



**Department of Electrical and Computer Engineering
North South University**

Directed Research

BRAIN TUMOR DETECTION USING NEURAL NETWORK AND IMAGE PROCESSING

SHAMS SANIAT	ID# 1821848042
MD. RAFSAN KHAN	ID# 1812403042
MD. SHARUKH FARDIN	ID# 1821463042
ANIKA FERDAUSY	ID# 1512232042

Faculty Advisor:
MR. RIFAT AHMED HASSAN
Lecturer
ECE Department

Spring, 2023

LETTER OF TRANSMITTAL

July, 2023

To

Dr. Rajesh Palit
Chairman,
Department of Electrical and Computer Engineering
North South University, Dhaka

Subject: **Submission of Directed Research on “Brain Tumor Detection Using Neural Network And Image Processing”**

Dear Sir,

With due respect, we would like to submit our **Directed Research** on “**Brain Tumor Detection Using Neural Network And Image Processing**” as a part of our BSc program. The report deals with Brain Tumor. This project was very much valuable to us as it helped us gain experience from Deep Learning Techniques and apply in medical field. We tried to the maximum competence to meet all the dimensions required from this report.

We will be highly obliged if you kindly receive this report and provide your valuable judgment. It would be our immense pleasure if you find this report useful and informative to have an apparent perspective on the issue.

Sincerely Yours,

.....
Shams Saniat
ECE Department
North South University, Bangladesh

.....
Md. Rafsan Khan
ECE Department
North South University, Bangladesh

.....
Md. Sharukh Fardin
ECE Department
North South University, Bangladesh

.....
Anika Ferdausy
ECE Department
North South University, Bangladesh

APPROVAL

Shams Saniat (ID # 1821848042), Md. Rafsan Khan (ID # 1812403042), Md. Sharukh Fardin (ID # 1821463042) and, Anika Ferdausy (ID # 1512232042) from Electrical and Computer Engineering Department of North South University, have worked on the Directed Research titled “Brain Tumor Detection Using Neural Network And Image Processing” under the supervision of Mr. Rifat Ahmed Hassan partial fulfillment of the requirement for the degree of Bachelors of Science in Engineering and has been accepted as satisfactory.

Supervisor’s Signature

.....

Mr. Rifat Ahmed Hassan

Lecturer

Department of Electrical and Computer Engineering

North South University

Dhaka, Bangladesh.

Chairman’s Signature

.....

Dr. Rajesh Palit

Professor

Department of Electrical and Computer Engineering

North South University

Dhaka, Bangladesh.

DECLARATION

This is to declare that this project is our original work. No part of this work has been submitted elsewhere partially or fully for the award of any other degree or diploma. All project related information will remain confidential and shall not be disclosed without the formal consent of the project supervisor. Relevant previous works presented in this report have been properly acknowledged and cited. The plagiarism policy, as stated by the supervisor, has been maintained.

Students' names & Signatures

1. Shams Saniat

2. Md. Rafsan Khan

3. Md. Sharukh Fardin

4. Anika Ferdausy

ACKNOWLEDGEMENTS

We would like to express our heartfelt gratitude towards our project and research supervisor, Mr. Rifat Ahmed Hassan, Lecturer, Department of Electrical and Computer Engineering, North South University, Bangladesh, for his invaluable support, precise guidance and advice pertaining to the experiments, research and theoretical studies carried out during the course of the current project and also in the preparation of the current report.

Furthermore, we would like to thank the Department of Electrical and Computer Engineering, North South University, Bangladesh for facilitating the research. We would also like to thank our loved ones for their countless sacrifices and continual support.

