and frameworks used: Keras, which sits on top of TensorFlow, for building and defining the CNN architecture.
-The feature maps obtained from the CNN are typically 3D tensors with dimensions (height, width, channels). To use them as input to the SVM classifier, they need to be flattened into a 1D vector.
-The reshape operation is applied to flatten the feature maps into a 1D vector with dimensions (num_samples, num_features), where num_samples is the number of samples (images) and num_features is the total number of elements in the flattened feature maps.
-During testing, the trained SVM classifier is used to predict labels for new unseen images.
-First, the images are passed through the CNN to extract features. Then, these features are flattened and fed into the trained SVM classifier for prediction.

This comprehensive methodology outlines the steps involved in processing medical image data, building a deep learning model, and training it for brain tumor classification. Key libraries and frameworks, including OpenCV, PIL, scikit-learn, Keras, and TensorFlow, are employed to achieve the project's objectives.

Chapter 6: Testing of the Proposed System

6.1 . Introduction to testing

The Project Testing Phase is a crucial stage in the Software Development Lifecycle (SDLC) that aims to investigate and examine the progress of a project. The goal of this phase is to provide stakeholders with an independent view of the project's performance and quality to help them evaluate potential risks of project failure or mismatch. By evaluating and testing declared requirements, features, and expectations of the project before delivery, stakeholders can ensure that the project matches the initial requirements stated in specification documents.

Testing is one of the most critical processes in the SDLC, as it enables companies to perform a comprehensive assessment of software and ensure that the final product meets the client's needs. By identifying all the bugs and errors in the software before the implementation phase begins, companies can avoid the adverse effects that unresolved bugs can have on the client's business. Delaying the detection of these issues can result in substantial costs, as attempting to resolve them at a later stage can be more challenging and expensive.

There are several ways to perform testing in the software development lifecycle, and the techniques used can vary depending on the software development model, the stage of the process, and the objectives of the testing procedure. Overall, the Project Testing Phase plays a vital role in ensuring the success of software development projects by enabling stakeholders to evaluate the project's performance, quality, and adherence to initial requirements.

6.2. Types of tests Considered

Pre-testing and Post-testing are essential components of evaluating the effectiveness of a disease prediction system that uses machine learning.

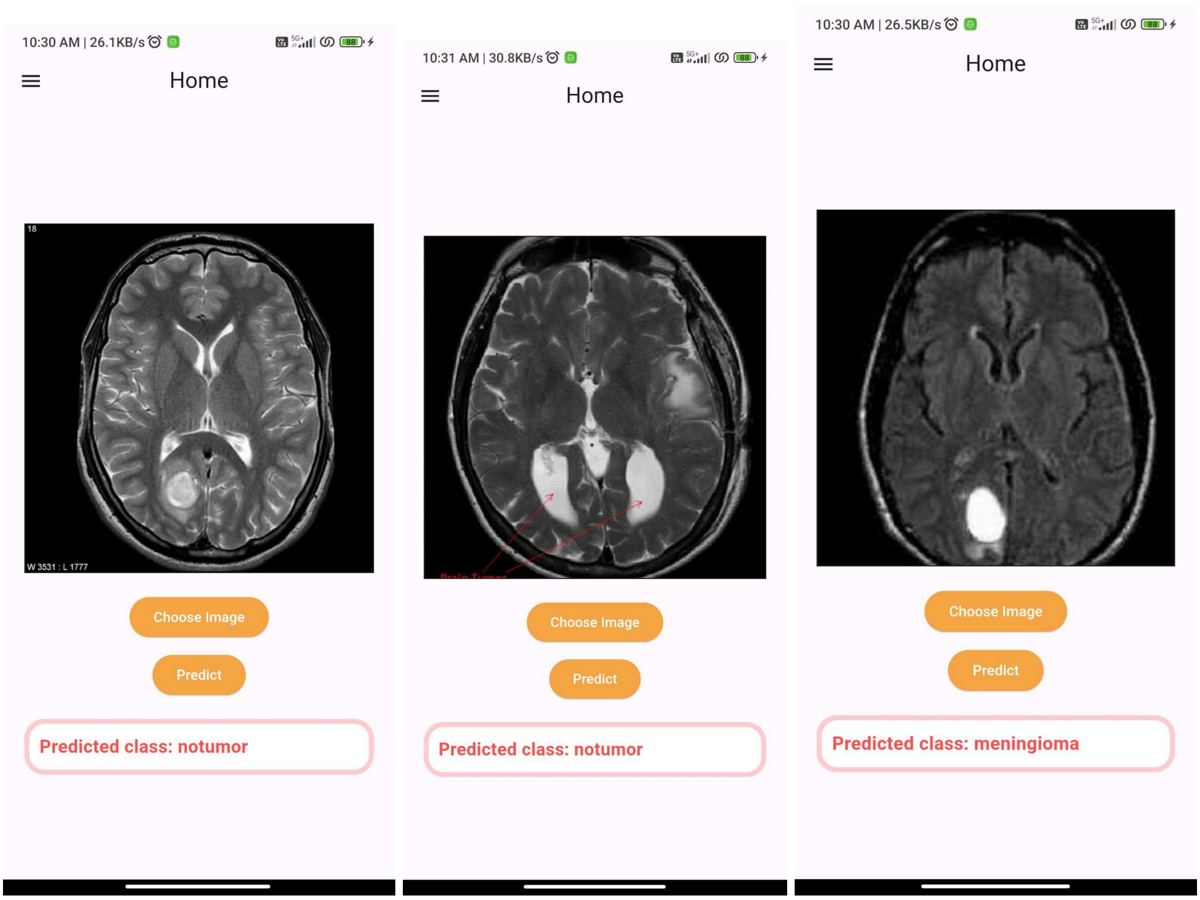
A. Pre-testing phase :

