NISAR AHMAD

BRAIN TUMOR DETECTION USING CONVOLUTIONAL NEURAL NETWORK

MASTER THESIS

2022

NEAR EAST UNIVERSITY

INSTITUTE OF GRADUATE STUDIES

DEPARTMENT OF SOFTWARE ENGINEERING

BRAIN TUMOR DETECTION USING CONVOLUTIONAL NEURAL NETWORK

MASTER THESIS

Nisar AHMAD

Nicosia

August, 2022

NEAR EAST UNIVERSITY

INSTITUTE OF GRADUATE STUDIES

DEPARTMENT OF SOFTWARE ENGINEERING

BRAIN TUMOR DETECTION USING CONVOLUTIONAL NEURAL NETWORK

MASTER THESIS

Nisar AHMAD

Supervisor

Assoc. Prof. Dr. Kamil DİMİLİLER

Nicosia

August, 2022

APPROVAL

We certify that we have read the thesis submitted by Nisar AHMAD titled **"BRAIN TUMOR DETECTION USING CONVOLUTION NEURAL NETWORK"** and that in our combined opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Educational Sciences.

Examining Committee	Name-Surname	Signature
Head of the Committee:	Assoc. Prof. Dr. Boran ŞEKEROĞLU	.....
Committee Member:	Assist. Prof. Dr. Elbrus İMANOV	.....
Supervisor:	Assoc. Prof. Dr. Kamil DİMİLİLER	.....

Approved by the Head of the Department

11../10./2022

Assoc. Prof. Dr. Boran ŞEKEROĞLU
Head of Department

Approved by the Institute of Graduate Studies

...../...../20...

Prof. Dr. Kemal Hüsnü Can Başer
Head of the Institute

DECLARATION

I hereby declare that all information, documents, analysis, and results in this thesis have been collected and presented according to the academic rules and ethical guidelines of the Institute of Graduate Studies, Near East University. I also declare that as required by these rules and conduct, I have fully cited and referenced information and data that are not original to this study.

Nisar AHMAD

10/08/2022

ACKNOWLEDGMENTS

I would like to gratefully and sincerely thank Assoc. Prof. Dr. Kamil Dimililer for his direction, understanding, patience, and most importantly, his supervision during my graduate studies at Near East University. His supervision was paramount in providing a well-rounded experience consistent with my long-term career goal. He encouraged me to not only grow as an experimentalist but an independent thinker.

I would also like to thank my co-supervisor Assoc. Prof. Dr. Boran Şekeroğlu for his support, determination, and supervision gave me all throughout my studies for this thesis.

I would also like to thank NEU Grand library administration members since it provided me with the appropriate environment for conducting my research and writing my thesis. Additionally, I am very grateful to my family, in particular, my father for his help throughout my life.

Nisar AHMD

ABSTRACT

Brain Tumor Detection Using Convolutional Neural Network

Ahmad, Nisar

Master, Department of Software Engineering

August 2022, 87 pages

Brain tumors have a wide range of appearances and share many features with normal tissues, making the extraction of tumor areas from images difficult. Furthermore, when there is a vast quantity of data to be handled manually, it becomes a time-consuming effort. The data augmentation approach, CNN model, and pre-trained model were used as part of this thesis study to build a model for detecting brain tumour from two-dimensional MRI scans. The Kaggle dataset of tumors of various sizes, shapes, and intensities was used in the experimental work.

According to some, deep learning may help overcome the difficulties connected with detecting and treating brain tumors. The deep learning approach, one of the most modern technical methods for classifying and detecting tumors, has long been influential in removing aberrant tumor areas from the brain. Neural Network classification algorithms and Advanced Artificial Intelligence may be successful in the early diagnosis of brain cancers. Convolutional neural network (CNN) models were built using the VGG 16 to find brain tumors, and parameters were selected to train this model to assess the literature solutions. Because of its simplicity, VGG16 is one of the most effective CNN models. In addition, the research found a way to quickly, efficiently, and precisely diagnose brain cancers using MRI. VGG 16 was utilized as the leading network to create feature maps, which were categorized into tumor regions, using the Faster CNN. Performance was evaluated based on the accuracy of the predictions. Using a dataset of 253 MRI brain scans, with 155 revealing tumors, my proposed technique was tested on this dataset. My method may help detect tumors in MR images of the brain. Good accuracy of VGG 16 99.94 percent, CNN 97 percent, RestNet50 45.75 percent, and InceptionV3 48.85 percent.

Keywords: Brain tumor, image processing, artificial networks, convolutional neural network

ÖZET

Evrişimsel Sinir Ağı Kullanarak Beyin Tümörü Tespiti

Ahmad, Nisar

Yüksek Lisans, Yazılım Mühendisliği Bölümü

Ağustos 2022, 87 sayfa

Beyin tümörlerinin çok çeşitli görünümleri vardır ve normal dokularla birçok özelliği paylaşırlar, bu da görüntülerden tümörlü alanların çıkarılmasını zorlaştırır. Ayrıca, elle işlenecek çok büyük miktarda veri olduğunda, zaman alan bir çaba haline gelir. Veri büyütme yaklaşımı, CNN modeli ve önceden eğitilmiş model, iki boyutlu MRI taramalarından beyin tümörünü saptamak için bir model oluşturmak için bu tez çalışmasının bir parçası olarak kullanıldı. Deneysel çalışmada, çeşitli boyut, şekil ve yoğunluktaki tümörlerin Kaggle veri seti kullanıldı.

Bazılarına göre, derin öğrenme, beyin tümörünü tespit etme ve tedavi etme ile ilgili zorlukların üstesinden gelmeye yardımcı olabilir. Tümörleri sınıflandırmak ve tespit etmek için en modern teknik yöntemlerden biri olan derin öğrenme yaklaşımının, beyindeki anormal tümör alanlarını temizlemede etkili olduğu uzun zamandır gösterilmiştir. Sinir Ağı sınıflandırma algoritmaları ve Gelişmiş Yapay Zeka, beyin kanserlerinin erken teşhisinde başarılı olabilir. Beyin tümörünü bulmak için VGG 16 kullanılarak konvolüsyonel sinir ağı (CNN) modelleri oluşturuldu ve literatür çözümlerini değerlendirmek üzere bu modeli eğitmek için parametreler seçildi. Basitliği nedeniyle VGG16, en etkili CNN modellerinden biridir. Ayrıca araştırma, MRI kullanarak beyin kanserlerini hızlı, verimli ve hassas bir şekilde teşhis etmenin bir yolunu buldu. VGG 16, Faster CNN kullanılarak tümör bölgelerine göre sınıflandırılan özellik haritaları oluşturmak için ana ağ olarak kullanıldı. Performans, tahminlerin doğruluğuna göre değerlendirildi. 155'i açığa çıkaran tümörle birlikte 253 MRI beyin taramasından oluşan bir veri kümesini kullanarak, önerdiğim teknik bu veri kümesi üzerinde test edildi. Yöntemim, beynin MR görüntülerindeki tümörleri tespit etmek için faydalı olabilir. VGG 16 yüzde 99,94, CNN yüzde 97, RestNet50 yüzde 45,75 ve InceptionV3 yüzde 48,85'in iyi doğruluğu.

Anahtar Kelimeler: Beyin tümörü, görüntü işleme, yapay ağlar, evrişimli sinir ağı

Table Of Contents

CHAPTER II

Chapter III

Chapter IV

Chapter VI

Chapter VII

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CNN:	Convolutional Neural Network
JPEG:	Joint Photographic Experts Group
PNG:	Portable Network Graphics
ReLU:	Rectified Linear Unit
TP:	True Positive
TN:	True Negative
FP:	False Positive
FN:	False Negative
AI:	Artificial Intelligence
MRI:	Magnetic Resonance Imaging
ML:	Machine Learning
OS:	Operating System
VGG:	Visual Geometry Group
ResNet:	Residual Network
ANN:	Artificial Neural Network
ILSVRC:	ImageNet Large Scale Visual Recognition Challenge
IARC:	International Agency for Research on Cancer
CT:	Computed Tomography

CHAPTER I

INTRODUCTION

Medical image categorization is one of the most challenging and lucrative fields in image processing. Medical imaging methods are exclusively utilized to examine the inside of the patient in order to identify medical conditions. Tumor identification and cancer detection are the most prevalent medical image categorization difficulties. According to statistics, the fatality rate of brain tumors suggests that it is one of the most prevalent and essential cancer forms in the human body. According to the International Agency for Research on Cancer (IARC), about one million individuals are diagnosed with brain tumors, with a rising death rate. It is the second leading cause of cancer-related death among children and young adults under 34. (McKinney, 2004)

Doctors have been using new technology to detect more uncomfortable tumors for patients in recent years. CT (Computed Tomography) and MRI (Medical Reasoning Imaging) scans are two good ways of analysing problems in various parts of the body. MRI-based medical image processing in brain tumor analyses has recently attracted much attention due to the growing need for a speedy and objective examination of enormous entireties of medical data. Because there are so many image data kinds to analyse, it is necessary to utilize specific computational quantification and visualization tools. Consequently, automated brain tumor identification using MRI images would be crucial in this situation, as it will eliminate the need for human data processing.

Brain Anatomy

Cancer can be a disease of the brain resulting in cancerous cells spreading throughout the entirety of the brain. The brain tumor is the most common and, as a result, one of the most dangerous brain tumors, affecting and destroying countless lives worldwide. Each year, more than 100 thousand people worldwide are diagnosed with brain tumors, according to a new cancer study. Despite ongoing efforts to solve the challenges of brain tumors, data indicate that tumor patients have poor results. To tackle this, scientists are using computer vision to understand better malignancies in their early stages and how to treat them with more effective therapies. The most common two

examinations to screen for the presence of tumors and determine their location for consideration of appropriate treatment are MR and CT scans of the brain. These two scans are still widely utilized due to their convenience and capacity to produce high-definition images of diseased tissues. A variety of alternative treatments for tumors are available, including surgery and therapies, for example, radiation therapy including chemotherapy. The extent, type, and evaluation of the tumor visible within MR imaging are all elements that decide which treatment to use. It is also in charge of determining whether cancer has spread to other body areas.

Brain Tumor

A brain tumor arises as abnormal cells grow inside the brain, according to Ilhan et al. (NCI, 2019). Some brain tumors are benign (noncancerous), while others are malignant (cancerous), and yet others are precancerous. Cancerous tumors are divided into two types. (Dimililer K. A., 2016)

Classification of Brain Tumor

Brain tumors can be either benign or malignant. The first form of tumor is a noncancerous benign tumor, while the second type of tumor is a malignant tumor, sometimes known as a cancerous tumor. (Nunn, Silverstein, & Silverstein, 2006)

Benign Tumor

Benign brain tumors are often characterized as a clump of similar cells that do not undergo normal cell division and development, resulting in the formation of a mass of cells that do not exhibit the characteristic look of cancer under a microscope (Nunn, Laura Silverstein 2006).

Benign tumors have the following characteristics:

- It grows slowly, does not infiltrate nearby tissues or can expand to various further organs, and frequently has an edge visible on CT images.
- Benign may be deceiving since they might squeeze brain tissues and other tissues within the skull.

Malignant Tumor

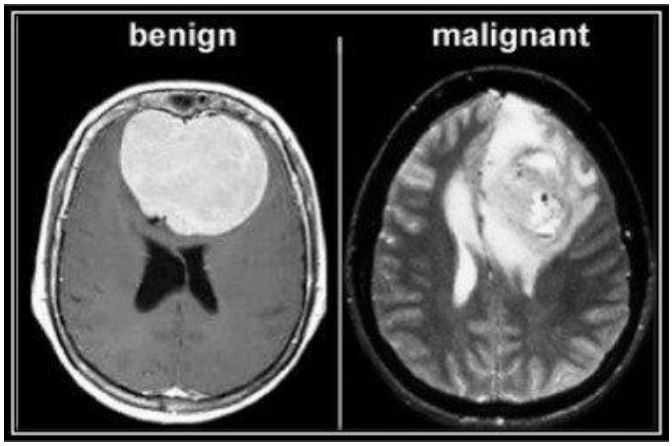
Cancer cells are present in malignant brain tumors, which regularly lack defined borders. These are supposed to be dangerous since they grow speedily and attack adjacent brain tissues (Strayer, Rubin, & Rubin, 2008).

A malignant tumor has the following characteristics:

- Cancer sweeps rapidly to further parts of the brain and spine.
- Malignant brain tumors are classed as grade 3 or 4, although benign or noncancerous tumors are classified as grade 1 or 2.
- These are typically more serious and, in many cases, deadly threats to life.

Figure 1.

Left Side Is A Benign Tumor & Right Side Is A Malignant Tumor (Radiologic and Clinical Outcomes with Special Reference to Tumor Involvement Pattern after Stent Placement for Malignant Bronchial Obstructions , 2022).



Scope

The primary goal of my study is to develop an approach to estimate whether or not medical images include a tumor and identify its medicinal features. Neurosurgeons and healthcare specialists can use the system. Because brain tumor datasets are limited and exceedingly difficult to get, dataset gathering is the most important task when working on medical imaging. I proposed a solution built on mutual classifiers and Convolutional Neural Networks that would support the identification and segmentation of brain cancers without human intervention. Finally, I evaluated all of the trial findings to determine which model performs best in standings of correctness, understanding, and

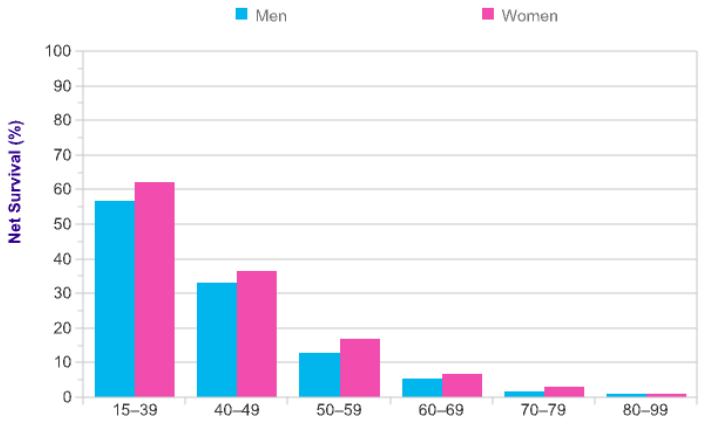
other presentation parameters. Convolutional Neural Networks significantly influenced the revolution of Artificial Intelligence and Deep Learning, and it has become a basic model for image classification processes. (Kamil Dimililer E. T.-E., 2019)

Problem Statement

Tumour diagnosis in medical images is time spent since it is determined by human observation. Based on recent figures on the death rate due to brain tumors, I chose brain tumor identification, part of medical image examination. This field's professionals, including specialized doctors, examine CT scans and MRIs and conclude which therapy is based on. This entire procedure takes time. Because computers do it, automated medical image examination can assist in minimizing the time and energy required and the workload.

Figure 2.

Brain Tumor-Related New Cases And Survival Rates (surv 5yr age brain 0.pdf, 2022)



In the above figure 2, mortality from a brain tumor is more common than death from other malignancies. Early identification of brain tumors can help to minimize mortality. When using digital technology to deliver medical care in remote locations, automated image analysis will substantially aid in providing speedier communication.

If I look at global statistics from the last decade, I can see that in 2020, an estimated 18,094,716 million instances of cancer will be detected. In 2020, the age-standardized rate for all cancers was 190 per 100,000 for men and women combined. Men had a greater rate (206.9% per 100,000) than women (178.1 per 100,000) (Khurram Shehzad, Efficient Brain Tumor Detec- tion Using Image Processing Techniques, 2016).

Thesis Contribution

To fully ascertain and understand the present state of brain tumor segmentation methodologies. I employed Convolutional Neural Networks in my thesis to locate the tumor in the brain. The model's accuracy in locating the tumor was used to assess its overall performance. The solution to this issue was deduced from brain MRI scans. There are two folders in this dataset. One for "yes," one for "no." MRI images of tumors may be found in the "Yes" folder and all the tumor images are 153. No folder contains 98 tumor images that are not visible. We sorted the data into three separate categories. A training set is one of them. The second is the test set. The Validation set is the third category. It is the training set that the model is trained on. Overfitting may occur when a validation set is utilized while in training. The last and most critical set is the test set, which is an unobserved set by the model.

To find tumors, I used two kinds of classifications. Deep learning and a Convolutional Neural Network were used to classify tumors, and the three models' performance was compared to see which one was better.

Thesis Structure

My thesis book has six chapters that are based on my study. Then I shall briefly examine the foundations of these chapters.

Chapter 1 Introduction. The preliminary section of my research thesis. I will briefly summarize the thesis's objective, goal, and contribution in this Chapter. I will also characterize and classify Brain tumors shortly.

Chapter 2 Literature Review. . I will cover some previous related work and their working approach, benefits, and drawbacks.

Chapter 3 Image Processing Techniques. This Chapter contains all the necessary background information for my thesis topic. Image processing approaches will be discussed.

Chapter 4 Artificial Intelligence. This Chapter contains all the necessary information for my thesis topic. Artificial Intelligence, traditional machine learning classifiers, and deep learning will be discussed.

Chapter 5 Proposed Methodology. The methodology that has been proposed in this Chapter discusses my suggested methodology for segmenting the tumor utilizing basic image processing methods and detecting the tumor utilizing Deep learning and Convolutional Neural Networks.

Chapter 6 Evaluation of Experiment Results. I will describe the performance measurements, and create a CNN model from the scratch after implementing it with pre-trained models like VGG16, RestNet50, and InceptionV3, and experimental results in this Chapter.

Chapter 7 Conclusion. Finally, my work will close with the last Chapter. I will discuss the limitations of my work and future directions for this division to work and improve along with further development.

Summary

This Chapter provides a brief overview of brain tumors in addition to their sub-area with a literature review. I have gone over the many forms of brain tumors and their characteristics.

I briefly discussed the motive for my work and the goal of completing it. Finally, I briefly described my thesis contribution and the structure of the thesis.

CHAPTER II

LITERATURE REVIEW

Medical Image Processing attracts experts in image processing and machine learning. Medical image processing has seen several changes in recent years. I combed several recent articles to identify the most beneficial and novel tactics used in recent investigations. In all, I read many study pieces. Because these articles and their working practices are pertinent to my study, I will go through them in depth in this Chapter.

The current improvement of artificial intelligence, in conjunction with the collecting of massive quantities of medical pictures, offers up new opportunities for designing CAD systems for medical applications. Artificial intelligence technologies (including, for example, shallow learning and deep learning) essentially replace the feature extraction and illness categorization procedures in traditional CAD systems. Artificial intelligence algorithms have also been frequently applied in the segmentation of brain X-ray images. Machine learning is a set of mathematical approaches that allow computers to generate data-driven predictions based on massive volumes of data. These approaches have a wide range of uses, many of which are medically customized.

Traditional machine learning techniques for MRI segmentation of normal (e.g., white matter and gray matter) and diseased brain tissues have been developed with considerable effort. Deep learning (DL), a novel machine learning approach, has the potential to overcome the limits of standard machine learning algorithms. Deep learning is the term used to describe multi-layered neural networks (usually more than five layers) that extract a hierarchy of characteristics from the original input pictures. It is a novel and popular machine learning approach that extracts a complicated hierarchy of characteristics from images rather than literal features like typical machine learning algorithms do. They obtain good outcomes and generalization by training a big quantity of data. Alzheimer's disease identification, lesion segmentation (e.g., tumors, white matter lesions, microbleeds), and brain tissue categorization are all uses of deep learning algorithms in brain image analysis. Much of the current research is centered on brain segmentation, with little attention paid to tumor grading. Because of the capabilities of DL-based algorithms to automatically extract features, they have lately acquired increased attention and accuracy in contrast to classic classification techniques for medical imaging. Many lives can be spared if cancer is diagnosed and graded

correctly utilizing rapid and low-cost diagnostic procedures. As a result, developing rapid, non-invasive, and cost-effective diagnostic tools is critical. DL algorithms are presented in this study as a solution for speeding up, automating, and improving the interpretation of different imaging exams.

