



American International University Bangladesh

Enhancing the Identification of Brain Tumors Using CNN Ensemble Model

Shifar Tanjim (20-42827-1)

Mohi Uddin Anando (20-43585-1)

Rabbi Hasan Himel (20-42204-1)

Ahatesham Rabbi (19-41722-3)

*A Thesis submitted for the degree of Bachelor of Science
(BSc) in Computer Science and Engineering (CSE) at
American International University Bangladesh in January 2024
Faculty of Science and Technology (FST)*

Declaration by author



Shifar Tajam
20-42827-1
Computer Science & Engineering



Mohi Uddin Anando
20-43585-1
Computer Science & Engineering



Rabbi Hasan Himel
20-42204-1
Computer Science & Engineering



Ahatesham Rabbi
19-41722-3
Computer Science & Engineering

Abstract

This thesis provides an examination of Convolutional Neural Network (CNN) architectures including Scratch CNN, InceptionV3, Xception, EfficientNetB0, ResNet50 and VGG19 and their effectiveness, in detecting brain tumors from MRI images. The study evaluates these models based on accuracy and log loss metrics using machine learning classifiers such as AdaBoost, Random Forest, Support Vector Machine (SVM) K Nearest Neighbors (KNN) and Softmax. Among them EfficientNetB0, Xception and InceptionV3 demonstrate accuracy with the use of methods to enhance overall performance. The research highlights the importance of selecting models and classifiers in image classification tasks in medical diagnostics.

The thesis delves into the significance of early detection of brain tumors while discussing techniques for classification and feature extraction from MRI images. These techniques include analyzing texture, color information, tumor region location identification as edge detection. The document explains the methodology employed in the study which involves preprocessing techniques like augmentation, Gaussian Blurring and Sobel Edge Detection to improve image quality, for better feature extraction and classification.

The results show that deep learning models— EfficientNetB0, Xception and InceptionV3—outperform others significantly in terms of accuracy and reliability. The study explores combining these models to enhance capabilities while discussing the promising outcomes achieved through methods.

The thesis also recognizes the constraints, such, as the types of tumors considered and the necessity, for clinical confirmation. It proposes avenues that involve enhancing precision expanding datasets and employing a variety of models to propel the field of brain tumor detection forward.

Keywords

Brain Tumor, MRI, CNN, Feature Extraction, Classification, KNN, Filter

Table of Contents

Declaration by author	2
Abstract.....	3
Keywords.....	3
List Of Figure	5
List Of Table	6
Chapter 1.....	7
1. Introduction	7
1.1 Thesis Topic.....	7
1.2 Introduction	7
1.3 Medical Images	7
1.4 Motivation.....	8
1.5 Objective	8
1.6 Orientation.....	9
Chapter 2.....	10
2. Literature Review	10
Chapter 3.....	14
3. Method	14
3.1 Dataset	14
3.2 Data Pre-processing	14
3.3 Filtering:	15
3.4 Data Augmentation:.....	15
3.5 Feature Extractors:.....	16
3.5.1 Proposed Scratch CNN	17
3.5.2 Modified Xception.....	17
3.5.3 Modified ResNet50	18
3.5.4 Modified EfficientNetB0.....	18
3.5.5 Modified InceptionV3	19
3.5.6 Modified VGG19:	20
3.6 Classifier	21
3.6.1 Adaboost	21
3.6.2 Random Forest.....	22
3.6.3 SVM.....	22

3.6.4	Softmax	22
3.7	Tools and Libraries Used	23
3.7.1	TensorFlow.....	23
3.7.2	Kaggle.....	23
3.7.3	Scikit-Learn.....	23
3.7.4	OpenCV	24
3.7.5	NumPy.....	24
3.7.6	Joblib	25
3.7.7	Matplotlib.....	25
Chapter 4.....		26
4.	Result	26
Chapter 5.....		32
5.	Discussion.....	32
5.1	Limitations.....	32
5.2	Limited Dataset Size	32
5.3	Tumor-type Scope	32
5.4	Limited Clinical validity	32
5.5	Limited Comparison with Existing Method.....	32
5.6	Limited Clinical Validation.....	32
Chapter 6.....		33
6.	Conclusion.....	33
6.1	Future Work.....	33
Bibliography		34

List Of Figure

Figure 1: Selection Process of CNN Architectures for a Two-Stage Ensemble Model in Image Processing	14
Figure 2: Image comparison for Gaussian filter.....	15
Figure 3: Comparison of Sobel Edge Filter.....	15
Figure 4: Data Pre-processing Stages.....	16
Figure 5: Flowchart of Ensemble Machine Learning Pipeline for Brain MRI Analysis with Two-Stage Model Deployment	21
Figure 6: Softmax, Figure 7: SVM, Figure 8: RF.....	26
Figure 9: KNN, Figure 10: AdaBoost.....	26
Figure 11: Scratched CNN, Figure 12: Xception	27
Figure 13: InceptionV3, Figure 14: RestNet50.....	27

Figure 15: EfficientNetB0, Figure 16: VGG19.....	27
Table 1: EfficientNetB0 & Xception, Figure 17: Average Accuracy and Loss	28
Table 2: EfficientNetB0 & InceptionV3, Figure 18: Average Accuracy and Loss.....	29
Table 3: InceptionV3 & Xception, Figure 19: Average Accuracy and Loss	29
Table 4: EfficientNetB0, Xception & InceptionV3, Figure 20: Average Accuracy and Loss.....	30
Figure 21: Code base Interface	31

List Of Table

Table 1: EfficientNetB0 & Xception.....	28
Table 2: EfficientNetB0 & InceptionV3	29
Table 3: InceptionV3 & Xception	29
Table 4: EfficientNetB0, Xception & InceptionV3	30

Chapter 1

1. Introduction

1.1 Thesis Topic

Enhancing the Identification of Brain Tumors using CNN Ensemble Model.

1.2 Introduction

In the field of diagnostics and healthcare, detecting brain tumors early and accurately has become a challenge. In diagnostics and healthcare, the early and precise detection of brain tumors has emerged as a formidable task. Brain tumors, regardless of their nature—benign or malignant, can have severe repercussions if not diagnosed and treated promptly [1]. The demand for efficient detection methods has never been more urgent, considering how quickly the disease can progress and affect patient's lives.

Historically, the identification of brain tumors heavily rested on the expertise of radiologists who methodically analyzed medical pictures, such as those produced from magnetic resonance imaging (MRI) and computed tomography (CT) scans [2]. While these professionals possess knowledge and experience, the sheer volume of medical imaging data and the need for accuracy calls for an efficient and reliable approach. Deep learning techniques, which are a subset of artificial intelligence (AI), come into play in this context.

Deep learning, which falls under the category of machine learning, has made advancements in recent times [3]. Its ability to autonomously learn patterns from datasets has paved the way for groundbreaking developments across various fields, including medical image analysis [4]. In brain tumour detection, deep learning is promising in improving accuracy, speed and consistency.

1.3 Medical Images

Medical imaging holds significant importance in assessing and managing diverse health conditions [5]. Specifically, when examining the complex structure of the human brain, several imaging techniques offer unique perspectives. Magnetic Resonance Imaging (MRI) stands out as a cornerstone in neuroimaging, delivering exceptional comprehensive images of the brain's interior architecture through the application of magnetic fields and radio waves. These high-resolution images permit clinicians to spot small abnormalities, such as brain tumors, with amazing precision [6]. Computed Tomography (CT) scans, utilizing X-rays to generate cross-sectional pictures, supplement MRI by providing crucial information regarding tissue density. Despite its slightly reduced resolution, CT serves a significant role in the overall evaluation of brain health.

Positron Emission Tomography (PET) imaging offers a metabolic component to the diagnosis of brain malignancies. By tracing the spread of a radiotracer, PET scans highlight areas of heightened metabolic activity, aiding in detecting and characterising malignancies. While not as common in brain imaging, Ultrasound remains a flexible modality. Though it is commonly linked with prenatal imaging, ultrasonography can be applied in specific neuroimaging circumstances, particularly in measuring blood flow and detecting anomalies [7]. The development of Functional MRI (fMRI) significantly advances our understanding of the brain's dynamic activities, allowing doctors to correlate structural abnormalities, such as tumors, with changes in neural activity. As technology progresses, so does our capacity to utilize imaging modalities like Diffusion Tensor Imaging (DTI), which studies water diffusion in tissues to determine white matter pathways. In the field of brain tumor detection, these different imaging tools collectively contribute to a comprehensive diagnostic

strategy, establishing the framework for integrating cutting-edge technology, including deep learning approaches, to automate and refine this delicate process.

1.4 Motivation

Primary brain or spinal cord tumors arise in these tissues. Primary malignant brain and spinal cord tumors will affect 24,810 Americans (10,530 women and 14,280 men) in 2023. This kind of tumor is rare, less than 1%. Brain tumors account for 85%–90% of first CNS malignancies. Worldwide, 308,102 instances of primary brain or spinal cord tumors were expected in 2020. US CNS tumor diagnoses in children under 20 are expected to reach 5,230 in 2023. The rest of this manual covers' adult primary brain tumors. Brain and nerve system malignancies are incurable and the ninth leading cause of death for men and women. Primary malignant brain and central nervous system tumors will kill 18,990 Americans in 2023 (seven,970 women and 11,020 men). Primary malignant brain and central nervous system cancers killed 251,329 people worldwide in 2020. Under-15s had a 75% 5-year relative survival rate. The 5-year relative survival rate for 15-39-year-olds is 72%. The 5-year relative survival rate was 21% for those over 40. Doctors calculate brain tumor survival rates every five years [8].

About 120 types of brain tumors affect different brain tissues. Benign or noncancerous brain tumors might be dangerous owing to their size or location. Brain and nerve cancers affect 30 per 100,000 Americans. Brain tumors damage healthy brain tissue by pressing on or spreading into it. Certain brain tumors may become cancerous. If they obstruct cerebral fluid flow, skull pressure may increase. Certain cancers may spread via spinal fluid to distant spine or brain regions [9].

Global cancer observatory (2020) ranks 1,284 new cases 22nd with a cumulative risk of 0.82% and a rank of 0.09. The sickness caused 1,144 deaths, ranking 19th, with a cumulative risk of 1.0% and 0.08. These numbers show the prevalence of brain and CNS cancer. In all age categories, 2,898 cases occurred over five years, resulting in a 1.76 per 100,000 rates [10].

In Bangladesh, according to research by Sarkar et al. (2021), Treatment of brain tumors is in requirement of joint efforts by several professionals from neurosurgery, neuroradiology, neuropathology, oncology, and radiation. The result is poorer in underdeveloped nations compared to developed countries because of shortcomings in adequate registration, lake of awareness of patients, failure of prompt diagnosis, lack of availability and co-ordination of numerous professionals for complete care and high abandonment rates [11].

Brain tumors have decreased, although they remain a medical concern. Brain tumors may profoundly impact an individual's quality of life and well-being. Brain tumors must be detected early to optimize therapy and patient survival.

Early detection and therapy are crucial for the cure of brain tumors. Risky, untreated brain tumors raise healthcare expenses and suffering. However, early discovery and proper therapy may improve brain cancer. Early brain tumor detection improves treatment outcomes and lowers disease severity. So, the goal is to use image processing and machine learning techniques to identify brain cancers early, benefiting the medical business.

1.5 Objective

The final sub-goal is to provide a rationale for our method selection from a broad range of options, detailing their functionality and our development process.

- **Sub-Objective 1:** To collect a suitable dataset of brain tumor images, then enhancing the image clarity and characteristics through various augmentation and filtering techniques.

- **Sub-Objective 2:** To process the gathered data, training it with chosen Convolutional Neural Network models, and categorizing them using specific classifiers. This approach is crafted to be both efficient and effective.
- **Sub-Objective 3:** To develop a model capable of detecting brain tumors from imaging data, focusing on enhancing the model's precision.

1.6 Orientation

The second chapter of the paper explores earlier academics' studies on the same subject, as well as the concepts of deep learning, Image processing and Brain tumor identification. The third part explains the implementation, data preparation, augmentation, filtering, feature extraction such scratch CNN, Xception, ResNet50, EfficientNetB0, InceptionV3, VGG19 model train classifications, and a concise review of the methodologies utilized to this model.

The tools and libraries required to create this model and the train-test component are provided in the following section.

The specifics are summed up in Chapter 4 by analyzing all of the hurdles and comparing the outcomes of all the models to acquire the best accuracy.

In chapter 5 we addressed the article and the restrictions of our model.

Finally, in Chapter 6 there are a few closing observations and future investigations stated for our research.

