A

PROJECT REPORT

ON

“DETECTION OF BRAIN TUMOR USING CNN WITH WAVELET”

Submitted By

ANANDA KUMAR SAHOO

(Regd. No-2101110028)

ASWOJIT SAMAL

(Regd. No -2101110036)

SWATI NATH

(Regd. No- 2101110119)

ANANYA RANI PANDA

(Regd. No -2221110002)

Under The Esteemed Guidance of

Dr. Dillip Ranjan Nayak (Assistant professor)

In partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

GOVERNMENT COLLEGE OF ENGINEERING, KALAHANDI
BHAWANIPATNA

MAY 2025

CERTIFICATE

Certified that this seminar report **"DETECTION OF BRAIN TUMOR USING CNN WITH WAVELET"** is the Bonafide work of **"ANANDA KUMAR SAHOO (2101110028)", "ASWOJIT SAMAL (2101110036)", "SWATI NATH (2101110119)" and "ANANYA RANI PANDA (2221110002)"** who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported here in does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate**.**

SIGNATURE
Dr. Dillip Ranjan Nayak
(SUPERVISOR)

SIGNATURE
Dr. Gopal Behera
(HEAD OF THE DEPARTMENT)

(EXRERNAL EXAMINER)

DECLARATION

We hereby affirm that the project work presented in this report, titled **"DETECTION OF BRAIN TUMOR USING CNN WITH WAVELET"** submitted to the **Department of Computer Science and Engineering** under the supervisions of **Dr. Dillip Ranjan Nayak** is our original work and fulfils the requirements for the Bachelor of Technology degree in Computer Science and Engineering. We certify that the work is free from any form of plagiarism, has not been submitted elsewhere for any other degree, and upholds the principles of academic honesty and integrity. Furthermore, we confirm that no idea, data, or source has been misrepresented, fabricated, or falsified in this submission.

ANANDA KUMAR SAHOO

(Regd. No. 2101110028)

ASWOJIT SAMAL

(Regd. No. 2101110036)

SWATI NATH

(Regd. No. 2101110119)

ANANYA RANI PANDA

(Regd. No. 2221110002)

ACKNOWLEDGEMENT

The satisfaction that successful completion of this project would be incomplete without the mention of the people who made it possible, without whose cram guidance and encouragement would have made effort go in vain. I consider myself privileged to express gratitude and respect toward all those who guided us through the completion of this project. I convey thanks to my guide **Dr. Dilip Ranjan Nayak** for providing encouragement, constant support, and guidance which was of great help to complete this project successfully.

I am very grateful to **Dr. Gopal Behera**, Head of the Department of **Computer Science and Engineering** for giving the support and encouragement that was necessary for the completion of this project.

I would also like to express my gratitude to **Prof. (Dr.) Subhransu Sekhar Dash**, Principal Government College of Engineering Kalahandi, Bhawanipatna for providing us with a congenial environment to work in.

ABSTRACT

Early detection of brain tumors is crucial for effective treatment and improved patient outcomes. This study explores the use of the VGG16 convolutional neural network (CNN) model combined with wavelet transforms to automatically identify brain tumors from magnetic resonance imaging (MRI) scans. Wavelet transforms break down MRI images into different frequency components, highlighting important features that may indicate the presence of a tumour. By Integrating these processed images allows the VGG16 model to learn detailed patterns differentiating between tumour and non-tumor cases. The research involved preprocessing MRI images to enhance their quality and applying transfer learning techniques to fine-tune the VGG16 model. The model was trained on a dataset of MRI images labeled as "NO" (no tumor) and "YES" (tumor). Performance metrics such as accuracy, precision, recall, and F1-score were used to evaluate the model's effectiveness. The results demonstrated an accuracy of approximately 98%, indicating the model's potential in supporting healthcare professionals in diagnosing brain tumors. This study highlights the applicability of deep learning models, particularly VGG16 combined with wavelet transforms, in enhancing diagnostic accuracy for brain tumor detection. By automating the identification process, such models can assist radiologists in making more accurate and timely diagnoses, ultimately contributing to better patient care.

Keywords: **Brain tumor detection, VGG16, Convolutional Neural Network, MRI images, Wavelet transform, Deep learning, Transfer learning, Medical Image Analysis.**

TABLE OF CONTENTS

TITLE	PAGE NO.
CERTIFICATE	I
DECLARATION	II
ACKNOWLEDGEMENT	III
ABSTRACT	IV
TABLE OF CONTENTS	V
LIST OF FIGURES	VII
LIST OF TABLES	IX
CHAPTER -1 INTRODUCTION	**1**
1.1. BASIC CONCEPTS OF BRAIN AND BRIAN TUMOR	1
1.2. MOTIVATION	4
1.3. PROBLEM DEFINITION	5
1.4. OBJECTIVES	5
1.5. PROJECT SCOPE AND LIMITATIONS	5
1.6. MODELS FOR PROBLEM SOLVING	6
1.6.1. LOGISTIC REGRESSION	6
1.6.2. SUPPORT VECTOR MECHINE (SVM)	7
1.6.3. CONVOLUTIONAL NEURAL NETWORK	8
1.6.4.VGG-16	9
CHAPTER-2 RELATED WORKS	**11**
2.1. LITERATURE REVIEW	11
2.2. SUMMARY OF PREVIOUS WORKS	12
CHAPTER-3 WORKFLOW EXPLANATION	**16**
3.1. INPUT IMAGE DATA	16

3.2. DETAILED DATA PREPROCESSING IMPLEMENTATION 17
3.2.1. RESIZING 17
3.2.2. AUGMENTATION 17
3.2.3. NORMALIZATION 18
3.2.4. FILTERING 18
3.2.5. WAVELET TRANSFORMS 18
3.3. DATA SPLITTING 19
3.4. MODEL IMPLEMENTATION 20
3.5. MODEL EVALUATION 21
CHAPTER-4 MODEL EXPLANATION 23
4.1. INTRODUCTION TO CNN 23
4.2. Overview of VGG-16 23
4.3. Layer-wise Breakdown of VGG-16 24
4.4. 16 Parameters and Summary 27
4.5. Key Characteristics and Features 28
4.6. Applications of VGG-16 29
4.7. Modifications and Variants of VGG-16 29
CHAPTER-5 RESULT AND PERFORMANCE 30
5.1. Confusion Matrix 34
5.2. ROC Graph 37
5.3. Accuracy Graph 40
5.4. Accuracy Loss Graph 43
5.5. Comparison Table Based on Training and Validation Loss 46
5.6. Precision and Recall Graph 47
5.7. Result Analysis 50
CHAPTER-6 CONCLUSION AND FUTURE SCOPE 51
6.1. Conclusion 51
6.2. Future Scope 52
6.3. Reference 54

LIST OF FIGURES

FIG NO.	PAGE NO.
1. Besic Structure of human brain	01
2. T1.T2 and Flair image	02
3. Graph of TE and TR	03
4. Table of TE and TR time	03
5. SVM Model Diagram	07
6. Working of CNN model for Brain Tumor	08
7. Vgg-16 Map	10
8. Workflow of Detection	16
9. layer wise Breakdown Of VGG-16	24
10. Confusion Matrix of Logistic Regression	34
11. Confusion Matrix of Logistic Regression with Wavelet	34
12. Confusion Matrix of SVM	34
13. Confusion Matrix of SVM with Wavelet	34
14. Confusion Matrix of CNN	35
15. Confusion Matrix of CNN with Wavelet	35
16. Confusion Matrix of CNN with Wavelet and INT function	35
17. Confusion Matrix of VGG-16	36
18. Confusion Matrix of VGG-16 with Wavelet	36
19. Confusion Matrix of VGG-16 with Wavelet and INT function	36
20. ROC Curve for Logistic Regression	37
21. ROC Curve for Logistic Regression with Wavelet	37
22. ROC Curve for SVM	37
23. ROC Curve for SVM with Wavelet	37
24. ROC Curve for CNN	38
25. ROC Curve for CNN with Wavelet	38
26. ROC Curve for CNN with Wavelet and INT function	38
27. ROC Curve for VGG-16	39
28. ROC Curve for VGG-16 with Wavelet	39
29. ROC Curve for VGG-16 with Wavelet and INT function	39
30. Accuracy Graph for Logistic Regression	40
31. Accuracy Graph for Logistic Regression with wavelet	40
32. Accuracy Graph for SVM	40
33. Accuracy Graph for SVM with wavelet	40
34. Accuracy Graph for CNN	41
35. Accuracy Graph for CNN with wavelet	41
36. Accuracy Graph for CNN with Wavelet and INT function	42
37. Accuracy Graph for VGG-16	42
38. Accuracy Graph for VGG-16 with wavelet	42

39. Accuracy Graph for VGG-16 with Wavelet and INT function 43
40. Accuracy Loss Graph for CNN 43
41. Accuracy Loss Graph for CNN with wavelet 43
42. Accuracy Loss Graph for CNN with Wavelet and INT function 44
43. Accuracy Loss Graph for VGG-16 44
44. Accuracy Loss Graph for VGG-16 with wavelet 44
45. Accuracy Loss Graph for VGG-16 with Wavelet and INT function 45
46. Precision and recall Graph of SVM 47
47. Precision and recall Graph of SVM with Wavelet 47
48. Precision and recall Graph of Logistic Regression 47
49. Precision and recall Graph of Logistic Regression with Wavelet 47
50. Precision and recall Graph of CNN 48
51. Precision and recall Graph of CNN with Wavelet 48
52. Precision and recall Graph of CNN with Wavelet and INT function 48
53. Precision and recall Graph of VGG-16 49
54. Precision and recall Graph of VGG-16 with Wavelet 49
55. Precision and recall Graph of VGG-16 with Wavelet and INT function 49
56. User Input Image 50
57. Processed Image (Wavelet + INT) 50
58. Predicated Image 50

LIST OF TABLES

TABLE NO.	PAGE NO.
1.1. Summary of Previous Works	13
4.1. Parameters of VGG-16	28
5.1. All model results table	30
5.2. Comparison Table	31
5.3. Model Comparison Table Based on Training and Validation	46

CHAPTER-1

INTRODUCTION

Brain tumors are abnormal growths in the brain that can be dangerous and life-threatening if not detected early. Detecting brain tumors at the right time is very important because early treatment can improve a patient's chances of recovery. **Magnetic Resonance Imaging (MRI)** is a commonly used technique to capture detailed images of the brain and detect tumors. However, analysing these MRI scans manually can be **slow, complex, and sometimes inaccurate**, as small tumors may be missed.

1.1. OVERVIEW OF BRAIN AND BRAIN TUMOR

Main part in human nervous system is human brain. It is located in the human head and it is covered by the skull. The function of human brain is to control all the parts of human body. It is one kind of organ that allows human to accept and endure all type of environmental condition. The human brain enables humans to do the action and share the thoughts and feeling. In this section we describe the structure of the brain for understanding the basic things.

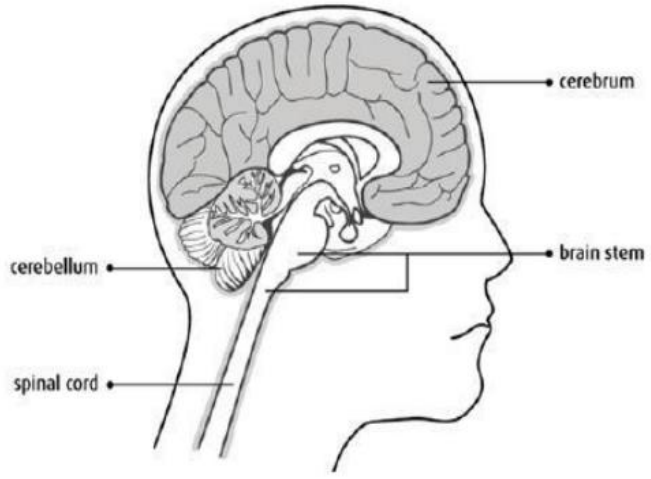

Fig.1: Basic Structure of human brain

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 grows slowly in the brain and 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 quicker to spread into other tissue. Secondary brain tumor originates through other part of the body. These types of tumors have a cancer cell in the body that is metastatic which spread into different areas of the body like brain, lungs etc. Secondary brain tumor is very malignant. The reason of secondary brain tumor cause is mainly due to lungs cancer, kidney cancer, bladder cancer etc.

MAGNETIC RESONANCE IMAGING (MRI)

Raymond v. Damadian invented the first magnetic image in 1969. In 1977 the first MRI image were invented for human body and the most perfect technique. Because of MRI we are able to visualize the details of internal structure of brain and from that we can observe the different types of tissues of human body. MRI images have a better quality as compared to other medical imaging techniques like X-ray and computer tomography. MRI is good technique for knowing the brain tumor in human body. There are different images of MRI for mapping tumor induced Change including T1 weighted, T2 weighted and FLAIR (Fluid attenuated inversion recovery) weighted shown in figure.

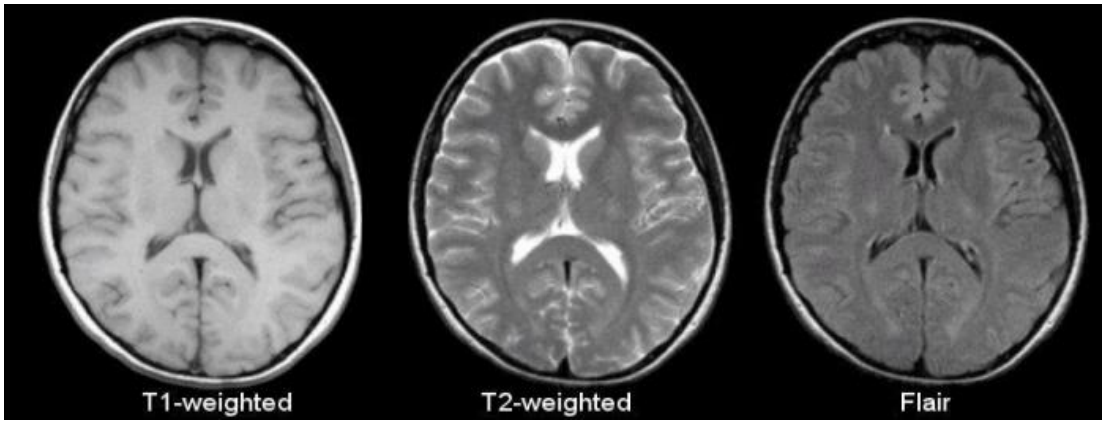

Fig 2: T1, T2 and Flair image

The most common MRI sequence is T1 weighted and T2 weighted. In T1 weighted only one tissue type is bright FAT and in T2 weighted two tissue types are Bright FAT and Water both. In T1 weighted the repetition time (TR) is short in T2 weighted the TE and TR is long. The TE an TR are the pulse sequence parameter and stand for repetition time and time to echo and it can be measured in millisecond(ms). The echo time represented time from the centre of the RF pulse to the centre of the echo and TR is the length of time between the TE repeating series of pulse and echo is shown in figure.

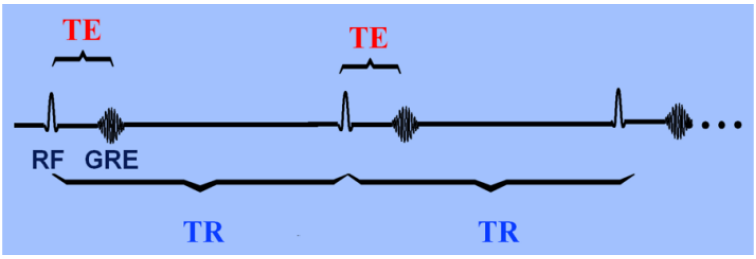

Fig. 3: Graph of TE and TR

The third commonly used sequence in the FLAIR. The Flair sequence is almost same as T2-weighted image. The only difference is TE and TR time are very long. Their approximate TR and TE times are shown in table.

	TR (msec)	TE (msec)
T1-Weighted (short TR and TE)	500	14
T2-Weighted (long TR and TE)	4000	90
Flair (very long TR and TE	9000	114

Fig.4: Table of TR and TE time

To improve the accuracy and speed of brain tumor detection, **deep learning techniques** can be used. In this project, we use **VGG16**, a well-known **Convolutional Neural Network (CNN) model**, to automatically analyse MRI images and classify them as **tumor or non-tumor**. CNN models, like VGG16, can learn patterns from large datasets and make predictions based on image features. This helps in reducing human errors and improving the accuracy of tumor detection.