Shehzad et al. (Khurram Shehzad, Efficient Brain Tumor Detec tion Using Image Processing Techniques, 2016) proposed an algorithm for detecting brain tumors and calculating tumor areas from MR images. The algorithm can identify and remove tumors of any form, size, severity, or location.

This article has the following approaches used.

- Grayscale images are created from MRI images.
- A Gaussian low pass filter blurs the image, then overlaid on the original. Median filters remove noise.
- Dilation and erosion produce a morphological gradient.
- Each pixel's binary is compared to the threshold value.
- Erosion is carried out again for dilution of the image with enlargement over to remove fragments of the tumor back.
- Tumous are identified by paralleling the original and dilated images.
- The part of the tumor is therefore considered

Sankari et al. (Sankari et al. 2017), alongside other researchers, developed a model for brain tumor malignancy diagnosis, which is the most challenging task. The researchers utilized convolutional neural networks to tackle the challenge.

This article has the following approaches used.

- Researchers proposed image denoising, luminance standardization, and error propagation approaches for the pre-processing image problem. The noise in the MRI is removed using the bilateral filter.
- Noise in the MRI is removed using the bilateral filter.
- The image is enhanced, features are extracted using Histogram Equalization, and the images are then classified using CNN.

Base et al. (Vrushali Borase, 1 Jan. 2017)used Artificial Neural Network to spot cancer and categorize tumors utilizing computer-based processes. Borase deployed MRI images during the testing phases.

- This article has the following approaches used.
- High pass filters reduce noise and pre-process.

- After segmentation, the section expanding technique provides a unique region label to each connected collection of pixels with equal grey level values.
- For categorization, an Artificial Neural Network is utilized.
- The tumor is identified by paralleling and associating the original and expanded images.
- Erosion is employed to reduce the remaining noise in the retrieved tumor image.

Morphological procedures using pixel deduction, as well as threshold-based segmentation and image filtering, are employed. Pixel subtraction is utilized for pre-processing phase to separate the skull from the brain image more efficiently. Their devised thresholding approach is used instead of the usual one, followed by filtering. The system might benefit from some improvements, the first of which is the incorrectly identified tumor photos—different classification approaches, such as neuro-fuzzy, considered and used. Additionally, is that the proposed method is not an automated technique. (Ilhan, 2017)

This article has the following approaches used.

- The image found during the initial procedure was deducted after the unique original image, separating the skull from it. Thereby removing any unwanted grey pixel. Finally, remove the cleaned skull image from the original image and get the resulting image.
- They introduced a segmentation threshold approach that divided unique pixel values minus zeros (black pixels) by the total number of unique pixel values.
- After segmentation, the median filter is employed.
- The proposed threshold technique has a 96 percent accuracy.

In hunnur et al. (B. Devkota A. A., 7-8 December 2017) authors suggested a technique for detecting brain cancers based on morphological operations, including a Thresholding methodology. It also determines the tumor area and displays the tumor stage.

This article has the following approaches used.

- The images are turned into grayscale images using MRI images of the brain as input.

- Any noise in the transformed images is removed using a high pass filter.
- The impulse noise is removed using the median filter.
- Thresholding selects a threshold value to extract the object from the backdrop.
- The tumor location is identified, and the image is reduced to remove any unnecessary details.
- The tumor area is finally calculated, and the tumor patient's stage is displayed.

Mustaqeem et al. (Anam Mustaqeem, 2012) use a cross-segmentation method that combines Watershed and Thresholding-based segmentation techniques (Mustaqeem et al.,2012). The value of the scanned images is first improved, and then morphological procedures and their proposed hybrid segmentation are used to detect the tumor. The proposed system is simple to implement and thus manage.

This article has the following approaches used.

- MRI images are pixel-based two-dimensional matrices.
- Salt-and-pepper noise iedemaced using a Gaussian low-pass filter and averaging filters.
- The value of each pixel in the filter is swapped with its neighbours' values.
- A Gaussian high pass filter is employed to improve the image daries.
- The grayscale image is converted to a binary image representation via threshold segmentation.
- Watershed Segmentation groups pixels in an image based on their intensity levels.
- To separate the tumor section of the image, morphological operations are applied to the transformed binary image.

K-means and the SVM approach were employed by Joseph et al. for segmentation and keeping the pattern for future usage, respectively. They discovered a link between the skull masking approach and SVM. They use skull masking with K-means segmentation and SVM algorithms to improve the output. Maintaining the tumor pattern was another demanding computer problem that they achieved and also backed with the expected concept.

This article has the following approaches used.

- On the MRI images, they use K-means segmentation with pre-processing.
- The image has been converted to grayscale. On the brain MR image, a 3*3 median filter is used to reduce noise, followed by a high pass filter to detect edges.
- Skull Masking is used to locate the area of interest.
- The unsupervised Support Vector Machine (SVM) is used to create and maintain the pattern for future use.

Devkota et al. (B. Devkota A. A., 2018) proposed a complete segmentation process established on Mathematical Morphological Operations, and a spatial FCM method reduces calculation time. It has a 92 percent accuracy in detecting cancer and an 86.6 percent accuracy in classifying it. However, the proposed method has yet to be evaluated.

Yantao et al. (Yantao, 2016) A segmentation technique utilizes histograms. The brain tumor segmentation challenge is a three-class classification problem employing the FLAIR and T1 modalities (tumor comprises necrosis, tumor, and oedema in addition to normal tissue). A region-based active contour model using the FLAIR modality was used to identify aberrant areas. With a Dice coefficient of 73.6 percent and a sensitivity of 90.3 percent, the k-means method was applied to discriminate edema and tumor tissues in aberrant areas utilizing contrast enhancement T1 imaging.

Dina et al. (Dina Aboul Dahab, 2012.)recommended a Learning Vector Quantization model based on the Probabilistic Neural Network model. The model was tested on 64 MRI images, with 18 of them provided as per the test set, and the rest kept as the training set. The images were round with the Gaussian filter. The modified PNN method reduced the processing time by 79 percent.

Rajendran et al (A. Rajendran, 2011)Achieved 95.3% and 82.1% ASM and Jaccard Index using an Enhanced Probabilistic Fuzzy C-Means model with morphological methods based on a region-based Fuzzy Clustering and deformable model.

Summary

I evaluated a variety of research articles and discussed their methodologies. In this chapter, the literature study reveals that several research papers on brain tumor classification and detection have been published.

CHAPTER III

IMAGE PROCESSING TECHNIQUES

Medial Image

Imagery and visualization for examination of various human body tissues or organs for medical analysis and therapy, including witnessing illness growth, is referred to as medical imaging. For example, Radiology, Optical imaging, and image-guided intervention are all examples of imaging methods and Systems. (Dimililer K. &., 2021)

Medical Image Analysis Prominence

Digital image dispensation is an advanced technology that processes images and videos. Image processing can be used in any perspective, ideally for personal or security personal details. X-ray is commonly the most critical use of Digital Image Processing nowadays. Before the invention of x-rays, examining human bone in the body was extremely difficult since the doctor had to create incisions into the patient's skin and tissue to investigate/determine if any bones were fractured or broken.

Modern advances in imaging studies have demonstrated how technological advancements can progress and modify a wide range of clinical practices. Wilhelm Röntgen discovered the X-ray in 1895, and now we have MRI and CT. As time moves onward, image technology will continue to evolve. However, today's systems are becoming away from medical imaging's focus on image creation and collection and toward image post-processing and data management. The necessity expedited ensures the most efficient use of data currently motivates this.

Machine Learning in Medical Image Analysis

Artificial intelligence is and can be utilized within various radiological imaging techniques, including but not limited to risk assessment, defect detection, diagnosis, different prognosis, therapy response, and rapid disease detection, because of the improvements in both imaging and computing. Machine learning methods, for example, computer algorithms that "learn" certain tasks given specific input data) can use computer-extracted (radio-mic) characteristics as input (Kamil Dimililer E. T., 2022). Several radio-mic features can be combined into a variable, like a tumor identity, which can then be linked to disease status probability using machine learning

techniques (Shasidhar, 2011). Over the years, several reviews of machine learning have been published, including those currently employed as tutorials, as new investigators are brought into the arena. (Avanzo, 2017)

Image Processing for Brain Tumor Segmentation

The brain image is ambitious. Two of the most challenging processes are segmenting the region of interest like an entity and segmenting the tumor from an MRI. Brain tumor segmentation from MRI is the most complex medical image processing task since it requires enough data. Additionally, tumors that have soft tissue borders may be imprecise. Therefore, accurately segmenting tumors from the human brain is a complex process.

Image processing helps enhance MRI images and feature extraction and categorization. Image processing for brain tumor segmentation includes stages, for example, but is not limited to head dissemination, pre-processing, and tumor contouring, among others.

Skull Stripping

Skull removal is the most effective and crucial stage for identifying a brain tumor and mapping its characteristics. It is the way to eliminate wholly non-brain tissues attained from MRI imageries. In addition, various cerebral matters, including fat, skin, and skull, can be removed from the images obtained using this procedure. Some prominent skull removal approaches are image contour, segmentation-based skull stripping, morphological operations, and histogram analysis with a threshold value (Clinic, 2022).

Pre-processing

Various types of noise image processing can taint medical images. It's critically essential to find accurate imageries to enable authentic interpretations of the application used (What is the difference between MRI scans, 2022). Filtering and morphological operations are the main steps in Pre-processing.

Filtering: Smoothing, sharpening, removing noise, and edge detection are all instances where image filtering is proper. Filters are described by a kernel, where grain is a minor array linked to pixels and their neighbours inside an image. By a

kernel, where a kernel is a little array that is applied to pixels and their neighborss inside an image. Convolution is a technique for using filters to images in the spatial or frequency domain. According to their outputs, spatial domain filters may be divided into smoothing and sharpening filters.

Smoothening Filter. Noise reduction and blurring procedures are performed using Spatial Smoothing filters. Box, Gaussian blur, Median blur, and bilateral filters are included.

Box Filtering. Box filtering is a technique for images that use the average of the pixels in the surrounding area.

Gaussian Blur Filtering. Gaussian Blur Filtering is a 2D convolution operator for eliminating information and noise from images. Deviation of the Gaussian determines the degree of smoothing. It's comparable to the mean filter in this aspect, although it employs a considerably different kernel to mimic the profile of a Gaussian hump.

Median Blur Filter. It is typically employed to decrease image noise. Instead of just changing the pixel value with the mean of nearby pixel values, the medium of the values is utilized alternatively.

Bilateral Filter. This non-linear image smoothing filter conserves edges while dropping noise. Sharp edges are preserved.

Sharpening Filter

Seek to ascertain specific fine details. Sharpening filters use spatial distinction in effect. It eliminates image blur and highlights edges. It contains Laplacian, Sobel, and different filters.

Laplacian Filters. These are derivative filters utilized in images to identify areas of fast change (edges).

Sobel Filters. These are usually utilized for edge recognition.

Difference filters. Difference filters increase their particulars within the path relevant to the particular mask.

Morphological Operations

Morphology is the study of pixel shapes. Using morphological techniques removes the goal of eliminating image structure flaws (Ravi Srisha, December 2013).

Dilation and erosion are critical morphological operations. Dilation adds pixels to object borders in an image, whereas erosion removes pixels from object borders. Islands and little things are removed via morphological erosion, leaving only substantial objects. Opening, closing, hit and miss transforms, and other morphological procedures are also included.

$$I \oplus H = \{(p+q) \mid for\ every\ p \in I,\ q \in H\} \quad (1)$$

The original image is I, and the structural element is H.

Erosion is the polar opposite of dilation. The output pixel's relevance is determined by the sum of its maximum relevant neighboring pixels' values. A pixel is set to 1 if any neighbors are similar to 1. Morphological dilatation increases visibility and fills gaps.

$$I \ominus H = \{p \in Z^2 \mid (p+q) \in I,\ forever\ q \in H\} \quad (2)$$

The original image is I, and the structural element is H.

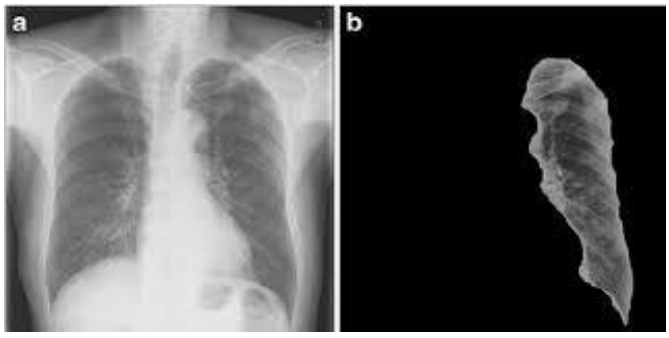
Morphological operations are required to eliminate soft tissue borders and affect tumor dissection in the brain tumor portion.

Segmentation

This is the procedure of splitting an image into distinct sections and separating objects from their surroundings. The ability to accurately segment things of attention shown in an image makes future examination of these substances much easier. Segmentation techniques range from the edge, thresholding, region, and clustering-based to various other methods.

Figure 3.

The Segmentation Of Objects From Their Background (Overflow, 2022).



Fuzzy C-Means (FCM) Clustering

The Fuzzy C-Means method is a data collection technique that allows a single set of information to correspond to two or more clusters. Pattern recognition usually uses this technique. The Fuzzy C-means clustering (FCM) Algorithm is the most extensively utilized clustering technique. Clustering is also known as cluster analysis and is the process of gathering data points into clusters, enabling items in the same collection to be as similar as possible. In contrast, those in additional sets are as distinct as feasible. Comparison measurements are utilized to recognize clusters. Distance, connectivity, and intensity are samples of comparison measurements (Ismail Saied, 2022). It minimizes the objective function: Where m is any real integer higher than 1, Ui j is just the degree of membership of xi in cluster j, xi is the ithe of dimensional measured data, and cj is the d-dimension cluster centre.

Figure 4.

Figure Show FCM Segmented Image (Camilus S. , 2022).

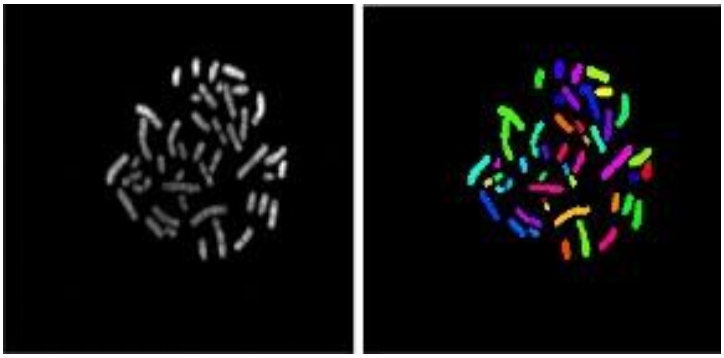
K-Means Clustering:

Unsupervised learning is a form of K-Means clustering. K-means clustering detects unidentified data groups. This approach sees K groups in data (Rohini Paul Joseph, March, 2014). This approach assigns data points to K groups based on their characteristics. K-Means Clustering labels new and training data with cluster centers of K clusters (each data point has a set). Unlabelled data may be clustered using K-Means. A cluster is a group of similar data items. The centroid represents the cluster's center. Each cluster's centroid features define the resulting groupings. The centroid feature

weights may qualitatively describe each cluster's group. The segmentation was performed using the K-means clustering approach, as shown in Figure 5.

Figure 5.

Segmentation By K-Means Clustering (Camilus S. , 2009)



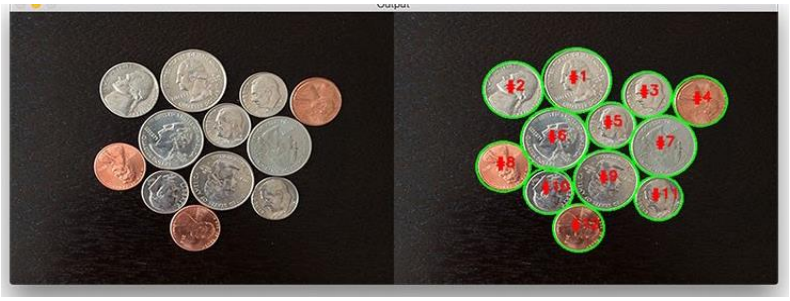
Watershed Segmentation:

This traditional image segmentation method separates various articles in an image. This approach is used to segment an image after two areas of interest are close, for example: when their edges contact. This approach considers the image as a topographic map, with the strength of each pixel indicating the height. Dark spots, for example, are automatically seen as 'lower' in height and might represent troughs. Watershed is a grey-scale image modification.

On the other hand, bright areas may be classified as 'upper,' and could serve as hills or a mountain range. It is one of the most effective ways of grouping pixels in an image according to their intensities. Pixels with the same intensity level are clustered together. For perspective, I can isolate a particular area of the image (Anam Mustaqeem, 2012).

Figure 6.

Demonstrates How Watershed Segmentation Affects The Environment (Rosebrock A. , 2022).



Summary

In this section of my thesis, I provide a concise explanation of all of the issues that are pertinent to the work that I am doing in my view and their working procedures and other relevant information. To get started, I tried explaining some fundamental image processing concepts.

CHAPTER IV

ARTIFICIAL INTELLIGENCE

Machine learning

In artificial intelligence, a machine's capacity to simulate intelligent anthropoid behavior is broadly characterized. Simulated intelligence structures can carry out difficult tasks in a manner analogous to the way individuals solve problems. Machine Learning is a highly sophisticated technological application used in various fields of research.

Agreeing with Boris Katz, a foremost research scientist and the leader of InfoLab Group at CSAIL, AI's ultimate goal is to construct various electronic computer systems entailing the demonstration of "intelligent behavior" to ourselves. This refers to apparatuses that can detect visual scenes and read ordinary linguistic writings while acting within a physical environment.

Machine learning can be utilized by using AI. Arthur Samuel, an AI pioneer, characterized it within the decade of the 50s as "the branch of research that provides computers the capability to learn after expressly being programmed."

Machine learning is divided into three subsections:

Supervised Machine Learning

Supervised machine learning models are constructed and designed to allow for future progress over time to utilize labeled data sets. For example, algorithms may be taught using photographs of animals and other objects already recognized by ourselves and others. In addition, any device would be able to adjust to acknowledge/recognize individual images of animals. Today, the most popular kind is supervised machine learning.

Unsupervised Machine Learning

Programs search for patterns contained within unlabelled data in this learning process. Unsupervised machine learning will detect patterns in addition to trends that humans are not looking for. An unsupervised learning algorithm, for example, may look

through a wide range of online sales data and identify different types of consumers' purchases that are distinct.

Reinforcement machine learning

By rewarding wise decisions, reinforcement learning may teach computer programs to play video games or teach autonomous cars to drive.

Convolutional neural network

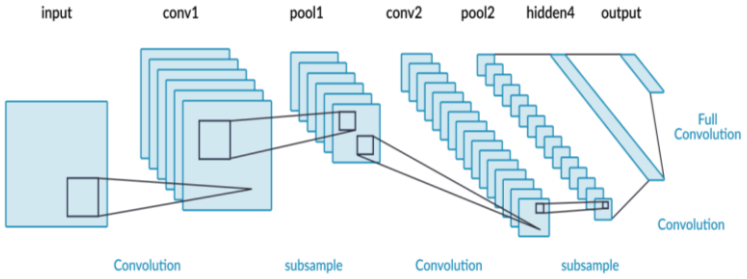
Convolutional neural networks, artificial neural networks created expressly to analyse pixel data, are used in this image processing application. Convolution is a linear operation used in mathematics by this network to interpret pixel data. A convolutional network is a neural network that employs convolution rather than standard matrix multiplication in at least one layer.

As neurons do in the human brain, neural networks are hardware or software systems that model their behaviour. In humans and other animals, the frontal lobe is the region responsible for processing visual input, so CNN's "neurons" are structured more like that. The whole visual field is covered by the layers of neurons, resolving the problem with traditional neural networks' fragmented image processing.