ABSTRACT

Brain Tumor Detection Using Neural Network And Image Processing

Brain tumors are a major cause of death and disability worldwide. Early detection and treatment of brain tumors is essential for improving patient outcomes. A brain tumor is an abnormal cell development in the brain, some of which can progress to cancer. Tumors are abnormal cell growths in the brain, and cancer is a term used to describe malignant tumors. CT or MRI scans are commonly used to detect cancerous areas in the brain. Timely and prompt disease detection and treatment plan leads to improved quality of life and increased life expectancy in the brain tumor patients. Magnetic resonance imaging (MRI) is a widely used imaging technique for detecting brain tumors. However, manual analysis of MRI images is time-consuming and error-prone. Automated analysis can be a great time efficient approach to detect tumor cell from MRI images.

In this paper, we propose a deep learning-based approach for brain tumor detection from MRI images. We use a convolutional neural network (CNN) model, to learn features from MRI images. Our CNN model has 13 convolutional layers, followed by 3 dense layers. The convolutional layers use 2D convolutions to extract features from the input image, while the dense layers classify the extracted features. We train the model on a dataset of MRI images with and without brain tumors. We evaluate the performance of the model on a test set of MRI images. The CNN model achieves a good accuracy on the test set. Our results suggest that deep learning can be used to develop accurate and efficient methods for brain tumor detection from MRI images.

TABLE OF CONTENTS

LETTER OF TRANSMITTAL	2
APPROVAL	4
DECLARATION	5
ACKNOWLEDGEMENTS	6
ABSTRACT	7
LIST OF FIGURES	10
LIST OF TABLES	11
Chapter 1 Introduction	12
1.1 Background and Motivation	12
1.2 Purpose and Goal of the Project	12
1.3 Organization of the Report	13
Chapter 2 Research Literature Review	14
2.1 Existing Research and Limitations	14
Chapter 3 Methodology	18
3.1 Dataset	18

3.2 Data preprocessing	19
3.3 Data Augmentation	21
3.4 Convolutional Neural Network	22
3.5 Transfer Learning	23
Chapter 4 Tools and Libraries.....	24
Chapter 5 Investigation/Experiment, Result, Analysis and Discussion.....	26
5.1 Introduction	26
5.2 Experiment	26
5.2.1 Epochs.....	26
5.2.2 Parameters.....	26
5.2.3 Hyperparameters	27
5.2.4 Optimizer	28
5.2.5 Learning rate	28
5.3 Technique	29
5.3.1 Accuracy of the model	30
5.3.2 Loss of the model.....	30
5.4 Discussion and Future improvements	32
Chapter 6 Conclusions	33
6.1 Summary	33
6.2 Limitations	33
6.3 Future Improvement.....	33
References	34

LIST OF FIGURES

Figure 1: Sample of MRI images in the dataset.....	18
Figure 2: Classification pipeline.....	20
Figure 3: Augmented images.....	21
Figure 4: Generic CNN model architecture.....	22
Figure 5: Techniques.....	29
Figure 6: Accuracy of the model.....	30
Figure 7: Loss of the model.....	31

LIST OF TABLES

Table 1: Tools we used.....	24
Table 2: Libraires we used in codes.....	27
Table 3: Hyperparameter.....	24

Chapter 1 Introduction

1.1 Background and Motivation

A Tumor is an abnormal cell mass that forms in the body. There are around 200 different types that can harm humans. According to the American Cancer Society, a brain tumor is a severe condition in which irregular brain tissue growth compromises brain function. Brain tumors are abnormal growths of cells in the brain. They can be either benign (non-cancerous) or malignant (cancerous). Malignant brain tumors are the most common type of brain tumor, and they can be very aggressive. The symptoms of a brain tumor can vary depending on the size and location of the tumor. Some common symptoms include headaches, nausea, vomiting, seizures, and vision problems. The diagnosis of a brain tumor is usually made with a combination of MRI, CT scan, and biopsy. According to the National Brain Tumor Foundation (NBTF), the number of people who have died as a result of brain tumors has increased by 300 percent in the last three decades. Untreated brain tumors can result in death. The complexity of brain tumors makes it difficult for healthcare personnel to diagnose and treat affected patients. Early identification and treatment of brain tumors are critical factors in these individuals' survival rates [1]. Brain tumor biopsy is more difficult than biopsy of other regions of the body since it requires surgery. As a result, another way for precise diagnosis that does not require surgery is critical. Magnetic Resonance Imaging (MRI) is the most effective and widely used method for diagnosing brain malignancies. Brain tumor biopsy is more difficult than biopsy of other regions of the body since it requires surgery. As a result, another way for precise diagnosis that does not require surgery is critical. Magnetic Resonance Imaging (MRI) is the most effective and widely used method for diagnosing brain malignancies [2]. It provides detailed images of the brain, and it can be used to identify the size, location, and type of tumor.