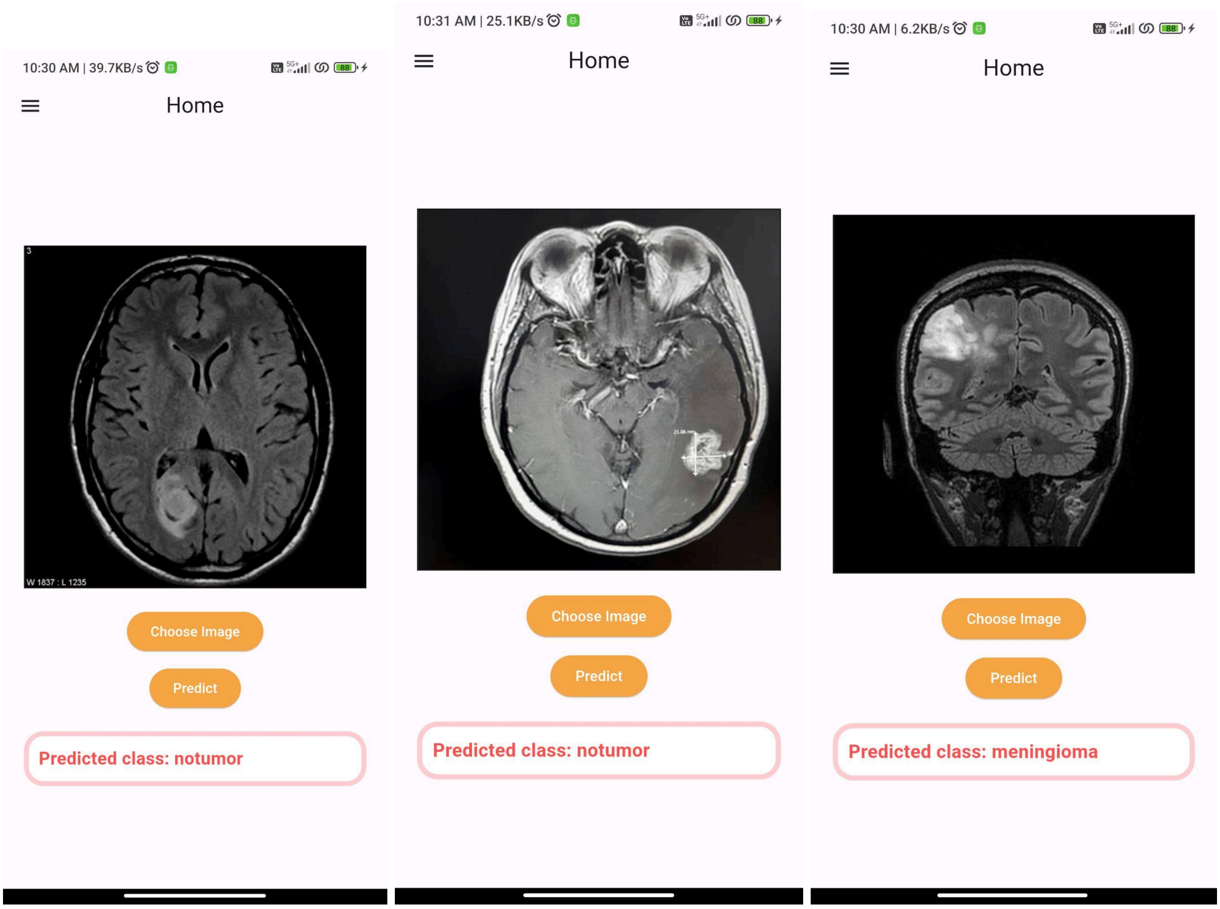
In the pre testing phase, the system's developers would design and implement the machine learning algorithms, using appropriate data sets, and developing relevant features that could help the system accurately predict the occurrence of a specific disease. Here the accuracy and precisions will be tested