Multi-layered perceptron is used in CNNs, which are optimized for low processing demands. A CNN is illustrated in Figure 1 by its input, output, and hidden layers. Layers such as convolutions, groupings, and related layers are the most common. There are many types of coatings in convolutional neural networks, such as layers that are fully connected, layers that are normalizing, and layers that are pooling. However, a hidden layer can also incorporate multiple convolutional neural networks. Further examples are batch-levelling layers, leak layers, and ReLU layers. By eliminating restrictions and improving image processing efficiency, a system for natural language processing and image processing limits can be made more efficient and easier to train (Leon A. Gatys, 2015).

Figure 7.

Convolutional Neural Network Architecture (Hagyeong Lee, 30, Nov 2019)



Convolutional Layer

Convolutional layers, often known as Conv layers, are a group of trainable filters that make up a convolutional network. Filters have lower heights and weights than their inputs. The neurons' activation maps are determined by combining each filter's input size. the spatial locations between the information and the filter where the point products were calculated. The width, height, and input dimensions are used to filter the filter. The activation maps of each filter are stacked along the depth dimension to define the output size of the convolutional layer. Each neuron in the map is only linked to a very tiny portion of the input size because the width and height of each filter are intended to be less than the input; in other words, the projected field size of each neuron is small and equal to the size of the filter. The anatomy of the animal's visual cortex, where the receptive cell fields are narrow, controls local contact. A local convolutional connection allows the convolutional layer of the network to identify which filters are sensitive to a particular input area based on the spatial correlation of the inputs (for example, the pixels in the input image are closer together).

Near neighbouring pixels (in contrast to pixels further away). If the filter and input are changed to create the activation map, all local placements will have the same filter settings. Weight distribution reduces the number of variables needed for good generalization, learning, and expression. (Weston, 2008).

Pooling Layer

The pooling layer's goal is to progressively reduce the representation's altitudinal size and the network's computational and parameter needs.

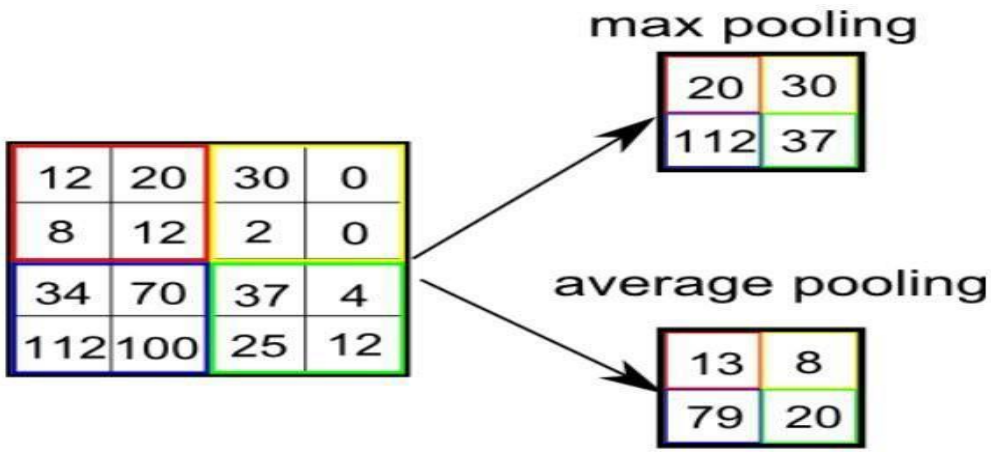
Most pooling methods often apply the regular or maximal function to the subset under consideration.

It is possible to apply aggregations in a variety of forms. The average pool and the full pool are the most commonly used. It is determined by the kind of aggregation

used whether the nucleus is converted into inputs that serve as inputs for the middle collection to calculate the average parameters. As shown in Fig., the Max pool determines each parameter's maximum value. The input sample is acquired due to layer grouping, which minimizes the number of calculations required (P, Mar 5th - 7th, 2019.).

Figure 8.

The Operation Of The Maximum And Average Pooling (Jan Egger, 2021).



Rectified Linear Unit Layer:

A pooling layer can reduce the network's spatial size and computational requirements by gradually reducing the size of the representation. Typically, when pooling data, a maximum or average function is applied to the subset in question.

Aggregates come in a wide variety of forms that might be used. The two most often utilized pools are the average pool and the maximum pool. The average parameters are calculated as outputs depending on the aggregation we utilize, and the nucleus is turned into inputs that create outputs in the average pool. As shown in Fig.2, Using the Max pool, maximum values are selected for each kernel parameter. It has been found that layer grouping results in input sampling, which reduces the number of computations required (Brownlee, 2020).

Dropout Layer

The dropout layer regularization method avoids complex co-adaptations on training data, which reduces overfitting in neural networks. It is a very efficient

technique for averaging models in neural networks. Units that depart a hidden and visible neural network are referred to as "dropouts." Leakage layers function by aiding in feature adaptation by being disconnected during training. Co-pairing happens when many neurons are trained in the same attributes after network training with a small number of images.

As a consequence, a network that solely contains functional components is constructed using training images. The phrase "leakage layer" refers to the fact that during the designated training period, atomization creates leaky modules that have no effect on the network. The ability of the units to be separated is specified by leakage layers. This unit split occurs only during training, producing a number of weakest networks. As a result, the network is randomly generated, and on average, a decent model is produced (Nitish Srivastava, 2022).

Batch Normalization Layer

Artificial neural networks may be made quicker, more efficient, and more stable by using batch normalization. This function normalizes the input layer by changing and scaling the activations. While batch normalization's effects are obvious, its basic reasons for success are still up for discussion. It was believed that it may aid with the internal covariate shift problem. The network's learning rate is affected by the initialization of parameters and variations in the input distributions to each layer. Batch normalization enhances performance by smoothing the objective function as opposed to eliminating internal covariate shift, according to recent study. Other research has demonstrated that batch normalization results in length-direction decoupling, speeds up neural networks, and accomplishes this goal. (Ergey Ioffe, 2015).

Fully Connected Layer

Fully connected (FC) layers are required for CNNs, which are particularly effective for detecting and classifying images for computer vision. The initial phases in the CNN process are convolution and pooling, which partition the image into features and evaluate each one independently. This process's result, the decision to categorize the data, is controlled by a fully linked neural network structure. (CS231n, 2022) (Bouchard, 2020).

SoftMax

The SoftMax loss is frequently used in CNNs due to its simple and conditional interpretation. We may use it for multi-class categorization if the classes are mutually exclusive. However, while dealing with binary classification issues, we employ the (usual) Logistic Regression model. (Developers, 2022).

Backpropagation

Backpropagation algorithms use a gradient descent method based on the chain rule to train artificial neural networks (ANNs) effectively. Backpropagation employs the chain rule to determine the contributions of each parameter to the ultimate loss value before traveling backward from the top layers to the bottom layers.

For many years to come, networks will be built using modern frameworks like Tensor Flow that provide symbolic differentiation. As a result, given a series of operations with a known derivative, they could use the chain rule to build a gradient function that transforms network parameter values into gradient values. When you have access to this gradient function, you may call it to conduct the backward pass (Heaton, 2017).

Adam optimization

This approach was created to employ adjustable learning rates for training deep neural networks. A stochastic gradient descent procedure may be substituted for iteratively updating network weights based on training data. A random regression ratio isn't Adam's operation.

Random gradient origins maintain a constant learning rate (called alpha) throughout the training process.

It is updated independently as learning advances to maintain track of the learning rate per network weight (parameter) (Kingma, 2014).

Deep Learning

A subfield of machine learning consists mostly of neural networks with more than three layers. To replicate the function of the human brain, these neural networks can "learn" from huge amounts of data.

Deep Learning excels in pattern recognition but needs a vast amount of data. Since it is executed using three or more layers of artificial neural networks, deep learning excels at detecting objects in images. Each layer is accountable for removing one or more characteristics from the image.

Deep Neural Networks with Transfer Learning

A deep learning framework for medical concerns must consider the amount and quality of data. Sample collection may be prohibitively costly or even impossible in certain situations. Transfer learning has been presented to allow deep learning architectures to be used in short dataset issues. Instead of gathering information, it uses a model that has already "seen" a wide range of complex and sophisticated data. Models that have been trained on a large number of datasets are assumed to be able to tackle a broad range of issues. It is possible to repurpose the pre-built convolutional neural networks, and only the top layers must be retrained. They are used extensively in natural language processing, signal processing, and computer vision. CNN's unique qualities, such as incremental feature extraction in consecutive layers, make it feasible to reuse a pre-trained model for a new job without retraining the whole network. Typically, the lower levels identify common patterns like lines and edges, the intermediate layers locate portions of things, and the final layers learn to recognize whole objects in various forms and locations. The primary idea is to retrain just the last few layers of a pre-trained model using fresh pictures. Formal transfer learning is defined in terms of domain and task. One of three methods is needed to retrain a pre-trained model.

Feature extraction (freezing CNN model base)

Pre-trained classifiers may be used to train new classifiers on top of them. Convolution layers' weights remain fixed, and only the last fully linked layer is trained.

Fine-tuning (training also some convolution layers)

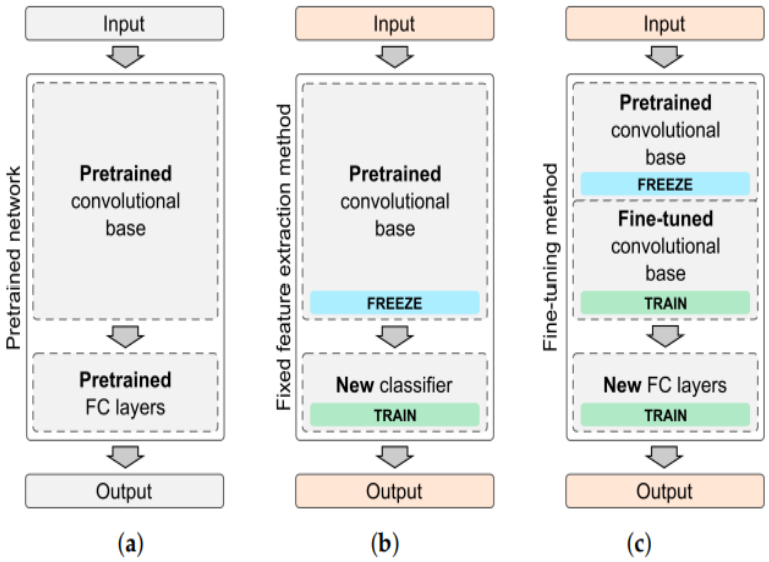
Retraining one or more convolution layers in addition to a fully linked classifier for fine-tuning (training additionally some convolution layers). As a starting point, the original weights of the convolution layer are employed. Only a new issue can be solved by unlocking convolution layers.

Train The Entire Model

It's best to employ the pre-trained model's implemented architecture and run it through its paces on your dataset. Start using values from a pre-trained model instead of random weights.

Figure 9.

Three Learning Transfer Scenarios (Andrzej Brodzicki, 2020)



The performance of deep learning models grows as the number of data increases. As a result, huge datasets are often not easily accessible for training most designs. Open-source models with pre-trained weights are available for the most popular models. The ImageNet dataset is used to train them, which contains 14,197,122 images spanning 21,841 different real-life categories. Competitors throughout the globe utilize this dataset to find better algorithms every year.

Pre-Trained Deep Learning Architectures

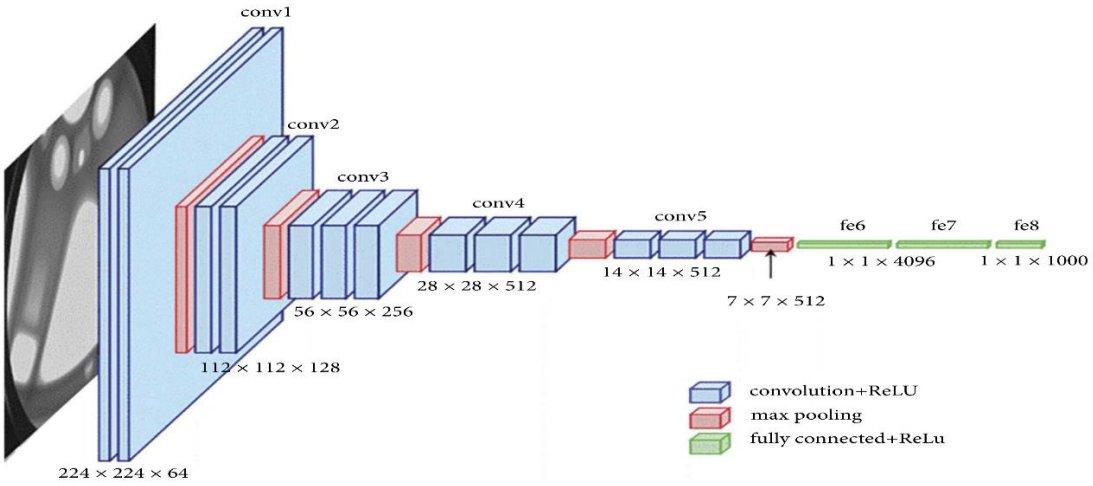
In recent years, pre-trained neural networks have dominated the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Advanced designs, including ResNet50, Xception, VGG-19, and DenseNet121, have been among the most effective in recent years, following the well-known VGG-19 from 2012. The Keras library provides these models, pre-trained weights, and handy functions.

VGG16

VGG16 uses the ImageNet dataset to train a neural network to classify raw images. Having a fixed input size of 224 224, Simonyan and Zisser-man introduced the VGG16 architecture. An array of convolutional (convs) layers is used to process the images, each containing an extremely small receptive filter of size 3. In addition, convolution filters are employed when linear changes in input channels are required (followed by non-linear changes). Five maximum-pooling layers work together to retain the spatial resolution after convolution. With stride 2, a stack of convs layers, three fully connected (FC) layers—4096 channels in each of the first two FC levels and 1000 channels in the third FC layer—and a 22-pixel window, max pooling is performed. The SoftMax layer is the last layer of the architecture (Karen Simonyan, 2014).

Figure 10.

Vgg-16 Network Architecture Diagram With Layers. (Chetana Srinivas, 2022)

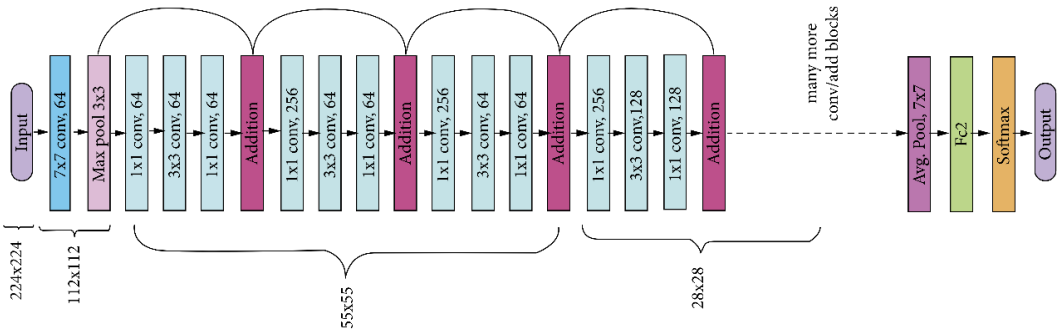


ResNet

The first deep learning network to be created in 2015, known as ResNet, was able to train hundreds or thousands of layers without experiencing the vanishing gradient issue. ResNet is based on residual learning, which sets it apart from other deep learning algorithms. In residual learning, the network's inputs from earlier layers are made more deeply available, increasing the network's accessibility to those inputs. In addition to connecting the remaining networks, connection shortcuts provide the building blocks for them. (He, Zhang, Ren, & Sun, 2016).

Figure 11.

ResNet-50 Model Architecture (Chetana Srinivas, 2022)



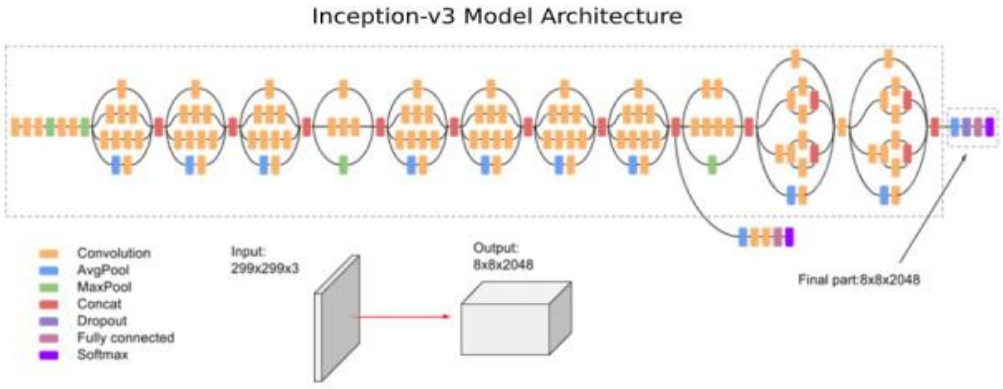
Inception

Inception v3 has attained more than 78.1 percent accuracy on the ImageNet dataset. It has been trained as a convolutional neural network using more than a million photos from the ImageNet collection. Keyboards, mice, pens, and various animals are among the 1000 things it can classify. It consists of 48 separate algorithms and has 48 layers. As a consequence, numerous photographs on the network include elaborate feature representations. The input pictures for the network are 299 by 299 pixels in size.

The model's symmetric and asymmetric building elements include convolutions, average pooling, maximum pooling, concatenations, dropouts, and linked layers. Throughout the whole model, the batch norm is applied to activation inputs. It is possible to calculate loss using SoftMax (Gao Huang, 2017).

Figure 12.

Inception-v3 Model Architecture (Chetana Srinivas, 2022)



Data Processing

Laplacian as a focus measure

Choosing the image that is the most focused when categorizing cells is crucial. It is possible to differentiate a normal cell from a tumor cell if the image's focus is inadequate. The Laplacian is an example of a focus gauge. In plasma mass, the second-degree derivative is known as high-frequency scrolling, which might indicate the image's sharp edges. The second-order derivative displays the rate of change. Quicker changes suggest a sharper edge. Laplacian can be expressed using a mask. It is possible to use asymmetric contrast as a posterior measure of picture focus. 3D object recovery can be improved by using the generalized Laplacian to evaluate focus for low-contrast images, which suggests certain, sharp edges implying noise (Subbarao, 1995).

Normalization

Data normalization is a method for preparing data for machine learning. Normalization involves converting numeric columns to a common scale to reduce repetition and maintain disparities in value ranges. Machine learning can be applied to any dataset.

Normalization is not necessary. Only when the ranges of characteristics vary is it essential (Freedman, Pisani, & Purves, 2007-02-20).

Augmentation

Augmentation reduces over-display by artificially enlarging a training dataset to create altered copies of each instance. Various methods can translate the conversion process, such as mirroring and rotating. A poster's original image remains unchanged when the image is increased. Adding a different version of the original image will be treated as a separate image with the same label according to the grid (Rosebrock A. , 8 July 2019).

Biomedical data augmentation

In comparison to many other non-medical domains, this one present unique obstacle, which are briefly discussed in various publications involving medical imaging and neural networks. The primary issue is a shortage of data, which leads to extremely tiny training sets and overfitting**.**

It is mentioned that the majority of academics use data augmentation to address the issue of tiny data sets. Typically, this is accomplished by the use of geometric transformations like rotation, scaling, translation, and flipping. The authors also discuss this and propose a technique called Generative Adversial Networks. This can be explained as a network that creates fictitious images and another that assesses them to determine if they are excellent or terrible. The article demonstrates how this strategy might enhance performance when utilizing a short training set. (M. Frid-Adar, 2018)

On the training samples, rotation, shift, and deformation augmentations are applied. Since deformations are the most frequent alterations in tissue, the authors contend that elastic deformation augmentations are particularly significant in biomedical segmentation tasks. (O. Ronneberger, 2015)

Similar data augmentations are used as in the earlier papers in another article by Asperti et al. In their experiments, rotation, translation, shearing, and scaling are used. Like in many other articles regarding neural network applications in medicine, they draw the conclusion that data augmentation improved performance while working with tiny data sets. (J. Akeret, 2017)

Performance Estimation

The next step in the classification challenge is to evaluate the performance of the classification model that has been learned. In the test set, the model is rated according to how well it identifies the samples. Using a confusion matrix [26], you can report whether samples were correctly or incorrectly classified.