Chapter 2

2. Literature Review

In this thesis paper, have developed methods for detecting and classifying brain tumors using learning models such as CNN, Xception, InceptionV3, ResNet50, EfficientNetB0, and VGG19. So many researchers and doctors work on it to improve brain tumor detection Using deep learning, artificial intelligence, image processing, and so many others. In the paper, they use Convolutional Neural Network (CNN), Xception, InceptionV3, ResNet50, EddicientNetB0, and VGG19. Researchers have found new ways and algorithms to improve accuracy of the detection of brain tumors.

Hafiz Muhammad Tayyab Khushi et al., deep learning models include InceptionV3, Resnet50, and VGG19. ResNet50 achieved the highest and best validation accuracy. That is 89.45%, with a validation loss of 0.28. The model InceptionV3 achieved a validated accuracy of 76.33%. The EfficientNetB7 model has a state-of-the-art model accuracy of 98.97%. That demonstrates excellent performance [12].

Muhammed Celik el al., explained brain tumor classification using MRI imaging and deep learning. Using CNN models and the proposed model hybrid, they achieved classification with 97.15% accuracy and a recall of 97%. The pre-trained models were EfficientNetB0, VGG19, ResNet50, InceptionV3, and Xception. Most of the ResNet accuracy was 96% and the accuracy of the CNN model proposed by Generic was 81.05 % accuracy [13].

Ahmeed Suliman Farhan el al., introduced brain tumor detection in MRI images. The models assessed the two datasets, which are state-of-the-art models, achieving 94.77% and 97.1% inception vacancies, respectively. Comparative models are used: VGG19, EfficientNetB0, InceptionV3, ResNet50, and Xception. VGG19 accuracy is 97%, 98%, and 99% for the modalities. In the second scenario, the Inceptionv3 model is used to extract different features from different Inception modules, which are fed into a softmax for brain tumor diagnosis. Total accuracy was 94.77% for a fast table. And the second table's accuracy was 97.1%. [14]

Ranit Sen et al., proposed a novel approach for datasets and categories of brain tumors and also employed state-of-the-art CNN-like architectures, like EfficientNetB0, Xception, ResNet50, MobileNetV2, and VGG16, using transfer learning then classified the three types of brain tumor. The MRI images across 4 classes and image enhancement methods. The EfficientNetB0 gave the best performance, with an accuracy of 97.61%. And ResNet50, Xception, MobileNetV2, and VGG16 accuracy are 96.26%, 96.64%, 96.90%, and 72.45%, respectively. The classification of abnormal brain pixels is crucial for finding distinct tumor types. [15]

R. Tamilarasi proposed delves into brain tumor highlight and detection to improve patients' quality of life. There are two models used to achieve high accuracy in brain tumor classification. One identifying tumor accuracy of 98.6% and another pituitary tumor accuracy of 98%. The proposed CNN models are ResNet-50, and Inceptionv3. Overall Multi-Classification accuracy was 98%. [16]

Yuting Xie et al., CNN to classify medical images, using distinct models tailored to specific classification challenges. The research implements CNN for medical image classification. It achieved impressive accuracies of 97.6% in tumor detection and 98% in tumor classification. This includes custom CNN models, VGG, ResNet, and EfficientNet. Deep learning techniques CNN-based were published on Scopus and PubMed from 2015 to June 2022. At last, it reaches a remarkable accuracy of 98% [17].

Saif Ahamad et al., seven transfer learning methods, such as VGG-19, InceptionV3, ResNet50, Inception, Xception, and ResNetV2. For instance, VGG19-SVM achieved a height accuracy of 99.39%. That is the accuracy of the height classification. On the other hand, the InceptionV3-Decision has an accuracy score of 75.67%, which is the lowest among the models. The section covers dataset data augmentation, description, CNN, and image pre-processing models [18].

Naeem Ullah proposed the critical need for the timely detection of brain tumors. To identify and classify three major brain tumor types: glioma, meningioma, and pituitary. These classifications are evaluated, including Inceptionv3, Xception, and Resnet50. The accuracy of Inceptionv3, Resnet50, Xception is 94.48%, 67.03%, and 98.37%, respectively. The time limit is too high. And resnet50 accuracy was not good [19].

Md Ishtyaq Mahmud et al., application of AI, specifically deep learning algorithms. This thesis paper used a dataset of 3264 MR images, with 80% of the data used and 20% for testing. Identify brain tumors using CNN architectures, also against established models like CNN, Inception V3, and ResNet-50. The CNN model brain tumor accuracy of 93.3%, on the other hand, an AUC of 98.43%, a recall of 91.19%, and less than 0.25. Using the assessment of deep learning models for brain tumor detection performance modes, ResNet-50, CNN, and Inception V3 are 93.30% accuracy, 81.10%, and 80.00%, respectively [20].

Ahmad Osman's proposal highlights the importance of early and easy brain tumor detection and its impact on mortality rates. VGG-19 is the top performer with 97% accuracy, while EfficientNetB7 has the lowest accuracy at 93%. On the other hand, ResNet-50's accuracy is 94%. The analysis of the implications of CNN used in stress and healthcare reveals both their potential benefits [21].

Muhammad Naeem Tahir extensively explores various classifications for MRI brain tumor photo analysis using classifications such as SOM, KNN SVM, and others. Focusing on brain tumor MRI images for tumor diagnosis, the study aims to extract and select features (texture, color, tumor region, location, and edge) from images. Pre-processing includes segmentation and image filtering, and post-processing includes resizing, classification, and tumor area calculation using DNN. Achieving 90% classification accuracy using DNN is possible and has revealed the efficiency of the algorithm [22].

Anjaneya Teja Sarma Kalvakolanu et al., focus on non-invasive brain tumors and classification using deep learning methods. They used registration and segmentation techniques to separate skull images from MRI images using grab-cut methods that validated tumor features in the processed images. Here, 3064 MRI images were used, especially with a dataset consisting of T1 flare MRI photos. The model achieved a high classification accuracy of 98.83% for training, 96.26% for validation, and 95.18% for the test set. Used ResNet50 as the base model for accurate classification of multiple tumor types. This method gave promising results compared to other studies, which showed that the method works well for brain tumor classification [23].

Gokila Brindha et al., introduce brain tumors using MRI scans of diagnosis. It emphasizes artificial neural network (ANN) and Convolutional neural network (CNN) machine learning for accurate detection and acceleration. The focus is on comparing the performance of artificial neural networks and convolutional neural network models on brain tumor MRI datasets. The main objective of doing this is to demonstrate the efficiency and effectiveness of this learning method in differentiating tumor-affected and normal brains, aiding in the rapid treatment and diagnosis of brain tumors in patients. The ANN model, trained over 50 times, exhibits a training accuracy of 97.13%, 71.51% accuracy of validation, and 80.77% accuracy of testing. Suggests optimization techniques for determining ideal layers and filters for this advanced model [24].

Sidra Sajid et al., introduce a deep learning-based method for segmentation using different MRI modalities. By observing the hybrid CNN architecture and contextual information, the data imbalance and overfitting problems were solved. Validation indicated improved segmentation performance compared to existing techniques on the BRATS 2013 dataset, emphasizing the efficacy of the method to accurately detect and segment brain tumors. The role of the hybrid CNN model for brain tumor segmentation is many. It achieved superior performance compared to state-of-the-art technology [25].

Abdullah A. Asiri et al., An advanced model combining CNN U-Net and ResNet50 for improved brain tumor detection segmentation and classification. Using public datasets for validation and training, the U-Net model performed best in accurately segmenting tumor regions compared to other models, in which the U-Net with Resnet 50 coefficient value came in at 95%. Besides, the proposal and validation for a hybrid model - CNN, U-Net, and ResNet50- for robust brain tumor detection, classification, and differentiation from MRI images. ResNet50 performed well in correctly detecting the presence of tumors. Contributed significantly to advances in medical analysis and patient treatment planning [26].

Aryan Verma et al., focused on early detection of brain tumors using deep learning techniques by MRI slices. Here, VGG 16, ResNet 50, and EfficientNet architectures were analyzed. ResNet 50 provides performance: determined accuracy of 99.37%. It surpasses many of the detection mechanisms that exist in the brain. The model performed very well in triaging patients and assisting clinicians in decision-making. A newly introduced dataset, JMCD (Jabalpur Medical College Dataset), including annotated MR sequences in 140 patients, supports research in targeted brain tumor detection. Its main objective was to improve brain tumor detection methods [27].

Abdullah A. Asiri et al., aimed to enhance brain tumor diagnosis using a computer-aided system and the challenge of manual segmentation from many magnetic resonance images for cancer analysis. They refine VGG19 with CNN architecture through a block-wise mechanism for precision, which proposes the BW-VGG19 architecture. Various Chinese hospitals used a contrast-enhanced MRI dataset between 2005-2020, achieving 98% accuracy using their method. Here, using CNN, VGG16, and VGG19 performed better against the method. Here, the BW-VGG19 model showed an exceptional accuracy of 0.98%, which outperformed other models like CNN and VGG19. This method played a significant role in the detection of brain tumors [28].

Krisna Nuresa Qodri et al., explore the classification of brain tumors in MRI images using Deep learning and transfer learning. Based on brain tumors in the United States and Indonesia, the research was done and found its flaws. In this study, 23,890 adults (10,300 females and 13,590 males) and 3,540 children under 15 years of age collected brain tumor photos. Within this, a public dataset consisting of 253 images (98 tumor-free and 155 tumor images) was used. The study employed residual networks (ResNet), NASNet, Xception, DenseNet, and VGG methods. Among them, ResNet50 and VGG16 achieved 96% accuracy. Despite the best accuracy of ResNet50 and VGG16, the Xception model achieves better accuracy in terms of specificity and sensitivity. These results were able to accurately detect MRI images for brain tumors, particularly highlighting the transfer learning performance demonstrated by ResNet50 and VGG16 [29].

Md. Tanvir Rouf Shawon et al., explore the brain tumor detection for deep neural networks from MRI images. CNN, ResNet50, InceptionV3, EfficientNetB0, and NASNet-Mobile were used here. A pipeline was developed to combine these five models and apply them to a dataset. Using cost-sensitive InceptionV3 and CNN models demonstrates an accuracy of 92.31%. Using the proposed InceptionV3 model achieved 99.33% accuracy. However, this introduces artificial intelligence (AI) to clarify decision-making. The results suggest the effectiveness of InceptionV3 and CNN for brain tumor detection, potentially providing diagnostic applications [30].

Abhishek Anil et al., explore the application of modern technology in brain tumor detection using deep learning from MRI images. This study, using deep learning methods, specifically developed a classification network that can distinguish between brain images with and without tumors, trained by migration. Demonstrates the potential to improve brain tumor detection accuracy and medical imaging efficiency. Here is a dataset created from multiple sources. Then split it into test and training sets. evaluated three networks—AlexNet, VGG16, and VGG19—for performance in tumor detection. VGG19 demonstrated the highest accuracy of 95.78%. Which also outperformed other models. The data used here was able to demonstrate larger MRI image sizes without data loss. [31]

Feature extraction from MRI images means picking out important information, such as unique patterns that indicate brain tumors. Classification is the process of sorting patterns into categories – identifying whether this is a tumor or not. Tried various ways to prove it. One way was to use custom-designed CNNs, which are specialized programs for rendering patterns. Another way would be to use models like Xception and EfficientNetB0 that already know something about the pattern. Taking these factors together, we found that some tumors were better at being identified accurately and with fewer errors. The way these subjects are selected and combined works very well for finding brain tumors from MRI images.

The datasets are used to teach computers about brain tumors and represent sets of MRI images. Think of this as a huge collection of images, each image showing a different aspect of the tumor, which works like a report card. They measure things like accuracy, which indicates how often the computer correctly detects the tumor, and confidence in the predictions. Here, various classifiers have been tested—AdaBoost, KNN, RF, SVM, and Softmax—using these metrics to see which methods would perform better on images for tumor detection. By examining different types of datasets and metrics, we can understand which classifiers performed better in correctly classifying brain tumors. These methods evaluate how well they learn from image sets and how confident they are in their judgments.

Challenges in this field include accurately distinguishing between tumors for which otherwise complex MRI data can be interpreted. Future directions involve different models to handle different tumors and sizes. Another challenge is to ensure that these models perform well across different MRI uses and settings. Advances in deep techniques can advance model sensitivity to subtle tumor characteristics, improving overall detection accuracy. Also ensures credibility. Interpretability will be very important in understanding how decisions are made using these models. The inclusion of more diverse datasets may strengthen the ability of these models to detect atypical tumors. Collaborative efforts between medical experts and AI researchers can address this situation and advance the accuracy and reliability of brain tumor detection using MRI.