B. Post-testing phase:

In the post-testing phase, the system would be implemented in real-world settings to determine its effectiveness in accurately predicting the disease. Data from actual patients would be collected, and the system's performance would be compared against the pretesting results. It will regarding th BTD (brain Tumor detection App) where authentication, Real MRI scans will be calculated

6.3 Various test case scenarios considered




10:30 AM | 39.7KB/s
Home
W 1837 : L 1235
Choose Image
Predict
Predicted class: notumor
10:31 AM | 25.1KB/s
Home
Choose Image
Predict
Predicted class: notumor
10:30 AM | 6.2KB/s
Home
Choose Image
Predict
Predicted class: meningioma

Chapter 7: Results and Discussion

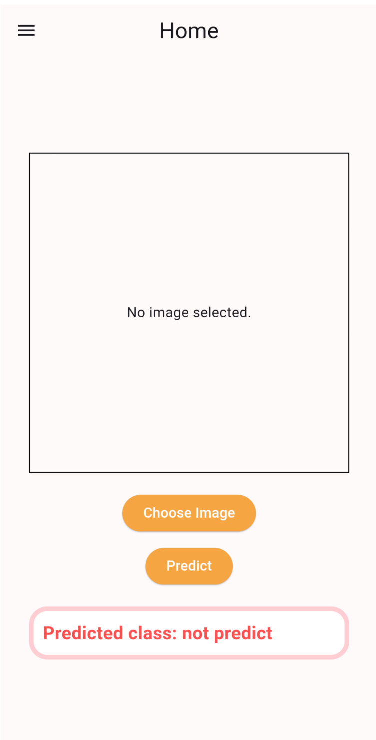
7.1. Screenshots of User Interface (UI) for the respective module

GUI Implementation Using SVM

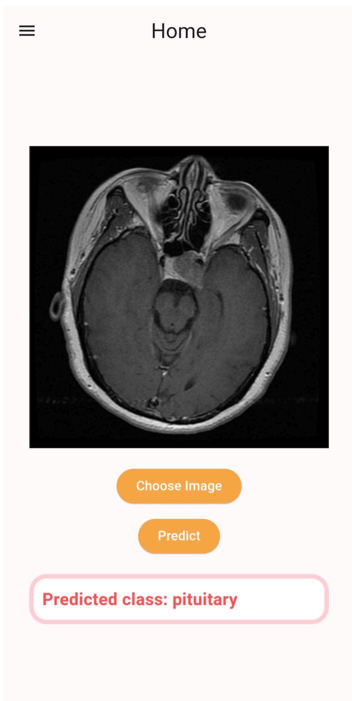
We opted to keep the UI design elegant and modest. The doctor will simply launch the app, select an image (MRI scan), and then click the "Predict" option. So we determined that this GUI can be simply created using Flutter (Dart), and the GUI will connect to the SVM model via FLASK (Python).

The GUI application should seamlessly integrate with our trained SVM model. When the user initiates tumor detection, the GUI :

- Loads the selected brain scan image.
- Preprocesses the image if necessary.
- Extracts relevant features from the image suitable for SVM classification.
- Pass the features to the trained SVM model for prediction.
- Retrieve the predicted tumor type or probability from the SVM model.
- Update the GUI display to present the results to the user.