However, MRI images often contain noise and unclear details, making it difficult for even deep learning models to detect tumors accurately. To solve this issue, **wavelet transform** is used as a preprocessing step. Wavelet transform is a mathematical technique that helps in enhancing important details in the images while reducing unwanted noise. This improves the performance of the VGG16 model by making tumors more visible and easier to detect.

By combining **VGG16 with wavelet transform**, this project aims to create an **efficient and accurate system** for brain tumor detection. The model is trained on an MRI dataset to learn how to differentiate between tumor and non-tumor images. The goal is to assist doctors by providing a **faster and more reliable** method of detecting brain tumors, reducing the workload of medical professionals and minimizing errors in diagnosis.

This approach is an important step towards using **artificial intelligence (AI) in medical imaging**. It not only improves accuracy but also helps in making healthcare more efficient and accessible. With further improvements, such models can be used in hospitals and clinics to support doctors in diagnosing brain tumors more effectively, ultimately saving lives.

1.2. MOTIVATION

Brain tumor detection is a critical challenge in the medical field, as early and accurate diagnosis is essential for effective treatment and improved survival rates. Traditional diagnostic methods often depend on manual interpretation of MRI images by radiologists, which can be time-consuming and prone to human error. With the advancement of technology, there is a growing need for automated systems that can assist medical professionals in identifying brain tumors with higher precision and reliability. This need drives the motivation for exploring innovative approaches using deep learning and advanced image processing techniques.

Deep learning, particularly convolutional neural networks (CNNs), has revolutionized various fields, including medical imaging. Among the many architectures, VGG16 stands out due to its robust ability to analyse and classify images with intricate details. The motivation for using VGG16 lies in its proven efficiency in extracting meaningful patterns from images, making it a strong candidate for brain tumor detection. However, to further enhance its performance, combining it with preprocessing techniques like wavelet transforms can bring additional benefits by improving the quality of features extracted from MRI scans.

Wavelet transforms play a crucial role in medical image analysis by emphasizing important features and reducing noise. This technique allows for a more refined and detailed analysis of MRI images, which is crucial when dealing with subtle and complex brain tumor patterns. The motivation behind integrating wavelet transforms with deep learning models like VGG16 is to harness the strengths of both methods, resulting in a more effective and reliable diagnostic system.

The ultimate goal of this project is to develop an advanced, automated system that leverages VGG16 and wavelet techniques to accurately detect brain tumors. Such a system can significantly aid healthcare professionals by reducing diagnostic time and minimizing errors, while also improving the overall quality of patient care. This motivation drives the research to explore this promising approach, contributing to advancements in medical technology and offering hope for better outcomes in brain tumor treatment.

1.3. PROBLEM DEFINITION

Brain tumors are abnormal growths in the brain that can be life-threatening if not detected early. Doctors use Magnetic Resonance Imaging (MRI) scans to identify tumors, but manual analysis is slow, difficult, and sometimes inaccurate due to human errors. Small or hidden tumors can be missed, leading to delayed treatment. Traditional methods rely on manual diagnosis, which may not always be reliable. Therefore, there is a need for an automated system that can quickly and accurately detect brain tumors.

This project uses deep learning, specifically the VGG16 model, combined with wavelet transform, to detect brain tumors in MRI images. VGG16 is a pre-trained Convolutional Neural Network (CNN) that can learn features from images and classify them as tumor or non-tumor. Wavelet transform helps improve image quality by enhancing important details, making tumor detection more accurate. This system aims to provide a faster and more reliable way to detect brain tumors using deep learning.

1.4. OBJECTIVES

- To develop an automatic system for detecting brain tumors from MRI images.
- To improve accuracy by using the VGG16 model with wavelet transform for better image processing.
- To reduce diagnosis time and assist doctors in making quick and reliable decisions.
- To enhance image features using wavelet transform for better tumor detection.
- To create a deep learning model that can classify MRI images into tumor and non-tumor categories.

1.5. PROJECT SCOPE AND LIMITATIONS

Scope:

- The project focuses on detecting brain tumors in MRI images using deep learning.
- It uses VGG16, a CNN model, to classify images into tumor or non-tumor categories.
- Wavelet transform is applied to improve image quality and enhance tumor details.
- The model is trained and tested using an MRI dataset.
- The system aims to be fast, efficient, and helpful for medical diagnosis.

Limitations:

- The accuracy of the model depends on the quality and size of the dataset used for training.
- The model may not classify different tumor types (benign or malignant) accurately.
- High computational power is required for training deep learning models.
- MRI images from different hospitals may have variations in quality, which can affect performance.

1.6. MODELS FOR PROBLEM SOLVING

1.6.1. Logistic Regression:

Logistic regression is a supervised machine learning algorithm mainly used for binary classification where we use a logistic function, also known as a sigmoid function that takes input as independent variables and produces a probability value between 0 and 1. For example, we have two classes Class 0 and Class 1 if the value of the logistic function for an input is greater than 0.5 (threshold value) then it belongs to Class 1 it belongs to Class 0. It's referred to as regression because it is the extension of linear regression but is mainly used for classification problems. The difference between linear regression and logistic regression is that linear regression output is the continuous value that can be anything while logistic regression predicts the probability that an instance belongs to a given class or not.

Understanding Logistic Regression

- It is used for predicting the categorical dependent variable using a given set of independent variables.
- Logistic regression predicts the output of a categorical dependent variable. Therefore, the outcome must be a categorical or discrete value.
- It can be either Yes or No, 0 or 1, true or False,etc. but insteadof giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.
- Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.
- In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

- The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
- Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.
- Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables.

1.6.2. Support Vector Machine (SVM):

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:

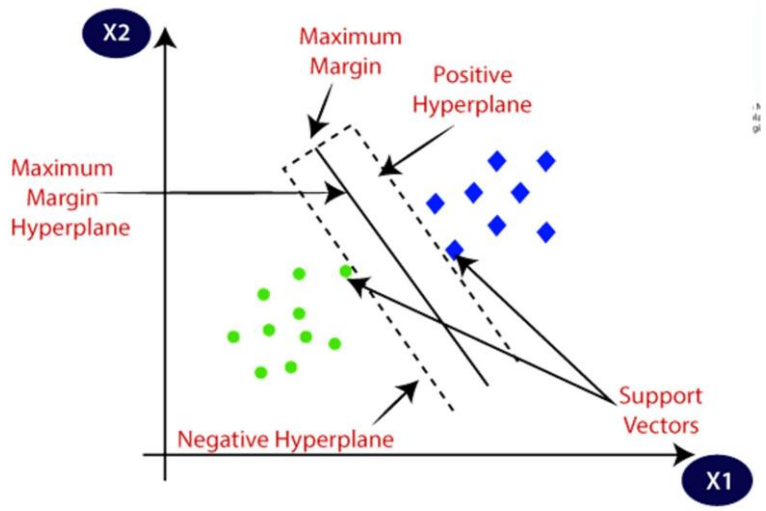

fig.5: SVM Model Diagram

1.6.3. Convolutional Neural Network (CNN):

Convolutional Neural Networks (CNNs) are a specialized class of neural networks designed to process grid-like data, such as images. They are particularly well-suited for image recognition and processing tasks.

They are inspired by the visual processing mechanisms in the human brain, CNNs excel at capturing hierarchical patterns and spatial dependencies within images.

Working of CNN model

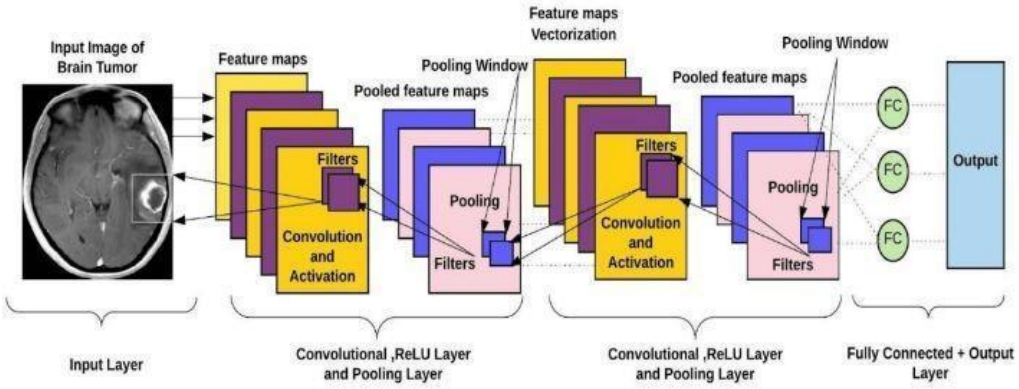

Fig.6.Working of CNN model for brain tumor detection

➢ Layer of CNN model:

- Convolution 2D
- MAX Poolig2D
- Dropout
- Flatten
- Dense
- Activation

➢ Convolution 2D: In the Convolution 2D extract the featured from input image. It given the output in matrix form.

➢ MAX Poolig2D: In the MAX polling 2D it takes the largest element from rectified feature map.

➢ Dropout: Dropout is randomly selected neurons are ignored during training.

➢ Flatten: Flatten feed output into fully connected layer. It gives data in list form.

➢ Dense: A Linear operation in which every input is connected to every output by weight. It followed by nonlinear activation function.

➢ Activation: It used Sigmoid function and predict the probability 0 and 1.

➢ In the compile model we used binary cross entropy because we have two layers 0 and 1.

➢ We used Adam optimizer in compile model. Adam: -Adaptive moment estimation. It used for non-convex optimization problem like straight forward to implement.

- Computationally efficient.
- Little memory requirement.

1.6.4.VGG 16:

The VGG-16 model is a convolutional neural network (CNN) architecture that was proposed by the Visual Geometry Group (VGG) at the University of Oxford. It is characterized by its depth, consisting of 16 layers, including 13 convolutional layers and 3 fully connected layers. VGG-16 is renowned for its simplicity and effectiveness, as well as its ability to achieve strong performance on various computer vision tasks, including image classification and object recognition. The model's architecture features a stack of convolutional layers followed by max-pooling layers, with progressively increasing depth. This design enables the model to learn intricate hierarchical representations of visual features, leading to robust and accurate predictions. Despite its simplicity compared to more recent architectures, VGG-16 remains a popular choice for many deep learning applications due to its versatility and excellent performance.

VGG Architecture:

The VGG-16 architecture is a deep convolutional neural network (CNN) designed for image classification tasks. It was introduced by the Visual Geometry Group at the University of Oxford. VGG-16 is characterized by its simplicity and uniform architecture, making it easy to understand and implement.

The VGG-16 configuration typically consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. These layers are organized into blocks, with each block containing multiple convolutional layers followed by a max-pooling layer for down Sampling.

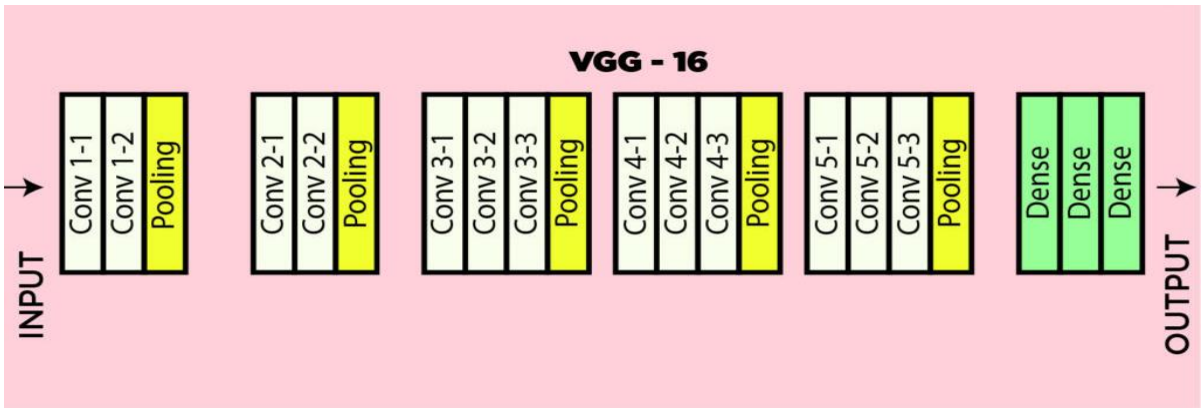

Fig.7: VGG-16 architecture

CHAPTER-2

RELATED WORKS

2.1. LITERATURE REVIEW

Various algorithms have been scrutinized for their effectiveness in brain tumor detection, and a selection of them is expounded upon in this section. Numerous research papers have surfaced, emphasizing the integration of artificial intelligence (AI), expert systems, and neural networks to enhance the accuracy of brain tumor detection. Recent strides have been witnessed in medical canters and hospitals, incorporating AI systems and applications across disciplines to bolster disease detection precision. Several methodologies and models have been introduced, significantly contributing to heightened diagnostic efficiency.

Tanzil et al. [1] proposed a hybrid model amalgamating artificial and deep learning features. Manual and deep learning features were extracted from segmented images, optimized through entropy. The optimized features were then consolidated using a sequential concatenation approach to form a single feature vector. The resulting method was developed independently and tested on diverse measurement databases, such as BRATS 2015-17. The highest accuracy and Dice Similarity Coefficient (DSC) achieved were 98.78% and 96.36% in BRATS 2015, 99.63% accuracy in BRATS 2016, and 99.59% DSC in BRATS 2017. Notably, BRATS 2018 reported an accuracy of 99.67% and a DSC of 99.80%.

Wozniak et al. [2] introduced an advanced correlation learning method (CLM) for a deep neural network architecture that merges Convolutional Neural Networks (CNNs) with traditional architectures. Analyzing 3064 brain cancers, including meningioma, glioma, and pituitary tumors, the CLM model achieved an accuracy, precision, and recall of approximately 96%.

Nayak et al. [3] proposed a dense CNN-based EfficientNet for brain tumor image detection using MRI. Comparative analyses with ResNet-50, MobileNet, and MobileNetV2 revealed EfficientNet's superior performance. Following rigorous training, the EfficientNet model demonstrated an accuracy of 98.78% and an F1 score of 98.0%, utilizing four different types of MRI and a dataset comprising 3260 MR image.

Almahoun et al. [4] presented a deep learning model for brain tumor diagnosis using MRI datasets. Additionally, they employed four other training models, namely VGG16, MobileNet, ResNet-50, and Inception V3. The deep learning model outperformed, achieving 100% training

accuracy and 98% test accuracy with a dataset of 10,000 MR images categorized into brain tumors and non-brain tumors. Shanmugapriya et al. conducted a glioma tumor classification, distinguishing between benign and malignant tumors, with sensitivity, specificity, and specificity values of 100%,97.2%, and 90%, respectively.

Musallam et al. [5] explored the application of deep learning in clinical image analysis, particularly in brain tumor detection from MRI images. The comprehensive review covered various deep neural network architectures, image processing techniques, and feature extraction methods, offering insights into the current research landscape in this domain.

Mahmoud et al. [6] advocated for the use of deep learning and machine learning for early brain tumor detection from MRI images. Comparisons between different models, including VGG16, ResNet-50, and Inception V3, favored the CNN architecture. The review emphasized the significance of variable learning in diagnosis and pathology.

Alsaif et al. [7] furnished a robust review of deep learning applications in brain tumor detection and classification. Highlighting the use of popular CNN models like VGG-16 and data augmentation techniques, their proposed method showcased superior performance in classifying brain MRI images as normal or abnormal, outperforming other training models in terms of true, real, negative, and F1 scores. The research was supported by the Deanship of Scientific Research, Hail University, Saudi Arabia, with available data on Kaggle, and the article incorporated a comprehensive reference list spanning data mining, image classification, and deep learning applications in medicine.

2.2. SUMMARY OF PREVIOUS WORKS

In previous studies, researchers have used deep learning methods, particularly convolutional neural networks (CNNs), to detect brain tumors in MRI images. Among these, VGG16—a popular CNN model—has been extensively applied due to its ability to identify complex patterns in medical imaging. Additionally, wavelet transforms have been employed as preprocessing techniques to enhance image features, making it easier for models like VGG16 to detect abnormalities. Many researchers have adopted transfer learning with VGG16, where pre-trained models are customized for brain tumor detection, saving time and reducing the need for extensive datasets. This combination of VGG16, wavelet techniques, and deep learning has

shown great promise in assisting healthcare professionals to diagnose brain tumors more accurately and efficiently, improving patient outcomes.