The literature review summarised has explored the six CNN models, Xception, InceptionV3, ResNet50, EfficientNetB0, and VGG19, in brain tumor classification using images. Softmax, SVM, RF, KNN, and AdaBoost are used in all these models. Using this classification, all models are given better results. The research paper explores CNN 92.07%, Xception 95.47%, InceptionV3 96.26%, ResNet50 87.03%, EfficientNetB0 97.86%, and lastly, VGG19 82.6%. Among these models, InceptionV3, EfficientNetB0, and Xception Accuracy have performed well. The three models incorporate a combination of features. After doing the combined feature, the result is that InceptionV3 and EfficientNetB0 have an accuracy of 95.86%; on the other hand, EfficientNetB0 and Xception have an average accuracy of 95.9%; and the last two, InceptionV3 and Xception, have an average accuracy of 96.87%. After doing these InceptionV3, EfficientNetB0, and Xception combined features. The top two accuracies were identical, that is, InceptionV3 and Xception, as well as EfficientNetB0 and Xception. Here found the correctness by combining these two methods. That is an ensemble classifier accuracy of 96.9%. Overall, all CNN models show promise in medical imaging. Their efficiency depends on computational resources.

Chapter 3

3. Method

Brain MRI input is followed by data preparation, including image scaling, augmentation, and filtering. Feature extraction is done using VGG19, a scratched CNN, Xception, InceptionV3, ResNet50, and EfficientNetB0. AdaBoost, k-Nearest Neighbors (kNN), Random Forest (RF), Support Vector Machine (SVM), and Softmax are used with each CNN.

These are merged into a two-stage ensemble model that seems to capture the best attributes of the separate models for better prediction. The model finishes with a classification result that is deployed using a simple code base with an interface for adding photos and checking results.

The techniques portion of a thesis might explain the CNN architectures, ensemble learning, and classification model deployment strategy for this approach as shown in the flowchart.

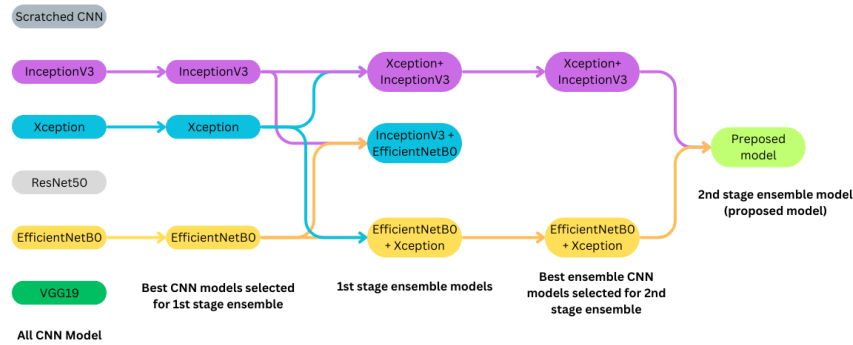


Figure 1: Selection Process of CNN Architectures for a Two-Stage Ensemble Model in Image Processing

3.1 Dataset

The dataset, Br35H (2020), is specifically designed for brain tumor localization. Accurate identification and categorization of brain tumors, which are severe conditions that impact individuals of all ages, are essential for appropriate medical therapies. Brain tumors account for a vast majority, around 85 to 90 percent, of primary Central Nervous System (CNS) tumors [1]. They affect over 11,700 persons each year. The dataset has three folders: "yes," "no," and "pred," which together include 3060 Brain MRI Images. In addition, the situation emphasizes the difficulties presented by the intricacies of brain tumors, particularly in areas where there is a shortage of highly trained neurosurgeons. The suggestion is to introduce a cloud-based automated solution to address these difficulties [32].

3.2 Data Pre-processing

Analysis and machine learning need data preparation, which cleans, transforms, and organizes raw data. Missing values, outliers, and category variables to numerical format are among its duties. This technique provides data consistency and relevance for analysis or model training. Noise, inconsistencies, and missing data are addressed to increase analysis and machine learning model

accuracy and dependability [33]. Standardization, feature engineering, and dimensionality reduction improve data quality during preprocessing [34].

3.3 Filtering:

In signal processing, image processing, and data analysis, filtering modifies or extracts certain data or signal while attenuating others. Filter type and settings rely on analysis or processing goals [35]. For the purpose of applying a filter, an input picture is processed by first applying Gaussian Blur, and then utilizing the OpenCV library to perform Sobel Edge Detection thereafter.

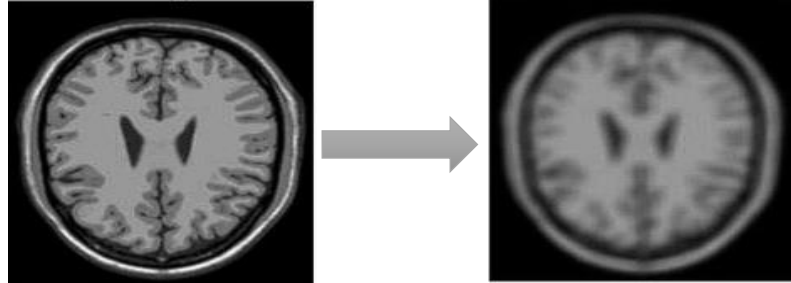


Figure 2: Image comparison for Gaussian filter.

The picture is first smoothed down by the Gaussian Blur function, and then Sobel edge detection is carried out in both the horizontal and vertical axes. Utilizing the Sobel outputs, one may arrive at the ultimate conclusion by performing a calculation to determine the magnitude of the gradients. In order to improve the appearance of edge characteristics in photographs, this function is helpful.

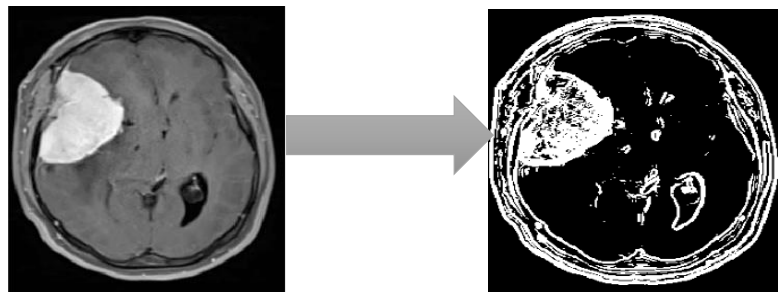


Figure 3: Comparison of Sobel Edge Filter

3.4 Data Augmentation:

The datasets undergo just a minimal amount of preparation, which consists of picture augmentation and scaling. The act of scaling an image is a frequent pre-processing procedure, particularly when dealing with convolutional neural networks (CNNs) or other deep learning models [36]. For the purpose of applying scaling to each picture in the list, it makes use of the OpenCV library. The 'image_size' argument is responsible for determining the particular dimensions that are associated with the resized pictures. This resizing process is frequently utilized for the purpose of standardizing or changing the size of photographs contained inside a dataset. All of the photographs

have been scaled to a resolution of 128*128. Finally, the dataset is subjected to the application of six distinct forms of augmentation procedures, which include horizontal flipping, rotation, height shift, width shift, and fill mode.

It guarantees that all of the photos that are entered have the same height and width. When the batch size is set to 32, it indicates that each training cycle will include the processing of 32 photos simultaneously. Choosing the appropriate batch size may have an effect on the amount of memory that is required for training. The usage of smaller batch sizes is common in the context of online or stochastic training, whilst the use of larger batch sizes allows for the utilization of parallelism and may result in more stable convergence [37].

Data normalization is a technique that is frequently utilized in machine learning applications, particularly when working with image data. The purpose of this technique is to guarantee that neural networks and other models have input ranges that are consistent and controllable. It does this by dividing each member in the array by 255.0, with the goal of bringing the values closer to the range of 0 to 1.

The dataset was loaded as input, and the data and label lists were set to empty. Following that, it goes through each folder in the given path and tries to load every image file in those folders using OpenCV [38]. If the picture loads properly, it is added to the "data" list, and the label (folder name) that goes with it is added to the "labels" list. An error message is shown if there is a problem loading the picture. This method is meant to load pictures from a dataset location and link each image to the label that goes with it.

To shuffle data and labels, use the `shuffle` function from an unnamed library, such as scikit-learn. The shuffle uses a 42-seed randomization to ensure reliability. Machine learning preparation sometimes involves shuffling [39] data samples to prevent misleading patterns from being learned based on input order.

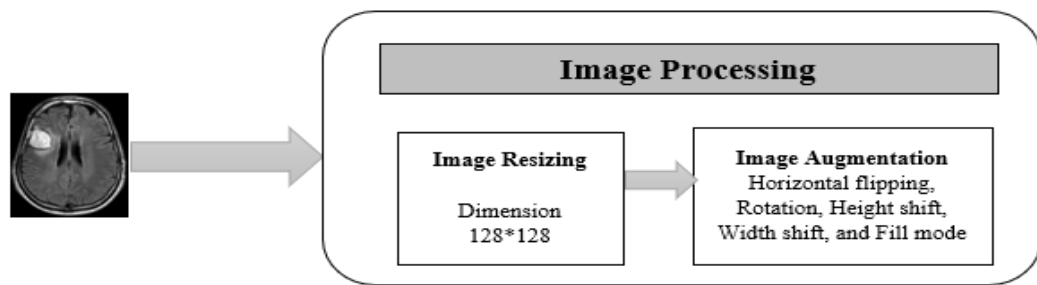


Figure 4: Data Pre-processing Stages

3.5 Feature Extractors:

In the field of machine learning and data analysis, one of the most essential processes is called feature extraction. This process entails translating raw data into a condensed and comprehensible representation that is referred to as features [40]. This crucial stage is of the utmost importance for a variety of applications, including as pattern recognition, natural language processing, and image and signal processing [41]. The key goals of feature extraction are improving the representation of data and lowering the dimensionality of the data in order to promote efficient analysis [42].

During the process of feature extraction, five pre-trained models and a CNN model that was created from scratch were taken into consideration. On account of the fact that it has previously been trained with an issue that is comparable, it has the benefit of requiring less time to train. Scratch CNN [43], Xception [44], InceptionV3 [45], ResNet50 [46], EfficientNetB0 [47], and VGG19 [48] are the five pre-trained CNN models that have been modified and used for feature extraction in this study. These models that had been pre-trained were used on the dataset that was described, and later tweaks were made to the models via the use of random search in order to at least partially offset the effects of overfitting [43].

3.5.1 Proposed Scratch CNN

An approach that is considered to be a pioneer in the area of deep learning for the purpose of medical image analysis is known as Scratch Convolutional Neural Networks (CNNs). The CNNs in question are constructed from the ground up, without relying on models that have been pre-trained [49]. Within the realm of brain magnetic resonance imaging (MRI), these networks have shown remarkable potential by making it feasible to automatically extract precise properties directly from raw pixel data [50]. Because the model is initialized from scratch in Scratch CNNs, it is feasible to design task-specific architectures that are able to recognize tiny patterns within brain MRI data. This is made possible by the fact that Scratch CNNs are available. This not only adds to a better knowledge of neurological illnesses but also increases the accuracy of diagnostic procedures [51].

There is a total of sixteen layers in this design, which includes three convolutional layers, three max pooling layers, one flattens layer, and a dropout layer. A default value of 128 x 128 is assigned to the first input size. One of the goals was to build a Convolutional Neural Network (CNN) For the objective of extracting features from two-dimensional pictures. By using the Keras Application Programming Interface (API), the function generates a sequential model that represents a linear stack of layers. The model is made up of three convolutional layers that are successively stacked, with each layer being followed by a max of pooling. The rectified linear unit (ReLU) activation function is used by the convolutional layers, which are initialized with 32, 64, and 128 filters of size (3, 3), respectively. This activation function enhances the non-linearity in the feature maps. By contributing to spatial down sampling and so lowering the dimensionality of the feature maps, the max-pooling layers, which have a pool size of (2, 2), are also crucial. Importantly, a flatten layer is introduced in order to convert the two-dimensional feature maps into a one-dimensional vector. This helps to get the data ready for the future layers that are completely linked. This function provides a fundamental framework for a CNN-based feature extractor, which is often used in image processing jobs. It is also capable of being expanded or customized to meet the needs of a particular application.

3.5.2 Modified Xception

Xception convolutional neural network (CNN) architecture dominates deep learning for brain MRI processing (Chollet et al., 2017). A 2017 extension to Inception, Xception, uses depth-wise separable convolutions to separate cross-channel and spatial correlations [52]. This unique approach reduces parameter count and enhances feature extraction, boosting computation efficiency without reducing predictive performance.

The Xception deep learning model and conventional machine learning classifiers classify brain MRI images. MRI brain pictures are preprocessed and improved. The Xception model trains and evaluates many classifiers utilising high-level picture attributes, including AdaBoost, K-Nearest

Neighbours (KNN), Random Forest (RF), Support Vector Machine (SVM), and Softmax Regression.