1.2 Purpose and Goal of the Project

We are at the starting point of the generation of data science. Machine learning and deep learnings are getting highly used in our daily life. In the medical sector we can see many great examples. Automated disease detection is a key aspect in which researchers of data science investing a lot of their time. Recent advancements in machine learning, particularly deep learning, have resulted in

the recognition and classification of medical imaging patterns. The ability to retrieve and extract knowledge from data rather than learning from experts or scientific publications is one example of success in this area. Machine learning is increasingly becoming a useful tool for enhancing performance in a variety of medical applications, including disease prognosis and diagnosis, identification of molecular and cellular structures, tissue segmentation, and image categorization. Convolutional Neural Networks (CNNs) are the most successful image processing algorithms now in use, as they have many layers and high diagnostic accuracy when the quantity of input images is large [3]. Autoencoders are an unsupervised learning method that uses neural networks to learn representations. Surprisingly, multiple deep learning and machine learning algorithms have been utilized to diagnose cardiovascular stenosis and identify cancers (such as lung tumors). Furthermore, performance tests have revealed that they have a high diagnostic accuracy [4]. Many studies on the identification of brain tumors using various approaches and models have been undertaken. However, some of these researches had flaws, such as a lack of a performance comparison between the suggested model and established machine learning approaches. In one investigation, the proposed model necessitated sophisticated computations. Without adding healthy patients, the bulk of relevant studies produced models for identifying three categories of brain tumors.

In this paper, we present a brain tumor detection method using CNN to develop a model that will detect brain tumors and classify the detected tumors into three different categories. The three categories are glioma, meningioma and pituitary tumor. We trained the CNN model on a dataset of MRI brain images, and we evaluated the performance of the model on a test dataset. If the MRI scan finds no tumor cell our model will clarify that there's no tumor in the cell. Our dataset consists of around 500 images.

1.3 Organization of the Report

Chapter 1 Presents the **Introduction** which is this chapter. In the following chapters Chapter 2 Presents **Literature Review**, Chapter 3 Presents **Methodology**, Chapter 4 Presents **Result, Analysis and Discussion**, Chapter 5 Presents **Conclusion and Future Work**.

Chapter 2 Research Literature Review

2.1 Existing Research and Limitations

A CNN was utilized to classify glioma, meningioma, and pituitary tumors in a study undertaken by Bada and Barjaktarovic' in 2020. This study's network architecture included an input layer, two blocks "A," two blocks "B," a classification block, and an output layer, for a total of 22 layers. The k-fold cross-validation approach was used to assess network performance. The best value obtained in this investigation for the tenfold cross-validation approach was 96.56%. This study included 3064 T1-weighted contrast-enhanced MRI images from China's Nanfang Hospital, General Hospital, and Tianjin Medical University [5].

In 2018, capsule algorithm networks (DCNet) and varied capsule networks (DCNet++) were established. DCNet is simply a deeper convolutional network that leads to the learning of separate feature maps. DCNet++ learns using a hierarchical architecture, which makes it more efficient for learning complex data. They classified a dataset of 3064 MRI pictures of 233 brain tumor patients using just images of three types of brain tumors; a dataset of healthy people was not used for classification. The DCNet model was created by reducing the initial eight convolutional layers to four layers with 16 kernels and training it using eightfold cross-validation. The DCNet algorithm test accuracy was 93.04%, and the DCNet++ algorithm accuracy was 95.03% [6].