It consists of rows representing real classes and columns representing anticipated classes. The confusion matrix should now be described for each element (Saito, 2017).

Table 1.

Confusion Matrix Table for Performance Estimation

	Predicted: NO (Negative)	Predicted: YES (Positive)
Actual: NO	TN	FP
Actual: YES	FN	TP

A true positive sample contains positive results and is classified as such. False negatives are positive samples categorized as negative (false positives).

False Positives: Negative samples that appear positive but are, in fact, negative.

Negative samples (true negatives) are those that are categorized as such.

The confusion matrix can be used to determine how well a model performs, as well as to assess a variety of different performance metrics. One of the most basic and regularly used performance measures is accuracy.

This ratio is calculated by dividing the total number of samples in the test set by the number of instances that had been correctly identified. In the confusion matrix, the diagonal elements represent the number of correctly identified cases.

Evaluation Metrics

There are various metrics to evaluate the performance of the systems developed for the image segmentation tasks ((Al-antari, 2018)). In this study, six metrics are used: accuracy, sensitivity, specificity, precision, dice score, and Jaccard index to evaluate the performance of the proposed system.

The accuracy represents the overall success rate of the system and is defined as in Equation 3.

$$Acurracy = \frac{\mathrm{TP} + \mathrm{FN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{TP} + \mathrm{FN}} * 100 \quad (3)$$

The sensitivity (recall) measures the system's ability to segment tumorous pixels and is defined in Equation 4.

$$Sensitivity = \frac{\mathrm{TP} + \mathrm{FN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{TP} + \mathrm{FN}} * 100 \quad (4)$$

Contrary to the sensitivity, specificity shows the ability of the system to segment the non-tumorous pixels. The equation of specificity is given in Equation 5.

$$Specificity = \frac{\mathrm{TP} + \mathrm{FN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{TP} + \mathrm{FN}} * 100 \quad (5)$$

The precision represents the ratio of the correctly classified tumorous pixels over all pixels that are classified as tumorous by the system. This led researchers to analyze the success of the system in producing true positives over all pixels assigned as tumorous. It is defined in Equation 6.

$$Precision = \frac{\mathrm{TP} + \mathrm{FN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{TP} + \mathrm{FN}} \quad (6)$$

It is also possible to express the error rate as follows:

It is feasible to evaluate the performance of a model merely based on accuracy or error rate. However, these measures do not reveal the model's accuracy in highly skewed datasets. An imbalanced dataset in this instance shows that the majority of the samples fall into one of the classes. Assume you have a dataset with 70% of the examples in the positive category and 30% in the negative category. A classifier that successfully identifies all positive models achieves an accuracy of 98%, making it seem to be an excellent classifier even if it fails to identify even one negative example correctly.

Summary

The section briefly describes the issues that are applicable and relevant to my Thesis, as well as their functional procedures, benefits, drawbacks, and so on. I began by attempting to convey the concepts of a virtual machine and deep learning, followed by an explanation of simple CNN architecture and its many hyper-parameters.

CHAPTER V

WORKING METHODOLOGY

Introduction

We employed Convolutional Neural Networks in our thesis to locate the tumor in the brain. The model's accuracy in locating the tumor was used to assess its overall performance. The solution to this issue was deduced from brain MRI scans. There are two folders in this dataset. One for "yes," one for "no." MRI images of tumors may be found in the "Yes" folder, and all the tumor images are 153. No folder contains 98 tumor images that are not visible. We sorted the data into three separate categories. A training set is one of them. The second is the test set. The Validation set is the third category. It is the training set that the model is trained on. Overfitting may occur when a validation set is utilized while in training. The last and most critical set is the test set, which is an unobserved set by the model.

Dataset

A collection of brain MRI scan images was gathered for this study. Raw MRI images are typically measured in terms of pixel values, and around 256 distinct dimensions (width x height) are available. Images of MRI brain scans were picked from the Kaggle dataset for this project. JPEG (Joint Photographic Experts Group) images store the data. Based on the presence or absence of a tumor in an MRI brain image, an image database is divided into two groups: "Yes" and "No." Around 158 brain MRI images show benign tumors, while the remaining 98 images depict malignant growths. Our work's data are usually divided according to whether they are used for learning, testing, or validating purposes. Table 1 shows the categories of the MRI collection.

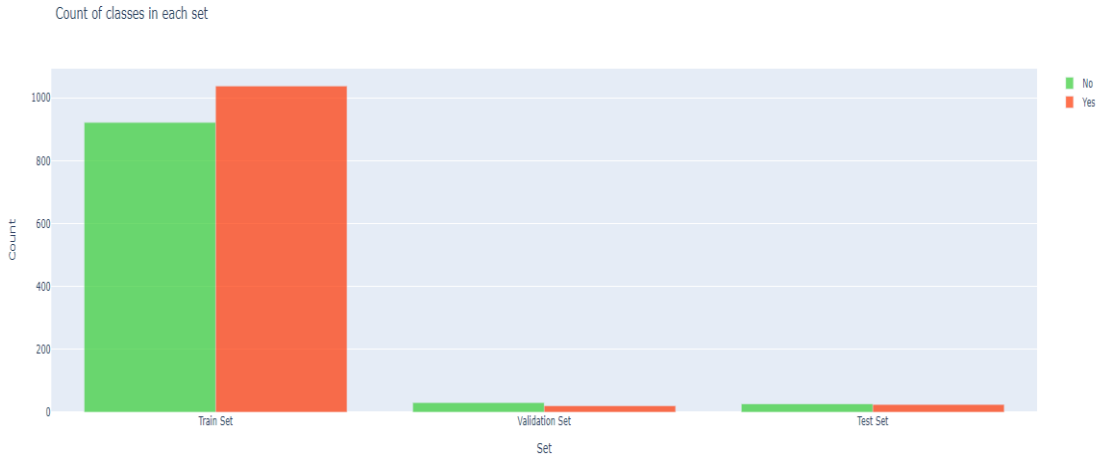
Table 2.

Splitting Images YES, And NO Folder from Dataset

Brain Tumor	Brain MRI Images
Yes	155
No	98
Total	253

Figure 13.

Count Of Classes In Each Set.



Language and tool used.

Python programming Language used to implement project work. The following are some of the primary motivations for implementing our research in Python:

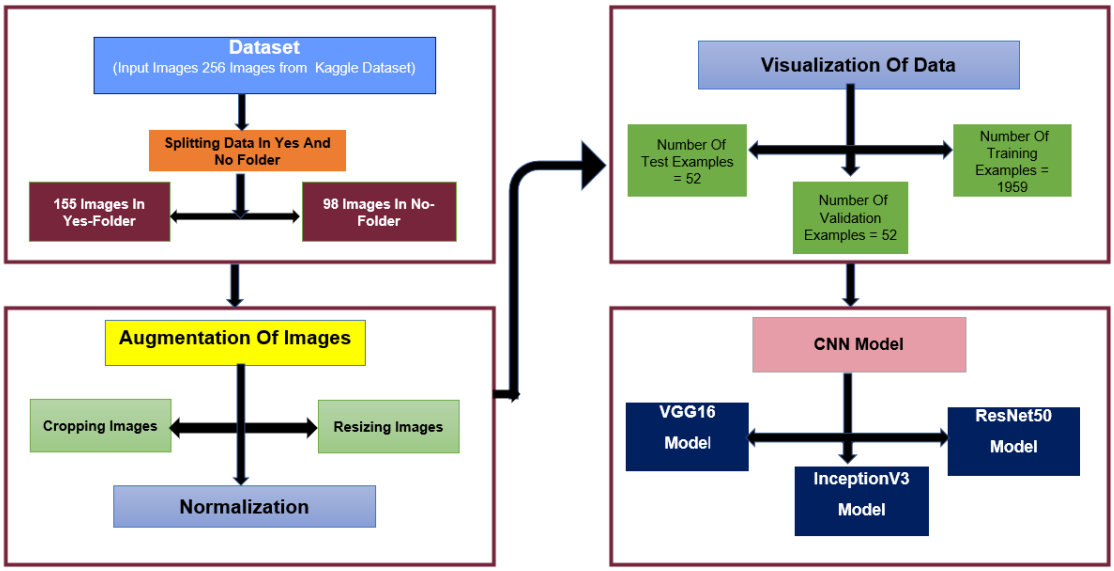
- Python has several benefits to MATLAB, including being free, open source, and providing additional datasets and visual packages. Python code is more readable than MATLAB code and offers more versatility in terms of code structure.
- The study team used various tools to build Python programs, including Google Colab. Colab by Google is a tool that allows you to train and search for a learning machine. You do not need to prepare anything before using it (GPU).
- A free research project called Google Colab can save and read all notes straight from Google Drive.
- Python supports many libraries, including the keras, shutil, fnmatch, and OS libraries.

Proposed Workflow

The detailed workflow for our suggested project is presented in this section. We are using the VGG-16 model, InceptionV3, and ResNet50 model. We investigated how to assess performance while detecting brain tumors accurately.

Figure 14.

Workflow of the Proposed Model.

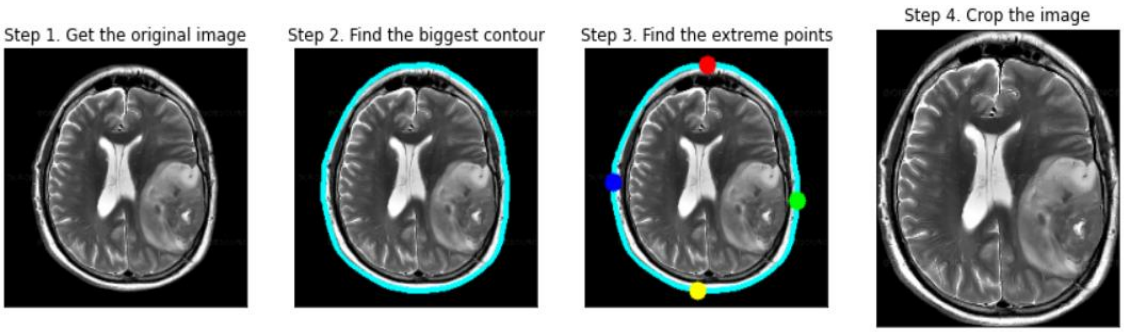


Pre-processing

In the following stage, images in the data set will be resampled to (224x224) and pre-processed using the VGG-16 model's input. Each image in the data set has been pre-processed. Normalization is a necessary initial step. To begin, a portion of the images of the brain must be removed. The pre-processing method is known as "normalization," which Explains the process of pre-processing step by step.

Figure 15.

Pre-Processing In A Step-By-Step



The first figure depicts the input image that must be pre-processed, while the second image depicts the second step of pre-processing in which the highest contour

must be identified. Here, we must identify the brain's boundaries. The third figure depicts the extreme places that must be identified to eliminate the brain from the image. As seen in the last figure, this is the final image.

Data Augmentation

Pre-processing procedures such as data augmentation are necessary. Because the data collection is small, we employ data augmentation to make it larger. Then we can train the model with a sufficient number of samples. The data imbalance will be lessened as a result.

In order to train massive neural networks, data augmentation methods including cropping, padding, and horizontal flipping are often used.

Regarding numbers: The no and Yes folders include 253 Brain MRI Images. There are 155 tumorous brain MRI images in the yes folder and 98 non-tumorous brains MRI images in the no folder. We will have 2065 samples after applying augmentation.

- There are a total of 1960 training examples.
- There are a total of 52 validation examples.
- There are a total of 52 test samples used in this study.

Figure 16.

Original Image

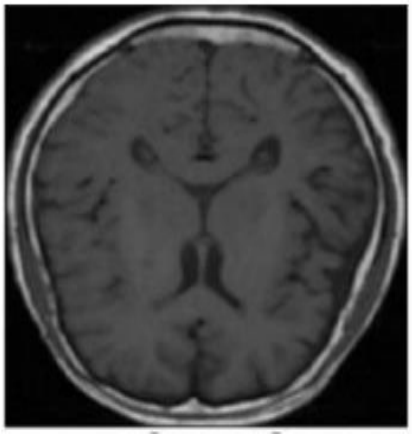
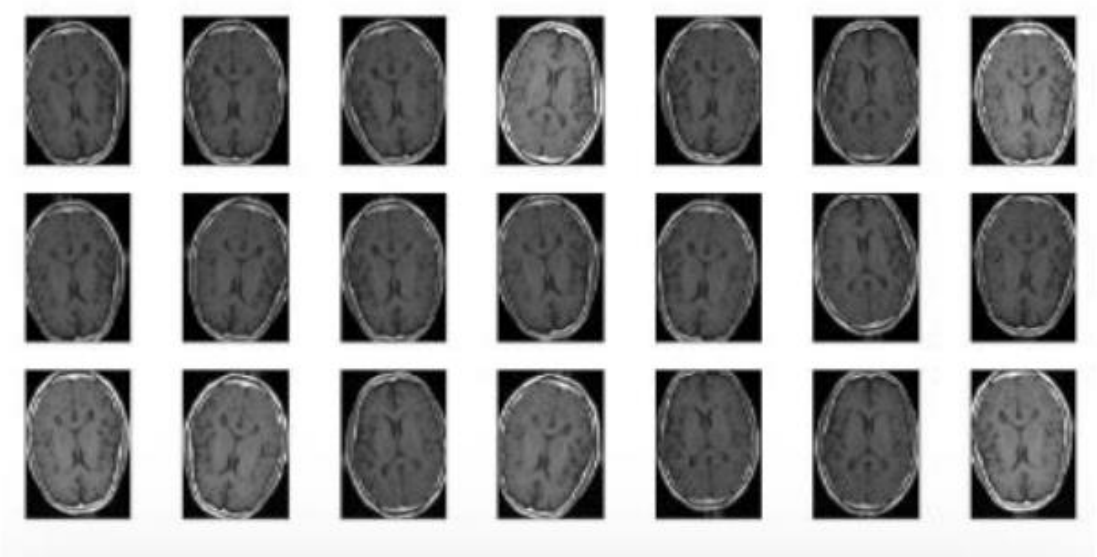


Figure 17.

Dataset After Data Augmentation



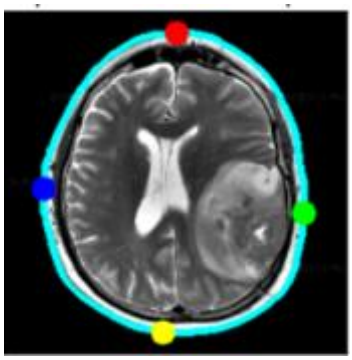
As a result of the data augmentation, here is what the data set looks like. The enhanced images may be seen in fig.16. After the data augmentation, we now have 253 pieces of data, 177 pieces of training data, and 76 testing data. Validation data is applied to the remaining images.

Normalization

OpenCV's cv2.Threshold, cv2.dilate functions can only be implemented by importing the cv2. Package. Two standard morphological operators, Dilate and Erosion, are employed most often. Based on image structure, these processes operate on images.

Figure 18.

Image Normalization



The original image is converted to grayscale using the above procedures, and the little, noisy dais ta is eliminated (Kamil Dimililer, August 2016,). Using the cv2 Find contour technique, we will get the maximum contour area. The intended image will have coordinates similar to a matrix (x, y). The essential advantage is noise reduction. Identifying intensity bumps or gaps in an image is a further advantage.

Plot Histogram

The whiteness intensity is low, reflecting that they do not have a tumor.

Figure 19.

The Whiteness Intensity Is Low, Reflecting

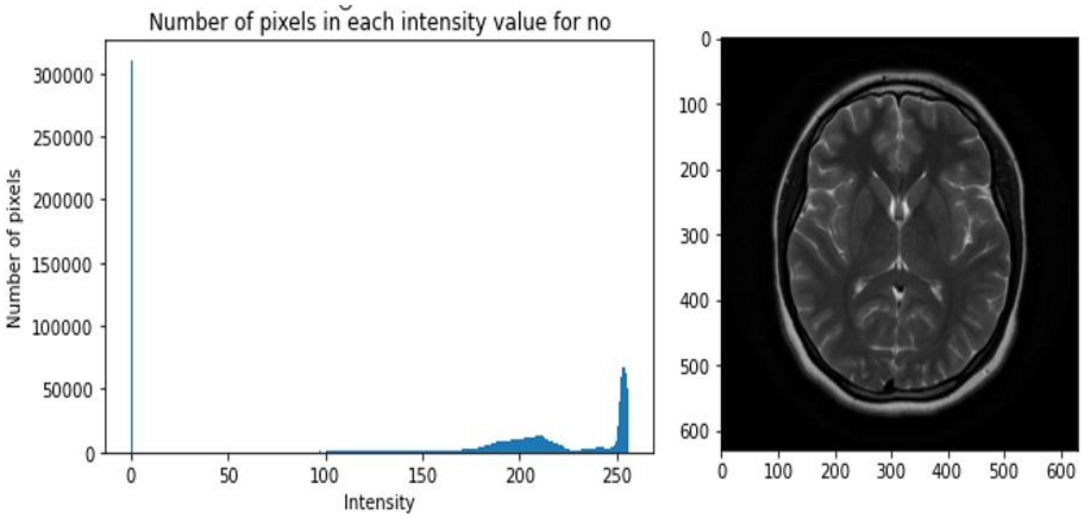
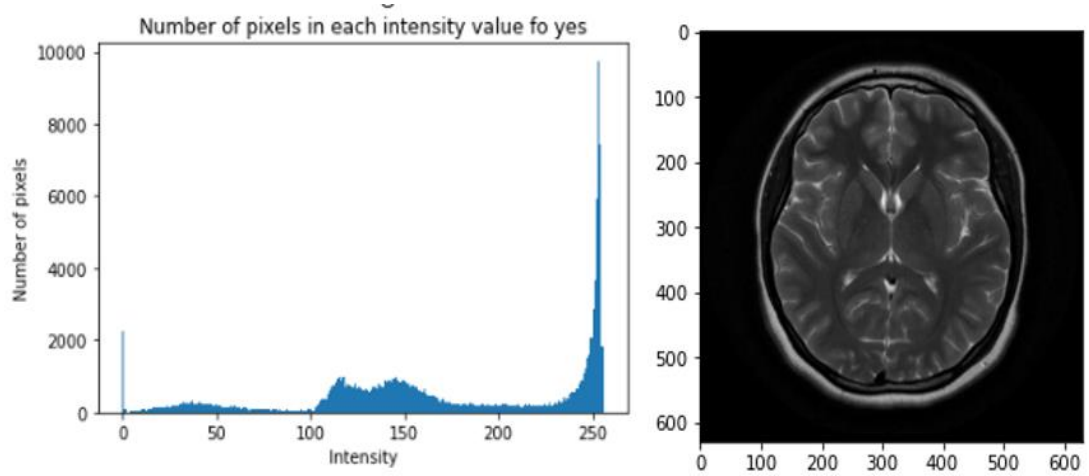


Figure 20.