The advanced pre-trained neural network Xception categorises images. Complex MRI patterns and representations are recorded. Brain tumour classifiers are trained using the collected data. The method assesses classifiers by accuracy and log loss, which measure prediction uncertainty. Using extracted features, AdaBoost, KNN, RF, and SVM classifiers classify brain MRI images. Logistic regression for multiclass classification, Softmax Regression, accurately classifies brain tumours.

This complete method classifies brain tumours using deep learning and standard machine learning, showing how sophisticated neural networks and classical classifiers work together. Average accuracy and log loss measures help evaluate and compare the models' medical picture categorization effectiveness. Researchers and practitioners benefit from the architecture's adaptability and feature extraction in medical imaging's ever-changing sector.

Xception excels at brain MRI image analysis. Bai et al. [53] employed Xception in a semi-supervised learning framework for network-based cardiac MR image segmentation, proving its adaptability and dependability in challenging medical imaging applications. Xception was used to locate cancer metastases on gigapixel pathology images by Liu et al. [54]. Prasoon et al. [55] segmented knee cartilage using Xception, demonstrating its deep feature learning in numerous anatomical circumstances. Brain MRI interpretation is difficult; thus, adaptation is crucial.

3.5.3 Modified ResNet50

ResNet50, a ground-breaking deep learning architecture, stands as a pivotal asset in the analysis of brain MRI images. This architecture, celebrated for its profound depth and innovative skip connections, adeptly addresses the challenges of training deep neural networks by mitigating the vanishing gradient problem. Comprising 50 layers, ResNet50 leverages residual learning to facilitate the smooth flow of information through the network, allowing for the extraction of intricate features essential for the nuanced interpretation of brain MRI scans. Its ingenious design, incorporating residual blocks and identity mappings, not only enhances model convergence but also enables the effective capture of subtle patterns indicative of neurological conditions [56].

ResNet50's effectiveness in medical imaging has been corroborated by subsequent works. In the classification phase, a diverse set of classifiers, comprising AdaBoost, K-Nearest Neighbors (KNN), Random Forest (RF), Support Vector Machine (SVM), and Softmax Regression, was employed. Each classifier underwent training utilizing features extracted by ResNet50, and subsequent evaluations were conducted based on accuracy and log loss metrics. Notably, the SVM classifier incorporated probability estimates for log loss calculation, while the Softmax Regression classifier adjusted its maximum number of iterations to enhance convergence which is explicitly set to 2000, indicating a sufficient number of iterations to ensure convergence. The methodology concluded with a comprehensive analysis, employing average accuracy and log loss as key metrics to assess the overall performance of the ResNet50-based approach across various classifiers. This ensures a comprehensive exploration of ResNet50's capabilities in the realm of brain MRI analysis, providing valuable insights for robust classification in medical imaging applications.

3.5.4 Modified EfficientNetB0

The goal of EfficientNet is to solve the problem of striking a balance between computing efficiency and model correctness. Traditionally, while developing Convolutional Neural Networks (CNNs), researchers have mostly concentrated on expanding the network structure in order to enhance

accuracy. Nevertheless, this often results in escalated computational expenses, posing a difficulty in implementing these models on devices with limited resources.

EfficientNet presents a new approach called compound scaling, which consistently adjusts the depth, breadth, and resolution of the network. This novel methodology distinguishes itself via its efficacy, attaining exceptional results in tasks such as picture classification, object identification, and segmentation, while showcasing better utilization of resources in comparison to previous methods. The architecture of EfficientNetB0 consists of recurring building blocks that use depth-wise separable convolutions, batch normalization, and non-linear activation functions such as Swish. The distinguishing feature of EfficientNet is its compound scaling mechanism, which consistently adjusts the depth, breadth, and resolution of the model. The use of specified coefficients in this balanced scaling guarantees a systematic method for getting the best possible performance of the model [47].

The brain MRI dataset is loaded and undergoes preprocessing, which involves shrinking and using EfficientNetB0's preprocessing algorithm. The EfficientNetB0 model, which has been pre-trained on ImageNet, is then used to extract features from both the training and testing datasets, yielding feature vectors. Afterwards, a varied range of classifiers, including AdaBoost, K-Nearest Neighbors (KNN), Random Forest (RF), Support Vector Machine (SVM), and Softmax Regression, are initialized and trained using the retrieved features.

The ``apply_filters`` function in image processing is crucial for improving the distinguishing characteristics in input pictures for brain MRI classification. To reduce noise and provide a cleaner visual representation, a Gaussian blur is applied using a kernel size of (5, 5). The procedure highlights the edges and important elements in the picture, which helps to extract relevant information for further processing and analysis. The function is executed using the OpenCV library, and its use as a first step is crucial in enhancing the input pictures for reliable brain MRI categorization.

Considering accuracy scores and log loss values, the assessment step entails evaluating each classifier's performance on the testing set. The Softmax Regression classifier notably uses probability estimations to calculate log loss. The calculation of average accuracy and average log loss provide a comprehensive evaluation of the feature extraction using EfficientNetB0 across several classifiers. This technique highlights the potential of EfficientNetB0 in analyzing brain MRI and provides useful insights for accurate classification in medical imaging applications.

The model's high level of efficiency and adaptability contribute to its widespread use in diverse applications, such as medical picture categorization and object recognition in autonomous cars. Pre-existing weights for EfficientNetB0 learned on large datasets, such as ImageNet, enhance the process of transfer learning for specific tasks, particularly when there is a scarcity of labelled data [57].

3.5.5 Modified InceptionV3

InceptionV3 is a significant convolutional neural network (CNN) structure specifically created for the purposes of image categorization and feature extraction [58]. The InceptionV3 architecture relies on the inception module, which combines filters of various sizes in a single layer. This allows the network to effectively capture hierarchical characteristics at different scales. The use of 1x1 convolutions helps to decrease computational complexity while maintaining crucial information. The entire design consists of many inception modules, resulting in an efficient and robust feature extraction process.

The InceptionV3 model will be used for extracting features in the specific setting of brain MRI classification, followed by an assessment of several classifiers. The InceptionV3 model is used as

a feature extractor to extract distinctive characteristics from the dataset. Afterwards, a number of classifiers, such as AdaBoost, K-Nearest Neighbors (KNN), Random Forest (RF), Support Vector Machine (SVM), and Softmax Regression, are initialized. Every classifier is trained separately using the characteristics retrieved by InceptionV3 and then assessed for accuracy. Log loss is computed for classifiers that provide probability estimates. The script closes by calculating the mean accuracy and mean log loss for all classifiers, offering a thorough summary of their performance.

Within the field of medical imaging, InceptionV3 has shown significant achievements, namely in tasks pertaining to MRI analysis. The capacity to record complex patterns and hierarchical structures is especially advantageous for identifying minor nuances in medical pictures, such as those seen in the identification of brain tumors [59]. InceptionV3 is a useful tool for medical image analysis applications, where labelled datasets are often scarce, because of its pre-trained weights on large-scale datasets and transfer learning capabilities.

3.5.6 Modified VGG19:

The University of Oxford Visual Geometry Group created the VGG19 convolutional neural network (CNN) for image categorization. Noteworthy features include 3x3 convolutional filters with stride 1 and padding 1 to preserve spatial information across the network. The model's ability to learn hierarchical visual data representations makes it versatile in computer vision applications. Fine-tuning VGG19 for MRI brain tumor diagnosis requires adjusting the model to brain image datasets and using transfer learning from pre-trained weights, such as ImageNet. Sequential convolutional layers and max-pooling layers with a 2x2 filter and stride of 2 improve feature extraction [60].

Starting with brain MRI dataset loading and preprocessing, it follows a methodical methodology. Shuffled, scaled, and normalized images ensure processing-ready input. As a feature extractor, the InceptionV3 model extracts discriminative characteristics from the dataset. After that, AdaBoost, K-Nearest Neighbors (KNN), Random Forest (RF), Support Vector Machine (SVM), and Softmax Regression are initialized. Individual classifiers are trained using InceptionV3 features and tested for accuracy. Additionally, the input layer is resized to 128×128 to align with the picture size. VGG19 is deep and uniform with 19 layers, 16 convolutional and 3 completely linked [43]. Additional sources supporting VGG19's effectiveness and use in various circumstances include [61].

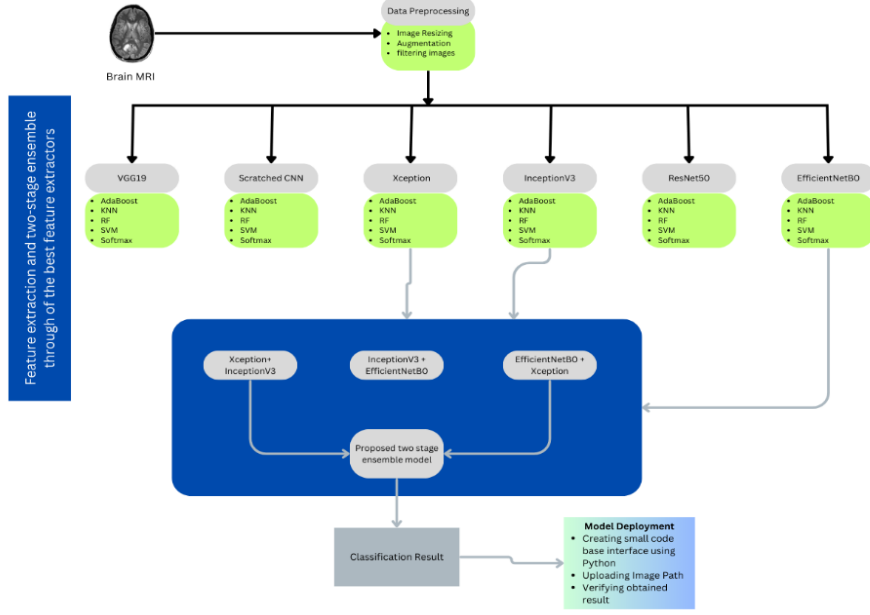


Figure 5: Flowchart of Ensemble Machine Learning Pipeline for Brain MRI Analysis with Two-Stage Model Deployment

3.6 Classifier

A classifier is a computer model or programme that classifies incoming data. It predicts new occurrences by learning patterns and characteristics from labelled training data. Machine learning uses classifiers for image identification, sentiment analysis, and spam detection. They map input characteristics to output classes using learnt patterns. Support vector machines, decision trees, and neural networks are classifiers [62].

3.6.1 Adaboost

Adaptive Boosting (AdaBoost) is a classification and regression ensemble learning technique. The predictions of numerous weak learners (usually basic models, called "weak classifiers") are combined to generate a strong classifier. The approach weights training examples and corrects weak learners' faults in subsequent rounds. Each cycle of boosting trains and evaluates a new weak learner. Misclassified cases gain weight, making them more impactful next iteration. This cycle continues until a certain number of poor learners or a perfect model is attained. AdaBoost is frequently utilized in machine learning applications because it improves weak model accuracy simply and effectively [63]. The sign function (1) and weighted error rate for weak classifier (3) and misclassified samples (2) for Adaboost classifier.

$$H(X) = \text{sign} \left(\sum_{t=1}^T \alpha_t \cdot h_t(X) \right) \quad (1)$$

$$D_i^{(t+1)} = \frac{D_i^{(t)} \cdot \exp(-\alpha_t \cdot y_i \cdot h_t(X_i))}{Z_t} \quad (2)$$

$$\epsilon_t = \sum_{i=1}^N D_i^{(t)} \cdot I(y_i \neq h_t(X_i)) \quad (3)$$

3.6.2 Random Forest

Random Forest is an ensemble learning technique that trains several decision trees and outputs the mode (classification) or mean prediction (regression) of the individual trees. Each tree in Random Forest is trained on a random portion of the training data, and each split considers a random sample of characteristics. By majority voting for classification tasks or average for regression tasks, all tree forecasts are combined to make the final prediction [64]. The approach uses randomization in sample selection (bagging) and feature selection during tree construction, improving robustness and generalization. Random Forest is accurate, overfit-resistant, and adaptable to varied datasets [65]. In random forest the most frequently predicted class is voting on (1) and the equation for margin function on (2).

$$f(x) = \arg \max_{y \in \mathcal{Y}} \sum_{j=1}^J I(y = h_j(x)) \quad (1)$$

$$mg(\mathbf{X}, Y) = av_k I(h_k(\mathbf{X}) = Y) - \max_{i \neq Y} av_k I(h_k(\mathbf{X}) = i) \quad (2)$$

3.6.3 SVM

SVM is a sophisticated supervised machine learning technique for classification and regression. SVM seeks a feature space hyperplane that maximally separates two classes. SVM maximizes the margin, the distance between the hyperplane and the nearest data point from either class. The equation of the hyperplane is given by $f(x) = \langle w, x \rangle + b$, where w is the weight vector, x is the input vector, and b is the bias term. The optimization problem for SVM involves minimizing $1/2 \cdot \|w\|^2$ subject to the constraint that each data point is on the correct side of the margin. The kernel technique maps input data into a higher-dimensional space for SVM to handle non-linearly separable data. The ultimate forecast is based on tree majority [66].