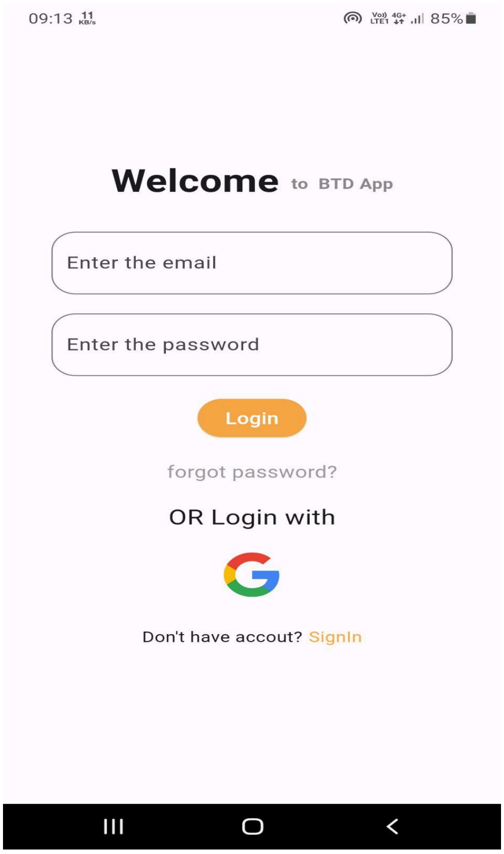


GUI Application



The above image shows the implementation part of Brain Tumor Detector Application where the user will be able to upload any Brain Scan MRI image from it's device.

After the image is uploaded the user will have to click on the predict. Now, the application predicts the type of tumor or no tumor of the Brain Scan image provided.
(testing, table accuracy and desc)



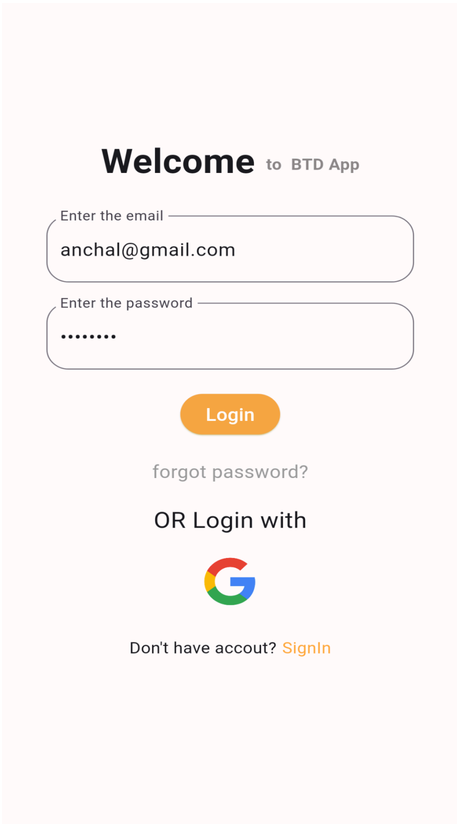


Fig 7.1.1

Fig 7.1.2

The above image shows the login page of brain tumor detector application.Here, the user needs to login with the registered valid email Id.Once the user has logged in Successfully the home page of the application is displayed as shown below.

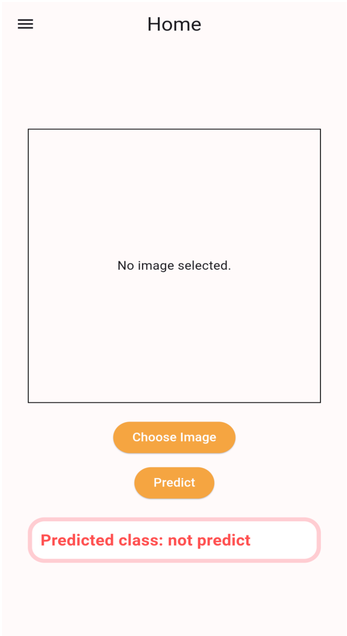
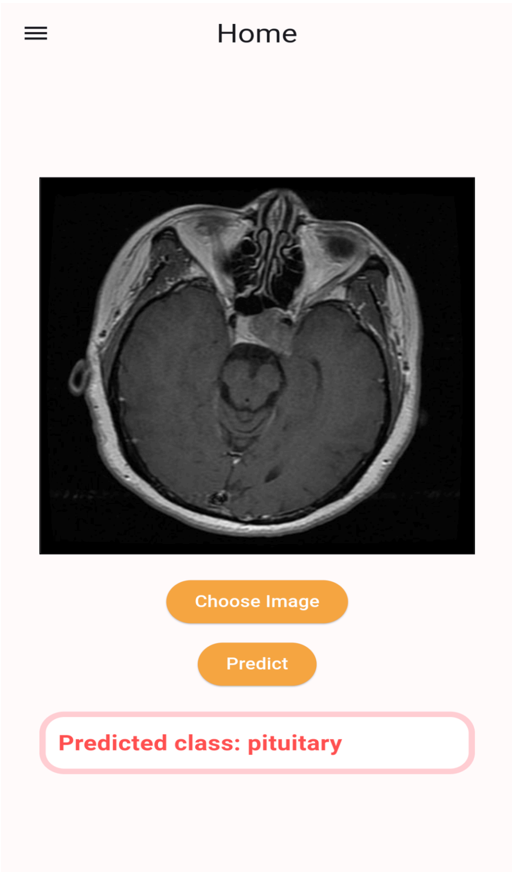


Fig 7.1.3

Once the user has logged in and now the user will be able to choose the image from it's device. Then the user needs to click on the predict button.And the tumor name or no tumor will be displayed on the page as shown below.