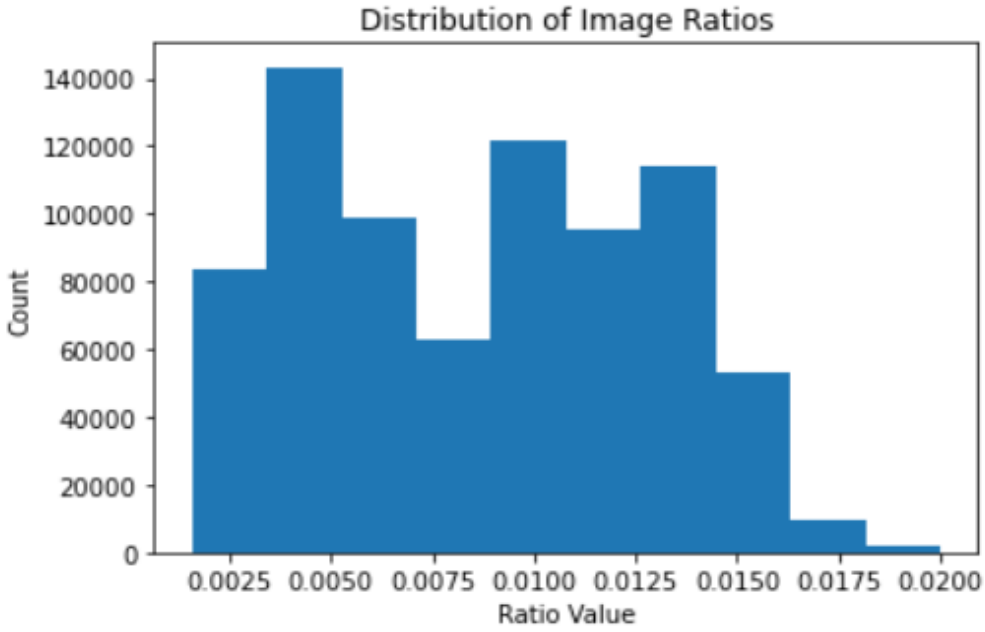
The Intensity of Whiteness Is Very Height



Distribution of image Ration

Figure 21.

Distribution Of Image Ration



Experimental Results

I will detail the experimental results of the suggested technique in this part, which I shall cover in detail.

Network Architecture

For our dataset on brain tumors, we trained VGG16, ResNet5, and InceptionV3 models from scratch, in addition to three pre-trained models.

Models Training and Validation

Create a Proposed Model

In order to create this model, 12 convolutional layers were built from scratch, in addition to a fully linked hidden layer (see figure 22). In the output layer, softmax activation is utilized since the probability for each class must be output. Adam optimization was used to maximize the network.

The model is applied to training sets of varied sizes as part of the training method. The network's weights will be changed automatically depending on the

gradients determined for each batch. Epochs are defined as one iteration across the whole training set. An individual is typically trained until he has experienced a consistent loss.

Checkpoints were included to maintain the model's second-highest level of validation accuracy. The network can overfit after several epochs, which is helpful. To implement these features, Keras uses the callback feature. At specific points in the training process, such as when an epoch ends, a sequence of routines is called a callback. Keras has two built-in features: model checkpointing and a learning rate schedule.

After training and verifying our model, we got 97% training accuracy and 94.28% validation accuracy. (as shown in figure 22).

Figure 22.

CNN Model Architecture Summary

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 240, 240, 3)]	0
zero_padding2d (ZeroPadding2D)	(None, 244, 244, 3)	0
conv2d (Conv2D)	(None, 238, 238, 32)	4736
bn0 (BatchNormalization)	(None, 238, 238, 32)	128
activation (Activation)	(None, 238, 238, 32)	0
max_pooling2d (MaxPooling2D)	(None, 59, 59, 32)	0
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 32)	0
flatten (Flatten)	(None, 6272)	0
dense (Dense)	(None, 1)	6273

Total params: 11,137
Trainable params: 11,073
Non-trainable params: 64

Figure 23.

CNN Model 22 Epoch With 32 Batches

```
Epoch 1/22
62/62 [==============================] - 191s 3s/step - loss: 0.8133 - accuracy: 0.6233 - val_loss: 0.6411 - val_accuracy: 0.65
38
Epoch 2/22
62/62 [==============================] - 158s 3s/step - loss: 0.5192 - accuracy: 0.7499 - val_loss: 0.5590 - val_accuracy: 0.75
00
Epoch 3/22
62/62 [==============================] - 152s 2s/step - loss: 0.4166 - accuracy: 0.8101 - val_loss: 0.4887 - val_accuracy: 0.84
62
Epoch 4/22
62/62 [==============================] - 154s 2s/step - loss: 0.4030 - accuracy: 0.8285 - val_loss: 0.4722 - val_accuracy: 0.75
00
Epoch 5/22
62/62 [==============================] - 155s 3s/step - loss: 0.3417 - accuracy: 0.8637 - val_loss: 0.4116 - val_accuracy: 0.80
77
Epoch 6/22
62/62 [==============================] - 145s 2s/step - loss: 0.3279 - accuracy: 0.8673 - val_loss: 0.3649 - val_accuracy: 0.82
69
Epoch 7/22
62/62 [==============================] - 149s 2s/step - loss: 0.3023 - accuracy: 0.8734 - val_loss: 0.3154 - val_accuracy: 0.88
46
Epoch 8/22
62/62 [==============================] - 148s 2s/step - loss: 0.2882 - accuracy: 0.8887 - val_loss: 0.3637 - val_accuracy: 0.82
69
Epoch 9/22
62/62 [==============================] - 149s 2s/step - loss: 0.2442 - accuracy: 0.9132 - val_loss: 0.2695 - val_accuracy: 0.88
46
Epoch 10/22
62/62 [==============================] - 157s 3s/step - loss: 0.2515 - accuracy: 0.8979 - val_loss: 0.7113 - val_accuracy: 0.71
15
Epoch 11/22
62/62 [==============================] - 154s 2s/step - loss: 0.2552 - accuracy: 0.9045 - val_loss: 0.4140 - val_accuracy: 0.82
69
Epoch 12/22
62/62 [==============================] - 165s 3s/step - loss: 0.2456 - accuracy: 0.9030 - val_loss: 0.2363 - val_accuracy: 0.86
54
Epoch 13/22
62/62 [==============================] - 152s 2s/step - loss: 0.2570 - accuracy: 0.8903 - val_loss: 0.3224 - val_accuracy: 0.82
69
Epoch 14/22
62/62 [==============================] - 148s 2s/step - loss: 0.1955 - accuracy: 0.9255 - val_loss: 0.2783 - val_accuracy: 0.88
46
Epoch 15/22
62/62 [==============================] - 149s 2s/step - loss: 0.1853 - accuracy: 0.9347 - val_loss: 0.2455 - val_accuracy: 0.90
38
Epoch 16/22
62/62 [==============================] - 150s 2s/step - loss: 0.1615 - accuracy: 0.9438 - val_loss: 0.4948 - val_accuracy: 0.80
77
```

```
Epoch 17/22
62/62 [==============================] - 149s 2s/step - loss: 0.1807 - accuracy: 0.9357 - val_loss: 0.1993 - val_accuracy: 0.90
38
Epoch 18/22
62/62 [==============================] - 159s 3s/step - loss: 0.1643 - accuracy: 0.9444 - val_loss: 0.2513 - val_accuracy: 0.90
38
Epoch 19/22
62/62 [==============================] - 163s 3s/step - loss: 0.1440 - accuracy: 0.9474 - val_loss: 0.8182 - val_accuracy: 0.71
15
Epoch 20/22
62/62 [==============================] - 156s 3s/step - loss: 0.1473 - accuracy: 0.9423 - val_loss: 0.2545 - val_accuracy: 0.90
38
Epoch 21/22
62/62 [==============================] - 154s 2s/step - loss: 0.1331 - accuracy: 0.9561 - val_loss: 0.1421 - val_accuracy: 0.96
15
Epoch 22/22
62/62 [==============================] - 152s 2s/step - loss: 0.1170 - accuracy: 0.9658 - val_loss: 0.1616 - val_accuracy: 0.94
23
```

In order to illustrate how the model's training and validation worked, we made graphs using Matplotlib that showed how accurate it was and how much loss it suffered.

Figure 24 depicts the network's learning curve during training and validation. There was an upward trend in validation and training accuracy, which culminated in both reaching 97 percent.

Figure 24.

Loss Curve During Training And Validation

In Figure 25, we show the loss curve during training and validation of the network. In conclusion, the validation and training losses decreased with the number of iterations, which proves the model's effectiveness.

Figure 25.

Accuracy Training And Validation

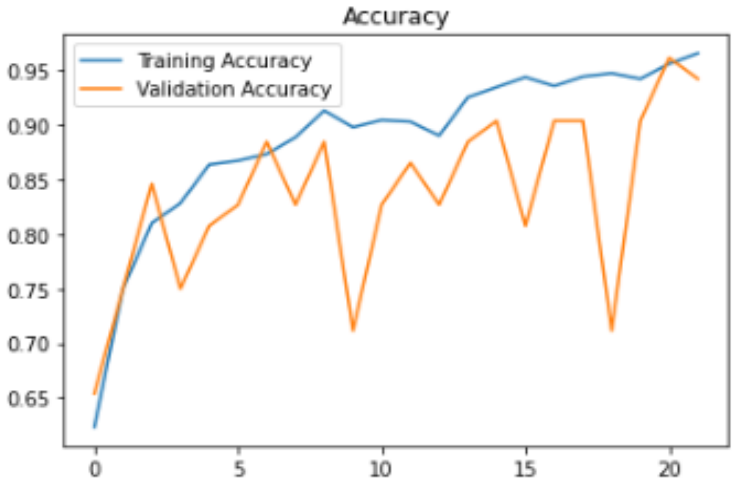


Figure 26.

CNN Model's Validation Accuracy = 0.94

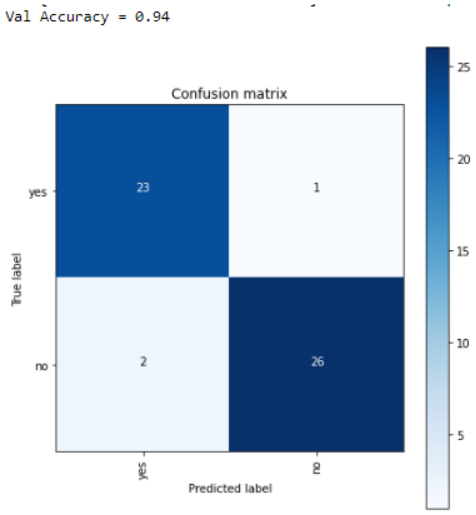


Figure 27.

CNN Model's Test Accuracy = 0.90

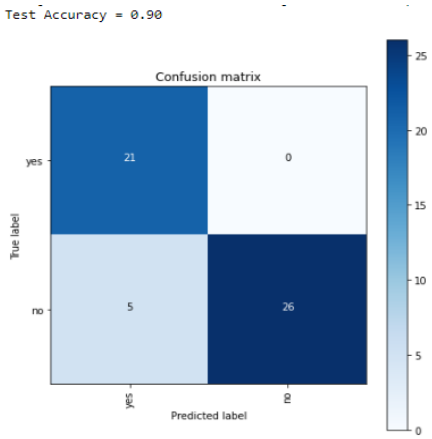
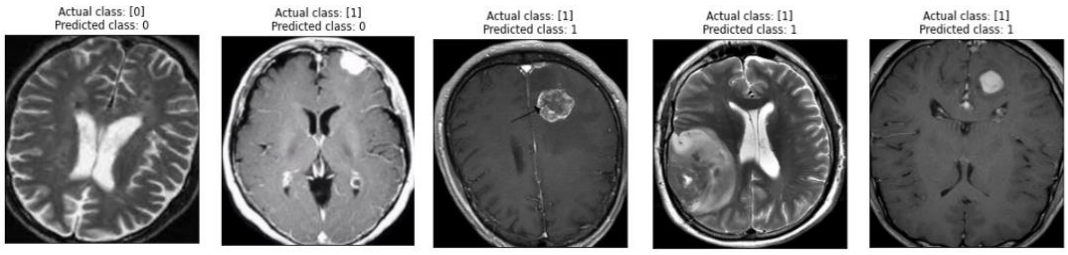


Figure 28.

Form Actual And Predicated Images From CNN Model



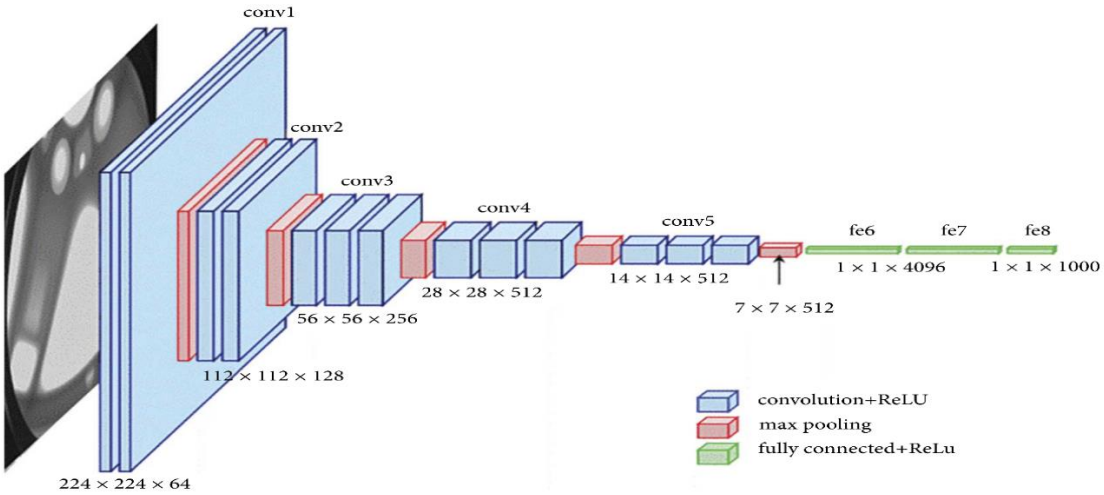
VGG16 model

The CNN model we used in our experiments was pre-trained on VGG-16. As a result of the small size of the images dataset, we fine-tuned our model by freezing parts of the convolution layers. Researchers created VGG-16, which has sixteen convolutional layers. MRI images of the brain with a dimension of 224 224 3 are fed into the model. It contains five max-pooling layers of dimension 2 2 and five Conv layers with kernels (filters) of fixed 3 3 filter size.

Moreover, it includes two fully linked layers with a SoftMax output layer and comprehensive ReLU activation features. There are around 140 million hyperparameters in the VGG-16 model. Stacking many convolutional layers improves the ability to learn features that have never been seen before. Hyperparameters govern model behavior. It minimizes a predetermined loss function to improve outcomes for a given dataset. The hyperparameters to be modified are neuron counts, epoch counts, SoftMax activation functions, learning rates, and optimizers. The second hyperparameter to tune is the number of convolution layers.

Figure 29.

A Standard VGG 16 Network Architecture (Chetana Srinivas, 2022)



Therefore, as ConvNet depth increases, so does the potential to learn hidden features at a lower cost. In figure 28, you can see the architecture of the VGG-16 ConvNet.

In addition to dropout layers, the VGG16 network is a pre-trained model with 16 convolutional layers and a fully connected hidden layer.

Figure 30.

VGG 16 Model Architecture Summary

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg16 (Functional)          (None, 7, 7, 512)         14714688

 flatten_1 (Flatten)         (None, 25088)             0

 dropout (Dropout)           (None, 25088)             0

 dense_1 (Dense)             (None, 1)                 25089

=================================================================
Total params: 14,739,777
Trainable params: 25,089
Non-trainable params: 14,714,688
```

Figure 31.

VGG16 Model 22 Epoch With 32 Batch