3.6.4 Softmax

The Softmax function is essential to machine learning, especially multiclass classification. It converts raw scores or logits into probabilities to help comprehend model predictions. The Softmax function converts input vector z to output vector $\sigma(z)$, where each element reflects class probability.

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \quad (1)$$

Here z is the input vector of raw scores or logits for each class, $\sigma(z)_i$ is the i -th element of the output vector, representing the probability of class i . K is the total number of classes. Softmax gives

a probability distribution over several classes for a given input in neural network output layers [67]. It improves model interpretability and utility in multiclass settings.

3.7 Tools and Libraries Used

3.7.1 TensorFlow

Deep learning for brain MRI processing requires TensorFlow, an open-source framework. Flexible library [68] enables neural networks and deep learning [69]. TensorFlow built advanced brain MRI image segmentation, sickness classification, and picture-generating solutions [70]. Due to its adaptability, community support, and hardware accelerator interoperability, TensorFlow is vital for brain disease deep learning research [71].

A large ecosystem of tools and extensions makes TensorFlow useful for brain MRI research [72]. TensorFlow's integration with Keras [69] expedites neural network design [73] and prototype [74]. Data preparation, model training, deployment, and monitoring in brain MRI deep learning applications are repeatable and scalable with TensorFlow Extended (TFX) [75][76]. Researchers can process giant brain MRI datasets with TensorFlow's GPU and TPU compatibility [77]. Brain MRI analysis has significantly increased since TensorFlow lets researchers explore new deep learning algorithms and accelerates data preparation and model deployment [78].

3.7.2 Kaggle

A popular platform for machine learning and data science is Kaggle. It does this by hosting tournaments. Creating room for over a million members [80]. The platform provides datasets, coding, and collaboration. Especially useful for brain MRI and deep learning [81]. Kaggle's kernel environment lets users develop, execute, and share code via a web interface for frameworks like TensorFlow [82].

This platform is known for its medical image analysis contests, especially brain MRI segmentation and classification. Participating in these tournaments has advanced medical image analysis. Additionally, Kaggle offers tools and a discussion platform that increases knowledge and execution of deep learning approaches in brain MRI analysis. Kaggle is useful for brain MRI and deep learning enthusiasts. It speeds research and application development by increasing community cooperation and competitiveness through data availability [83].

3.7.3 Scikit-Learn

The analysis of NeuroMRI using learning has made progress thanks to Scikit Learn [84] a well-known machine learning library. This Python package, which is source provides tools, for data preprocessing feature extraction, model selection and assessment in the field of machine learning. Researchers working with brain MRI data benefit from Scikit Learns image segmentation, classification and regression algorithms. Its interface is user friendly and well documented enabling researchers to experiment with machine learning models and conduct learning on brain MRI datasets.

Scikit Learns adaptability has played a role in the development of interpretable deep learning models for analyzing brain MRI data. It offers an ecosystem for creating deep learning pipelines by integrating with other libraries such as NumPy, SciPy and Matplotlib. Moreover, it also supports network architectures like TensorFlow and PyTorch. As a result, Scikit Learn has become a tool

for researchers who employ learning techniques to gain insights, into brain related diseases through MRI scans [85].

3.7.4 OpenCV

OpenCV, a used library and toolkit, for computer vision plays a role in deep learning research focused on brain MRI. With its functions and methods OpenCV simplifies tasks such as preprocessing feature extraction and image manipulation. These steps are vital in preparing MRI data for learning models [86]. Researchers have successfully employed OpenCV to address challenges related to image registration, noise reduction and the extraction of structures from brain MRI data. Thanks to its user interfaces and compatibility with programming languages OpenCV proves adaptable for both researchers and practitioners [87].

In the realm of learning analysis for brain MRI OpenCV is indispensable as it enables researchers to diversify their training datasets through data augmentation techniques [88]. By leveraging OpenCV's transformations brightness adjustments and noise injection on MRI data researchers can enhance training effectiveness. Promote better generalization of deep neural networks [89]. Data augmentation helps reduce overfitting issues while ensuring that deep learning models trained on brain MRI perform in real world scenarios [90].

Given its versatility feature set and support for programming languages; OpenCV is an essential tool for analyzing brain MRI with deep learning techniques [91]. Through its utilization, in data preprocessing, augmentation procedures and image manipulation tasks; neuroimaging researchers and medical professionals can leverage the power of learning to accurately analyze brain MRI scans efficiently [92].

3.7.5 NumPy

The NumPy package, a part of the Python ecosystem plays a role, in analyzing brain MRI using deep learning techniques. Known as Numerical Python NumPy is a library that allows researchers and practitioners to effectively handle multidimensional arrays and matrices. It also offers mathematical functions for performing array operations. By combining operations with data manipulation capabilities NumPy empowers researchers and practitioners to preprocess, analyze and modify MRI data within the learning pipeline.

When it comes to brain MRI analysis NumPy proves invaluable for tasks such as normalization, scaling and transformation. These operations prepare the input data for learning models. Moreover, NumPy seamlessly integrates with deep learning frameworks like TensorFlow and PyTorch simplifying data preparation well as model training and evaluation processes. Researchers heavily rely on NumPy's robust linear algebra capabilities and statistical calculations to extract features from brain MRI datasets and gain insights. The ability of NumPy to handle data manipulation along with its prowess makes it an essential tool for learning based brain MRI analysis [93].

Scientists and machine learning enthusiasts have an affinity towards NumPy due to its versatility, performance and extensive documentation. The simple syntax of the language coupled with its array operations makes it particularly attractive for brain MRI researchers dealing with image collections. Moreover, being an open-source library promotes cooperation and simplifies the sharing of code within the community as a whole. The collaborative nature of this environment resulted in creating a system that includes customized tools and procedures designed particularly for analyzing MRI data [94].

When it comes to analyzing brain MRIs using learning techniques NumPy is indispensable in enabling data manipulation along, with advanced mathematical operations.

With its abilities to manipulate data and perform operations along, with its seamless integration into well-known deep learning frameworks NumPy plays a crucial role as a valuable research tool in this domain. As scientists continue to make advancements in MRI based diagnosis and treatment NumPy continues to serve as a tool, for data preprocessing, analysis and extracting meaningful features [95].

3.7.6 Joblib

Joblib, a used Python package provides a solution, for computationally demanding tasks. It proves valuable when working with learning models. The Scikit learn ecosystem [96] makes use of Joblib to parallelize and cache Python routines. This feature greatly assists researchers and practitioners, in managing memory and computational resources [97]. With Joblib experts can easily serialize Python objects allowing them to store and retrieve machine learning models and data [98]. The integration of learning within this library significantly enhances the speed and reproducibility of brain magnetic resonance imaging research.

3.7.7 Matplotlib

Matplotlib, a used Python tool, enables the creation of high-quality data visualizations, including brain MRI patterns. This program offers users a user interface to create static, animated and interactive graphs and plots. Deep learning and neuroimaging researchers rely on Matplotlib to visually represent their findings, making it easier to understand the architecture and patterns of MRI brain scans.

One of the reasons Matplotlib is highly regarded is its versatility in customizing plot visuals and properties [99]. Researchers have the flexibility to adjust colors, line styles, markers and annotations in their visualizations to depict their results and meet publishing requirements accurately. Additionally, Matplotlib seamlessly integrates with deep learning frameworks like NumPy and Pandas to visualize numerical data.

Regarding learning-based brain MRI analysis, Matplotlib becomes indispensable. Its flexibility empowers researchers to create visually appealing graphics that simplify the communication of neuroimaging data. Medical image analysis researchers can enhance readability and impact by leveraging the capabilities offered by Matplotlib.

Chapter 4

4. Result

This thesis aimed to improve the precision of brain tumor detection using MRI images. Several deep learning models such, as CNN, InceptionV3, Xception, EfficientNetB0, ResNet50 and VGG19 were analyzed. To ensure feature extraction and classification the dataset underwent pre-processing techniques, like Augmentation, Gaussian Blurring and Sobel Edge Detection to optimize the quality of the images.

Confusion Matrix:

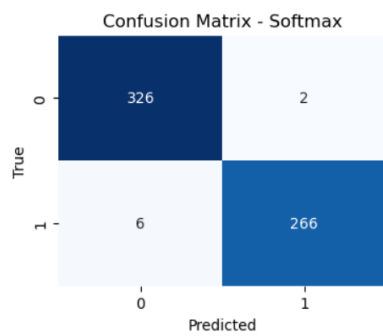


Figure 6: Softmax

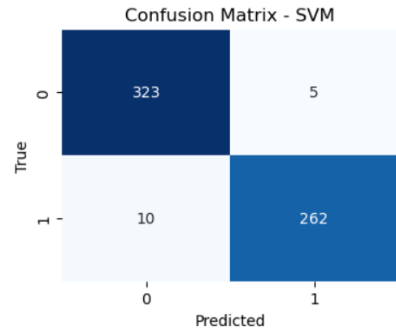


Figure 7: SVM

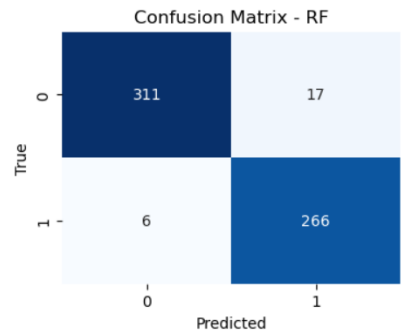


Figure 8: RF

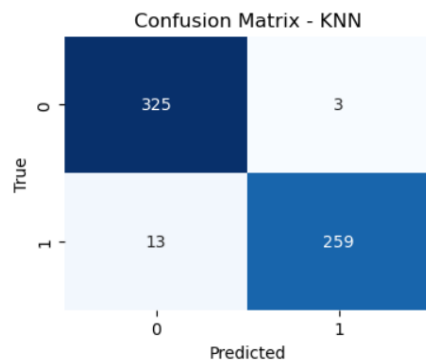


Figure 9: KNN

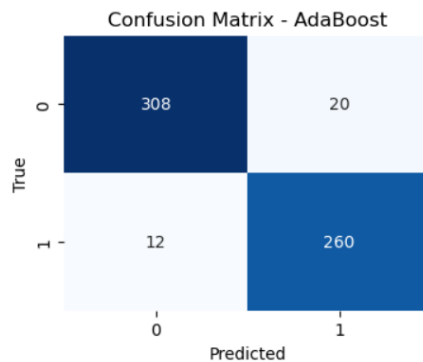


Figure 10: AdaBoost

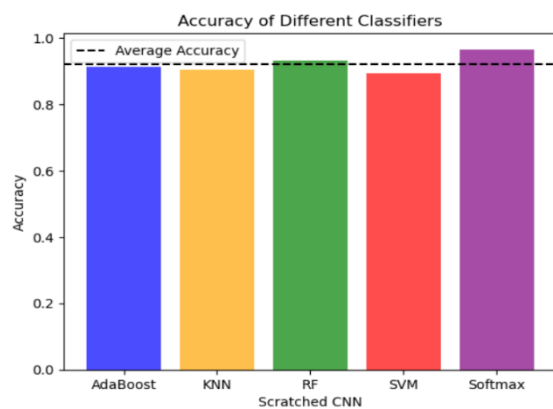


Figure 11: Scratched CNN

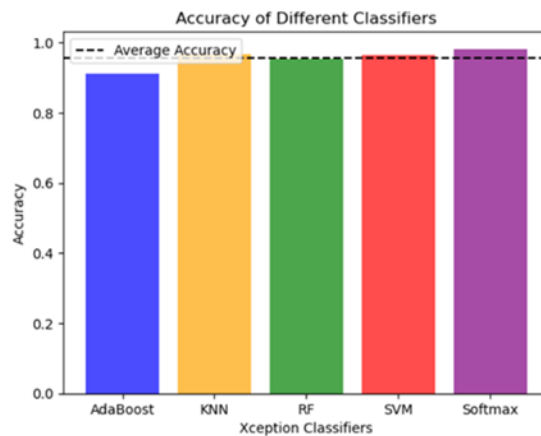


Figure 12: Xception

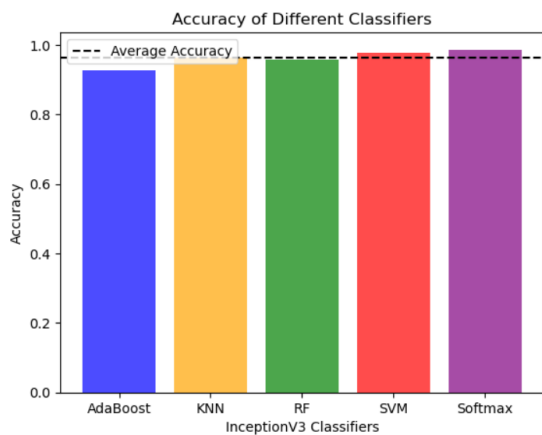


Figure 13: InceptionV3

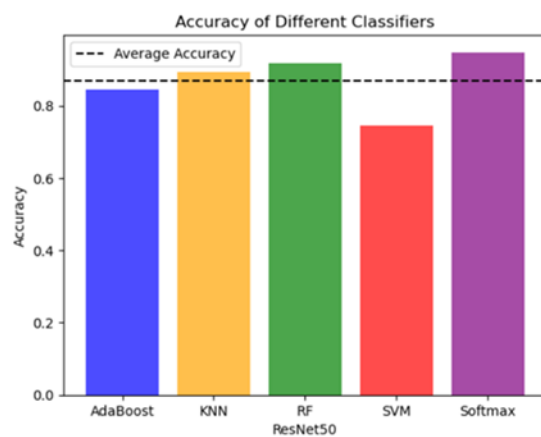


Figure 14: RestNet50

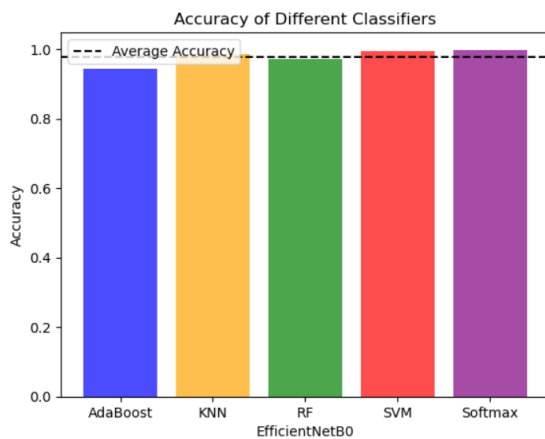


Figure 15: EfficientNetB0

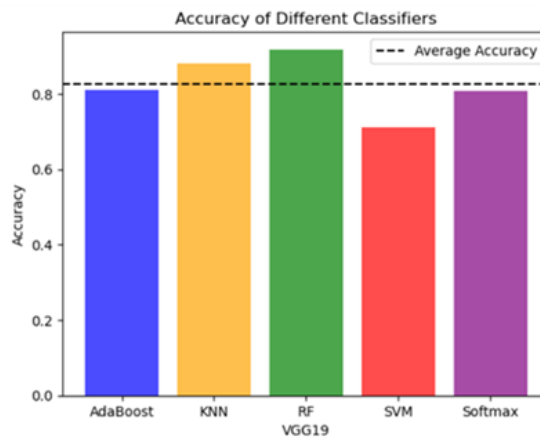


Figure 16: VGG19

Various machine learning classifiers were used to evaluate the performance of six neural network (CNN) architectures. These architectures included Scratched CNN, Xception, InceptionV3, ResNet50 EfficientNetB0 and VGG19. Among them EfficientNetB0 showcased classification capability with the average accuracy of 97.87% and the lowest average log loss of 0.2571. On the hand ResNet50 demonstrated suboptimal classification performance, with the average accuracy of 87.03% and the highest average log loss of 0.4786. Xception and InceptionV3 performed competitively with accuracies of 95.47% and 96.27% respectively highlighting their effectiveness.

AdaBoost and Random Forest classifiers consistently achieved performance across all CNN architectures analyzed in this study. Support Vector Machines (SVM) and k Nearest Neighbors (KNN) however showed outcomes in terms of their performance.

Regarding log loss Softmax consistently exhibited performance, across all CNN architectures evaluated.

These findings underscore the importance of selecting a CNN architecture and classifier when performing image classification tasks. Amongst the models evaluated in this study EfficientNetB0 stood out as the performing model.

Combining 3 best-performing models with each other EfficientNetB0 & Xception, InceptionV3 & EfficientNetB0, InceptionV3 & Xception.

	Accuracy	Loss
AdaBoost	92.33%	62.84%
KNN	97.00%	13.72%
RF	96.33%	18.01%
SVM	95.00%	13.55%
Softmax	98.83%	3.63%
Average	95.90%	22.35%

Table 1: EfficientNetB0 & Xception

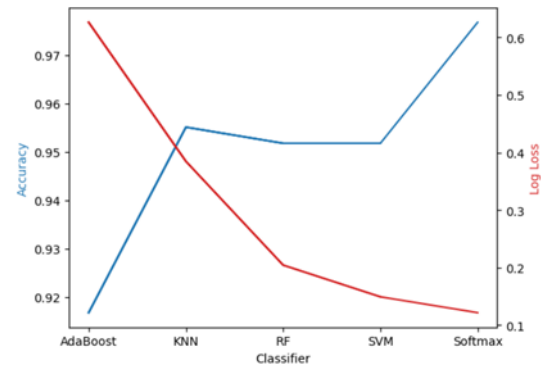


Figure 17: Average Accuracy and Loss

EfficientNetB0 and Xception were combined for analysis. AdaBoost achieved an accuracy of 92.33% and a log loss of 62.84%. KNN performed better with 97.00% accuracy and a lower log loss of 13.72%. Random Forest and SVM displayed performance reaching, over 95% accuracy with log loss values. Notably Softmax stood out with the accuracy at 98.83% and the lowest log loss of 3.63%. On average the models achieved an accuracy of 95.90% with a log loss of 22.35% indicating high performance and reliable predictions, across different methodologies.

	Accuracy	Loss
AdaBoost	93.83%	62.51%
KNN	96.33%	30.90%
RF	94.50%	20.84%
SVM	96.00%	9.05%
Softmax	98.67%	5.91%
Average	95.87%	25.84%

Table 2: EfficientNetB0 & InceptionV3

In combined EfficientNetB0 and InceptionV3 architectures, AdaBoost model achieved an accuracy of 93.83% with a log loss of 62.51%. The KNN, Random Forest, SVM and Softmax models all showed accuracies ranging from 94.50%, to 98.67% with the Softmax model leading the pack. On average across all models achieved an accuracy of 95.87% and a log loss of 25.84%. These results demonstrate that our models performed overall with some models showing exceptional performance that makes them well suited for tasks requiring high predictive accuracy.

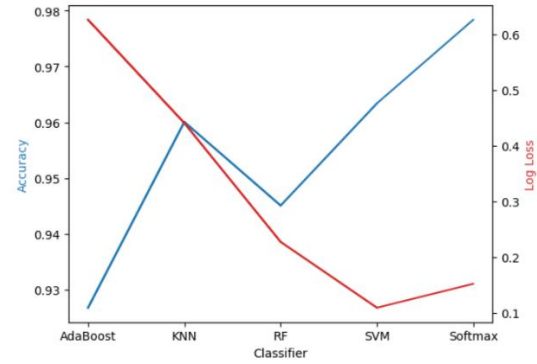


Figure 18: Average Accuracy and Loss

	Accuracy	Loss
AdaBoost	94.67%	62.18%
KNN	97.33%	17.41%
RF	96.00%	20.49%
SVM	97.50%	5.14%
Softmax	98.83%	4.46%
Average	96.87%	21.94%

Table 3: InceptionV3 & Xception

The combination of InceptionV3 and Xception, in machine learning models produced outcomes. AdaBoost demonstrated accuracy at 94.67% closely followed by KNN and SVM which achieved accuracies above 97%. Softmax performed well with an accuracy of 98.83%. The overall average accuracy of 96.87% and a low average log loss of 21.94% highlight the effectiveness of this architecture, in achieving accuracy and minimizing errors across different tasks.

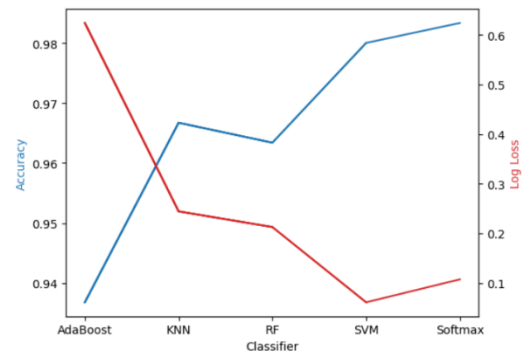


Figure 19: Average Accuracy and Loss

From these 3 models picking best 2 for final model EfficientNetB0 & Xception and InceptionV3 & EfficientNetB0.

	Accuracy	Loss
AdaBoost	94.67%	61.89%
KNN	97.33%	17.41%
RF	96.33%	17.78%
SVM	97.50%	6.13%
Softmax	98.67%	4.46%
Average	96.90%	21.53%

Table 4: EfficientNetB0, Xception & InceptionV3

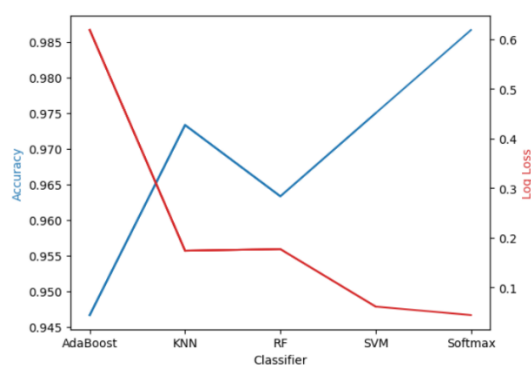


Figure 20: Average Accuracy and Loss

The combination of EfficientNetB0, Xception and InceptionV3, in machine learning models is a blend of deep learning architectures. This group of models has shown performance across classification tasks highlighting its effectiveness in achieving high predictive accuracy and minimizing errors.

AdaBoost, which is one of the methods achieved an accuracy rate of 94.67%. This demonstrates its capability to enhance the performance of learners. Additionally, the low log loss score of 61.89% indicates that it can provide accurate probability estimates with consistency.

K Nearest Neighbors (KNN) displayed an accuracy rate of 97.33% indicating its proficiency in classifying data points based on their proximity in feature space. The corresponding log loss score of 17.41% suggests that KNN provides calibrated probability estimates.

Random Forest (RF) a method achieved a strong accuracy rate of 96.33%. Its log loss score of 17.78% indicates reliable probability estimation abilities making it a dependable choice for classification tasks.

Support Vector Machine (SVM) exhibited an accuracy rate of 97.50%. With a log loss score, as 6.13% SVM not only demonstrates accurate classification but also provides calibrated probability scores effectively.

Softmax, a learning technique achieved an impressive accuracy rate of 98.67% and demonstrated a remarkably low log loss of 4.44%. This clearly emphasizes the capability of networks to capture intricate patterns, within data.

Ensemble methods have been utilized to combine the abilities of EfficientNetB0, Xception and InceptionV3 models. This combination has resulted in a performance that accurately reflects the capabilities of classifiers. The final accuracy metric, for this approach stands impressively at 96.90% highlighting the effectiveness of merging models to enhance accuracy.

```

# Counting 'Yes' and 'No' predictions and returning the majority
def majority_vote(results):
    yes_count = sum([1 for prediction in results.values() if prediction.lower() == 'yes'])
    no_count = sum([1 for prediction in results.values() if prediction.lower() == 'no'])

    return 'Yes' if yes_count > no_count else 'No'

# Example usage
image_path_to_predict = 'C:/Users/mohiu/OneDrive/Python/y1.jpg'
result = predict_image_class(image_path_to_predict, loaded_classifiers)

if result is not None:
    print("Predictions by individual classifiers :")
    for name, prediction in result.items():
        print(f"{name}: {prediction}")

    # Get the majority vote
    majority_prediction = majority_vote(result)
    print(f"Majority Prediction: {majority_prediction}")
else:
    print("Error loading the image.")

1/1 [=====] - 1s 1s/step
1/1 [=====] - 1s 1s/step
1/1 [=====] - 1s 570ms/step
Predictions by individual classifiers :
AdaBoost: yes
KNN: yes
RF: yes
SVM: yes
Softmax: yes
Majority Prediction: Yes

```

Figure 21: Code base Interface

Create a code base interface where the path to an MRI image was provided. The ensemble model effectively made predictions for image classifications by utilizing three pretrained neural networks; InceptionV3, EfficientNetB0 and Xception. The final prediction, for an image was determined by taking a majority vote from these classifiers ensuring a dependable method of classification.