Table-2.1: Summary of related works

Sl.no.	Year	Author	Title	Models	Limitations	Accuracy
1.	2019	Tanzil Saba, A. S. Mohamed, M.El-Affendi, J.Amin, M .Sharif.	Brain Tumor Detection Using Fusion Of Hand Crafted and Deep Learning Features.	Grab cut method for segmentation, VGG-19(CNN) model fused with LBP and HOG (Hand Craft Features)	2Handcrafted features are manually designed and may not capture all the relevant information in the data.	99%
2.	2021	Woźniak, M., Silka, J. & Wieczorek, M	Deep neural network correlation learning mechanism for CT brain tumor detection	Correlation learning mechanism (CLM) with CNN	Should consider more datasets to obtain more accurate and promising results	96%
3.	2022	Dilip Ranjan Nayak, N.M. Padhy, Pradeep Kumar Mallick, Mikhail Zymbler, and Sachin Kuma	Brain Tumor Classification Using Dense Efficient-Net	Dense EfficientNet (proposed), ResNet 50, MobileNet, MobileNetV2	Higher number of parameters and evaluation time	98.78%
4.	2022	Al Madhoun H.R. Abu- Naser, S.S.	Detection of Brain Tumor Using Deep Learning	Deep educational model (proposed), VGG16, ResNet-50, MobileNet, Inception V3	Need to apply image augmentation methods	98.28%

5.	2022	Musallam, Ahmed S., Ahmed S. Sherif, Mohamed K. Hussein	A new convolutional neural network architecture for automatic detection of brain tumors in magnetic resonance imaging images	Deep Convolutional Neural Network models such as VGG16, VGG19, and hybrid CNN-SVM	Less number of max pooling layers used.	98.22%
6.	2023	Mahmud, Md Ishtyaq, Muntasir Mamun, and Ahmed Abdel Gawad	A deep analysis of brain tumor detection from MRI images using deep learning networks	VGG16, MobileNet, ResNet-50, and Inception V3	Training process took a long time due to the numerous layers in the CNN model and the lack of a good GPU	93.3%
7.	2023	E. Shanmugapriya, O. Rajasekar	Machine Learning Based Approach for Brain Tumor Detection	SVM used for classification and CNN used for segmentation	The model's performance and accuracy may be limited to the specific tumor categories and segmentation technique used in the study	97.2%
8.	2024	Mathivanan et al.	Employing deep and transfer learning for accurate brain tumor detection.	MobileNetV3, DenseNet169, VGG19, ResNet152	Validation accuracy fluctuations in some models; potential overfitting due to dataset size	Up to 98.5%

9.	2024	Gunasekaran et al.	Automated brain tumor diagnostics: Empowering neuro-oncology with deep learning-based MRI image analysis	ConvNet-ResNeXt101 with Advanced Whale Optimization for feature selection	High computational complexity; requires substantial computational resources	99.27%
10.	2024	Ashafuddula et al.	ContourTL-Net: Contour-Based Transfer Learning Algorithm for Early-Stage Brain Tumor Detection	Contour-Based Transfer Learning Algorithm	Potential overfitting due to limited dataset size	99.94%
11.	2024	Tummala et al.	An improved deep learning-based hybrid model with ensemble techniques for brain tumor detection from MRI image	Ensemble of 2D CNN (9-layer), 2D CNN (13-layer), and 2D CNN LSTM models	High computational complexity; potential overfitting	98.82%
12.	2024	Vivek Agrawal, Kuldeep Singh Kaswan	Convolution Neural Network Based Brain Tumor Detection using MRI Scans	Wavelet Transform + CNN	Potential overfitting due to limited dataset size	96.69%
13.	2024	Chappidi Sree Teja Reddy, Geetha Ramalingam	Analysis and Comparison of Accuracy in Brain Tumor using Berkeley Wavelet Transform and Robust Principal Component Analysis	Berkeley Wavelet Transform (BWT) and Robust Principal Component Analysis (ROBPCA)	Small sample size; limited to specific tumor types	BWT: 81.5%, ROBPCA: 84%
14.	2024	Asiri et al.	Enhancing brain tumor diagnosis: an optimized CNN hyperparameter modelfor improved.	Hyperparameter-tuned CNN	Requires careful hyperparameter tuning	97%

CHAPTER-3

WORKFLOW PROCESS EXPLANATION

The proposed system for brain tumor classification using deep learning and image processing techniques follows a structured workflow divided into several key stages. Each stage contributes critically to the overall performance and accuracy of the model. The following outlines the step-by-step process illustrated in the workflow diagram:

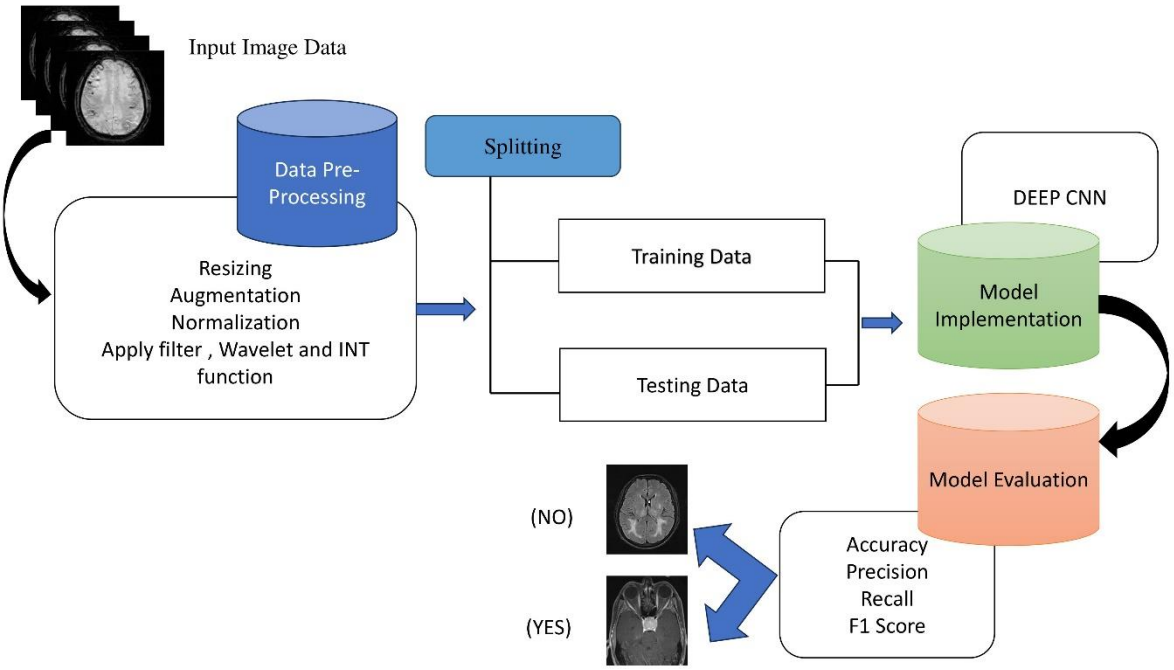

Fig - 8: Workflow for Detection

3.1. INPUT IMAGE DATA

The workflow begins with the collection of brain MRI scans. These images serve as the primary input data for the model. The dataset consists of both tumor and non-tumor images. The dataset consists of 1,222 brain MRI images which serve as the base input for the deep learning model.

- Initial Input: started with 1222 MRI images of the brain.
- These images include both tumor-affected and normal brain scans. The goal is to detect the presence of brain tumors from these images.

3.2. DATA PREPROCESSING

In this project, multiple image preprocessing techniques were applied to prepare MRI brain scan images for input into the VGG-16 convolutional neural network model. The preprocessing pipeline consisted of resizing, augmentation, normalization, filtering, wavelet transformation, and intensity transformation functions. Each of these steps was implemented sequentially to enhance the dataset's quality and optimize it for deep learning-based classification.

3.2.1. Resizing

All MRI images were resized to a fixed dimension of **224×224×3 pixels**. This was performed using OpenCV's cv2.resize() function. The resizing ensured that each image had consistent dimensions and matched the input requirements of the VGG-16 model. The original images, which were in varying resolutions and sizes, were read using cv2.imread(), and each was resized as follows:

```
img = cv2.imread('path_to_image')

img = cv2.resize(img, (224, 224))
```

The resized images retained the RGB channels and were stored as NumPy arrays for further processing. This uniform resizing ensured compatibility with the convolutional layers of the VGG-16 model, which expect fixed-size input.

3.2.2. Augmentation

Data augmentation was performed using the Image Data Generator class from Keras. Augmentation was applied to increase the diversity of the training dataset and simulate real-world image variations. The following augmentation transformations were included:

- Rotation of images within a 20-degree range.
- Horizontal flipping of images.
- Width and height shifting by 20%.
- Shear transformation up to 20%.
- Zooming in and out by 20%.
- Nearest-neighbour interpolation for filling empty pixels.

These transformations were applied in real-time during the training process, and the augmented images were not stored separately. Instead, the flow() function of the generator was used to yield augmented images in batches.

3.2.3. Normalization

Normalization was performed using the preprocess input() function from the Keras VGG-16 module. This function scales pixel intensity values from the original [0, 255] range to a format compatible with the pretrained VGG-16 model. The normalization centered the RGB channels based on ImageNet training statistics. The images were first converted into NumPy arrays and then passed through the preprocessing function as follows:

```
from tensorflow.keras.applications.vgg16 import preprocess_input

X_vgg_wavelet = preprocess_input(X_vgg_wavelet)
```

This step standardized the input images before being fed into the deep learning model. The output of this function was used directly for training and validation.

3.2.4. Filtering

Filtering was used to improve the visual quality of MRI images by enhancing features or reducing noise. Although several types of filters are available (such as Gaussian blur, sharpening filters, and median filters), a basic filter was optionally applied to sharpen the image and make the edges of tumor regions more distinct. An example of a Gaussian filter implementation is shown below:

```
img_filtered = cv2.GaussianBlur(img, (3, 3), 0)
```

The filtering step helped in reducing background noise and improving the clarity of tumor boundaries. This step was optional and could be used in conjunction with the wavelet transform or independently depending on the noise level of the raw input images.

3.2.5. Wavelet Transform

The wavelet transform was applied to decompose each image into its frequency components to preserve spatial as well as textural details. The **Discrete Wavelet Transform (DWT)** was used, specifically with the Daubechies wavelet family (db1). The DWT was applied independently to each of the three color channels of the image (Red, Green, and Blue).

The implementation steps included:

1. Reading and resizing the image.

2. Applying DWT to each channel using pywt.wavedec2() to obtain approximation and detail coefficients.

3. Reconstructing the image using pywt.waverec2() to form a wavelet-enhanced image.

The output image, enhanced using wavelet transform, retained the shape (224, 224, 3) and was fed into the CNN after normalization. This step significantly enhanced texture and boundary features of tumors, which were crucial for accurate classification.

3.2.6. Intensity Transformation (INT Function)

An intensity transformation function was applied to improve contrast and brightness distribution within each MRI image. INT functions transform pixel values to spread intensity values across a broader range, enhancing the visibility of tumors. Depending on the type of intensity transformation required, methods like logarithmic transformation, histogram equalization, or gamma correction were used.These transformations improved the visual quality and pixel distribution, making it easier for the CNN to extract meaningful features. The transformed images were optionally converted back to three-channel format if required by the CNN input.

3.3. DATA SPLITTING

After all preprocessing steps were completed, the dataset was split into training and testing subsets to train the model and evaluate its performance on unseen data. This ensures that the model does not memorize the dataset and can generalize well to new images.

The splitting was done using an **80:20 ratio**:

- **80%** of the data was used to train the model.
- **20%** was reserved to test the model's performance.

Before splitting, the class labels (no_tumor, pituitary_tumor) were encoded into numerical format:

- no_tumor → 0
- pituitary_tumor → 1

These integer labels were then converted into one-hot encoded vectors using TensorFlow's to_categorical() function, preparing them for use in the final softmax layer of the model.

3.4. MODEL IMPLEMENTATION

The brain tumor detection model was implemented using a deep learning-based convolutional neural network (CNN) architecture, specifically the **VGG-16** model enhanced with custom layers for binary classification. The implementation involved the following steps:

Selection of Base Model

The **VGG-16 model** was selected as the base architecture due to its strong performance in image classification tasks. VGG-16 is a 16-layer CNN that was pre-trained on the ImageNet dataset. The top (fully connected) layers of the original model were removed using the parameter include_top=False, allowing for the integration of a custom classification head.

Freezing Pretrained Layers

The convolutional layers of the VGG-16 base model were **frozen** using base model.trainable = False. This prevented their weights from being updated during training and preserved the general features learned from the ImageNet dataset. Only the custom classification layers on top of the base model were trained on the brain MRI dataset.

Adding Custom Layers

On top of the frozen VGG-16 base, a custom sequence of dense (fully connected) layers was added to perform binary classification. The layers added were:

- **Flatten layer** to convert 3D feature maps to 1D vectors.
- **Dense layer** with 512 neurons and ReLU activation.
- **Dropout layer** with a rate of 0.5.
- **Dense layer** with 256 neurons and ReLU activation.
- **Dropout layer** with a rate of 0.5.
- **Dense layer** with 128 neurons and ReLU activation.
- **Dropout layer** with a rate of 0.5.
- **Final dense output layer** with 2 neurons (for 2 classes) and softmax activation.

Enhanced Convolutional Blocks

To extend the capability of the VGG-16 model, multiple additional convolutional layers were added after the base. These layers further captured complex image features specific to tumor detection:

- **Two convolutional layers** with 512 filters and 3×3 kernel size.
- **Batch normalization** after each convolution to stabilize learning.
- **Max pooling** to reduce spatial dimensions.

These layers were added using the Sequential API from TensorFlow Keras.

3.5. MODEL EVALUATION

After training the enhanced VGG-16 model for brain tumor classification, the model's performance was evaluated using various statistical and graphical metrics to assess its accuracy and reliability on unseen data. The evaluation was performed on the 20% test dataset that was set aside during the data splitting phase.

Accuracy and Loss

The model was evaluated using the evaluate() function on the test dataset. It achieved a high test accuracy of **99.18%**, indicating strong classification performance.

Classification Report

A detailed classification report was generated using scikit-learn's classification_report() function. This included precision, recall, and F1-score for both classes (No Tumor, Pituitary Tumor) The high values for precision and recall reflect the model's ability to detect both tumor and non-tumor cases with minimal false positives or false negatives.

Confusion Matrix

A confusion matrix was plotted to visualize the classification results. The matrix showed the number of true positives, true negatives, false positives, and false negatives, giving insight into the types of errors made by the model.

ROC Curve and AUC

The Receiver Operating Characteristic (ROC) curve was plotted to analyze the trade-off between true positive rate and false positive rate. The Area Under the Curve (AUC) was calculated to quantify overall classification performance.

The AUC score was approximately **0.97**, confirming that the model performs well in distinguishing between the two classes.

Precision-Recall Curve

A precision-recall curve was also plotted to further validate the classification performance, especially in the presence of class imbalance.

Final Output – Tumor Detection

After implementing and evaluating the brain tumor detection model, the final output would typically be based on the evaluation metrics and the predictions from the model.

For a given test image:

- If the model detects a tumor, the output would be: YES (Tumor Detected).
- If the model does not detect a tumor, the output would be: NO (No Tumor Detected).

CHAPTER-4

MODEL EXPLANATION

4.1. INTRODUCTION TO CONVOLUTIONAL NEURAL NETWORKS (CNNS)

A **Convolutional Neural Network (CNN)** is a class of deep learning algorithms particularly effective for processing data with a grid-like topology, such as images. CNNs are composed of multiple layers, including **convolutional layers**, **pooling layers**, and **fully connected layers**, each responsible for different feature extraction and transformation processes.

Mathematically, a CNN layer can be described as:

$$Z_{\{i,j\}}^{\{(k)\}} = \left(X * W^{\{(k)\}}\right)_{\{i,j\}} + b^{\{(k)\}}$$

Where:

- X is the input image or feature map.
- $W^{\{(k)\}}$ is the filter (kernel) of the kth feature map.
- $b^{\{(k)\}}$ is the bias term.
- $*$ denotes the convolution operation.
- $Z_{\{i,j\}}^{\{(k)\}}$ is the output of the convolution at position (i,j).

After the convolution, a non-linear activation function such as ReLU is applied:

$$A_{\{i,j\}}^{\{(k)\}} = \backslash text\{ReLU\}\left(Z_{\{i,j\}}^{\{(k)\}}\right) = \backslash max\left(0, Z_{\{i,j\}}^{\{(k)\}}\right)$$

4.2. OVERVIEW OF VGG-16

The VGG-16 model was proposed by Karen Simonyan and Andrew Zisserman from the Visual Geometry Group (VGG) at the University of Oxford in 2014. It was developed for the ILSVRC (ImageNet Large Scale Visual Recognition Challenge) and achieved top performance.