```
Epoch 1/22
62/62 [==============================] - 733s 12s/step - loss: 0.4799 - accuracy: 0.7667 - val_loss: 0.2552 - val_accuracy: 0.8
846
Epoch 2/22
62/62 [==============================] - 713s 11s/step - loss: 0.2211 - accuracy: 0.9234 - val_loss: 0.1319 - val_accuracy: 0.9
808
Epoch 3/22
62/62 [==============================] - 652s 10s/step - loss: 0.1670 - accuracy: 0.9474 - val_loss: 0.1073 - val_accuracy: 0.9
808
Epoch 4/22
62/62 [==============================] - 633s 10s/step - loss: 0.1228 - accuracy: 0.9678 - val_loss: 0.0901 - val_accuracy: 0.9
808
Epoch 5/22
62/62 [==============================] - 630s 10s/step - loss: 0.0967 - accuracy: 0.9791 - val_loss: 0.0681 - val_accuracy: 1.0
000
Epoch 6/22
62/62 [==============================] - 16984s 278s/step - loss: 0.0834 - accuracy: 0.9862 - val_loss: 0.0685 - val_accuracy:
0.9808
Epoch 7/22
62/62 [==============================] - 673s 11s/step - loss: 0.0682 - accuracy: 0.9883 - val_loss: 0.0569 - val_accuracy: 1.0
000
Epoch 8/22
62/62 [==============================] - 719s 12s/step - loss: 0.0584 - accuracy: 0.9898 - val_loss: 0.0466 - val_accuracy: 1.0
000
Epoch 9/22
62/62 [==============================] - 699s 11s/step - loss: 0.0520 - accuracy: 0.9918 - val_loss: 0.0470 - val_accuracy: 1.0
000
Epoch 10/22
62/62 [==============================] - 714s 12s/step - loss: 0.0468 - accuracy: 0.9923 - val_loss: 0.0557 - val_accuracy: 0.9
615
Epoch 11/22
62/62 [==============================] - 733s 12s/step - loss: 0.0388 - accuracy: 0.9949 - val_loss: 0.0420 - val_accuracy: 1.0
Epoch 12/22
62/62 [==============================] - 748s 12s/step - loss: 0.0365 - accuracy: 0.9969 - val_loss: 0.0470 - val_accuracy: 0.9
808
Epoch 13/22
62/62 [==============================] - 725s 12s/step - loss: 0.0338 - accuracy: 0.9964 - val_loss: 0.0372 - val_accuracy: 0.9
808
Epoch 14/22
62/62 [==============================] - 650s 10s/step - loss: 0.0280 - accuracy: 0.9980 - val_loss: 0.0381 - val_accuracy: 1.0
000
Epoch 15/22
62/62 [==============================] - 700s 11s/step - loss: 0.0248 - accuracy: 0.9974 - val_loss: 0.0594 - val_accuracy: 0.9
615
Epoch 16/22
62/62 [==============================] - 721s 12s/step - loss: 0.0218 - accuracy: 0.9995 - val_loss: 0.0593 - val_accuracy: 0.9
615
Epoch 17/22
62/62 [==============================] - 733s 12s/step - loss: 0.0216 - accuracy: 0.9964 - val_loss: 0.0347 - val_accuracy: 1.0
000
Epoch 18/22
62/62 [==============================] - 714s 12s/step - loss: 0.0260 - accuracy: 0.9949 - val_loss: 0.0282 - val_accuracy: 1.0
000
Epoch 19/22
62/62 [==============================] - 737s 12s/step - loss: 0.0198 - accuracy: 1.0000 - val_loss: 0.0323 - val_accuracy: 1.0
000
Epoch 20/22
62/62 [==============================] - 693s 11s/step - loss: 0.0196 - accuracy: 0.9985 - val_loss: 0.0296 - val_accuracy: 1.0
000
Epoch 21/22
62/62 [==============================] - 657s 11s/step - loss: 0.0170 - accuracy: 0.9990 - val_loss: 0.0295 - val_accuracy: 1.0
000
Epoch 22/22
62/62 [==============================] - 647s 10s/step - loss: 0.0164 - accuracy: 0.9995 - val_loss: 0.0324 - val_accuracy: 0.9
808
```

Dropout regularizes the networks to avoid overfitting by removing ReLU activations from the output layer.

There is a gradual rise in the accuracy number vallation accuracy. As a result, this shows that the created model is both practical and learnable.

Despite changes in the amount of loss and validation loss over time, it has a training accuracy rate of approximately 99.94%.

Figure 31 depicts the training and validation of the VGG16 model for brain tumor datasets. The brain tumor dataset's VGG16 model training and validation losses are shown in Figure 31.

Figure 32.

Vgg16 Training And Validation Accuracy Curve

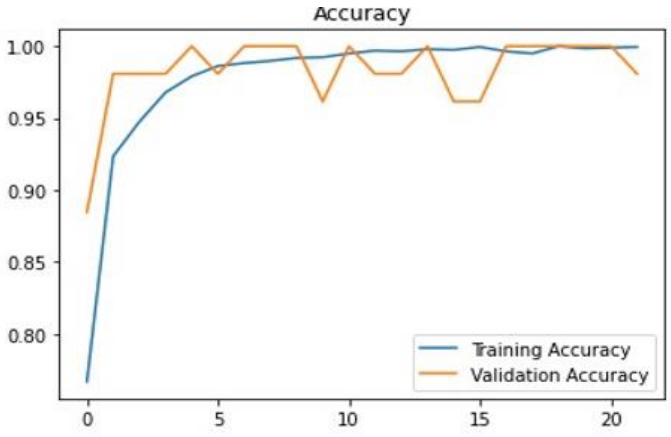


Figure 33.

Vgg16 Training And Validation Loss Curve

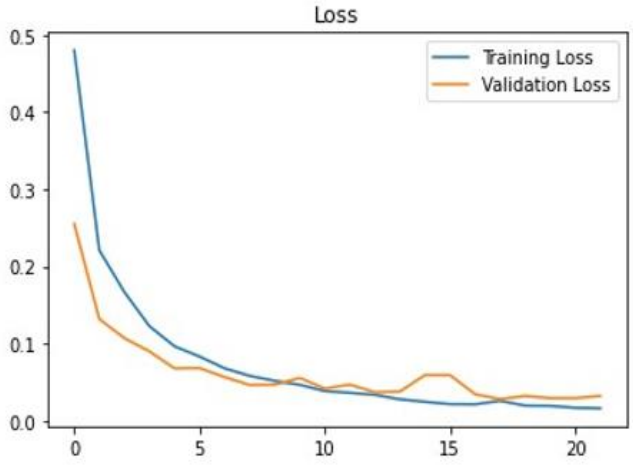


Figure 34.

Vgg16 Validation Accuracy

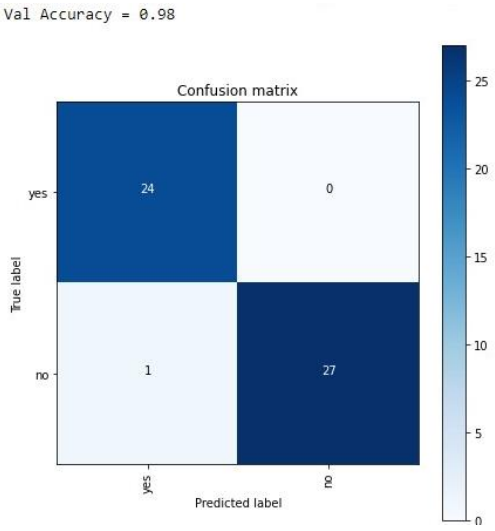


Figure 35.

VGG16 Test Accuracy

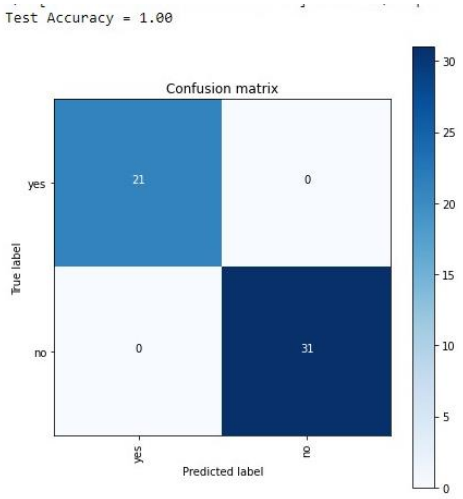
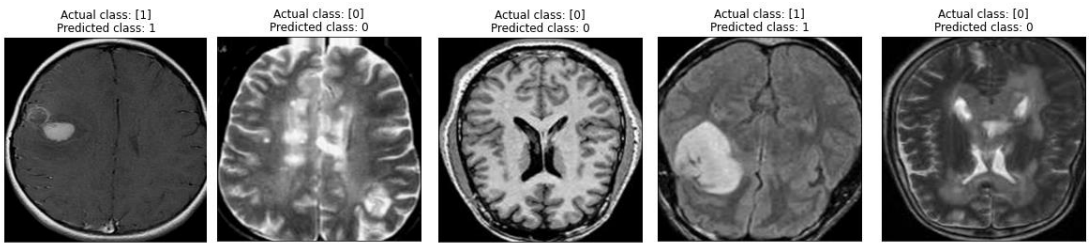


Figure 36.

For Actual And Predicated Images From VGG 16 Model

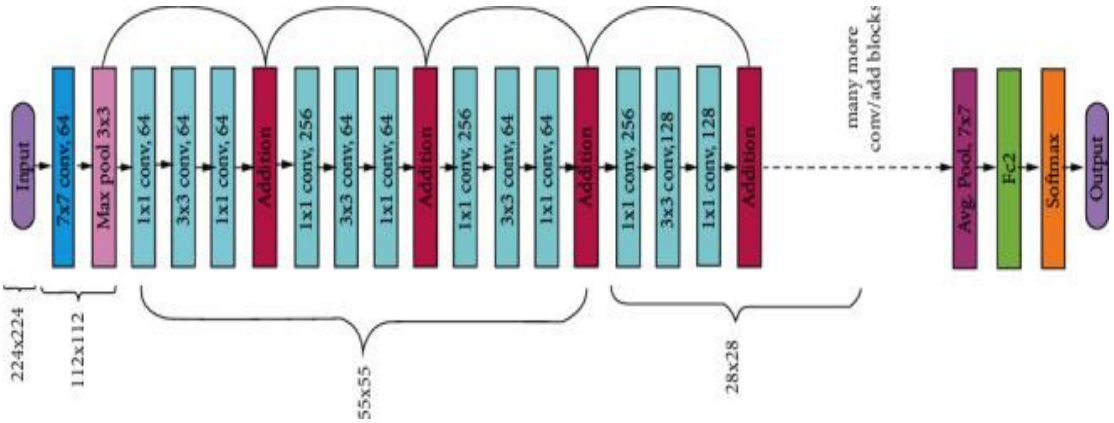


ResNet50 model

Kaiming He et al. at Microsoft Research, 2015 created a 50-layer Residual Network with 26M parameters for image identification and classification. The term "feature subtraction" is used to describe residual networks. Instead of acquiring new traits, they hone down on those excised from the source material. ResNet50 requires less time to train than conventional deep CNNs. The ImageNet database is used to train the pre-trained model. These networks also improve the accuracy of image classification. In addition, it has significant batch normalization and skips connections. Gated recurrent units or gated units are names given to specific skip connectivity. Nth layer input is directly connected to some (n+x)th layer input in the ResNet50 network, a deep neural network. Compared to the VGG16 and VGG19 models, this residual network has a lower temporal complexity. In our experiment, we employed a pre-trained and calibrated ResNet50 model. Figure 36 depicts ResNet50's pre-trained architecture.

Figure 37.

ResNet50's Pre-Trained Architecture (Chetana Srinivas, 2022)

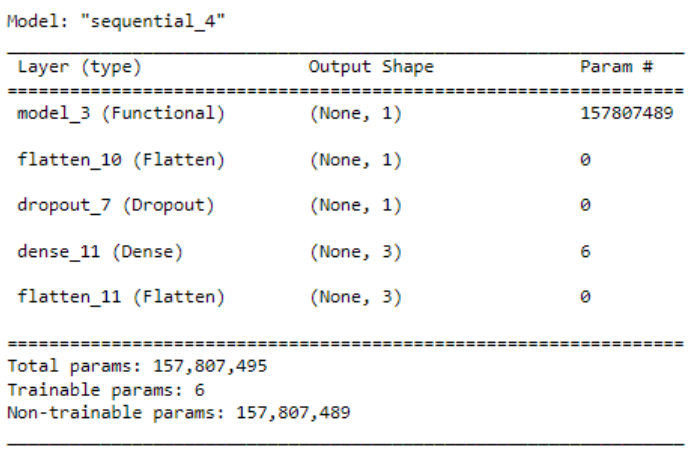


ResNet50 was the pre-trained model that we used. It contains 50 convolutional layers, one of which is a fully connected hidden layer, while the rest are dropout layers. Dropout prevents networks from overfitting, and all layers except the output layer employ ReLU activations.

With the pre-trained ResNet50 model, we obtained 48 percent validation accuracy for the brain tumor dataset.

Figure 38.

ResNet50 Model Architecture Summary



```
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 model_3 (Functional)        (None, 1)                 157807489

 flatten_10 (Flatten)        (None, 1)                 0

 dropout_7 (Dropout)         (None, 1)                 0

 dense_11 (Dense)            (None, 3)                 6

 flatten_11 (Flatten)        (None, 3)                 0

=================================================================
Total params: 157,807,495
Trainable params: 6
Non-trainable params: 157,807,489
_________________________________________________________________
```

Figure 39.

ResNet50 Model 22 Epoch With 32 Batch

```
Epoch 1/22
62/62 [==============================] - 182s 3s/step - loss: 0.7307 - accuracy: 0.2348 - val_loss: 0.7056 - val_accuracy: 0.01
92
Epoch 2/22
62/62 [==============================] - 174s 3s/step - loss: 0.7215 - accuracy: 0.2532 - val_loss: 0.7020 - val_accuracy: 0.00
00e+00
Epoch 3/22
62/62 [==============================] - 165s 3s/step - loss: 0.7171 - accuracy: 0.2532 - val_loss: 0.6993 - val_accuracy: 0.01
92
Epoch 4/22
62/62 [==============================] - 167s 3s/step - loss: 0.7136 - accuracy: 0.2547 - val_loss: 0.6971 - val_accuracy: 0.05
77
Epoch 5/22
62/62 [==============================] - 178s 3s/step - loss: 0.7112 - accuracy: 0.2665 - val_loss: 0.6955 - val_accuracy: 0.07
69
Epoch 6/22
62/62 [==============================] - 172s 3s/step - loss: 0.7070 - accuracy: 0.2782 - val_loss: 0.6943 - val_accuracy: 0.11
54
Epoch 7/22
62/62 [==============================] - 177s 3s/step - loss: 0.7054 - accuracy: 0.2741 - val_loss: 0.6934 - val_accuracy: 0.07
69
Epoch 8/22
62/62 [==============================] - 157s 3s/step - loss: 0.7034 - accuracy: 0.2879 - val_loss: 0.6928 - val_accuracy: 0.17
31
Epoch 9/22
62/62 [==============================] - 151s 2s/step - loss: 0.7019 - accuracy: 0.2869 - val_loss: 0.6925 - val_accuracy: 0.19
23
Epoch 10/22
62/62 [==============================] - 153s 2s/step - loss: 0.7010 - accuracy: 0.2986 - val_loss: 0.6921 - val_accuracy: 0.15
38
Epoch 11/22
62/62 [==============================] - 154s 2s/step - loss: 0.7002 - accuracy: 0.2976 - val_loss: 0.6920 - val_accuracy: 0.23
08
Epoch 12/22
62/62 [==============================] - 159s 3s/step - loss: 0.6993 - accuracy: 0.3068 - val_loss: 0.6918 - val_accuracy: 0.26
92
Epoch 13/22
62/62 [==============================] - 166s 3s/step - loss: 0.6982 - accuracy: 0.3078 - val_loss: 0.6917 - val_accuracy: 0.30
77
Epoch 14/22
62/62 [==============================] - 162s 3s/step - loss: 0.6977 - accuracy: 0.3088 - val_loss: 0.6916 - val_accuracy: 0.30
77
Epoch 15/22
62/62 [==============================] - 152s 2s/step - loss: 0.6973 - accuracy: 0.3170 - val_loss: 0.6917 - val_accuracy: 0.28
85
Epoch 16/22
62/62 [==============================] - 159s 3s/step - loss: 0.6968 - accuracy: 0.3104 - val_loss: 0.6916 - val_accuracy: 0.26
92
Epoch 17/22
62/62 [==============================] - 151s 2s/step - loss: 0.6961 - accuracy: 0.3206 - val_loss: 0.6915 - val_accuracy: 0.26
92
Epoch 18/22
62/62 [==============================] - 152s 2s/step - loss: 0.6959 - accuracy: 0.3119 - val_loss: 0.6915 - val_accuracy: 0.25
00
Epoch 19/22
62/62 [==============================] - 151s 2s/step - loss: 0.6957 - accuracy: 0.3104 - val_loss: 0.6915 - val_accuracy: 0.32
69
Epoch 20/22
62/62 [==============================] - 149s 2s/step - loss: 0.6954 - accuracy: 0.3155 - val_loss: 0.6914 - val_accuracy: 0.34
62
Epoch 21/22
62/62 [==============================] - 146s 2s/step - loss: 0.6947 - accuracy: 0.3272 - val_loss: 0.6914 - val_accuracy: 0.34
62
Epoch 22/22
62/62 [==============================] - 146s 2s/step - loss: 0.6948 - accuracy: 0.3170 - val_loss: 0.6914 - val_accuracy: 0.34
62
```

Figure 39 depicts the precision of ResNet50 model training and validation on the brain tumor dataset. Figure 40 shows the loss of the training and validation ResNet50 models for brain tumor data sets.

Figure 40.

ResNet50 Training And Validation Loss Curve

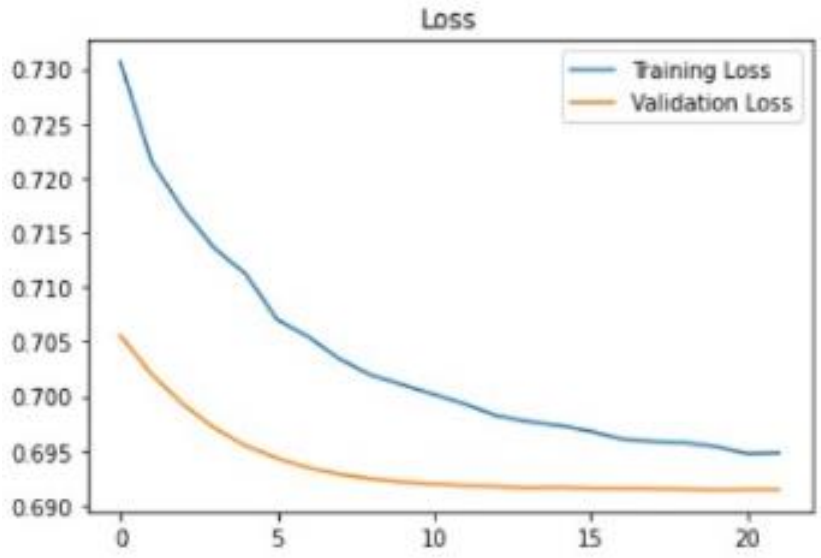


Figure 41.

ResNet50 Training and Validation Accuracy Curve

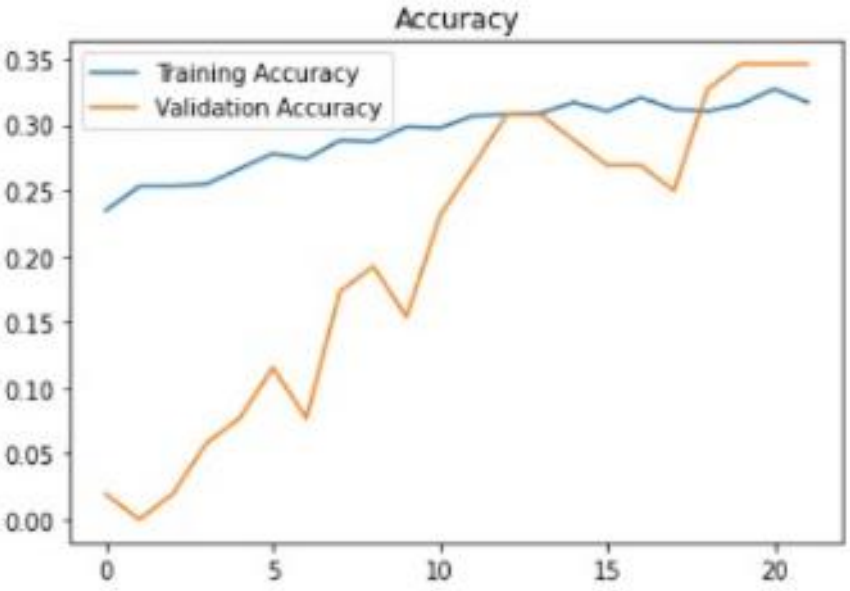


Figure 42.

ResNet50 Validation Accuracy

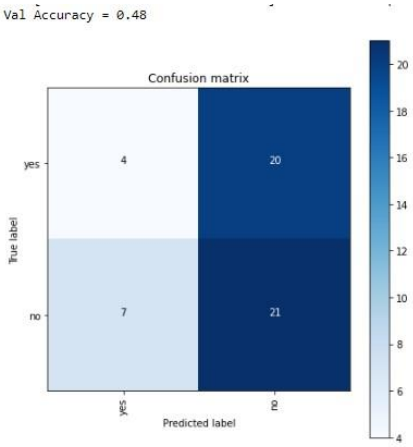


Figure 43.

ResNet50 Test Accuracy

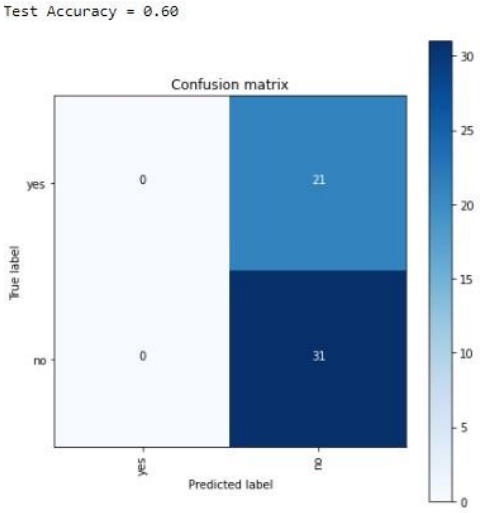
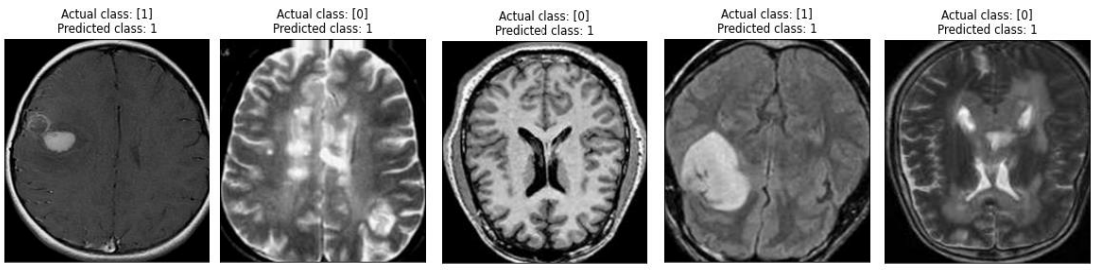


Figure 44.

For Actual and Predicated Images from ResNet50 Model



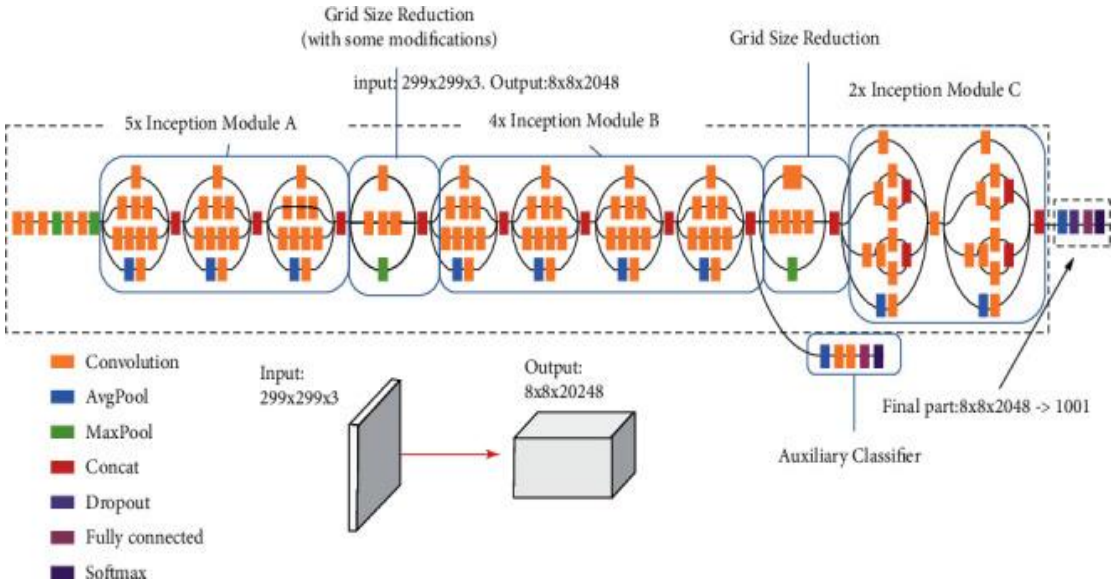
InceptionV3 model

Google Brain Team unveiled and constructed Inception in 2014. The ImageNet database served as the foundation for the GoogleNet CNN network. There were 22 layers of pre-trained networks with 5M parameters in Inception's 22-layer pre-trained network, with ten parameters for each layer. Reduced calculation time and network performance are the goals of utilizing 11-kernel filters. A new version of the Inception network, Inception-v3, was implemented, which allowed the Conv layers to scale down using hyperparameters. Deep neural networks classify MRI brain pictures into thousands of image classifications.

Self-learning models can detect and classify photographs depending on whether or not they exhibit features of handcrafted work. The Inception-v3 network requires an image with a resolution of 299 299 pixels as an input. It employs batch normalization, RMSprop, and image distortion to boost speed for computer vision challenges. The basic architecture is shown in Figure 44 of the Inception-v3 model.

Figure 45.

Inception-v3 Model Architecture (Chetana Srinivas, 2022)



Our pre-built model, InceptionV3, has 28 convolutional layers and a fully linked hidden layer to reduce overfitting. The output is the only layer that does not incorporate ReLU activations. As a result of the InceptionV3 model, the model's training was 48 percent accurate, and its validation was 44 percent accurate. Figure 45 illustrates this.

Figure 46.

Inceptionv3 Pertained Model Architecture Summary

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 model_1 (Functional)        (None, 1)                 97302305

 flatten_3 (Flatten)         (None, 1)                 0

 dropout_2 (Dropout)         (None, 1)                 0

 dense_4 (Dense)             (None, 1)                 2

=================================================================
Total params: 97,302,307
Trainable params: 2
Non-trainable params: 97,302,305
_________________________________________________________________
```

Figure 47.