It is worth noting that there can be conflicting outputs among classifiers when analyzing an MRI image, which can pose challenges to achieving predictive accuracy. Despite these obstacles the ensemble model offers an approach towards achieving accuracy in identifying brain tumors in MRI images. These results affirm that deep learning techniques advanced CNN architectures and ensemble methods play a role in improving healthcare outcomes for individuals suspected of having brain tumors.

This outcome demonstrates the effectiveness of learning models in detecting brain tumors. The high rates of accuracy attained by these models suggest their potential for applications. Further validation and refinement are necessary to ensure performance in real world scenarios. This research contributes insights into the application of AI, in diagnostics specifically within the critical field of brain tumor detection.

Chapter 5

5. Discussion

This study used image processing techniques with MRI images, which are a unique method for detecting brain tumors. Here, the filtering identifies the brain tumor part after removing the noise from the image. Here, the Softmax, SVM, RF, KNN, and AdaBoost classifiers were used. Here are six types of models such as: CNN, InceptionV3, Xception, EfficientNetB0, ResNet50, and VGG19. Among these models, EfficientNetB0, Xception, and InceptionV3 have good accuracy. Which has better performance and accuracy compared to other models. The combined accuracy is 96.9%. Also, good results were obtained by using an image.

5.1 Limitations

Considering the work done here and the good results, some limitations should be mentioned and acknowledged

5.2 Limited Dataset Size

A small data set can be used for the study. For which it is possible to bring better results. Using larger data sets can sometimes lead to errors. For which saturated results are not available. If the use of a small dataset has courage, it can bring accurate results

5.3 Tumor-type Scope

Studies may focus on a specific subset of brain tumor types, which may contribute to and limit the results. Moreover, the brain tumor image size can be better and clearer, so it is possible to get better results. In addition, some other techniques may improve utility by including different tumor types.

5.4 Limited Clinical validity

This procedure requires examining patients to see if they have a brain tumor. It is never possible to say exactly without testing. It would involve doctors and patients, and it would have to be used by people in hospitals to look for brain tumors. It must be checked whether it is safe. It's not enough for it to work well in the lab, it has to work properly when the doctor and patient use it. This new method is good enough for doctors to use in hospitals to help people with brain tumors.

5.5 Limited Comparison with Existing Method

Despite significant reliance on this approach, the use here may not directly correspond to other modern approaches. By comparing it with other previously proven techniques, a corresponding idea can be obtained and efficiency can be measured.

5.6 Limited Clinical Validation

The issues that are mentioned in this paper and the computational and hardware requirements may not be included in the study. The deployment and real-time implementation of such systems here may be hampered by resource or computational limitations.

Chapter 6

6. Conclusion

Brain tumors can be very scary and deadly, and it can lead to cancer in the long run. In this research, convolutional neural network (CNN) models accurately classify brain tumor detection using MRI images. There are six types of models. Such as CNN, InceptionV3, Xception, EfficientNetB0, ResNet50, and VGG19—in detecting brain tumors through MRI analysis. EfficientNetB0, Xception, and InceptionV3 have had better accuracy after combining them. Support Vector Machines (SVM), K-Nearest Neighbours (KNN), Softmax, Random Forest (RF), and AdaBoost, all of which create a new model using classification. Their final accuracy came to 96.9%. and reflected the accuracy of an ensemble classifier. Tried to get as much accuracy as possible.

However, this is highlighted by the conflicting output between classifiers for a single MRI image. Despite the predictive challenges, this model makes it easy to detect the presence of brain tumors in images. The models used here demonstrate promising accuracy rates and warrant further validation and refinement to ensure reliable performance. These machine-learning techniques are critical to improving healthcare outcomes for brain tumor diagnosis and patient care.

6.1 Future Work

In the future, Here will increase the accuracy of our models and aim to enhance them by using more advanced methods. The dataset is used from this source. Also, can also increase our dataset and add more images from here. Which identifies the better results. This research uses the models to ensure user-friendliness and ease in obtaining accuracy metrics. The use of these models can significantly benefit healthcare practitioners for brain tumors from MRI scans. Future developments will improve the user interface to make model deployment and clinicians easy access to precise predictions. In the future also work on Alexnet, Vgg-16, MobileNet, DenseNet121, EfficientNetB7, etc. If these models are well used, then it can get good results in the future. Also, work on more classifiers in this paper. Like Gradient Boosting Machines (GBM), Naive Bayes Classifiers, Decision Trees, Neural Networks, Logistic Regression, and Ensemble Techniques. Future work may focus on improving the algorithms for greater accuracy and performance, to improve results. Doctors should be able to easily detect brain tumors. Moreover, also can also work on the issue that patients do not have any kind of problems or dilemmas. Also, various parameters will be used, such as accuracy, specificity, time, efficiency, and many more. An automated system needs to be introduced that can detect the tumor at an early stage so that a better treatment plan can be made.

Bibliography

- [1] "Brain and spine tumors," NewYork-Presbyterian, <https://www.nyp.org/neuro/brain-spine-tumors> (accessed Dec. 7, 2023).
- [2] "Neurosurgery publishes new CNS guidelines for nonfunctioning pituitary adenomas," Neurosurgery Publishes New CNS Guidelines for Nonfunctioning Pituitary Adenomas | Wolters Kluwer, <https://www.wolterskluwer.com/en/news/neurosurgery-publishes-new-cns-guidelines-for-nonfunctioning-pituitary-adenomas> (accessed Nov. 2, 2023).
- [3] L. Zheng and Z. Li, "Virtual Namesake Point Multi-Source Point Cloud Data Fusion Based on FPFH Feature Difference," MDPI, Aug. 12, 2021. [Online]. Available: <https://www.mdpi.com/1424-8220/21/16/5441>
- [4] Y. Chopra, P. Kaushik, S. P. Singh Rathore, and P. Kaur, "Uncovering Semantic Inconsistencies and Deceptive Language in False News Using Deep Learning and NLP Techniques for Effective Management | International Journal on Recent and Innovation Trends in Computing and Communication," Aug. 18, 2023. [Online]. Available: <https://ijritcc.org/index.php/ijritcc/article/view/7256>
- [5] A. Stanzione, F. Verde, V. Romeo, F. Boccadifuoco, P. P. Mainenti, and S. Maurea, "Radiomics and machine learning applications in rectal cancer: Current update and future perspectives," Radiomics and machine learning applications in rectal cancer: Current update and future perspectives, Jan. 01, 2021. [Online]. Available: <https://www.iris.unina.it/handle/11588/877742>
- [6] Hindawi, A. El-Baz, G. Gimel'farb, and K. Suzuki, "Machine Learning Applications in Medical Image Analysis," Machine Learning Applications in Medical Image Analysis, Apr. 13, 2017. [Online]. Available: <https://www.hindawi.com/journals/cmmm/2017/2361061/>
- [7] "Three-dimensional printing of metals for biomedical applications," Three-dimensional printing of metals for biomedical applications - ScienceDirect, Aug. 20, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S259000641930050X?via%3Dihub>
- [8] "Brain Tumor - Statistics," Cancer.Net, Jun. 26, 2012. [Online]. Available: <https://www.cancer.net/cancer-types/brain-tumor/statistics>
- [9] "Brain Tumors and Brain Cancer," Brain Tumors and Brain Cancer | Johns Hopkins Medicine, Jan. 01, 2024. [Online]. Available: <https://www.hopkinsmedicine.org/health/conditions-and-diseases/brain-tumor>
- [10] "Cancer today," Cancer Today. [Online]. Available: <http://gco.iarc.fr/today/home>
- [11] D. I. Prasad, "Spectrum of Endocrine Disorders in Patients Attending Endocrine OPD in a Tertiary Care Hospital in Bangladesh," Journal of Medical Science And clinical Research, vol. 08, no. 03, Mar. 2020, doi: 10.18535/jmscr/v8i3.05. [Online]. Available: <http://dx.doi.org/10.18535/jmscr/v8i3.05>
- [12] "Improved Multiclass Brain Tumor Detection via Customized Pretrained EfficientNetB7 Model," Improved Multiclass Brain Tumor Detection via Customized Pretrained EfficientNetB7 Model | IEEE

Journals & Magazine | IEEE Xplore. [Online]. Available:
<https://ieeexplore.ieee.org/abstract/document/10287923>

- [13] “Development of hybrid models based on deep learning and optimized machine learning algorithms for brain tumor Multi-Classification,” Development of hybrid models based on deep learning and optimized machine learning algorithms for brain tumor Multi-Classification - ScienceDirect, Oct. 18, 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0957417423026611>
- [14] A. S. Farhan, M. Khalid, and U. Manzoor, “PRCnet: An Efficient Model for Automatic Detection of Brain Tumor in MRI Images,” *bioRxiv*, Oct. 01, 2023. [Online]. Available: <https://www.biorxiv.org/content/10.1101/2023.09.28.560042v1>
- [15] G. Balaji, R. Sen, and H. Kirty, “Detection and Classification of Brain tumors Using Deep Convolutional Neural Networks,” *arXiv.org*, Aug. 28, 2022. [Online]. Available: <https://arxiv.org/abs/2208.13264v1>
- [16] R. Tamilarasi and S. Gopinathan, “Inception Architecture for Brain Image Classification,” *Journal of Physics: Conference Series*, vol. 1964, no. 7, p. 072022, Jul. 2021, doi: 10.1088/1742-6596/1964/7/072022. [Online]. Available: <http://dx.doi.org/10.1088/1742-6596/1964/7/072022>
- [17] Y. Xie et al., “Convolutional Neural Network Techniques for Brain Tumor Classification (from 2015 to 2022): Review, Challenges, and Future Perspectives,” *MDPI*, Jul. 31, 2022. [Online]. Available: <https://www.mdpi.com/2075-4418/12/8/1850>
- [18] S. Ahmad and P. K. Choudhury, “On the Performance of Deep Transfer Learning Networks for Brain Tumor Detection Using MR Images,” *IEEE Access*, vol. 10, pp. 59099–59114, 2022, doi: 10.1109/access.2022.3179376. [Online]. Available: <http://dx.doi.org/10.1109/access.2022.3179376>
- [19] N. Ullah et al., “An Effective Approach to Detect and Identify Brain Tumors Using Transfer Learning,” *MDPI*, Jun. 02, 2022. [Online]. Available: <https://www.mdpi.com/2076-3417/12/11/5645>
- [20] M. I. Mahmud, M. Mamun, and A. Abdelgawad, “A Deep Analysis of Brain Tumor Detection from MR Images Using Deep Learning Networks,” *Algorithms*, vol. 16, no. 4, p. 176, Mar. 2023, doi: 10.3390/a16040176. [Online]. Available: <http://dx.doi.org/10.3390/a16040176>
- [21] B. Alsabbagh, “Image-classification for Brain Tumor using Pre-trained Convolutional Neural Network,” *Image-classification for Brain Tumor using Pre-trained Convolutional Neural Network*, Sep. 18, 2023. [Online]. Available: <https://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-336619>
- [22] M. N. Tahir, “Theseus: Classification and characterization of brain tumor MRI by using gray scaled segmentation and DNN,” *Theseus*, Jan. 01, 2018. [Online]. Available: <http://www.theseus.fi/handle/10024/151825>
- [23] A. T. Sarma Kalvakolanu, “Brain Tumor Detection and Classification from MRI Images,” *DigitalCommons@CalPoly*, May 03, 2021. [Online]. Available: <https://digitalcommons.calpoly.edu/theses/2267>
- [24] P. G. Brindha, M. Kavinraj, P. Manivasakam, and P. Prasanth, “Brain tumor detection from MRI images using deep learning techniques – IOPscience”, *IOPscience*, Feb. 01, 2021. [Online]. Available: <https://iopscience.iop.org/article/10.1088/1757-899X/1055/1/012115>