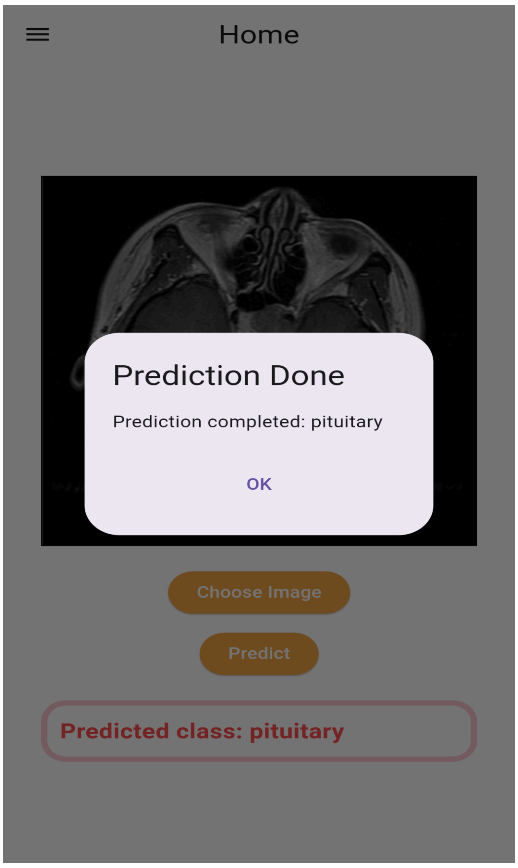


Fig 7.1.4

Fig 7.1.5

7.2. Performance Evaluation measures

A. Determination of efficiency

Precision is defined as the ratio of correctly classified positive samples (True Positive) to the total number of classified positive samples (either correctly or incorrectly).

$$\text{Precision} = TP / (TP+FP)$$

B. Determination of accuracy

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

$$\text{Accuracy} = (TP+TN) / (TP+FP+TN+FN)$$

C. Report on sensitivity analysis

Recall of positive class is also termed sensitivity and is defined as the ratio of the True Positive to the number of actual positive cases. It can intuitively be expressed as the ability of the classifier to capture all the positive cases. It is also called the True Positive Rate (TPR).

$$\text{Sensitivity} = TP / (TP+FN)$$

7.3. Input Parameters / Features considered

Following are some features considered :

1.Shape-Based Features:

The size of the tumor in the scan image that has been taken in the input.The solidity of the tumor, compactness and the area covered by the tumor.

2. Spatial Features:

The Location of the tumor in the image and the distance from the anatomical landmarks in the input has a lot of impact on output.

3.Textural features .

The texture of the image taken as the input and the intensity values of the consecutive pixels i.e the gray-level in the image impacts when the tumor is detected by the model.

7.4. Graphical and statistical output

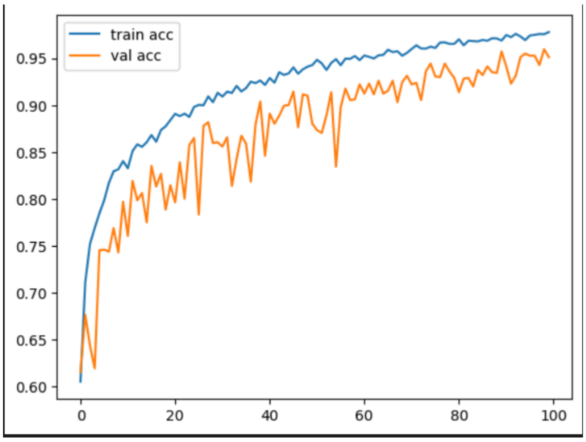


Fig. 7.4.1 Train and validation accuracy graph(SVM)