Inceptionv3 22 Epoch With 32 Batch

```
Epoch 1/22
62/62 [==============================] - 169s 3s/step - loss: 0.8368 - accuracy: 0.4916 - val_loss: 0.7585 - val_accuracy: 0.53
85
Epoch 2/22
62/62 [==============================] - 147s 2s/step - loss: 0.8118 - accuracy: 0.4957 - val_loss: 0.7506 - val_accuracy: 0.53
85
Epoch 3/22
62/62 [==============================] - 146s 2s/step - loss: 0.8014 - accuracy: 0.5064 - val_loss: 0.7439 - val_accuracy: 0.55
77
Epoch 4/22
62/62 [==============================] - 146s 2s/step - loss: 0.7872 - accuracy: 0.5054 - val_loss: 0.7384 - val_accuracy: 0.55
77
Epoch 5/22
62/62 [==============================] - 146s 2s/step - loss: 0.7807 - accuracy: 0.5013 - val_loss: 0.7340 - val_accuracy: 0.53
85
Epoch 6/22
62/62 [==============================] - 153s 2s/step - loss: 0.7804 - accuracy: 0.4936 - val_loss: 0.7304 - val_accuracy: 0.51
92
Epoch 7/22
62/62 [==============================] - 150s 2s/step - loss: 0.7761 - accuracy: 0.4855 - val_loss: 0.7274 - val_accuracy: 0.51
92
Epoch 8/22
62/62 [==============================] - 148s 2s/step - loss: 0.7592 - accuracy: 0.5048 - val_loss: 0.7252 - val_accuracy: 0.51
92
Epoch 9/22
62/62 [==============================] - 148s 2s/step - loss: 0.7486 - accuracy: 0.5043 - val_loss: 0.7234 - val_accuracy: 0.46
15
Epoch 10/22
62/62 [==============================] - 150s 2s/step - loss: 0.7564 - accuracy: 0.4936 - val_loss: 0.7220 - val_accuracy: 0.46
15
Epoch 11/22
62/62 [==============================] - 1049s 17s/step - loss: 0.7446 - accuracy: 0.4916 - val_loss: 0.7207 - val_accuracy: 0.
4038
Epoch 12/22
62/62 [==============================] - 151s 2s/step - loss: 0.7415 - accuracy: 0.4957 - val_loss: 0.7196 - val_accuracy: 0.42
31
Epoch 13/22
62/62 [==============================] - 156s 3s/step - loss: 0.7484 - accuracy: 0.4829 - val_loss: 0.7186 - val_accuracy: 0.42
31
Epoch 14/22
62/62 [==============================] - 151s 2s/step - loss: 0.7303 - accuracy: 0.4977 - val_loss: 0.7177 - val_accuracy: 0.44
23
Epoch 15/22
62/62 [==============================] - 151s 2s/step - loss: 0.7411 - accuracy: 0.4849 - val_loss: 0.7169 - val_accuracy: 0.44
23
Epoch 16/22
62/62 [==============================] - 153s 2s/step - loss: 0.7282 - accuracy: 0.4926 - val_loss: 0.7161 - val_accuracy: 0.42
31
Epoch 17/22
62/62 [==============================] - 152s 2s/step - loss: 0.7281 - accuracy: 0.4911 - val_loss: 0.7153 - val_accuracy: 0.44
23
Epoch 18/22
62/62 [==============================] - 154s 2s/step - loss: 0.7281 - accuracy: 0.4860 - val_loss: 0.7146 - val_accuracy: 0.46
15
Epoch 19/22
62/62 [==============================] - 150s 2s/step - loss: 0.7412 - accuracy: 0.4645 - val_loss: 0.7138 - val_accuracy: 0.46
15
Epoch 20/22
62/62 [==============================] - 150s 2s/step - loss: 0.7280 - accuracy: 0.4752 - val_loss: 0.7128 - val_accuracy: 0.46
15
Epoch 21/22
62/62 [==============================] - 151s 2s/step - loss: 0.7281 - accuracy: 0.4742 - val_loss: 0.7119 - val_accuracy: 0.46
15
Epoch 22/22
62/62 [==============================] - 161s 3s/step - loss: 0.7192 - accuracy: 0.4824 - val_loss: 0.7113 - val_accuracy: 0.44
```

InceptionV3 model's training and validation are shown in figure 47. Figure 48 illustrates the loss of the InceptionV3 model for the brain tumor dataset through training and validation.

Figure 48.

InceptionV3 Training And Validation Loss Curve

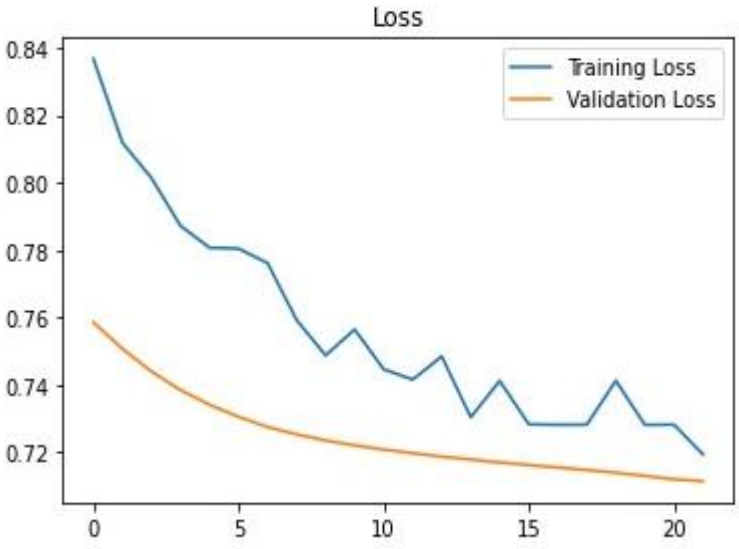


Figure 49.

InceptionV3 Training And Validation Accuracy Curve

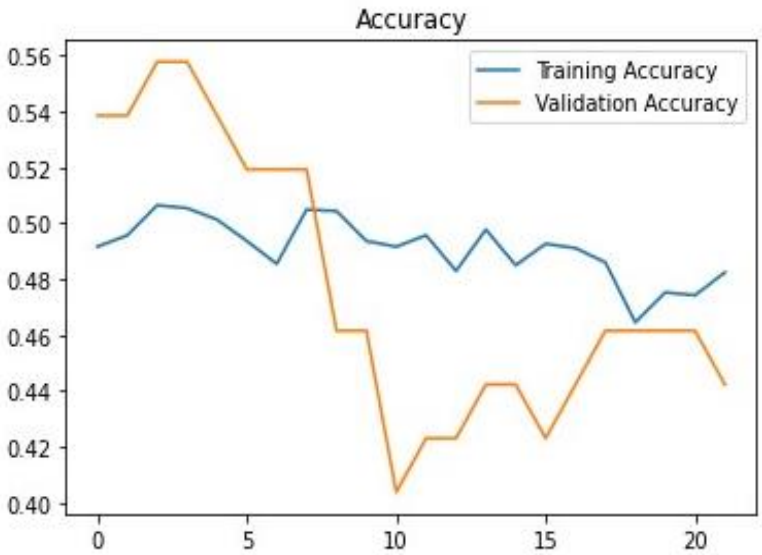


Figure 50.

InceptionV3 Validation Accuracy

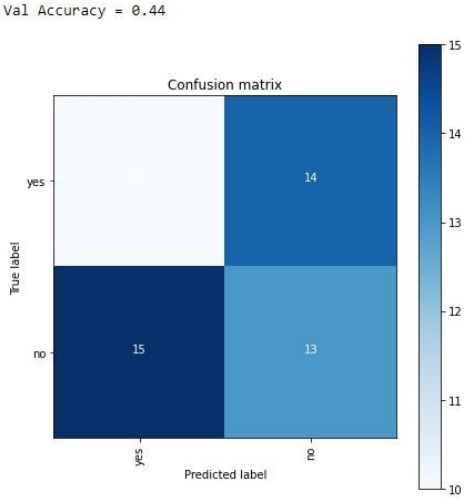


Figure 51.

InceptionV3 Test Accuracy

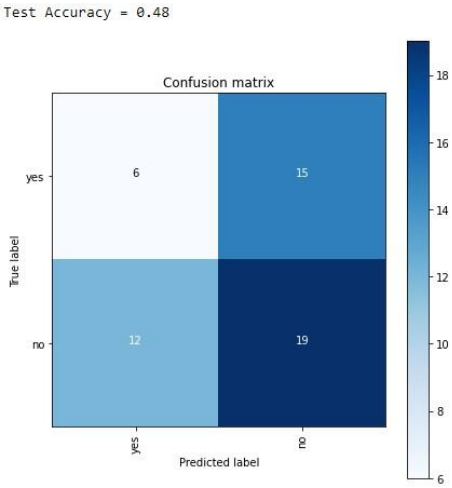
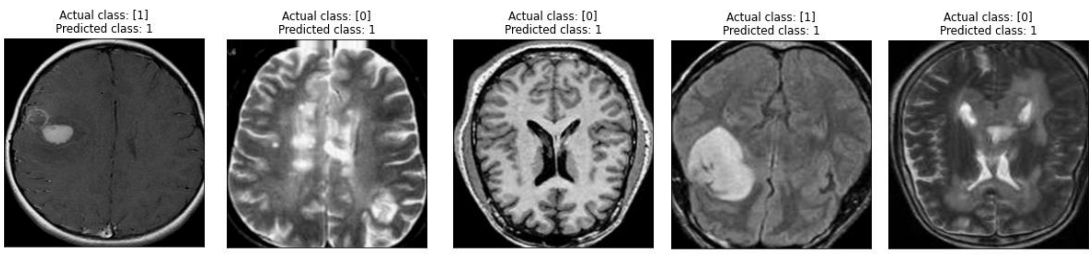


Figure 52.

For Actual And Predicated Images from InceptionV3 Model



Conclusion

In this chapter, we will go through a method for identifying and building CNN Model for brain tumors.

Brain MRI data is compared to deep pre-trained convolution neural networks for the automated detection of benign and malignant tumors. Pretrained CNN architectures, such as VGG16, Inception-v3, and ResNet50, are used in the image categorization. Each testing step includes a modification for the projected hyperparameter integral. Pretrained CNN architectures are expected to perform better if training and validation accuracy are increased. As training and validation accuracy improve, overfitting difficulties are expected. It is impossible for a model to accurately forecast a new dataset if just a tiny portion of its training data is used. According to assessment performance, CNN and VGG16 designs have a training accuracy of more than 0.9500 percent, with the highest validation accuracy of 0.9994 percent. All designs. We find that VGG-16 outperforms Inception-v3, ResNet50, and other pre-trained CNN models on both training and testing datasets. Loss and validation accuracy with less loss is much closer than validation accuracy. Images were classified by tumor kind using three pretrained models: VGG-16, Inception-V3, and ResNet50. It is possible to categorize brain tumors using a CNN model trained on VGG-16.

CHAPTER VI

EXPERIMENTAL RESULT & EVALUATION

Model Testing Data Set

The testing dataset (Brin-Tumor-test) contains 256 MRI brain tumor images that are distinct from the ones used in the training dataset and are divided into two categories: brain tumor and non-brain tumor.

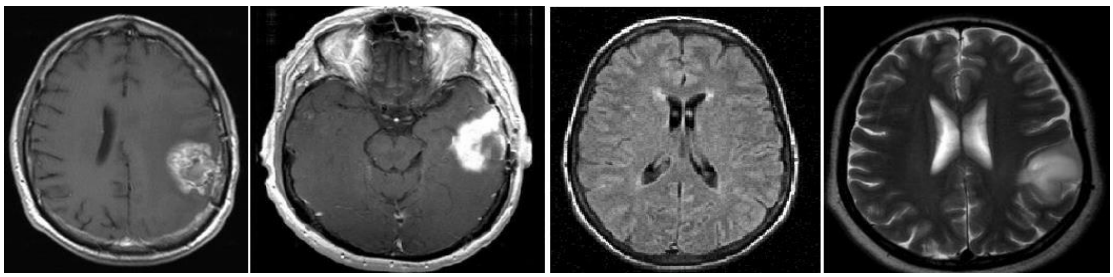
Table 3.

Distribution of The Test Dataset's Images

Categories	No testing images	No validation images	Image size
Non-brain Tumor	23	23	244x244 pixels
Brain Tumor	29	29	244x 244 pixels
Total	52	52	

Figure 53.

Example Of MRI Brain Tumor Images Used To Test The Four Models.



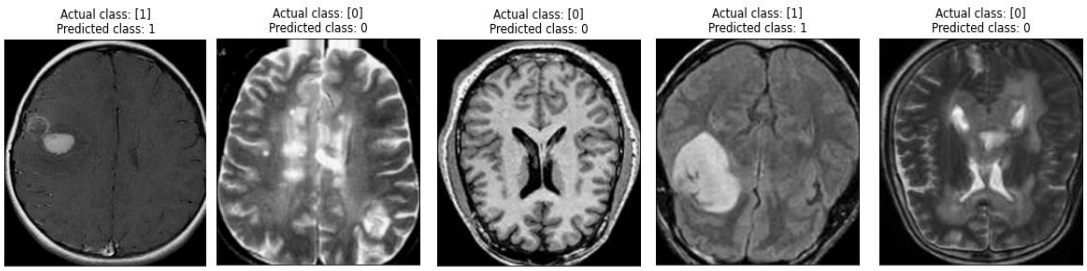
Testing the model

Using 98 MRI brain tumor images, including 155 tumors and 98 without tumors (normal), the network is trained and validated with 98 MRI brain tumor images.

I was using the model. Predict classes, and one may predict the classes of test photos after loading them. The following formula was used to determine the likelihood that each image belonged to a specific class: (1: normal) (0: brain tumor); after the prediction is complete, the model prints the first five results of each model, as shown in figure 54.

Figure 54.

The Initial Five Findings From Our Model Testing



CNN model was 97% correct, VGG16 99% correct, ResNet50 44% correct, and InceptionV3 48% correct during testing.

Comparative Analysis

The most current state-of-the-art investigations concerning the detection and monitoring of brain tumors must be examined in the study. This study analyzed recent studies that employed the CNN and VGG 16 algorithms to identify, classify, and classify brain cancers and were published within the previous ten years or so.

Performance Comparison

Brain MRI images are identified using multiple pre-defined procedures in current framework systems. An effective mechanism for detecting and classifying tumors in the brain using MRI imaging. Here are a few approaches and strategies for pursuing future goals. MRI brain scans are most commonly entered as input images. Depending on the architecture and memory limitations, the input might be 2D or 3D. Since this software greatly enhances image data, the input regarding the images to be entered is just as important as any other stage.

Brain MRI images are categorized using Deep Learning software without any prior processing. To find traits, a statistical or machine learning technique is applied. These returned properties are then used to train the Deep Learning system. Before choosing the optimal model, the main goal of each plan is to alter the Supervised Learning levels by practical guidelines.

Systems were developed using deep learning methods. Learning, testing, and verification sets are created from the dataset before any earlier approaches are used.

Convolutional Neural Networks (CNN) have received much acclaim and recognition in deep learning for their capacity to extract and recognize deep features from images automatically. All of the features of each study article previously given are condensed using a consistent comparison table (Table 4). The tactics employed thus far are summarized in this table. The comprehensive investigation's findings and limitations have been examined.

Table 4.

Performance Comparison Between the Existing Methodology

Ref	Years	Dataset	Type of tumors	Achieved outcomes
Anilkumar, B.; Kumar, P.R. Tumor classification using block wise fine tuning and transfer learning of deep neural network and K.N.N. classifier on M.R. brain images	2019	Kaggle	Detecting tumor or not tumor	92.3% acc.
Dequidt P., Bourdon P., Tremblais B., et al.	2021	BARTS-2020	HGG/LGG	84.1% acc.
Kesav N., Jibukumar M. Efficient and low complex architecture for detection and classification of brain tumor using cnn with two channel CNN.	2021	Kaggle	Glioma, meningioma, no tumor, and pituitary tumor	98.21% acc.
Das A., Mohapatra S. K., Mohanty M. N. Design of deep ensemble classifier with fuzzy decision method for biomedical image classification.	2022	Kaggle	Glioma, meningioma, no tumor, and pituitary tumor	95.24% acc.
Proposed method	2022	Kaggle	**Brain Tumor Detection**	**99.94%**

Result and Discussion

As a result of training our custom model using 2065 images for a total of 22 epochs and 32 batches, the following results had got it.

This thesis describes a method for determining whether images of brain tumors are MRIs or not. This was accomplished using the CNN model, the VGG16, the InceptionV3 ResNet50, and other architectures. After just 22 epochs of training, these

models could correctly predict MRI brain tumor images with a balanced accuracy of nearly 99 percent on the validation and test set.

Table 5.

Evaluation Results of the Validation Models.

Model	Accuracy (%)	Sensitivity (%)	Specificity (%)	Precision (%)
CNN Model	94.20%	92.00%	96.20%	95.80%
VGG16	**98.00%**	**95.40%**	**100%**	**100%**
ResNet50	48.00%	36.30%	51.20%	16.60%
InceptionV3	30.9%	0.00%	48.10%	0.00%

Table 6.

Evaluation Results of the Testing Models.

Model	Accuracy (%)	Sensitivity (%)	Specificity (%)	Precision (%)
CNN Model	90.30%	80.70%	100%	100%
VGG16	**100%**	**100%**	**100%**	**100%**
ResNet50	59.60%	0.00%	59.60%	0.00%
InceptionV3	48.00%	33.30%	55.88%	28.50%

Table 7.