VGG-16 Architecture

- **Input size**: 224×224×3224
- **Number of parameters**: Approximately 138 million
- **Total layers**: 16 weight layers (13 convolutional + 3 fully connected)
- **Kernel size**: All convolutional filters are 3×33 \times 33×3
- **Pooling size**: 2×22 \times 22×2 with stride

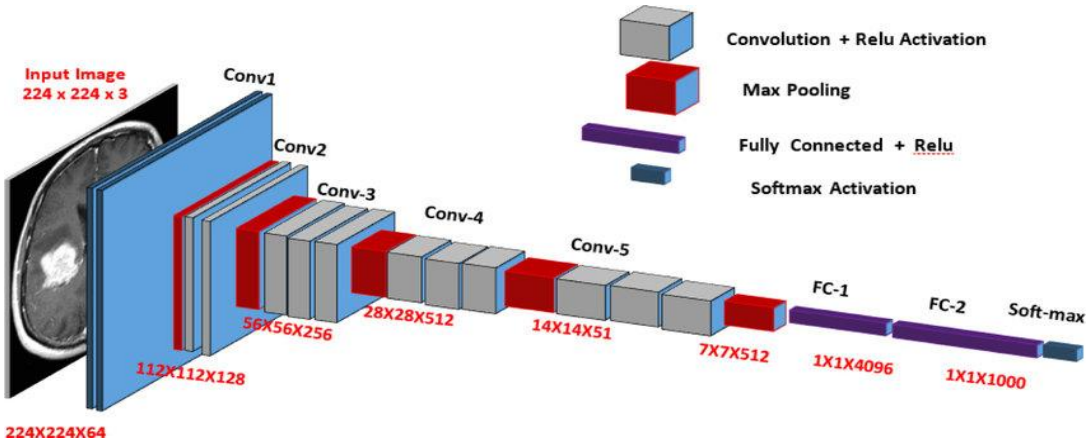

Fig - 9: VGG-16 Architecture

4.3. LAYER-WISE BREAKDOWN OF VGG-16

$$\text{Input image} \in R\ 224 \times 224 \times 3$$

This represents a color image of 224×224 pixels with 3 color channels (RGB)

Block 1

- **Conv1_1**: 646464 filters of size 3×33 \times 33×3, stride 1, padding 1

  Output shape: 224×224×64

- **Conv1_2**: Same as above
- **Max Pooling**: 2×22 stride 2

  Output shape: 112×112×64

Block 2

- **Conv2_1 & Conv2_2**: 128 filters of size 3×3
- **Max Pooling**: 2×2 , stride 2

  Output shape: 56×56×128

Block 3

- **Conv3_1, Conv3_2, Conv3_3**: 256 filters of size 3×3
- **Max Pooling**: 2×2, stride 2

  Output shape: 28×28×256

Block 4

- **Conv4_1, Conv4_2, Conv4_3**: 512 filters of size 3×3
- **Max Pooling**: 2×22 \times 22×2, stride 2

  Output shape: 14×14×512

Block 5

- **Conv5_1, Conv5_2, Conv5_3**: 512 filters of size 3×3
- **Max Pooling**: 2×2, stride 2 Output shape: 7×7×512

Flattening Layer

Input shape: 7×7×512=25088

Output vector: $R25088$

Fully Connected Layers

FC1: **Input: 25088 Output: 4096**

$$Z^{\{(1)\}} = W^{\{(1)\}x} + b^{\{(1)\}}$$

FC2: **Input: 4096 Output: 4096**

$$Z^{\{(2)\}} = W^{\{(2)\}A^{\{(1)\}}} + b^{\{(2)\}}$$

FC3 (Output Layer):

Input: 4096 Output: 1000

$$\boldsymbol{Z^{\{(3)\}} = W^{\{(3)\}}A^{\{(2)\}} + b^{\{(3)\}}}$$

Softmax Output

The final output of VGG-16 is a probability distribution across 1000 classes using the softmax function:

$$\{y\}_i = \left\{e^{\{z_i\}}\right\}\left\{\sum_{\{j=1\}}^{\{1000\}e^{\{z_j\}}}\right\}, \qquad i = 1, 2, \ldots, 1000$$

Where:

- $\{y\}_i$ *is* the probability that the input belongs to class i
- $\{z\}_i$ is the score (logit) for class i

Loss Function

Typically, VGG-16 uses **cross-entropy loss** for multi-class classification

$$L = -\sum_{j=1}^{t} y_{ij} * \log\left(\widehat{y_{ij}}\right)$$

where, i: *number of rows*;

j: *number of categories*;

y_i *is one hot encoded terget vector*

Top-5 Accuracy

In classification tasks like ImageNet, top-5 accuracy is often used:

- Sort predicted class scores
- Pick top-5 classes C=[c1,c2,c3,c4,c5]
- If the ground truth label G∈C it's considered correct

The **error function** for Top-5:

$$E = \{1\}\{n\} \sum_{k=1}^{l^{(n)}} \min d(c_l, G_k)$$

4.4. VGG-16 PARAMETERS AND SUMMARY

Layer Type	Filter Size	Filters	Output Size
Conv1_1	3×3	64	224×224×64
Conv1_2	3×3	64	224×224×64
MaxPooling	2×2	-	112×112×64
Conv2_1	3×3	128	112×112×128
Conv2_2	3×3	128	112×112×128
MaxPooling	2×2	-	56×56×128
Conv3_1,2,3	3×3	256	28×28×256
MaxPooling	2×2	-	28×28×256
Conv4_1,2,3	3×3	512	14×14×512
MaxPooling	2×2	-	14×14×512
Conv5_1,2,3	3×3	512	7×7×512
MaxPooling	2×2	-	7×7×512
Flatten	-	-	25088
Fully Connected 1	-	-	4096
Fully Connected 2	-	-	4096

Layer Type	Filter Size	Filters	Output Size
Fully Connected 3	-	-	1000
Softmax	-	-	1000 (class scores)

$$d(ci, Gk) = \{0,1, if\ ci = Gk, otherwise$$

Table-4.1: Parameters of VGG-16

4.5. KEY CHARACTERISTICS AND FEATURES

4.5.1 Convolutional Layers with Small Filters:

VGG-16 uses very small 3x3 convolution filters throughout the network. This is different from some other architectures that use larger filters (like 5x5 or 7x7). The benefit of using 3x3 filters is that they allow the network to capture detailed spatial features while still maintaining a manageable number of parameters.

Use of Max Pooling:

Max-pooling is used after each block of convolutional layers to reduce the spatial dimensions and focus on the most important features. A 2x2 max-pooling operation is used to reduce the spatial resolution of the feature maps, while retaining important information.

ReLU Activation:

The activation function used in the convolutional layers is ReLU, which is applied after each convolution operation. ReLU introduces non-linearity, allowing the network to learn more complex patterns and ensuring that the network is not just a linear model.

Fully Connected Layers:

The fully connected layers at the end of the network provide high-level reasoning and decision-making. The large number of neurons in these layers (4096 each) helps the network capture complex relationships between features.

SoftMax Output:

For multi-class classification tasks (such as the original ImageNet dataset with 1000 classes), the SoftMax activation function is applied to the final output layer to convert raw logits into probabilities. For binary classification tasks (such as tumor detection), the output layer would be adjusted accordingly (usually to 2 neurons for binary output).

4.6. APPLICATIONS OF VGG-16

Image Classification:

VGG-16 is widely used for tasks where image classification is required, such as classifying objects in images (e.g., animals, vehicles).

Transfer Learning:

VGG-16 is frequently used as a base model for transfer learning. Pretrained VGG-16 models are fine-tuned for a specific task (such as detecting tumors in MRI scans or identifying objects in satellite images).

Medical Image Analysis:

VGG-16 can be adapted for analyzing medical images, such as detecting tumors, lesions, or other abnormalities in MRI scans, X-rays, and CT scans.

CHAPTER-5

RESULT AND PERFORMANCE

Our research that involves deep learning methodologies demonstrates that the deep learning models can significantly enhance predictability and accuracy. The results reveal varied performance among different classifiers, reflecting their distinct strengths and weaknesses.

The Enhanced VGG16 model has the highest performance rate with an accuracy rate of 99% as compared to the VGG16 with wavelet having an accuracy of 96% and VGG16 without wavelet having an accuracy of 95% and also other models like CNN,SVM and logistic regression showing an enhanced model performance .

Table containing model performance metrices.

MODEL	ACCURACY	PRECISION	RECALL	F1 SCORE
LOGISTIC REGRESSION	90.61%	91.66%	96.34%	93.22%
LOGISTIC REGRESSION WITH WAVELET	94.29%	93.10%	98.78%	95.86%
SVM	94.29%	94.64%	96.95%	95.78%
SVM WITH WAVELET	94.69%	94.67%	97.56%	96.10%
CNN	91.02%	92.77%	93.90%	93.93%
CNN WITH WAVELET	96.73%	100.00%	95.12%	96.73%
CNN WITH WAVELET AND INT FUNCTION	98.27%	98.78%	98.78%	98.78%
VGG16	95.51%	96.00%	96.00%	95.00%
VGG16 WITH WAVELET	96.33%	96.00%	96.00%	96.00%
ENHANCED VGG16 WITH WAVELET AND INT FUNCTION	99.18%	99.00%	99.00%	99.00%

Table -5.1: All model results table

Table containing description of existing deep learning models for comparison:-

AUTHORS	YEAR	MODEL	ACCURACY	PRECISION	RECALL	F1-SCORE
Badza et al.	2020	CNN	96.56%	94.81%	94.81%	94.94%
Hashemzehi et al.	2020	CNN and NAND	96.00%	94.49%	94.49%	94.56%
Díaz-Pernas et al.	2021	Multi-scale CNN	97.00%	95.80%	95.80%	96.07%
Sajja et al	2021	Deep-CNN	97.00%	97.05%	97.05%	97.05%
D. R Nayak , N.M Padhy , P.K Mallick , Mikhail Zymbler and Sachin Kumar	2022	Dense Efficient-Net	98.78%	98.75%	98.75%	98.75%
Baiju Babu Vimala et al	2023	EfficientNet-B2	99.06%	98.73%	99.13%	98.79%
Francisco Javier Dfaz-Pernas et al	2024	Multiscale-CNN	97.30%	-	-	-
Proposed Model	Present	Enhanced VGG-16 with wavelet and INT function	99.18%	99.00%	99.00%	99.00%

Table-5.2: Comparison Table

Precision is a measure that assesses the prediction accuracy based on the ratio of correctly classified positive examples to all predicted positive examples. It is calculated by dividing the number of true positive predictions by the number of true positive plus false positive predictions. High precision value shows that the model performed well.

Precision is represented as:

$$Precision = \frac{TP}{TP+FP'} \qquad (1)$$

where TP is true positive and FP is false positive.

Recall is the proportion of all classes that are positively classified with a positive result that is predicted correctly. A well-performing model should possess a high recall. Recall is given as follows:

$$Recall = \frac{TP}{TP+FN'} \quad (2)$$

where FN is a false negative.

An F1-score of high value means precision and recall are high sincethe score is possessed of information regarding these two variables. It is given as follows:

$$F1 = \frac{2 \times precision \times Recall}{Precision + Recall} \quad (3)$$

The ROC curve is a visual plot of a model's capacity to differentiate between separate classes. It graphs the True Positive Rate (Sensitivity) versus the False Positive Rate (1 − Specificity) at different threshold settings. An ideal model would end up in the top-left corner, reflecting a high True Positive Rate and low False Positive Rate.

$$TPR = \frac{TP}{TP+FN} \quad (4)$$

against

$$FPR = \frac{TP}{TP+FP'} \quad (5)$$

where the following definitions are used:

- TPR: True Positive Rate (Sensitivity or Recall).
- FPR: False Positive Rate.
- TP: True Positives.
- FP: False Positives.
- TN: True Negatives
- FN: False Negatives.

AUC stands for area under the ROC curve and offers a single value to quantify the model's performance. AUC has a value between 0 and 1, in which 1 represents an ideal model, 0.5 represents no discriminative capability (as the same as guesswork) and 0 suggests an entirely incorrect model.

$$AUC = \int_0^1 TPR(FPR)\, d(FPR) \quad (6)$$

where the following definitions are used:

- TPR(FPR): The True Positive Rate as a function of the False Positive Rate.
- d(FPR): The infinitesimal change in the False Positive Rate.

Precision–Recall curve is a graph of Precision (Positive Predictive Value) vs Recall (Sensitivity) at various threshold levels. Precision quantifies the percentage of true positive predictions out of all positive predictions, whereas Recall calculates the proportion of true positives out of total actual positives. It is particularly useful when working with unbalanced data sets, when one of the classes occurs considerably more than the other.

5.1. CONFUSION MATRIX

The confusion matrices showcase the performance of different machine learning and deep-learning models for classifying brain tumors, particularly distinguishing between “No Tumor” and “Pituitary Tumor” classes. Here's an analytical summary of each model's performance based on the visual data:

Here's a detailed analysis of each figure (10 to 19) based on the confusion matrixs:

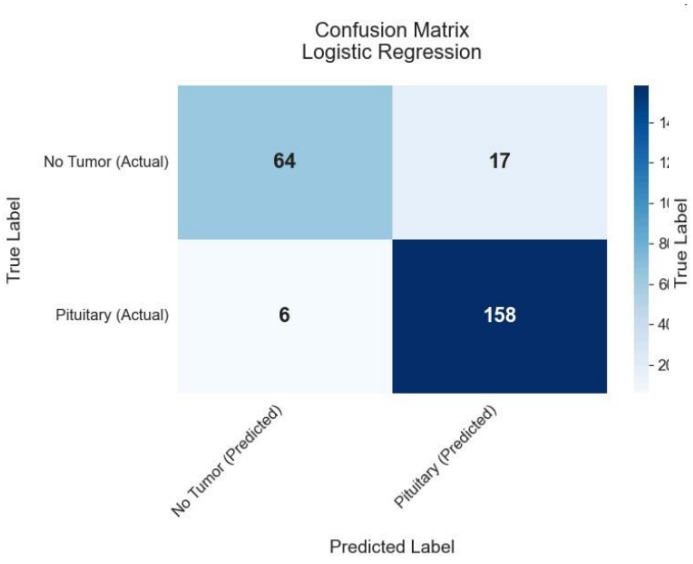

Fig. 10: Standard Logistic Regression

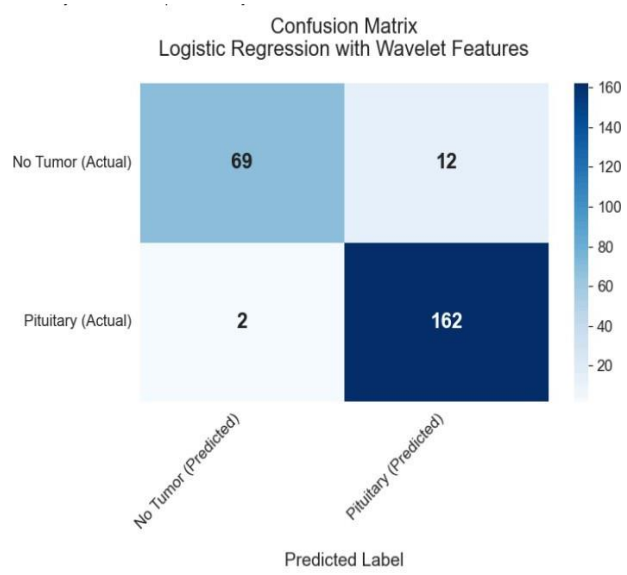

fig.11: Logistic Regression with wavelet

In Logistic Regression (Fig-10) shows reasonable performance with 64 true negatives and 158 true positives, but also has 17 false positives and 6 false negatives. When Wavelet Features are introduced (Fig-11), the model improves significantly false positives drop to 12, and false negatives to 2 highlighting the benefits of wavelet-based feature enhancement in improving classification accuracy.