- [25] S. Sajid, S. Hussain, and A. Sarwar, "Brain Tumor Detection and Segmentation in MR Images Using Deep Learning - Arabian Journal for Science and Engineering," SpringerLink, Jun. 28, 2019. [Online]. Available: <https://link.springer.com/article/10.1007/s13369-019-03967-8>
- [26] A. A. Asiri et al., "Brain Tumor Detection and Classification Using Fine-Tuned CNN with ResNet50 and U-Net Model: A Study on TCGA-LGG and TCIA Dataset for MRI Applications," MDPI, Jun. 26, 2023. [Online]. Available: <https://www.mdpi.com/2075-1729/13/7/1449>
- [27] A. Verma, N. Gupta, P. Bhatele, and P. Khanna, "JMCD Dataset for Brain Tumor Detection and Analysis Using Explainable Deep Learning - SN Computer Science," SpringerLink, Nov. 03, 2023. [Online]. Available: <https://link.springer.com/article/10.1007/s42979-023-02308-9>
- [28] A. A. Asiri et al., "Block-Wise Neural Network for Brain Tumor Identification in Magnetic Resonance Images," Computers, Materials & Continua, vol. 73, no. 3, pp. 5735–5753, 2022, doi: 10.32604/cmc.2022.031747. [Online]. Available: <http://dx.doi.org/10.32604/cmc.2022.031747>
- [29] K. N. Qodri, I. Soesanti, and H. A. Nugroho, "Image Analysis for MRI-Based Brain Tumor Classification Using Deep Learning," IJITEE (International Journal of Information Technology and Electrical Engineering), vol. 5, no. 1, p. 21, Jun. 2021, doi: 10.22146/ijitee.62663. [Online]. Available: <http://dx.doi.org/10.22146/ijitee.62663>
- [30] M. T. Rouf Shawon, G. M. Shahariar Shibli, F. Ahmed, and S. K. Saha Joy, "Explainable Cost-Sensitive Deep Neural Networks for Brain Tumor Detection from Brain MRI Images considering Data Imbalance," arXiv.org, Aug. 01, 2023. [Online]. Available: <https://arxiv.org/abs/2308.00608v1>
- [31] A. Anil, A. Raj, H. Aravind Sarma, N. C. R, and D. P L, "Brain Tumor detection from brain MRI using Deep Learning," International Journal of Innovative Research in Applied Sciences and Engineering, vol. 3, no. 2, p. 458, Aug. 2019, doi: 10.29027/ijirase.v3.i2.2019.458-465. [Online]. Available: <http://dx.doi.org/10.29027/ijirase.v3.i2.2019.458-465>
- [32] "Br35H :: Brain Tumor Detection 2020," Br35H :: Brain Tumor Detection 2020 | Kaggle. [Online]. Available: <https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection>
- [33] T. Hastie, J. Friedman, and R. Tibshirani, "The Elements of Statistical Learning," Springer Series in Statistics, 2001, doi: 10.1007/978-0-387-21606-5. [Online]. Available: <http://dx.doi.org/10.1007/978-0-387-21606-5>
- [34] Raschka, S., & Mirjalili, V. (2019). Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow 2, Packt Publishing Ltd.
- [35] R. C. Gonzalez, R. E. Woods, and B. R. Masters, "Digital Image Processing, Third Edition," Journal of Biomedical Optics, vol. 14, no. 2, p. 029901, 2009, doi: 10.1117/1.3115362. [Online]. Available: <http://dx.doi.org/10.1117/1.3115362>
- [36] N. F. Aurna, M. A. Yousuf, K. A. Taher, A. K. M. Azad, and M. A. Moni, "A classification of MRI brain tumor based on two stage feature level ensemble of deep CNN models," Computers in Biology and Medicine, vol. 146, p. 105539, Jul. 2022, doi: 10.1016/j.compbiomed.2022.105539. [Online]. Available: <http://dx.doi.org/10.1016/j.compbiomed.2022.105539>. [Accessed: Jan. 08, 2024]

- [37] J. Brownlee, "How to control the stability of training neural networks with the batch size," MachineLearningMastery.com, <https://machinelearningmastery.com/how-to-control-the-speed-and-stability-of-training-neural-networks-with-gradient-descent-batch-size/> (accessed Jan. 8, 2024).
- [38] Bradski, G., & Kaehler, A. (2008). "Learning OpenCV: Computer vision with the OpenCV library", O'Reilly Media, Inc.
- [39] T. Linh, T. Cao, and Y. Hoshino, "Data Pre-processing for a Neural Network Trained by an Improved Particle Swarm Optimization Algorithm," International Journal of Computer Applications, vol. 154, no. 1, pp. 1–8, Nov. 2016, doi: 10.5120/ijca2016912022. [Online]. Available: <http://dx.doi.org/10.5120/ijca2016912022>. [Accessed: Jan. 08, 2024]
- [40] Belhumeur, P. N., Hespanha, J. P., & Kriegman, D. J. (1997). "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection." IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7), 711-720. DOI: 10.1109/34.598228
- [41] Jurafsky, D., & Martin, J. H. (2019). "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition" 3rd ed. Pearson, 2019.
- [42] Guyon, I., & Elisseeff, A. (2003). "An Introduction to Variable and Feature Selection." Journal of Machine Learning Research, 3, 1157-1182. DOI: 10.5555/944919.944968
- [43] Aurna, N. F., Yousuf, M. A., Taher, K. A., Azad, A. K. M., & Moni, M. A. (2022). "A classification of MRI brain tumor based on two stage feature level ensemble of deep CNN models". Computers in biology and medicine, 146, 105539. <https://doi.org/10.1016/j.compbimed.2022.105539>
- [44] F. Chollet, "Xception: deep learning with depthwise separable convolutions", in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258
- [45] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, no. 1, Feb. 2017, doi: 10.1609/aaai.v31i1.11231. [Online]. Available: <http://dx.doi.org/10.1609/aaai.v31i1.11231>. [Accessed: Jan. 01, 2024]
- [46] K. He, X. Zhang, S. Ren, J. Sun, "Deep residual learning for image recognition", in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778
- [47] M Tan, Q Le, "Efficientnet: rethinking model scaling for convolutional neural networks", in: International Conference on Machine Learning, PMLR, 2019, pp. 6105–6114
- [48] K. Simonyan, A. Zisserman, "Very deep convolutional networks for largescale image recognition", 2014 arXiv, preprint, arXiv:1409.1556.
- [49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
- [50] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2015.

- [51] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2012.
- [52] Chollet, F. (2017), "Xception: Deep learning with depthwise separable convolutions", In "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)", (pp. 1251-1258).
- [53] Bai, W., Oktay, O., Sinclair, M., Suzuki, H., Rajchl, M., Tarroni, G., ... & Rueckert, D. (2018). "Semi-supervised learning for network-based cardiac MR image segmentation", in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, (pp. 253-260).
- [54] Liu, Y., Gadepalli, K., Norouzi, M., Dahl, G. E., Kohlberger, T., Boyko, A., ... & Hipp, J. D. (2017). "Detecting cancer metastases on gigapixel pathology images", *arXiv preprint, arXiv:1703.02442*.
- [55] Prasoon, A., Petersen, K., Igel, C., Lauze, F., Dam, E., & Nielsen, M. (2013). "Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network". In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)* (pp. 246-253).
- [56] He, K., Zhang, X., Ren, S., & Sun, J. (2016). "Deep Residual Learning for Image Recognition". In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 770-778).
- [57] Xie, S., Girshick, R., Dollar, P., Tu, Z., & He, K. (2017). "Aggregated Residual Transformations for Deep Neural Networks." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp. 1492-1500).
- [58] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). "Rethinking the Inception Architecture for Computer Vision." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2818-2826). DOI: <https://doi.org/10.48550/arXiv.1512.00567>
- [59] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & Sanchez, C. I. (2017). "A Survey on Deep Learning in Medical Image Analysis." *Medical Image Analysis*, 42, 60-88. DOI: <https://doi.org/10.1016/j.media.2017.07.005>
- [60] Simonyan, K., & Zisserman, A. (2014). "Very Deep Convolutional Networks for Large-Scale Image Recognition". *arXiv preprint arXiv:1409.1556*.
- [61] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2016). "Learning deep features for discriminative localization". In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp. 2921-2929).
- [62] Hastie, T., Tibshirani, R., & Friedman, J. (2009). "Statistical Learning Elements (2nd ed.)". Springer.
- [63] Freund, Y., & Schapire, R. E. (1997). "A decision-theoretic generalization of on-line learning and an application to boosting: *Journal of Computer and System Sciences*, 55(1), 119-139.
- [64] Breiman, L. (2001). "Random forests". *Machine learning*, 45(1), 5-32.
- [65] Liaw, A., & Wiener, M. (2002). "Classification and regression by randomForest". *R news*, 2(3), 18-22.

- [66] Cortes, C., & Vapnik, V. (1995). "Support-vector networks". Machine learning, 20(3), 273-297.
- [67] Bridle, J. S. (1990). "Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition". In Neurocomputing, Springer. (pp. 227-236).
- [68] "An Open-Source Machine Learning Framework for Everyone," TensorFlow, [Online]. Available: <https://www.tensorflow.org/>.
- [69] F. Chollet et al., "Keras: The Python Deep Learning API," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016
- [70] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in Proceedings of the 32nd International Conference on Machine Learning (ICML), 2015.
- [71] G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv preprint, arXiv:1704.04861, 2017.
- [72] "TensorFlow Addons," TensorFlow. [Online]. Available: <https://www.tensorflow.org/addon> [Accessed: December 6, 2023].
- [73] A. G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", arXiv preprint, arXiv:1704.04861, 2017.
- [74] T. C. He et al., "Bag of Tricks for Image Classification with Convolutional Neural Networks", arXiv preprint, arXiv:1812.01187, 2018
- [75] S. Whang et al., "TFX: A TensorFlow-Based Production-Scale Machine Learning Platform," in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017.
- [76] TensorFlow Extended (TFX), "TFX: An End-to-End Machine Learning Platform," [Online]. Available: <https://www.tensorflow.org/tfx>. [Accessed: December 6, 2023].
- [77] L. Yang et al., "Scalable TensorFlow Learning for Large-Scale Data," in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017
- [78] H. Greenspan et al., "Guest Editorial Deep Learning in Medical Imaging: Overview and Future Promise of an Exciting New Technique," IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1153-1159, 2016
- [79] "Kaggle: Your Machine Learning and Data Science Community," Kaggle: Your Machine Learning and Data Science Community. [Online]. Available: <https://www.kaggle.com/>. [Accessed: Dec. 08, 2023]
- [80] "About Kaggle," Kaggle, [Online]. Available: <https://www.kaggle.com/about>. [Accessed: Dec 8, 2023].
- [81] "Kaggle Competitions," Kaggle Competitions. [Online]. Available: <https://www.kaggle.com/competitions>.

- [82] "Run Data Science & Machine Learning Code Online | Kaggle," Run Data Science & Machine Learning Code Online | Kaggle. [Online]. Available: <https://www.kaggle.com/notebooks>. [Accessed: Jan. 08, 2024].
- [83] "Find Open Datasets and Machine Learning Projects | Kaggle," Find Open Datasets and Machine Learning Projects | Kaggle. [Online]. Available: <https://www.kaggle.com/datasets>. [Accessed: Nov. 11, 2023]
- [84] "scikit-learn: machine learning in Python — scikit-learn 1.3.2 documentation," scikit-learn: machine learning in Python — scikit-learn 1.3.2 documentation. [Online]. Available: <https://scikit-learn.org/stable/>. [Accessed: Nov. 11, 2023]
- [85] harvardgazette, "Making the worms turn — Harvard Gazette," Harvard Gazette, Feb. 02, 2012. [Online]. Available: <https://news.harvard.edu/gazette/story/2012/02/making-the-worms-turn/>
- [86] "Home," OpenCV, Jan. 03, 2024. [Online]. Available: <https://opencv.org/>
- [87] Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools
- [88] Pérez, L., & Granger, B. E. (2007). "IPython: A System for Interactive Scientific Computing". Computing in Science & Engineering, 9(3), 21-29
- [89] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2818-2826.
- [90] Shorten, C., & Khoshgoftaar, T. M. (2019). "A survey on image data augmentation for deep learning". Journal of Big Data, 6(1), 60
- [91] Zheng, S., Zhang, X., Zhang, X., & Huang, K. (2019). "A survey of deep learning-based object detection." arXiv preprint arXiv:1907.09408.
- [92] Simonyan, K., & Zisserman, A. (2014). "Very deep convolutional networks for large-scale image recognition". arXiv preprint arXiv:1409.1556
- [93] Harris, C.R., Millman, K.J., van der Walt, S.J. et al. (2020) "Array programming with NumPy." Nature, 585(7825), 357-362
- [94] Oliphant, T.E. (2006) "A Guide to NumPy." Trelgol Publishing.
- [95] McKinney, W. (2010) "Data Structures for Statistical Computing in Python." Proceedings of the 9th Python in Science Conference, 51-56
- [96] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830
- [97] "Joblib: running Python functions as pipeline jobs — joblib 1.4.dev0 documentation," Joblib: running Python functions as pipeline jobs — joblib 1.4.dev0 documentation. [Online]. Available: <https://joblib.readthedocs.io/en/latest/>

- [98] Dubois, M., Pedregosa, F., Cournapeau, D., & Varoquaux, G. (2018). Scikit-learn: Machine learning without learning the machinery. *Nature Methods*, 17(4), 303-305.
- [99] “Matplotlib — Visualization with Python,” *Matplotlib — Visualization with Python*, Sep. 15, 2023. [Online]. Available: <https://matplotlib.org/>