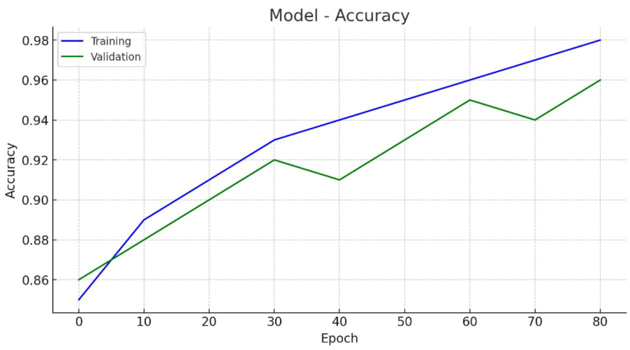


Fig. 7.4.2 Accuracy of the proposed SVM model

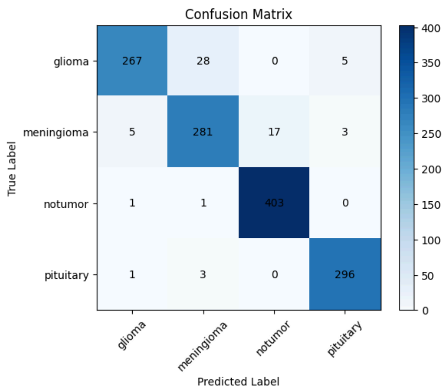


Fig 7.4.3 Confusion Matrix Graph (SVM)

7.5. Comparison of results with existing systems

This mini project contributes to a comprehensive understanding of the current methods used for brain tumor detection, emphasizing their limitations and research gaps. The current landscape showcases several notable systems, such as BraTS, DeepMedic, and TumorNet, each with their disadvantages. Furthermore, the project acknowledges the pressing need for improved healthcare services, especially in the context of rising mental health issues and complex diseases like brain tumors.

Our project addresses these challenges by leveraging a limited dataset of 4500 images and focusing on detecting small tumors efficiently. As of 2023, our project is hardware-efficient, capable of running on entry-level graphics cards. We employ various algorithms, including VGG16, random forests, CNN, and SVM, and select the most effective one. Additionally, our project integrates a mobile application feature, enabling individuals to detect tumors quickly from anywhere and report them immediately to healthcare facilities.

Moreover, our project accomplishes the task of categorizing images into various types of tumors, facilitating prompt action by patients and doctors. This multifaceted approach aims to enhance the effectiveness and accessibility of brain tumor detection, contributing to improved healthcare outcomes.

7.6. Inference drawn

- The performance evaluation metrics and graphical/statistical outputs affirm the model's capability to serve as a reliable diagnostic tool, potentially enhancing clinical decision-making processes.
- Compared to existing systems, the SVM approach offers distinct advantages in terms of computational efficiency and the ability to handle high-dimensional data, making it particularly suited for medical imaging analysis.

Result Analysis of CNN Model

Training the model with Epoch=10:

```
PS C:\Users\amand\OneDrive\Desktop\Projects\Brain_Tumor_Classification-main\BrainTumor
Classification DL> python -u "c:\Users\amand\OneDrive\Desktop\Projects\Brain_Tumor_Clas
sification-main\BrainTumor Classification DL\mainTrain.py"
2023-10-20 22:29:34.614240: I tensorflow/core/platform/cpu_feature_guard.cc:182] This T
ensorFlow binary is optimized to use available CPU instructions in performance-critical
 operations.
To enable the following instructions: SSE SSE2 SSE3 SSE4.1 SSE4.2 AVX AVX2 AVX512F AVX5
12_VNNI FMA, in other operations, rebuild TensorFlow with the appropriate compiler flag
s.
Epoch 1/10
150/150 [==============================] - 8s 45ms/step - loss: 0.5559 - accuracy: 0.72
00 - val_loss: 0.4488 - val_accuracy: 0.7800
Epoch 2/10
150/150 [==============================] - 11s 76ms/step - loss: 0.4075 - accuracy: 0.8
271 - val_loss: 0.3508 - val_accuracy: 0.8333
Epoch 3/10
150/150 [==============================] - 12s 78ms/step - loss: 0.3068 - accuracy: 0.8
758 - val_loss: 0.2639 - val_accuracy: 0.8983
Epoch 4/10
150/150 [==============================] - 11s 74ms/step - loss: 0.2290 - accuracy: 0.9
121 - val_loss: 0.2002 - val_accuracy: 0.9250
Epoch 5/10
 31/150 [=====>........................] - ETA: 9s - loss: 0.1545 - accuracy: 0.9395
```

Figure 7.6.1

Model creation:

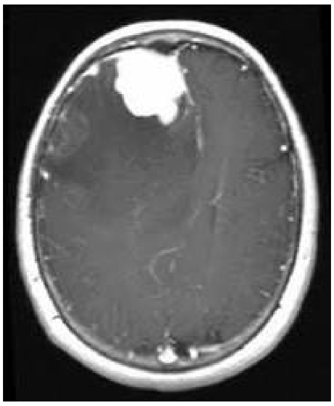
Figure 7.6.2 Training the model CNN

Testing the model i/p=14 :

```
[[0 0]
 [3 4]]
PS C:\Users\amand\OneDrive\Desktop\Projects\Brain_Tumor_C
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 32ms/step
Accuracy: 0.5714285714285714
Precision: 1.0
Recall: 0.5714285714285714
Confusion Matrix:
[[0 0]
 [6 8]]
```

Figure 7.6.3 Accuracy of CNN

Single image test :



```
PS C:\Users\amand\OneDrive\Desktop\Projects\Brain_Tumor_Classification-main\
BrainTumor Classification DL> python -u "c:\Users\amand\OneDrive\Desktop\Pro
jects\Brain_Tumor_Classification-main\BrainTumor Classification DL\test1.py"
2023-10-20 23:14:14.164058: I tensorflow/core/platform/cpu_feature_guard.cc:
182] This TensorFlow binary is optimized to use available CPU instructions i
n performance-critical operations.
To enable the following instructions: SSE SSE2 SSE3 SSE4.1 SSE4.2 AVX AVX2 A
VX512F AVX512_VNNI FMA, in other operations, rebuild TensorFlow with the app
182] This TensorFlow binary is optimized to use available CPU instructions i
n performance-critical operations.
To enable the following instructions: SSE SSE2 SSE3 SSE4.1 SSE4.2 AVX AVX2 A
VX512F AVX512_VNNI FMA, in other operations, rebuild TensorFlow with the app
ropriate compiler flags.
1/1 [==============================] - 0s 154ms/step
Predicted Class: 1
```

Fig 7.6.4 Input and Output of the test case

Random Forest Algorithm

The above image shows the result of implementation of random forest algorithm i.e the accuracy of the model

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

Chapter 8: Conclusion

8.1 Limitations

The development and implementation of a brain tumor detection system based on Support Vector Machines (SVM) has shown great potential for enhancing diagnostic accuracy and efficiency. However, some constraints have been observed during this project:

- **Data Availability and Quality:** The system's performance is highly dependent on the quantity and quality of the medical imaging data used for training the SVM models. Limited access to diverse, high-quality datasets may constrain the system's ability to generalize across different types of brain tumors and patient demographics.
- **Computational Resources:** The lack of high-tech computer systems with sufficient computational power can limit the system's ability to process large datasets or implement complex models, potentially affecting overall performance and scalability.
- **Algorithmic Complexity:** While SVMs are powerful tools for classification tasks, their effectiveness in highly complex and nuanced applications like brain tumor detection can be affected by the choice of kernel, the tuning of hyperparameters, and the handling of imbalanced datasets.
- **Clinical Integration:** The practical application and integration of the system into clinical workflows present challenges, including user training, system compatibility with existing medical records systems, and meeting stringent healthcare regulatory standards.

8.2 Conclusion

Throughout this project, we explored and compared several machine learning and deep learning models, including Support Vector Machines (SVM), Random Forest, VGG-16, and Convolutional Neural Networks (CNN), to identify the most effective approach for brain tumor detection. After rigorous testing and analysis over 100 epochs, SVM emerged as the model with the highest accuracy. This finding underscores the suitability of SVM for handling the complexities and nuances involved in brain tumor detection. In addition, to provide a practical outlet for brain tumor diagnosis, we created an app with a flawless user experience.

Despite the challenges and limitations encountered, the system has laid a solid foundation for the use of advanced analytics in healthcare, offering a new avenue for supporting and improving decision-making processes by medical professionals. In a nutshell, this project demonstrates how advanced analytical models, particularly Support Vector Machines, can significantly enhance the accuracy and efficiency of brain tumor detection, paving the way for faster, more reliable medical diagnostics and patient care.

8.3 Future Scope

Looking forward, there is substantial room for further research and development to enhance the capabilities and overcome the limitations of the current system:

- **Data Enrichment:** Acquiring more comprehensive and diverse datasets, possibly through collaborations with medical institutions, could improve model robustness and accuracy.
- **Computational Advancements:** Leveraging cloud computing and parallel processing technologies could address computational limitations, enabling the handling of larger datasets and more complex models.
- **Algorithmic Improvements:** Exploring advanced machine learning and deep learning techniques.
- **Clinical Trials and Validation:** Conducting extensive clinical trials to validate the system's effectiveness and user acceptance in real-world settings is crucial for its adoption in clinical practice.