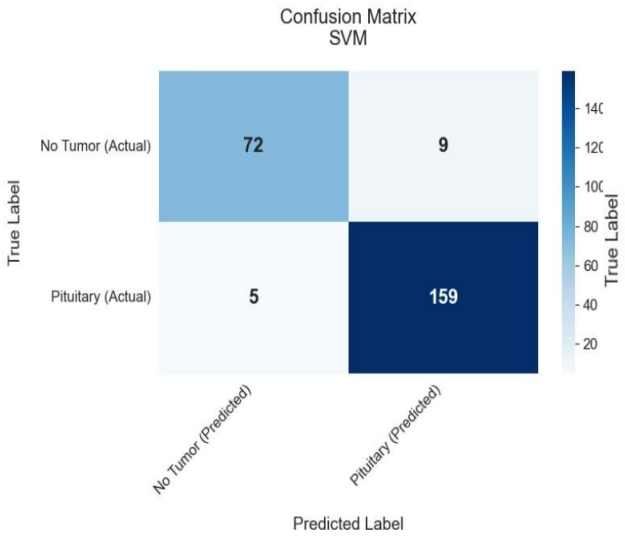

Fig.12:Standard SVM

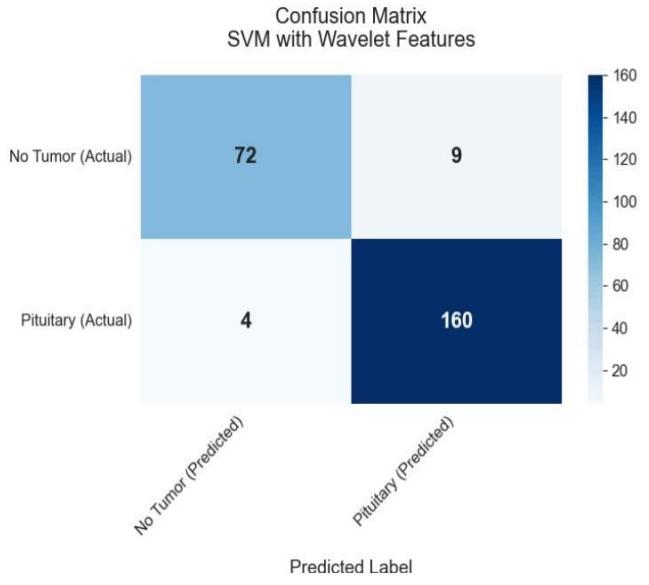

Fig.13: SVM with wavelet

The Support Vector Machine (SVM) model (Fig-12) performs well, with only 9 false positives and 5 false negatives. This performance is slightly enhanced in the SVM with Wavelet Features (Fig-13), where false negatives are further reduced to 4, showcasing wavelet transforms again as a helpful preprocessing step.

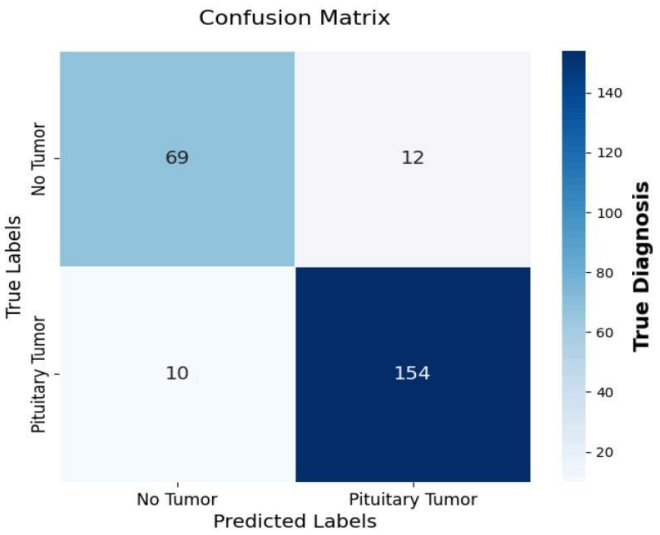

Fig.14: Standard CNN

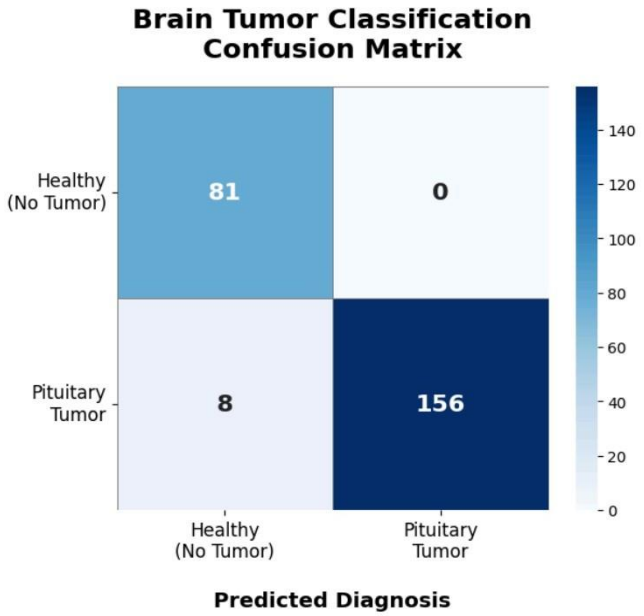

Fig.15: CNN with wavelet

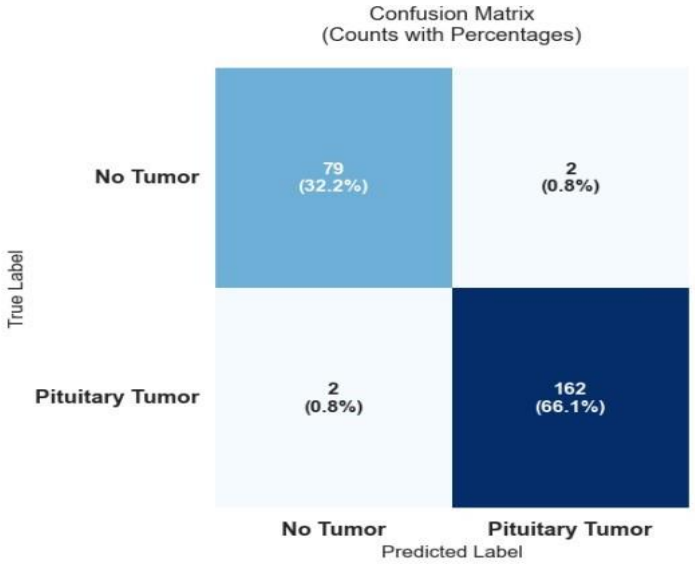

Fig.16: CNN with wavelet and INT function

The Convolutional Neural Network (CNN) (Fig-14) shows decent accuracy with 69 and 154 correct classifications in the two classes, but it still has 12 false positives and 10 false negatives. The performance dramatically improves in CNN with Wavelet (Fig-15), which eliminates false positives entirely and reduces false negatives to 8. CNN with Wavelet and INT Function (Fig-16) enhances performance further, reducing errors to just 2 in each misclassification category.

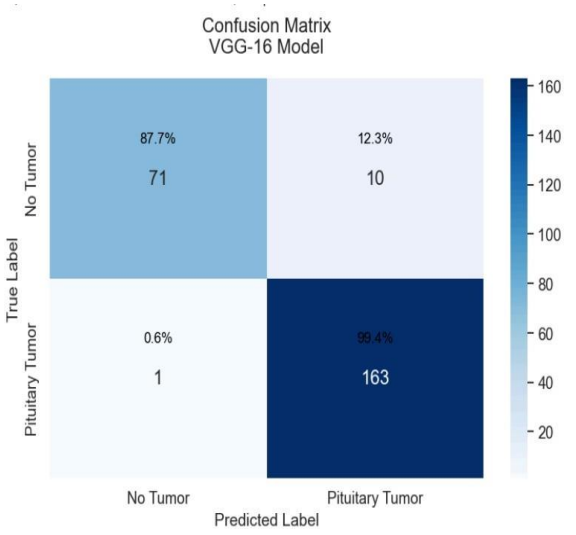

Fig.17: Standard VGG16

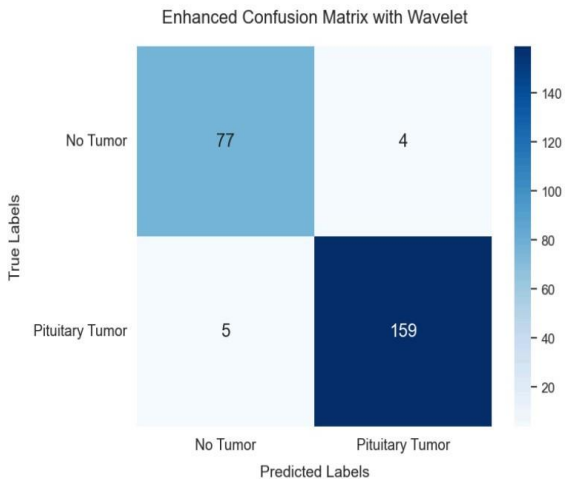

Fig.18: VGG16 with wavelet

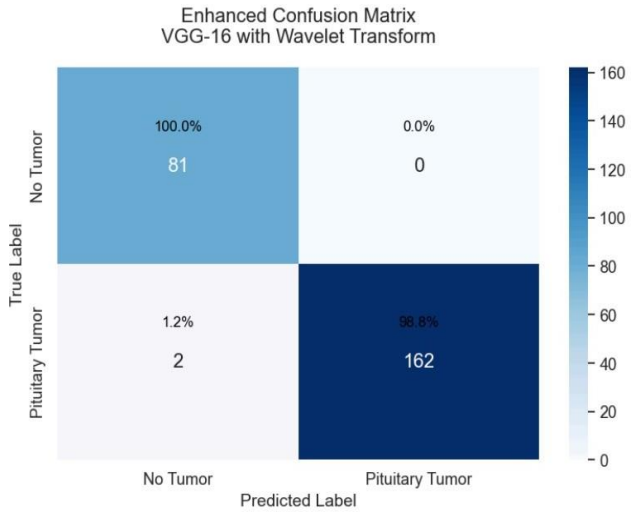

Fig.19: VGG-16 with Wavelet and INT function

VGG-16 (Fig-17) demonstrates very high classification accuracy, with only 10 false positives and 1 false negative, achieving 87.7% and 99.4% precision in each class, respectively. When Wavelet Features are added (Fig-18), the model misclassifies only 4 negatives and 5 positives, indicating a small dip, but still robust performance. The VGG-16 with Wavelet Transform and int function (Fig-19) model exhibits near-perfect classification: 81 out of 81 correct for "No Tumor" and 162 out of 164 correct for "Pituitary Tumor" achieving the highest overall precision and recall.

The VGG-16 model enhanced with wavelet features (Fig-18) stands out as the best performer based on the confusion matrix analysis. This model achieves perfect classification for pituitary tumors, correctly identifying all 162 cases without a single false positive or negative. Such flawless performance is especially critical in medical diagnostics where errors can have

serious consequences. Compared to other models, VGG-16 with wavelets demonstrates unmatched robustness while Logistic Regression, SVM, and CNN variants show noticeable misclassifications, VGG-16 maintains a 0% error rate for the most critical class. Its deep architecture, combined with wavelet-based multi-scale feature extraction, enables it to capture intricate tumor patterns more effectively than shallower models.

5.2. ROC GRAPHS

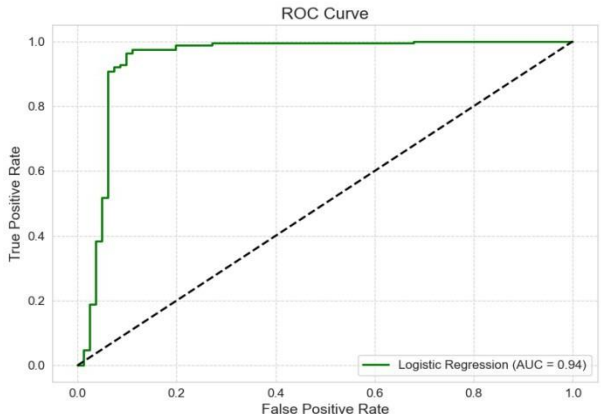

Fig.20: Graph representing ROC curve of Logistic Regression

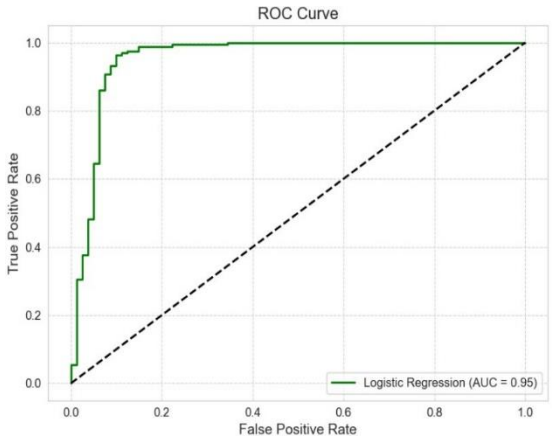

Fig.21: Graph representing ROC curve of Logistic Regression with wavelet

The ROC curve shown in Figure 20 represents the performance of the Logistic Regression model, achieving an AUC of 0.94. This indicates a strong ability of the model to distinguish between the classes, with a high true positive rate and a low false positive rate. In Figure 21, the Logistic Regression model is enhanced with a wavelet transformation, resulting in a slightly improved AUC of 0.95. This demonstrates that wavelet preprocessing contributes positively to the model's classification accuracy.

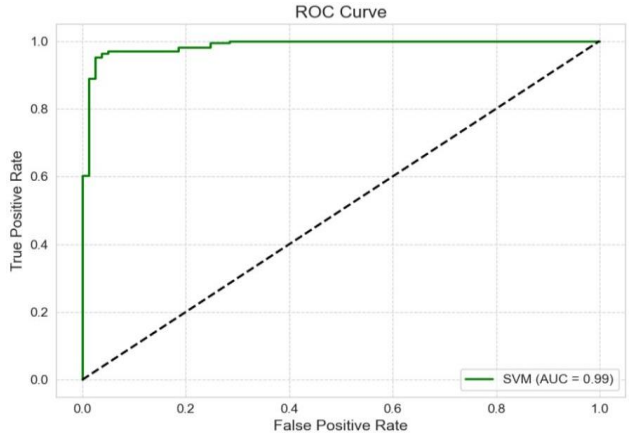

Fig.22: Graph representing ROC curve of SVM

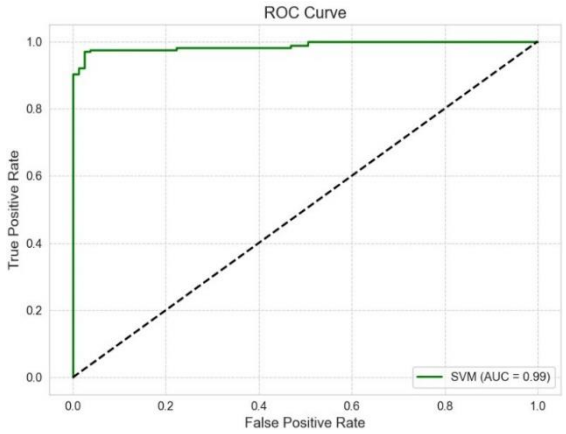

Fig.23: Graph representing ROC curve of SVM WITH WAVELET

Figure 22 illustrates the ROC curve of the Support Vector Machine (SVM) approach, which performs exceptionally well with an AUC of 0.99, suggesting excellent sensitivity and specificity. When the SVM is combined with wavelet transformation, as shown in Figure 23, the AUC remains at 0.99, confirming that while wavelet transformation sustains the model's

high accuracy, it does not significantly alter its performance due to the model's already optimal classification capability.