Analyses of the Training, Validation, And Testing Models

Criterion	VGG16	CNN Model	ResNet50	InceptionV3
Training Accuracy	99.94%	97%	45.75%	48.85%
Validation Accuracy	98.00%	94%	48.00%	44.00%
Training loss	0.002	0.0025	0.6948	0.7192
Validation loss	0.0030	0.052	0.6914	0.7113
Testing Accuracy	1.00%	90%	60.00%	48.00%

This accuracy was achieved using strategies including Adam optimization, data augmentation, dropout, and others. Overfitting is the term used to describe a trained model that closely fits the training data but does not generalize well to new cases. This is especially apparent when the size of the training sample is small. By analysing the plotted training, one may evaluate the chance of overfitting. The curve clearly shows that the data loss for the training and validation datasets is similar. If there were overfitting, the loss on the training data would be significantly more significant than the validation data. In addition, three groups were created from the instances for this reason (training, validation, and test).

Larger training datasets, more image augmentation methods, and additional deep learning algorithms with more ensembles could all help to enhance this.

CHAPTER VII

CONCLUSION AND RECOMMENDATIONS

Conclusion

This study identified a deep, simple, rapid, and successful learning approach that assisted doctors in making judgments on whether MRI images contained a brain tumor, a common challenge encountered by doctors.

An instructional paradigm that addresses this issue has been developed. Python has several benefits, including being free, open source, and providing additional datasets and visual packages. Python code is more readable than MATLAB code and offers more versatility in terms of code structure.

We employed a model we built from the base, and three pre-build deep learning models: InceptionV3, vgg16, and ResNet50, to train our dataset on a brain tumor. A model is constructed to train a dataset on brain tumors using three previously trained deep learning models and a newly created model (InceptionV3, VGG16, and ResNet50).

A validation accuracy of 98.00 percent was achieved, while a training accuracy of 99.94 percent was achieved. A brain tumor may be detected using the findings that training and validation accuracy were improving. Despite my best efforts, I encountered several issues in my study where tumor could not be discovered or were incorrectly identified. This dataset, as well as the images inside it, will be my primary focus. Consequently, I want to experiment with alternative deep learning approaches in the future to improve my results.

Recommendations

There will be many more chances for me to improve my research in the future. To begin, further improve accuracy, more conventional classifiers may be used.

Secondly, more images may be added. The model is better trained when there are more images to choose from. I want to do further work in the future using 3D images.

After detecting the tumor, I will attempt to determine if it is benign or malignant. Finally, many more versions of deep learning approaches may be evaluated.

REFERENCES

A. Rajendran, R. D. (2011). MRI brain image segmentation using modified fuzzy C-means clustering algorithm. *International Conference on Commnication Systems and Network Technologies. 2011*

A. Sankari, S. V. (23-24 March 2017). Automatic tumor segmentation using convolutional neural networks. *2017 Third International Conference on Science Technology Engineering & Management (ICONSTEM).* Chennai, India: IEEE.

Al-antari, M. A.-m.-T.-M.-S. (2018, April 5). *A fully integrated computer-aided diagnosis system for digital X-ray mammograms via deep learning detection, segmentation, and classification*. Retrieved from https://www.sciencedirect.com/science/article/abs/pii/S1386505618302880?via%3Dihub: https://doi.org/10.1016/j.ijmedinf.2018.06.003

Anam Mustaqeem, A. J. (2012). Efficient Brain Tumor Detection Algorithm Using Watershed & Thresholding Based Segmentation. *I.J. Image, Graphics and Signal Processing*, (pp. 10,34-39).

Andrzej Brodzicki, J. J.-K. (2020, September 30). *Pre-Trained Deep Convolutional Neural Network for Clostridioides Difficile Bacteria Cytotoxicity Classification Based on Fluorescence Images*. Retrieved from https://www.mdpi.com/1424-8220/20/23/6713/htm: https://doi.org/10.3390/s20236713

Avanzo, M. S. (2017). Beyond imaging: The promise of radiomics. *Physica Medica*, 38, 122-139.

B. Devkota, A. A. (2018). Image Segmentation for Early Stage Brain Tumor Detection using Mathematical Morphological Reconstruction,". *6th International Conference on Smart Computing and Communications,* (pp. 125, 115-123). Kurukshetra, India: ICSCC 2017.

B. Devkota, A. A. (7-8 December 2017). mage Segmentation for Early Stage Brain Tumor Detection using Mathematical Morphological Reconstruction. *6th International Conference on Smart Computing and Communications,* (pp. 115-123). Kurukshetra, India.: ICSCC 2017.

Bouchard, L. (2020, Nov 25). *Introduction to Convolutional neural networks (CNNs) | The most popular deep learning architecture*. Retrieved from Medium: https://medium.com/what-is-artificial-intelligence/introduction-to-

convolutional-neural-networks-cnns-the-most-popular-deep-learning-architecture-b938f62f133f

Brownlee, J. (2020, Aug 20). *A gentle introduction to the rectified linear unit (ReLU)*. Retrieved from Machine Learning Mastery: https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/

Camilus, S. (2009, Oct 09). *Fuzzy C-means segmentation*. Retrieved from MathWorks - Makers of MATLAB and Simulink - MATLAB & Simulink: https://www.mathworks.com/matlabcentral/fileexchange/25532-fuzzy-c-means-segmentation,

Camilus, S. (2022, August 11). *Fuzzy C-means segmentation. MathWorks - Makers of MATLAB and Simulink* . Retrieved from MATLAB & Simulink: https://www.mathworks.com/matlabcentral/fileexchange/25532-fuzzy-c-means-segmentation

Chetana Srinivas, N. P. (2022, March 7). *Deep Transfer Learning Approaches in Performance Analysis of Brain Tumor Classification Using MRI Images*. Retrieved from www.hindawi.com: https://www.hindawi.com/journals/jhe/2022/3264367

Clinic, M. (2022, August 11). *Tests and procedures - Tests and procedures - Mayo Clinic*. Retrieved from Mayo Clinic - Mayo Clinic: https://www.mayoclinic.org/tests-procedures

CS231n. (2022). *CS231n Convolutional Neural Networks for Visual Recognition*. Retrieved from CS231n Convolutional Neural Networks for Visual Recognition: https://cs231n.github.io/convolutional-networks/

CS231n Convolutional neural networks for visual recognition. (n.d.). Retrieved from CS231n Convolutional Neural Networks for Visual Recognition: https://cs231n.github.io/convolutional-networks/

Developers, G. (2022, july 18). *Multi-Class Neural Networks: Softmax*. Retrieved from Google Developers: https://developers.google.com/machine-learning/crash-course/multi-class-neural-networks/softmax

Dimililer, K. &. (2021, February 20). *Image Enhancement in Healthcare Applications: A Review. . Artificial Intelligence and Machine Learning for COVID-19, 111–140*. Retrieved from https://doi.org/10.1007/978-3-030-60188-1_6: https://doi.org/10.1007/978-3-030-60188-1_6

Dimililer, K. A. (2016, Aug 2). *Effect of Image Enhancement on MRI Brain Images with Neural Networks.* Retrieved from 12th International Conference on Application of Fuzzy Systems and Soft Computing, ICAFS: https://www.sciencedirect.com/science/article/pii/S1877050916325479

Dina Aboul Dahab, S. S. (2012.). Automated Brain Tumor etection and Identification using Image Processing and Probabilistic Neural Network Techniques., (pp. IJIPVC, Vol. 1, No. 2, pp. 1-8,).

Ergey Ioffe, C. S. (2015, Feb 11). *Batch normalization: Accelerating deep network training by reducing internal Covariate shift*. Retrieved from arXiv.org: https://arxiv.org/abs/1502.03167

Freedman, D., Pisani, R., & Purves, R. (2007-02-20). Statistics. *Statistics Fourth International Student Edition*, ISBN.

Gao Huang, Z. L. (2017). *CVPR 2017 open access repository*. Retrieved from CVF Open Access: https://openaccess.thecvf.com/content_cvpr_2017/html/Huang_Densely_Connected_Convolutional_CVPR_2017_paper.html

Hagyeong Lee, J. S. (30, Nov 2019). Introduction to convolutional neural network using Keras; an understanding from a statistician. *Communications for Statistical Applications and Methods*, Vol. 26, No. 6, 591–610. Retrieved from Communications for Statistical Applications and Methods2019, .

He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep Residual Learning for Image Recognition.* Retrieved from Computer Vision and Pattern Recognition : arXiv:1610.02357

Heaton, J. I. (2017). Genetic Programming and Evolvable Machines. *Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning*, 19, 305-307. Retrieved from Genetic Programming and Evolvable Machines: 10.1007/s10710-017-9314-z

Ilhan. (2017). Brain tumor segmentation based on a new threshold approach. *Procedia Computer Science* (pp. 580-587). https://doi.org/10.1016/j.procs.2017.11.282.

Ismail Saied, H. F. (2022, August 11). *K-means clustering algorithm for myocardial infarction classification*. Retrieved from Annals of Advanced Biomedical Sciences: https://doi.org/10.23880/aabsc-16000113

J. Akeret, C. C. (2017). Full length article: Radio frequency interference. *Astronomy and Computing*, 18:35 – 39,. Retrieved from Astronomy and Computing,.

Jan Egger, A. P. (2021, Nov 17). *Deep learning—a first meta-survey of selected reviews across scientific disciplines, their commonalities, challenges and research impact*. Retrieved from PeerJ Computer Science: https://peerj.com/articles/cs-773/

Kamil Dimililer, E. T. (2022, September 25). *Machine learning for expert systems in data analysis*. Retrieved from Scholar.google.com. : Scholar.google.com.

Kamil Dimililer, E. T.-E. (2019, 10 11). *A Pre-study on the Layer Number Effect of Convolutional Neural Networks in Brain Tumor Classification.* Retrieved from 2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT): https://doi.org/10.1109/INISTA52262.2021.9548599

Kamil Dimililer, E. Y. (August 2016,). Intelligent eye tumour detection system. *12th International Conference on Application of Fuzzy Systems and Soft Computing, ICAFS*, (pp. 325 – 332). Vienna, Austria. Retrieved from 12th International Conference on Application of Fuzzy Systems and Soft Computing, ICAFS .

Karen Simonyan, A. Z. (2014, Sep 4). *Very deep Convolutional networks for large-scale image recognition*. Retrieved from arXiv.org: https://arxiv.org/abs/1409.1556

Khurram Shehzad, I. S. (2016). Efficient Brain Tumor Detec tion Using Image Processing Techniques. *IEEE International Journal of Scientific & Engineering Research*.

Khurram Shehzad, I. S. (2016, December). *Efficient Brain Tumor Detec- tion Using Image Processing Techniques*. Retrieved from Cancer Research UK International Journal of Scientific & Engineering Research: https://www.cancerresearchuk.org

Kingma, J. B. (2014, December 22). *Adam: A method for stochastic optimization*. Retrieved from arXiv.org: https://arxiv.org/abs/1412.6980v8?hl=ja

Leon A. Gatys, A. S. (2015, Aug 26). *A neural algorithm of artistic style*. Retrieved from arXiv.org: https://arxiv.org/abs/1508.06576

M. Frid-Adar, I. D. (2018). *GANbased Synthetic Medical Image Augmentation for increased CNN Performance in Liver Lesion*. Retrieved from https://lup.lub.lu.se/luur/download?func=downloadFile&recordOId=8952747 &fileOId=8952748

McKinney. (2004, June 1). *Brain tumours: Incidence, survival, and aetiology PubMed Central (PMC).* . Retrieved from (PMC). https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1765660/

Mrs.A.Sankari, D. (2017). *Automatic Tumor Segmentation Using Convolutional Neural Networks.* Third International Conference on Science Technology Engineering & Management (ICONSTEM).

Nitish Srivastava, G. H. (2022, August 11). *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*. Retrieved from https://jmlr.org/: https://jmlr.org/papers/v15/srivastava14a.html#:~:text=Dropout%20is%20a%20technique%20for,%C3%A2%E2%82%AC%C5%93thinned%C3%A2%E2%82%AC%20networks.

Nunn, L. S., Silverstein, A., & Silverstein. (2006). *Cancer. Brookfield, Conn: Twenty-First Century Book. pp. 11-12.* Virginia: ISBN 0-7613-2833-5.

O. Ronneberger, P. F. (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*.

Overflow, S. (2022, August 11). *Segmenting an object from background using MATLAB based on feature points. (n.d.).* Retrieved from Stack Overflow: https://stackoverflow.com/questions/44179793/segmenting-an-object-from-background-using-matlab-based-on-feature-points/44180904#44180904

P, D. H. (Mar 5th - 7th, 2019.). A practical approach to Convolutional Neural Networks . *Inverted CERN School of Computing, .*

Radiologic and Clinical Outcomes with Special Reference to Tumor Involvement Pattern after Stent Placement for Malignant Bronchial Obstructions . (2022, August 10). Retrieved from Scientific Figure on ResearchGate: https://www.researchgate.net/figure/Example-of-type-III-tumor-involvement-A-49-year-old-man-with-lung-cancer-A-A-selective_fig3_38042515

Ravi Srisha, A. K. (December 2013). Morphological Operations for Image Processing : Understanding and its Applications. *National Conference on VLSI, Signal processing & Communications.*

Rohini Paul Joseph, C. S. (March, 2014). BRAIN TUMOR MRI IMAGE SEGMENTATION AND DETECTION. *IJRET: International Journal of Research in Engineering and Technology,* eISSN: 2319-1163 | pISSN: 2321-7308.

Rosebrock, A. (2022, August, 11). *Watershed OpenCV*. Retrieved from PyImageSearch: https://pyimagesearch.com/2015/11/02/watershed-opencv/

Rosebrock, A. (8 July 2019). Keras to Kubernetes®. *Deep learning using Keras*, 111-129.

Saito, T. &. (2017, Sep 13). *Basic evaluation measures from the confusion matrix*. Retrieved from Classifier evaluation with imbalanced datasets: https://classeval.wordpress.com/introduction/basic-evaluation-measures/

Shasidhar, M. R. (2011). MRI Brain Image Segmentation Using Modified Fuzzy C-Means Clustering Algorithm,. *nternational Conference on Communication Systems and Network Technologies,*, pp. 473-478. Retrieved from pp. 473-478: https://doi.org/10.1109/csnt.2011.102

Strayer, D. L., Rubin, R., & Rubin, E. (2008). *Rubin's pathology, Clinicopathologic foundations of medicine : The American Journal of surgical pathology*. Retrieved from LWW: https://journals.lww.com/ajsp/Citation/2008/09000/Rubin_s_Pathology,_Clinicopathologic_Foundations.20.aspx

Subbarao, M. &. (1995). Accurate recovery of three-dimensional shape from image focus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17,3, 266=274.

surv 5yr age brain 0.pdf. (2022, August 10). Retrieved from wcrf: https://www.wcrf.org/int/cancer-facts-figures/worldwide-data,World

Vrushali Borase, G. N. (1 Jan. 2017). Brain MR Image Segmentation for Tumor Detection using Artificial Neural. *International Journal Of Engineering And Computer Science ISSN: 2319-7242* (pp. 20160-20163 Volume 6). Dombivli ,India: http://www.ijecs.in/index.php/ijecs.

Weston, J. R. (2008). Deep learning via semi-supervised embedding. *Proceedings of the 25th international conference on Machine learning - ICML '08*, 10.1145/1390156.1390303.

What is the difference between MRI scans. (2022, August 11). Retrieved from researchgate: https://www.researchgate.net/post/What is the difference between MRI scans, Accessed:

Yantao, S. Y. (2016). A novel brain tumor segmentation from multi-modality MRI via a level-set-Based model. *Journal of Signal Processing Systems*, (pp. 87, 2,249-2

Appendix A

Ethical Approval Document

ETHICAL APPROVAL DOCUMENT

Date: 22/09/2022

To the Institute of Graduate Studies.

For the thesis project entitled **"BRAIN TUMOR DETECTION USING CONVOLUTIONAL NEURAL NETWORK"** the researcher declare that they did not collect any data from human/animal or any other subjects. Therefore, this project does not to go through the ethics committee evaluation.

Title: Assoc. Prof. Dr.

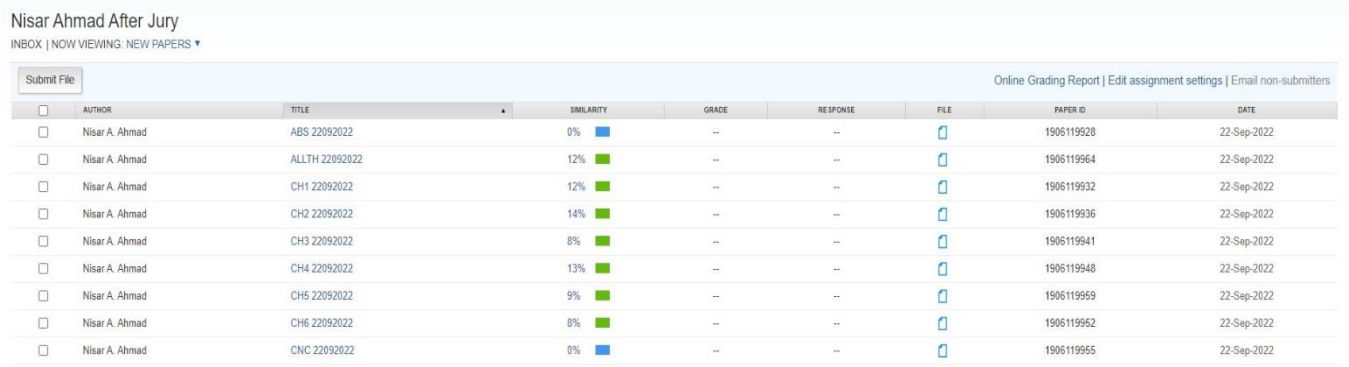
Name: Kamil DİMİLİLER

Signature:

Role in the Research Project: Supervisor

Appendix B

Similarity Report



Nisar Ahmad After Jury

INBOX | NOW VIEWING: NEW PAPERS ▾

Submit File

Online Grading Report | Edit assignment settings | Email non-submitters

	AUTHOR	TITLE ▲	SIMILARITY	GRADE	RESPONSE	FILE	PAPER ID	DATE
☐	Nisar A. Ahmad	ABS 22092022	0%	--	--		1906119928	22-Sep-2022
☐	Nisar A. Ahmad	ALLTH 22092022	12%	--	--		1906119964	22-Sep-2022
☐	Nisar A. Ahmad	CH1 22092022	12%	--	--		1906119932	22-Sep-2022
☐	Nisar A. Ahmad	CH2 22092022	14%	--	--		1906119936	22-Sep-2022
☐	Nisar A. Ahmad	CH3 22092022	8%	--	--		1906119941	22-Sep-2022
☐	Nisar A. Ahmad	CH4 22092022	13%	--	--		1906119948	22-Sep-2022
☐	Nisar A. Ahmad	CH5 22092022	9%	--	--		1906119959	22-Sep-2022
☐	Nisar A. Ahmad	CH6 22092022	8%	--	--		1906119952	22-Sep-2022
☐	Nisar A. Ahmad	CNC 22092022	0%	--	--		1906119955	22-Sep-2022

Assoc. Prof. Dr. Kamil DİMİLİLER

Supervisor

Appendix C

Code

```
Setting up the Environment
import numpy as np
import pandas as pd
import imutils
import os
from os import listdir
import tensorflow as tf
from keras.preprocessing.image import ImageDataGenerator
import cv2
import matplotlib.pyplot as plt
%matplotlib inline
from tensorflow.keras.models import Model,load_model
from tensorflow.keras.layers import
Conv2D,Input,ZeroPadding2D,BatchNormalization,Flatten,Activation,Dense,MaxPo
oling2D
from sklearn.model_selection import train_test_split
from sklearn.utils import shuffle #shuffling the data improves the model

Making directory for augmented images
os.makedirs('./output/kaggle/working/augmented-images', exist_ok = True)
os.makedirs('./output/kaggle/working/augmented-images/yes', exist_ok = True)
os.makedirs('./output/kaggle/working/augmented-images/no', exist_ok = True)
augmented_data_path ='./output/kaggle/working/augmented-images/'
augmented_yes =augmented_data_path+'yes'
augmented_no = augmented_data_path+'no'
IMG_SIZE = (224,224)
```