4. Annexure

4.1 References:

[1] P. Tiwari, B. Pant, M.M. Elarabawy, M. Abd-Elnaby, N. Mohd, G. Dhiman, and S. Sharma, "CNN Based Multiclass Brain Tumor Detection Using Medical Imaging," Comput. Intell. Neurosci., vol. 2022, p. 1830010, 2022.

[2] A. Anaya-Isaza and L. Mera-Jiménez, "Data Augmentation and Transfer Learning for Brain Tumor Detection in Magnetic Resonance Imaging," IEEE Access, vol. 10, pp. 23217–23233, 2022.

[3] V.S. Lotlikar, N. Satpute, and A. Gupta, "Brain Tumor Detection Using Machine Learning and Deep Learning: A Review," Curr. Med. Imaging, vol. 18, pp. 604–622, 2022.

[4] Y. Xie, F. Zaccagna, L. Rundo, C. Testa, R. Agati, D.N. Manners, C. Tonon, "Convolutional neural network techniques for brain tumor classification (from 2015 to 2022): Review, challenges, and future perspectives," Diagnostics, vol. 12, p. 1850, 2022.

[5] H.R. Almadhoun and S.S. Abu-Naser, "Detection of Brain Tumor Using Deep Learning," Int. J. Acad. Eng. Res. (IJAER), vol. 6, pp. 29–47, 2022.

[6] J. Amin, M. Sharif, A. Haldorai, M. Yasmin, and R.S. Nayak, "Brain tumor detection and classification using machine learning: A comprehensive survey," Complex Intell. Syst., vol. 8, pp. 3161–3183, 2022.

[7] D.R. Nayak, N. Padhy, P.K. Mallick, M. Zymbler, and S. Kumar, "Brain Tumor Classification Using Dense Efficient-Net," Axioms, vol. 11, p. 34, 2022.

[8] M.R. Obeidavi and K. Maghooli, "Tumor Detection in Brain MRI using Residual Convolutional Neural Networks," Proceedings of 2022 IEEE International Conference on Machine Vision and Image Processing (MVIP), Ahvaz, Iran, 23–24 February 2022, pp. 1–5.

[9] S.A. Qureshi, S.E.A. Raza, L. Hussain, A.A. Malibari, M.K. Nour, A.U. Rehman, F.N. Al-Wesabi, and A.M. Hilal, "Intelligent Ultra-Light Deep Learning Model for Multi-Class

Brain Tumor Detection," Appl. Sci., vol. 12, p. 3715, 2022.

[10] M.M. Zahoor, S.A. Qureshi, S. Bibi, S.H. Khan, A. Khan, U. Ghafoor, and M.R. Bhutta, "A New Deep Hybrid Boosted and Ensemble Learning-Based Brain Tumor Analysis Using MRI," Sensors, vol. 22, p. 2726, 2022.

[11] M. Arabahmadi, R. Farahbakhsh, and J. Rezazadeh, "Deep Learning for Smart Healthcare—A Survey on Brain Tumor Detection from Medical Imaging," Sensors, vol. 22, p. 1960, 2022.

[12]D.V. Gore, and V. Deshpande, "Comparative study of various techniques using deep Learning for brain tumor detection," in Proceedings of the 2020 IEEE International Conference for Emerging Technology (INCET), Belgaum, India, 5–7 June 2020

[13] G.S. Tandel, M. Biswas, O.G. Kakde, A. Tiwari, H.S. Suri, M. Turk, J.R. Laird, C.K. Asare, A.A. Ankrah, N. Khanna, et al., "A review on a deep learning perspective in brain cancer classification," Cancers, vol. 11, p. 111, 2019.

[14] J. Amin, M. Sharif, M. Yasmin, and S.L. Fernandes, "Big data analysis for brain tumor detection: Deep convolutional neural networks," Future Gener. Comput. Syst., vol. 87, pp. 290–297, 2018.

[15] V.Y. Borole, S.S. Nimbhore, and D.S.S. Kawthekar, "Image processing techniques for brain tumor detection: A review," Int. J. Emerg. Trends Technol. Comput. Sci. (IJETTCS), vol. 4, no. 2, 2015.

[16]CN104834943A - Brain tumor classification method based on deep learning https://patents.google.com/patent/CN104834943A/en

[17] US20100027865A1 - Method and System for Brain Tumor Segmentation in 3D Magnetic Resonance Images (https://patents.google.com/patent/US20100027865A1/en)

[18] EP 4075380 B1 - MEDICAL IMAGE-BASED TUMOR DETECTION AND DIAGNOSTIC DEVICE(https://data.epo.org/publication-server/rest/v1.0/publication-dates/20230315/patents/EP4075380NWB1/document.pdf)