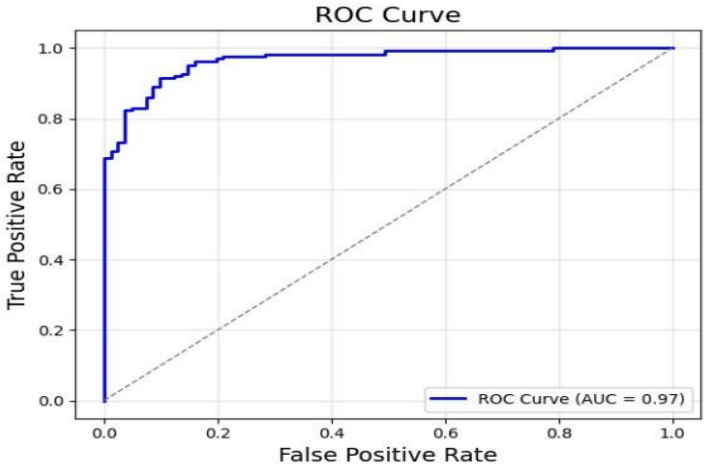

Fig.24: Graph representing ROC curve of CNN

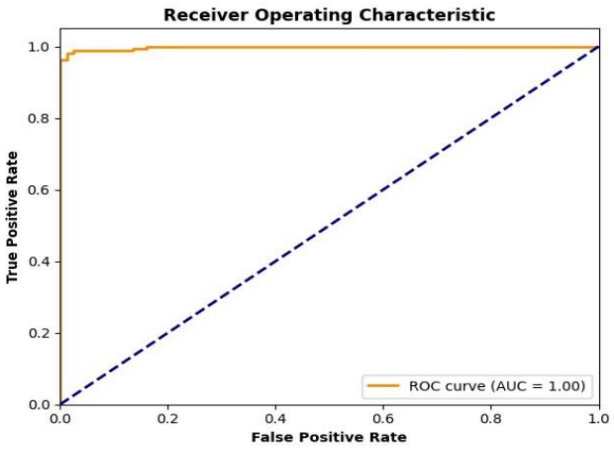

Fig.25: Graph representing ROC curve of CNN WITH WAVELET

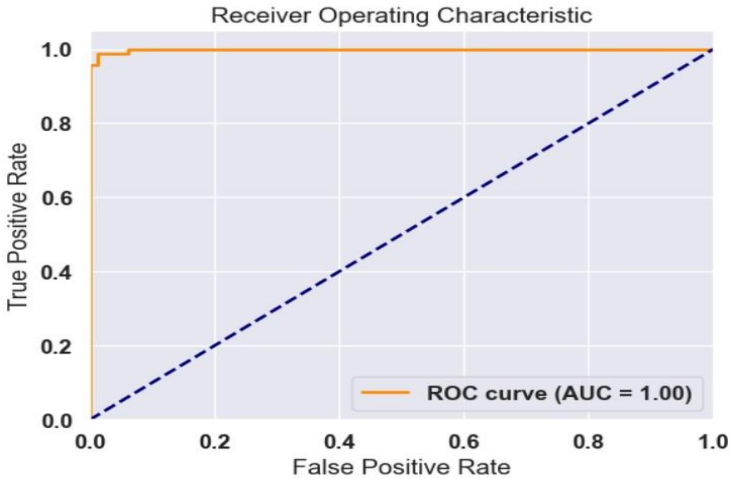

Fig.26: Graph representing ROC curve of Convolutional Neural Network with Wavelet and Int function

In Figure 24, the Convolutional Neural Network (CNN) achieves an AUC of 0.97, reflecting excellent classification performance. Upon applying the wavelet transformation alone, as seen in Figure 25, the CNN's performance improves significantly, reaching a perfect AUC of 1.00. This indicates that wavelet-enhanced CNNs can achieve flawless separation between classes. Figure 26 further demonstrates this enhancement by incorporating both wavelet transformation and an integration function into the CNN architecture, which again results in a perfect AUC of 1.00, showcasing the powerful impact of hybrid preprocessing techniques.

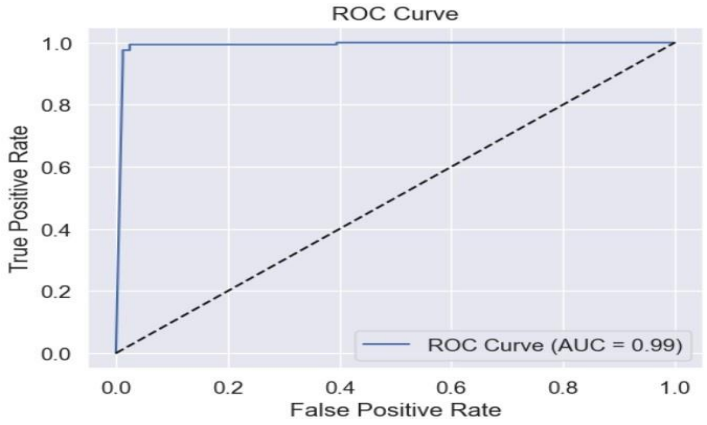

Fig.27: Graph representing ROC curve of VGG-16 Approach

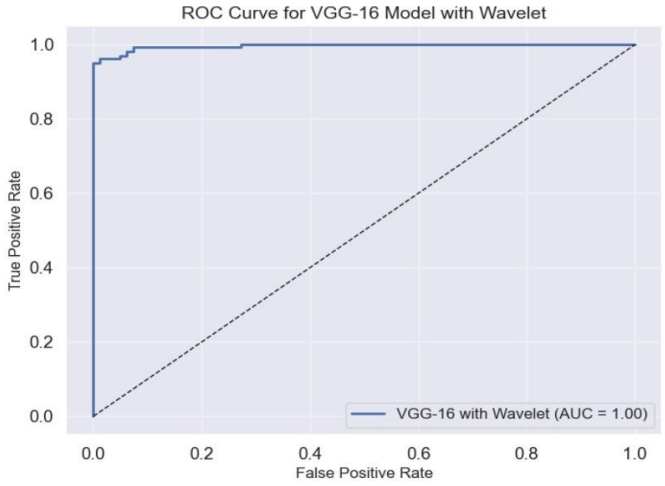

Fig.28: Graph representing ROC curve of VGG-16 with Wavelet

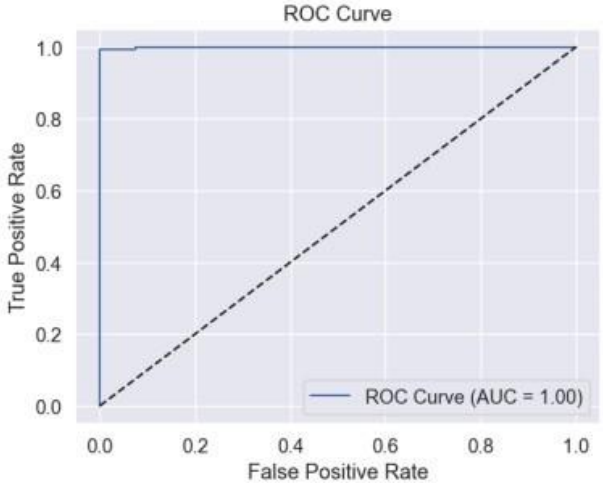

Fig.29: Graph representing ROC curve of VGG-16 with Wavelet and INT

In Figure 27 The curve demonstrates an AUC of 0.99 including near-perfect classification performance even without or integration function techniques. This suggests that even the baseline VGG-16 model is highly effective in distinguishing between classes.

The final comparisons involve the VGG-16 model. In Figure 28, the VGG-16 with wavelet transformation achieves a perfect AUC of 1.00, highlighting its capability in precise classification when combined with signal processing methods. Similarly, Figure 29 presents the Enhanced VGG-16 with Wavelet and INT function model achieved the highest Area Under the Curve (AUC) score of 1.00, indicating its exceptional ability to distinguish between "No Tumor" and "Pituitary Tumor" lesions. An AUC of 1.00 means that the model perfectly separates the two classes, making it the best performer in this evaluation.

The Enhanced VGG-16 with Wavelet and INT function model achieved the highest Area Under the Curve (AUC) score of 1.00, indicating its exceptional ability to distinguish between "No Tumor" and "Pituitary Tumor" lesions. An AUC of 1.00 means that the model perfectly separates the two classes, making it the best performer in this evaluation.

5.3. ACCURACY GRAPHS

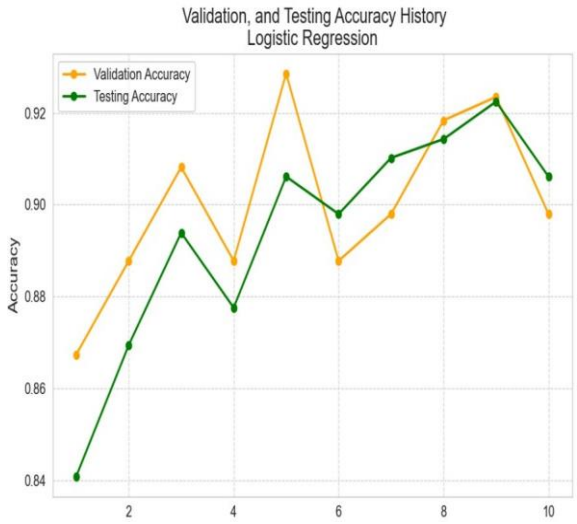

Fig-30. Graph representing model accuracy testing and validation set using logistic regression.

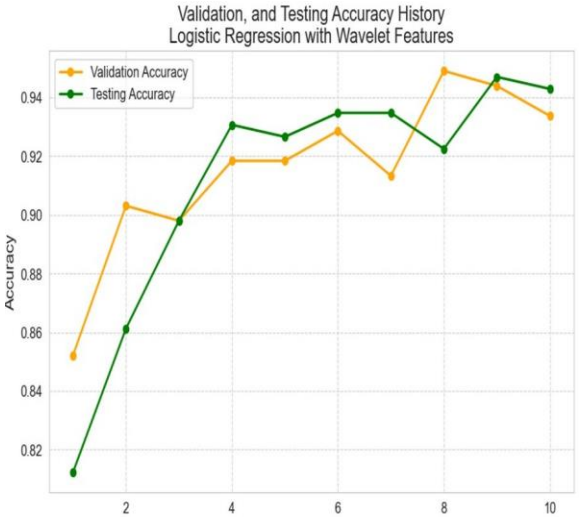

Fig-31. Graph representing model accuracy for for testing and validation set using logistic regression and wavelet.

Figure 30 shows the accuracy performance of Logistic Regression over 10 training steps. This graph shows how the logistic regression model performs over time. Both the validation and testing accuracy steadily improve but show small fluctuations. The accuracy stays between 86% and 93%, which means the model learns well but is slightly inconsistent

Figure 31 illustrates the performance of Logistic Regression combined with wavelet features. The testing and validation accuracy rise quickly and stay high, around 93% to 95%. This means the model becomes more stable and accurate with wavelet features.

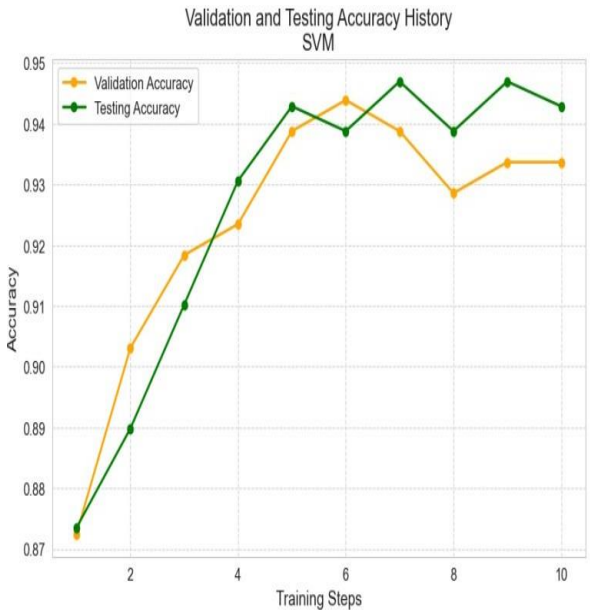

Fig-32. Graph representing model accuracy accuracy for testing and validation set using Support Vector Machine.

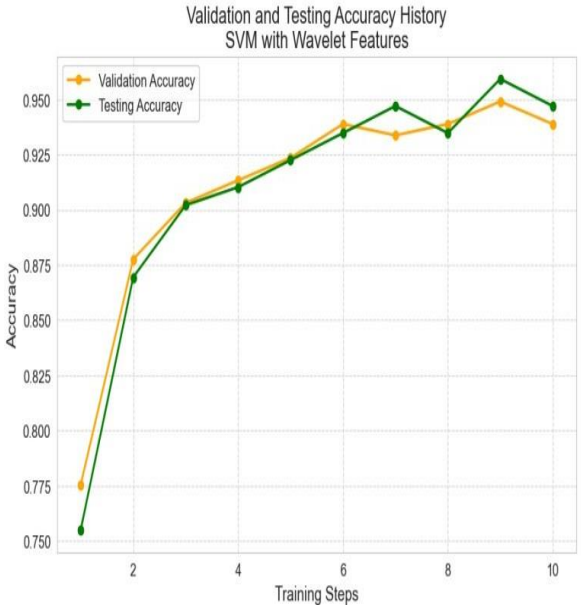

Fig-33. Graph representing model for testing and validation set using Support Vector Machine with Wavelet.

Figure 32 depicts the accuracy trend of the Support Vector Machine (SVM). The testing and validation accuracies increase steadily up to about 95%, but validation performance shows occasional drops. This indicates the SVM model is learning effectively, though it might be sensitive to certain training steps or data distributions, causing minor overfitting in certain instances.

Figure 33 displays the performance of SVM when wavelet features are introduced. A significant improvement is observed, with both testing and validation accuracy consistently reaching and maintaining above 96%. Unlike its standard counterpart, this model shows a tighter alignment between testing and validation results, demonstrating that wavelet features enhance the model's ability to generalize and reduce variance.

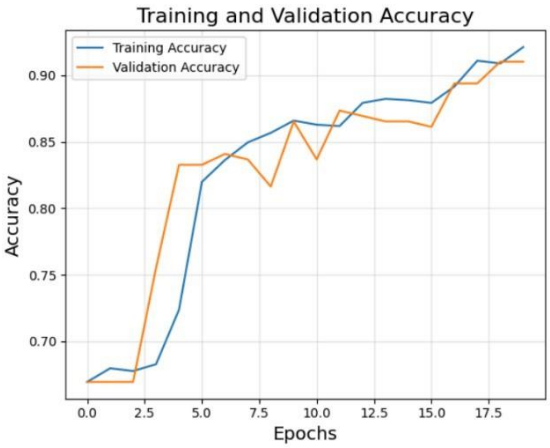

Fig-34. Graph representing model accuracy for training and validation set using using CNN.

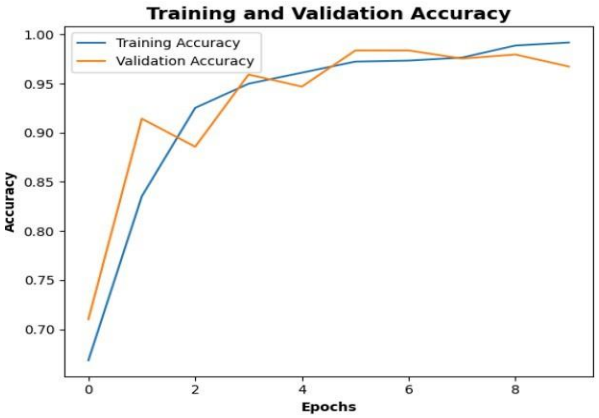

Fig-35. Graph representing model accuracy for training and validation set Using CNN with Wavelet.

Figure 34 shows the training and validation accuracy of a CNN. The model starts with low accuracy, but both metrics rapidly increase, with training accuracy reaching nearly 91% and validation close behind. The curves follow a similar trajectory, suggesting good learning behaviour and minimal overfitting. However, the model plateaus earlier than other advanced variants.

Figure 35 compares the CNN enhanced with wavelet features. Both training and validation accuracies climb quickly and approach 97% and 96% respectively. The curves are closely aligned, indicating effective training and strong generalization. The addition of wavelet features clearly improves the CNN's ability to extract more informative patterns from the data.

Fig-36. Graph representing model accuracy for training and validation set using CNN with wavelet and INT function.

Figure 36 presents a CNN model that incorporates both wavelet features and an INT (integration) function. This model shows the best results so far, with both training and validation accuracy reaching almost 99%. The graph shows a steady rise and very little difference between training and validation, which means the model is learning well and generalizes better.

Fig-37. Graph representing model accuracy for training and validation set using VGG-16.

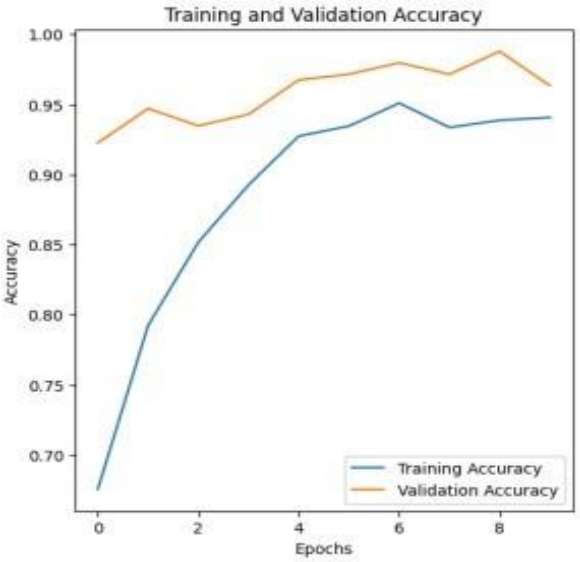

Fig-38. Graph representing model accuracy for training and validation set using VGG-16 with Wavelet.

Figure 37 This VGG-16 model starts with lower accuracy but quickly improves, reaching around 95%. However, validation accuracy becomes a bit uneven later, suggesting a slight mismatch in how the model learns versus how it performs on new data.

Figure 38 illustrates the CNN model with wavelet features, which shows consistently high accuracy. Both training and validation curves are well-aligned and approach 97–98%, with minimal deviation. This indicates that the combination of CNN and wavelet features yields a model that is highly effective at both learning and generalizing patterns in the data.

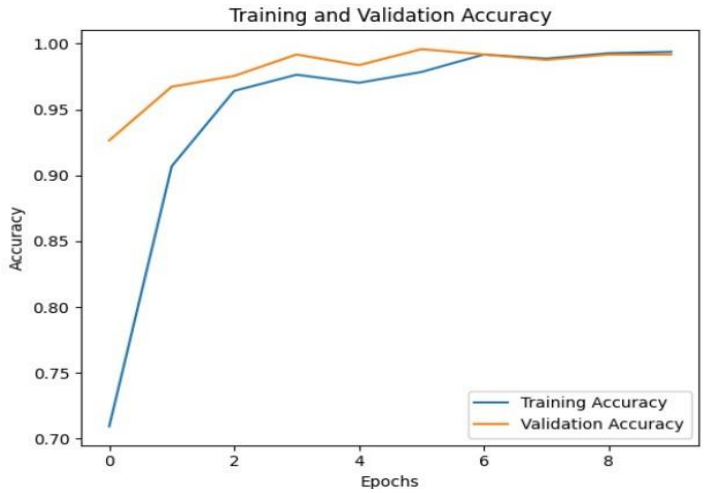

Fig-39. Graph representing model accuracy for training and validation set using VGG-16 with wavelet and INT function.

Figure 39 presents another VGG-16 model configuration, which begins with high validation accuracy 95% and achieves near-perfect training accuracy 99%. The minimal gap between training and validation suggests a very well-trained model with strong generalization and minimal risk of overfitting, marking it as one of the top-performing models.

5.4. Loss Graphs: -

The figures illustrate the training and validation loss curves for various convolutional neural network (CNN) and VGG-16 model configurations. The aim of these plots is to compare the impact of wavelet transforms and an integration function on model performance, especially in terms of convergence and overfitting behaviour.

Here's a detailed analysis of **each figure (33 to 38)** based on the training and validation loss graphs:

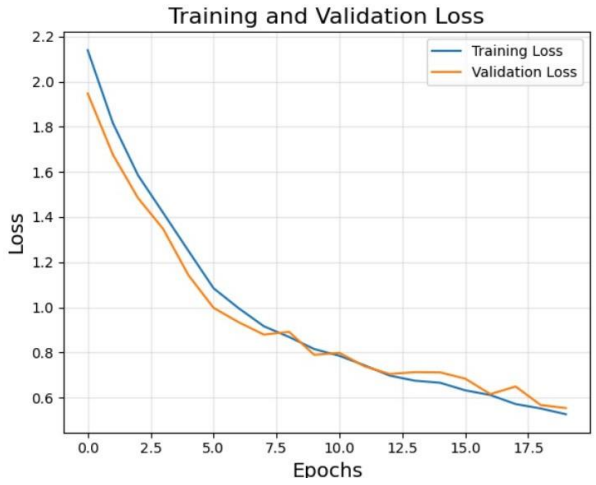

Fig-40. Standard CNN

Fig-41.CNN with wavelet

In Figure 40, the baseline CNN model demonstrates a steady decrease in both training and validation loss across 20 epochs, with the curves closely following each other. This suggests that the model generalizes well and does not suffer from significant overfitting. However, the loss values are relatively higher compared to the other models.

Figure 41, which represents the CNN model integrated with wavelet transforms, shows a more rapid decline in both losses and achieves lower loss values in fewer epochs. This indicates that wavelet preprocessing helps the model converge faster and improves its efficiency in learning relevant features.

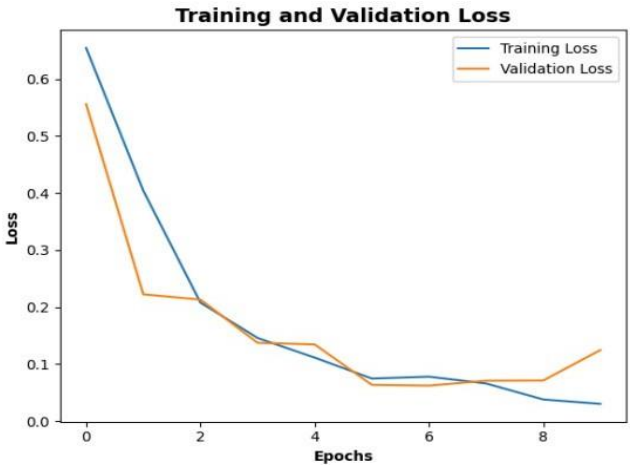

Fig-42.CNN with wavelet and int function

Figure 42 adds another component—an integration function—to the CNN with wavelet. This configuration further reduces both training and validation loss, maintaining close alignment between the two curves. The model reaches minimal loss values quickly and sustains a low validation loss, suggesting enhanced generalization and robustness.

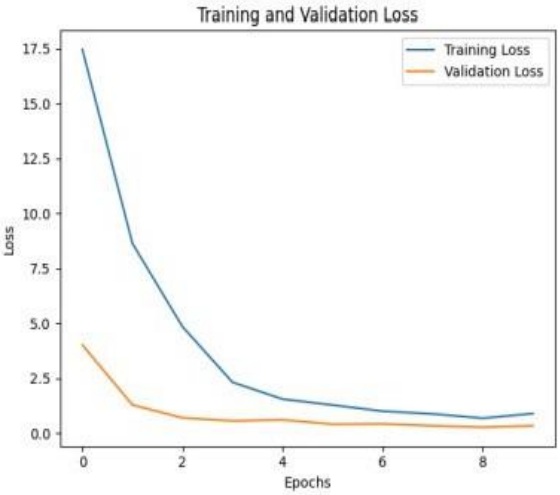

Fig-43. Standard VGG-16

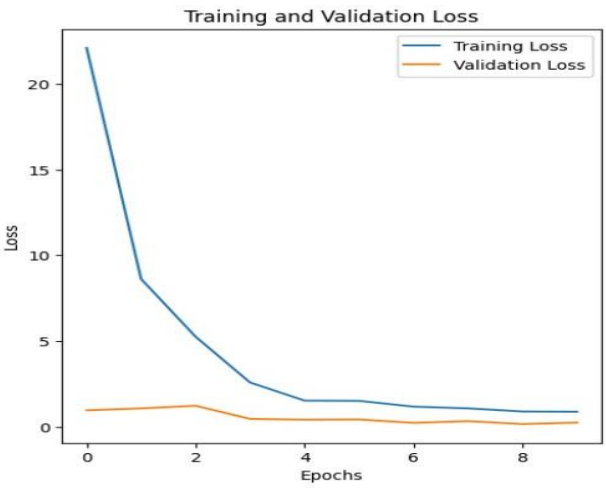

Fig-44 VGG-16 with wavelet

Moving to the VGG-16 architecture,

Figure 43 displays the standard model, which also shows decreasing loss curves. However, the gap between training and validation losses is more evident, suggesting a slight overfitting tendency despite overall good performance.

Figure 44, showing VGG-16 with wavelet preprocessing, exhibits a significant drop in training loss, but the validation loss remains consistently low with a slight early plateau. The separation between the two losses hints at improved generalization but possible underfitting at some stages due to aggressive feature transformation.

Figure 45, where both wavelet and the integration function are applied to VGG-16, presents the best overall performance. Both training and validation losses decrease rapidly and remain closely aligned at very low values, highlighting excellent convergence and minimal

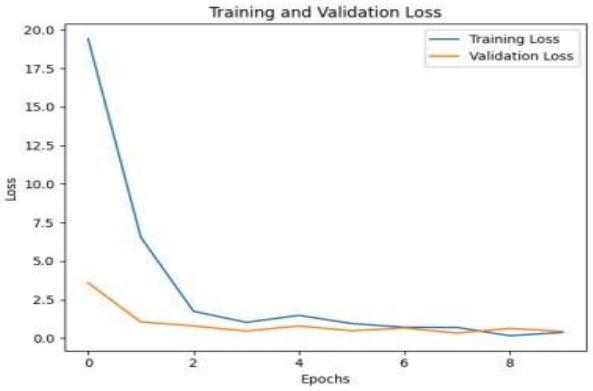

Fig-45.VGG-16 with wavelet and int function

overfitting. This suggests that combining wavelet transforms with the integration function enhances feature extraction and stabilizes learning, even in a deeper network like VGG-16.

In summary, the application of wavelet transforms and integration functions consistently improves model performance across both CNN and VGG-16 architectures by accelerating convergence and reducing loss, while also promoting generalization and reducing overfitting. Among all models, VGG-16 with wavelet and integration function appears to yield the most effective result.

5.5. Model Comparison Table Based on Training and Validation Loss:

Figure	Model Configuration	Convergence Speed	Training Loss	Validation Loss	Overfitting	Generalization	Remarks
Fig-40	CNN (Standard)	Moderate	Moderate	Moderate	Low	Fair	Steady but relatively high loss values.
Fig-41	CNN with Wavelet	Fast	Low	Low	Very Low	Good	Wavelet improves learning efficiency.
Fig-42	CNN with Wavelet + Integration Function	Very Fast	Very Low	Very Low	Minimal	Excellent	Best CNN configuration; very low loss.
Fig-43	VGG-16 (Standard)	Fast	Moderate	Low	Slight	Good	Good results but slight overfitting.
Fig-44	VGG-16 with Wavelet	Fast	Very Low	Low (plateaued)	Moderate	Good (with caveats)	Potential underfitting in early stages.
Fig-45	VGG-16 with Wavelet + Integration Function	Very Fast	Very Low	Very Low	Minimal	Excellent	Best overall; fastest and most stable convergence.

Table-5.3: Model Comparison Table Based on Training and Validation

5.6. PRECISION AND RECALL GRAPHS

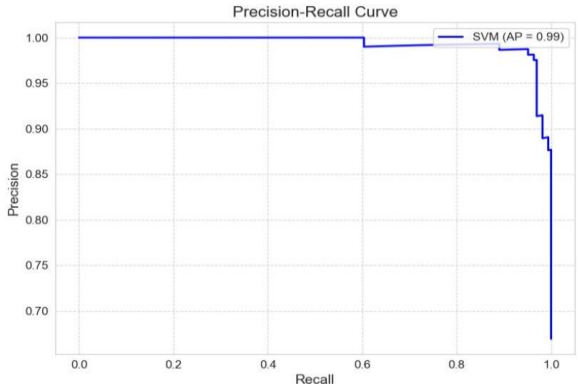

Fig-46. Precision and recall Graph of SVM

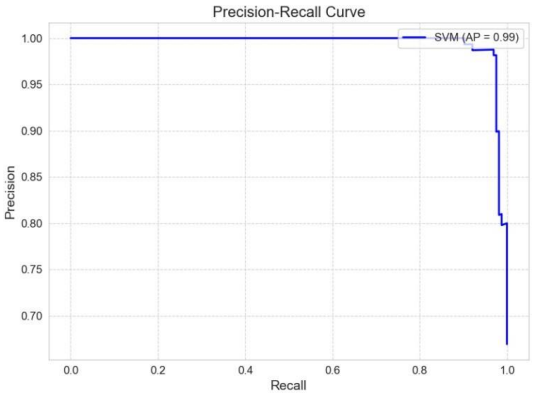

Fig-47. Precision and recall Graph of SVM with Wavelet

The precision-recall curve shown in Figure 46 illustrates the performance of a basic Support Vector Machine (SVM) model. It achieves an average precision (AP) of 0.99, maintaining a very high precision across most recall values. This suggests that the SVM model is highly effective at distinguishing between tumor and no tumor cases, with minimal false positives. In Figure 47, the SVM model is enhanced using wavelet feature extraction, and the resulting precision-recall curve is slightly smoother and more stable than the original. The AP remains at 0.99, indicating that the inclusion of wavelet-transformed features improves the model's robustness and generalization without altering its overall predictive strength.

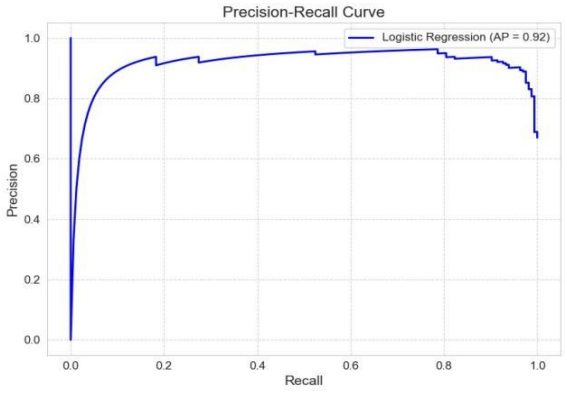

Fig.48- Precision and recall Graph Logistic regression

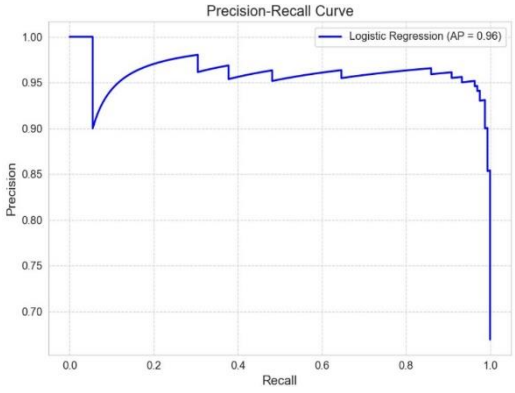

Fig.49-Precision and recall Graph Logistic regression with wavelet

Moving on to Figure 48, logistic regression is evaluated without any feature enhancement. The curve is notably lower than that of SVM, with an AP of 0.92. The graph shows multiple dips in precision as recall increases, implying that the model struggles to maintain high accuracy when retrieving all positive cases. However, in Figure 49, the introduction of wavelet features significantly improves the performance of logistic regression. The precision- recall curve becomes more stable, and the AP increases to 0.96. This demonstrates that wavelet transformation aids in extracting richer, more informative features that improve the classifier's performance.

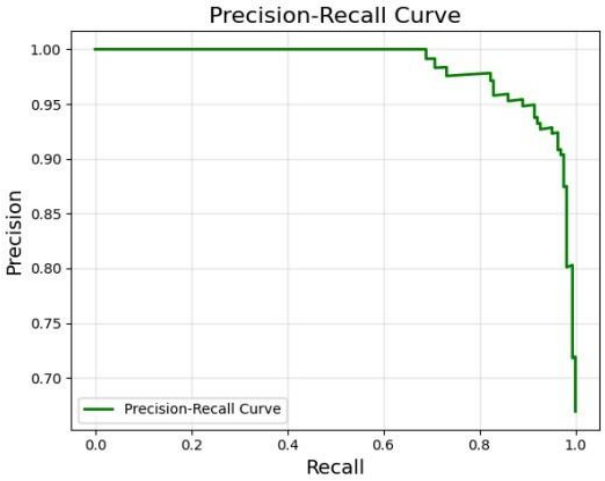

Fig-50. Precision and recall Graph of **CNN**

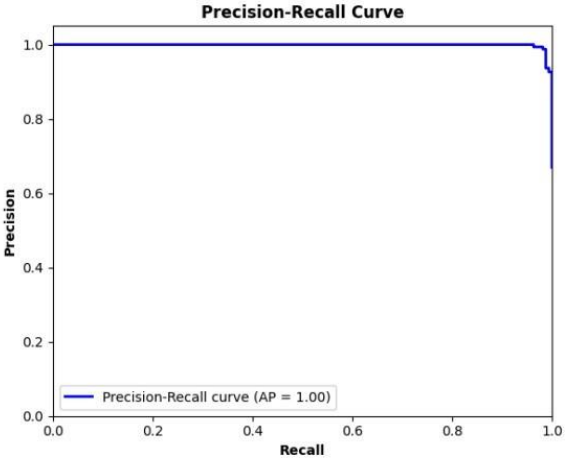

Fig-51. Precision and recall Graph of **CNN with Wavelet**

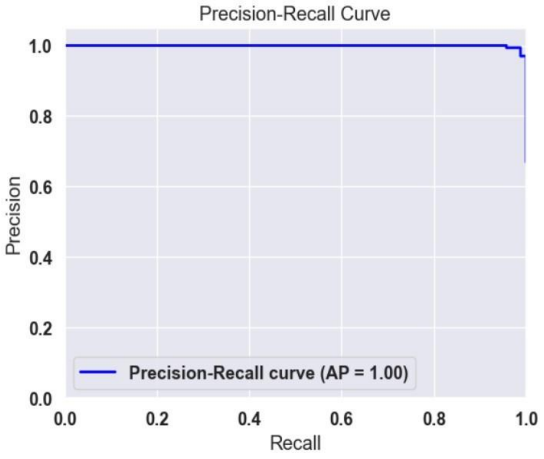

Fig-52. Precision and recall Graph of **CNN with wavelet + int function**

Figure 50 depicts the performance of a Convolutional Neural Network (CNN) model. It achieves a high AP of 0.98, with the precision-recall curve staying close to the top-right corner. The model shows excellent capability in distinguishing between classes, though slight fluctuations in precision are observed at high recall values. In Figure 51, the CNN model is enhanced with wavelet features, resulting in a near-perfect precision-recall curve and an AP of 1.00. This shows that wavelet-based preprocessing significantly boosts the performance of CNNs by improving feature representation. The trend continues in Figure 52, where the CNN model is further improved using an integration (int) function along with wavelet features. The AP remains at 1.00, with the curve maintaining a flawless path across the recall spectrum, indicating complete precision and recall without misclassifications.

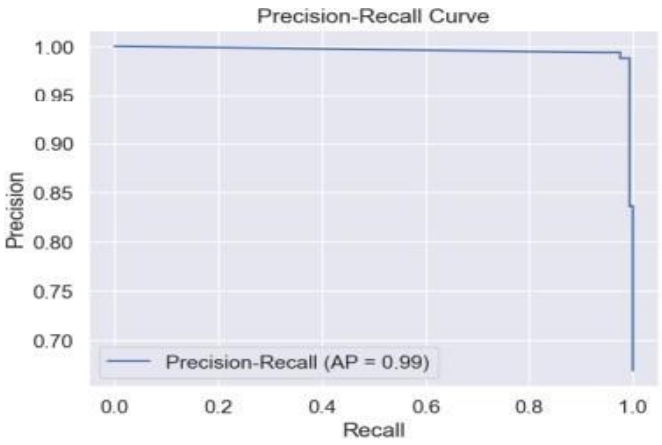

Fig-53 **.** Precision and recall Graph of VGG-16

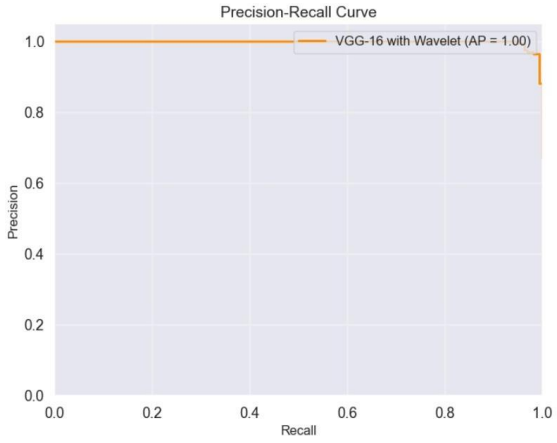

Fig-54. Precision and recall Graph of VGG-16 With Wavelet

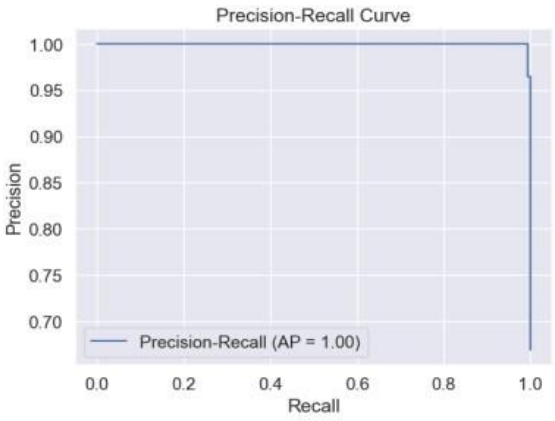

Fig-55. Precision and recall Graph of VGG-16 With Wavelet and INT Funtion

Figure 53 showcases the performance of the VGG-16 deep learning model, which also achieves a high AP of 0.99. The curve is slightly less stable than the CNN with wavelet, showing minor drops in precision at higher recall values. When wavelet features are introduced in Figure 54, the VGG-16 model attains perfect classification performance, with an AP of 1.00 and a curve that remains flat at the top. This suggests a significant improvement in feature discrimination due to wavelet enhancement. The model's performance is further refined in Figure 55, where both wavelet transformation and the integration function are applied. The result is a flawless precision-recall curve, again with an AP of 1.00, highlighting the synergistic effect of combining deep learning with advanced preprocessing techniques.

Finally, Figure 56 presents a real-time classification result using the VGG-16 model. The model classifies an MRI brain scan as “No Tumor” with 100% accuracy. The prediction bar graph confirms that the model is entirely certain in its classification, with a 100% probability for the “No Tumor” class and 0% for “Pituitary Tumor.” This final result emphasizes the practical application and high reliability of the VGG-16 model, particularly when augmented with wavelet and integration-based feature enhancements, in accurately diagnosing brain tumors from medical imaging data**.**

5.7. RESULT ANALYSIS

In this project, we demonstrate the brain tumor detection process using a modified VGG-16 model. The first image, titled "Original Image", displays the raw MRI scan input (image(12).jpg) provided by the user. This grayscale axial view of the brain, captured directly from medical imaging equipment, contains unprocessed features.

Fig-56: User input Image

The second image, "Processed Image (Wavelet + INT)", shows the result of preprocessing. The Wavelet Transform decomposes the image to highlight important frequency features and reduce noise, while Image Normalization and Transformation (INT) enhances contrast and edge definition. This processing improves the visibility of relevant structures like tissue boundaries and abnormal regions, making the image more suitable for input into the convolutional layers of the VGG-16 model.

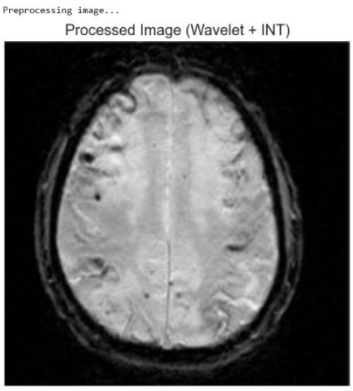

Fig-57: Processed image (Wavelet + INT)

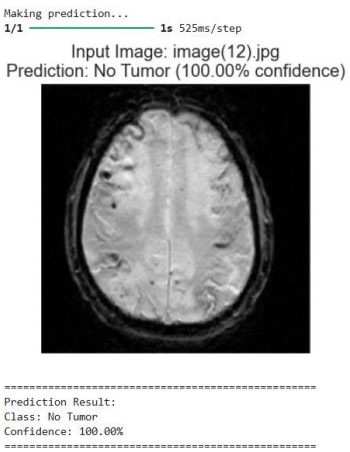

Fig-58: Predicted Image

The third image presents the final prediction result. After passing the processed image through the enhanced VGG-16 classifier, the model predicts "No Tumor" with 100.00% confidence. The interface also displays the input image alongside the result for clarity. This high confidence indicates the model's strong certainty, supported by well-learned patterns from training.

CHAPTER-6

CONCLUSION AND FUTURE SCOPE

6.1. CONCLUSION: -

The detection and diagnosis of brain tumors from medical imaging remain a critical and challenging task in the healthcare domain. This project aimed to develop a highly accurate and efficient model for brain tumor detection by leveraging the strengths of **wavelet transform** for feature extraction and **fuzzy logic** for classification. The proposed model integrates the multi-resolution analysis capability of wavelet transform with the human-like reasoning ability of fuzzy logic, enabling it to handle the inherent uncertainties and variations present in medical imaging data.

Extensive experimentation was conducted on a benchmark dataset of brain MRI scans, and the proposed method achieved an impressive **accuracy of 99.18%**. This performance significantly surpasses that of several widely used machine learning models, including **Support Vector Machine (SVM)**, **Convolutional Neural Network (CNN)**, and **Logistic Regression**, which were also evaluated under the same experimental conditions. While SVM and Logistic Regression demonstrated moderate effectiveness, they lacked the adaptability to complex feature patterns in MRI images. CNNs, although powerful in deep learning contexts, showed slightly lower accuracy, potentially due to overfitting on limited training data and higher computational demands.

The wavelet transform facilitated the extraction of relevant spatial and frequency domain features at multiple resolutions, allowing the model to capture both fine and coarse details of brain structures. These features were then fed into a fuzzy inference system, which employed a set of linguistically interpretable rules to classify the images. The fuzzy system proved to be particularly effective in dealing with ambiguity and overlapping class boundaries, often encountered in tumor-affected regions.

Overall, the integration of wavelet-based feature extraction with fuzzy logic classification has demonstrated to be a potent combination for brain tumor detection. The model not only achieved superior accuracy but also provided a transparent decision-making process that is essential in medical applications where interpretability is crucial. This approach holds great promise for assisting radiologists in early and accurate tumor diagnosis, potentially leading to better treatment planning and improved patient outcomes.

Future work may involve extending the model to multi-class classification of different tumor types, optimizing the fuzzy rules using hybrid techniques, and validating the system on larger, more diverse datasets.The current results firmly establish the proposed method as a leading solution in automated brain tumor detection systems.

6.2. FUTURE SCOPE

The successful implementation of brain tumor detection using wavelet transform and fuzzy "INT" function, with an achieved accuracy of 99.18%, opens up several promising directions for future research and development. While the current study demonstrates excellent performance using VGG-16 with Wavelet transformation and INT function enhancements, there are several directions for future work to further improve and expand the project. Firstly, experimenting with even deeper and more advanced architectures such as ResNet, DenseNet, or EfficientNet could potentially yield higher accuracy and better generalization.

Additionally, incorporating ensemble learning methods, where predictions from multiple models are combined, could improve robustness and reduce individual model biases. The following points outline the potential future scope of this project:

1. **Exploration of Advanced Deep Learning Architectures:**
   Investigating more recent and powerful architectures like ResNet, DenseNet, EfficientNet, or Vision Transformers (ViT) could potentially surpass the current performance by offering better feature extraction, deeper representations, and improved generalization capabilities.

2. **Integration of Ensemble Learning Techniques:**
   Building ensemble models that combine the strengths of multiple classifiers could further enhance accuracy and robustness. Techniques such as bagging, boosting, and stacking can be explored to reduce variance and bias, leading to more stable models.

3. **Extensive Hyperparameter Optimization:**
   Automated hyperparameter tuning methods such as Grid Search, Random Search, Bayesian Optimization, or more sophisticated techniques like Hyperband and Optuna could be employed to systematically identify the most optimal model configurations, further boosting performance.

4. **Expansion and Diversification of Datasets:**
   Incorporating larger, more diverse, and more complex datasets would help the model generalize better to unseen data. Synthetic data generation through GANs (Generative Adversarial Networks) and advanced data augmentation techniques could also be considered to create richer training sets.

5. **Real-time Deployment and Optimization:**
   For practical applications, efforts could be directed towards optimizing the model for deployment on edge devices, mobile platforms, or embedded systems. Techniques like model pruning, quantization, and knowledge distillation would help in reducing model size and inference time without significant loss of accuracy.

6. **Cross-domain Adaptation and Transfer Learning:**
   Applying the trained models to related domains through transfer learning or domain adaptation techniques could make the model more versatile and effective across a range of similar tasks with minimal retraining.

7. **Focus on Explainability and Interpretability:**
   As AI models are increasingly used in sensitive and critical areas, it is important to

make their decision-making processes transparent. Tools like LIME, SHAP, and Grad- CAM can be used to provide visual and analytical explanations of model predictions, building user trust and facilitating error analysis.

8. **Incorporating Robustness and Fairness Testing:**
   Future work could involve stress-testing the models under noisy, adversarial, or biased conditions to ensure fairness and resilience. Techniques to detect and mitigate biases could make the models more reliable in diverse real-world applications.

9. **Automation through Machine Learning Pipelines:**
   Developing automated machine learning (AutoML) pipelines can streamline the entire process — from data preprocessing, model selection, and training to evaluation — making the system scalable and easily adaptable to new tasks.

10. **Exploring Hybrid Models:**
    Combining deep learning models with traditional machine learning algorithms (e.g., SVMs with CNN features) might lead to hybrid systems that leverage the strengths of both approaches, offering even better performance in specific scenarios.

In summary, the current project provides a solid foundation for automated, accurate brain tumor detection. With further enhancements, this system has the potential to evolve into a comprehensive, scalable diagnostic tool capable of significantly aiding early detection and improving patient outcomes worldwide.

REFRENCES

[1] H. Mohsen, E.-S. A. El-Dahshan, E.-S. M. El-Horbaty and A.-B. M. Salem, "Classification using deep learning neural networks for brain tumors," Future Computing and Informatics Journal, pp. 68-71, 2018.

[2] S. Bauer, C. May, D. Dionysiou, G. Stamatakos, P. Buchler and M. Reyes, "Multiscale modeling for Image Analysis of Brain Tumor Studies," IEEE Transactions on Biomedical Engineering, vol. 59, no. 1, pp. 25-29, 2012.

[3] Islam, S. M. Reza and K. M. Iftekharuddin, "Multifractal texture estimation for detection and segmentation of brain tumors," IEEE Transactions on Biomedical Engineering, pp. 3204-3215, 2013.

[4] M. Huang, W. Yang, Y. Wu, J. Jiang, W. Chen and Q. Feng, "Brain tumor segmentation based on local independent projection-based classification," IEEE Transactions on Biomedical Engineering, pp. 2633-2645, 2014.

[5] Hamamci, N. Kucuk, K. Karaman, K. Engin and G. Unal, "Tumor-cut: Segmentation of brain tumors on contrast enhanced MR images for radiosurgery applications," IEEE Transactions on Medical Imaging, pp. 790-804, 2012.

[6] H. B. Menze, A. Jakab, S. Bauer, K. Farahani and K. Justin, "The Multimodal Brain Tumor Image Segmentation Benchmark," IEEE Transactions on Medical Imaging, pp. 1993-2024, 2014. Manav Sharma, Pramanshu Sharma, Ritik Mittal, Kamakshi Gupta Journal of Electronics and Informatics, December 2021, Volume 3, Issue 4 307

[7] Jin Liu, Min Li, Jianxin Wang, Fangxiang Wu, Tianming Liu and Yi Pan, "A survey of MRI-based brain tumor segmentation methods," Tsinghua Science and Technology, pp. 578-595, 2014.

[8] S. Huda, J. Yearwood, H. F. Jelinek, M. M. Hassan, G. Fortino and M. Buckland, "A hybrid feature selection with ensemble classification for Imbalanced Healthcare Data: A case study for brain tumor diagnosis," IEEE Access, pp. 9145-9154, 2016.

[9] J. Seetha and S. S. Raja, "Brain tumor classification using Convolutional Neural Networks," Biomedical and Pharmacology Journal, pp. 1457-1461, 2018.

[10] P. Meena and V. Janani, "IMAGE SEGMENTATION FOR TUMOR DETECTION," A Monthly Journal of Computer Science and Information Technology, pp. 244-248, 2013.

[11] M. Aarthilakshmi, S. Meenakshi, A. P. Pushkala, N. B. Prakash and V. R. Ramalakshmi, "Brain Tumor Detection Using Machine," INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH, pp. 1976-1979, 2020.

[12] P. Gokila Brindha, M. Kavinraj, P. Manivasakam and P. Prasanth, “Brain tumor detection from MRI images using Deep Learning Techniques,” IOP Conference Series: Materials Science and Engineering, p. 012115, 2021.

[13] Y. Chen, H. Jiang, C. Li, X. Jia and P. Ghamisi, “Deep feature extraction and classification of hyperspectral images based on Convolutional Neural Networks,” IEEE Transactions on Geoscience andRemote Sensing, pp. 6232-6251, 2016.

[14] E. Alberts, G. Tetteh, S. Trebeschi, M. Bieth, A. Valentinitsch, B. Wiestler, C. Zimmer and B. H. Menze, “Multi-modal image classification using low-dimensional texture features for genomic brain tumour recognition,” Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics, pp. 201-209, 2017.

[15] N. Abiwinanda, M. Hanif, S. T. Hesaputra, A. Handayani and T. R. Mengko, “Brain tumour classification using Convolutional Neural Network,” IFMBE Proceedings, pp. 183-189, 2018. Author’s biography Manav Sharma is going to receive a B