Gumaei et al. developed an automated method to help radiologists and physicians identify distinct forms of brain tumors. The research was divided into three stages: preprocessing of brain images, extraction of brain features, and categorization of brain tumors. Using a min-max normalization technique, brain pictures were converted into intensity brain images in the range of [0, 1] during the preprocessing step. The PCA-NGIST approach (a mix of normalized GIST descriptor and PCA) was then used to extract features from MRI images. The tumor kinds were identified and classified using Regularized Extreme Learning Machine (RELM) classification in the final stage. The researchers employed Cheng's dataset in their study, which consisted of 3064 MRI pictures from 233 patients separated into two groups, 70% for training and 30% for classifier testing; a fivefold cross-validation procedure was applied. The accuracy was reported to be 94.23%. However, there was no comparative evaluation with other procedures in the study, which can be considered a study limitation procedure was applied. The accuracy was reported to be 94.23%.

However, there was no comparative evaluation with other procedures in the study, which can be considered a study limitation [7].

Pashaei et al. created many methods for detecting meningioma, glioma, and pituitary cancers. A CNN was employed in their model to extract hidden features from photos and select features. Four convolutional layers, four pooling layers, one fully connected layer, and four batch normalization layers were included in the suggested model. The authors employed 10 epochs with 16 iterations per epoch and a learning rate of 0.01 in this model. This study also made use of the dataset supplied by Cheng. The suggested model's performance was assessed using a tenfold cross-validation method, with 70% and 30% of the data used for training and 30% used for system testing, respectively. The suggested method was compared to MLP, Stacking, XGBoost, SVM, and RBF in the study, and the findings showed that it was highly accurate (93.68%) [8].

In 2018, Abiwinanda used a CNN to diagnose the three most frequent types of brain cancers. The "adam" optimizer was utilized in the learning process, which is a stochastic optimization approach based on the stochastic gradient descent concept. The CNN was trained in the study using 3064 T-1 weighted CE-MRI images of brain tumors provided by Cheng. The collection includes 1426 meningioma photos, 708 glioma images, and 930 pituitary tumor images. Of the total number of photos available, 700 photographs from each class were used, 500 for the training phase and another 200 for the validation phase. All convolutional layers in the architectures employed 32 filters, ReLu as an activation function, the maxpool kernel size was 2 2, and all fully connected layers used 64 neurons in this model. The output layer contained three neurons and was activated using the softmax activation function. The best reported training and validation accuracy percentages were 98.51% and 84.19%, respectively [9].

Another study (2018) used CNNs to automatically diagnose brain cancers using magnetic resonance scans. The goal of this study was to distinguish between photographs of healthy brains and brain tumors. The diagnosis was made through a two-stage multi-model system. A CNN was used for preprocessing and feature selection in the first stage, and an Error-Correcting Output Codes Support Vector Machine (ECOC-SVM) was used for classification in the second stage. In

the first stage, three algorithms were used: AlexNet, VGG-16, and VGG-19, with AlexNet having the best performance (99.55% accuracy). For the brain tumor localization phase, BraTS (2013 dataset) was employed, and images taken from the standard Reference Image Database to Evaluate Response (RIDER) neuro MRI database were used for performance evaluation in the first phase [10].

Rehman et al. investigated three CNNs: AlexNet, GoogLeNet, and VGGNet. The major goal of the study was to differentiate three forms of brain tumors, meningioma, glioma, and pituitary, utilizing deep learning algorithms and MRI image processing. In the final phase, automated features were categorised using a linear classifier. To expand the sample size and limit the danger of over-fitting, data augmentation techniques were used. In comparison to other methods, the VGG16 technique had the best accuracy (98.69%), according to the evaluation data [11].

Medical picture segmentation is difficult because of the complexity of the images and the scarcity of anatomical models. A suggested technique leverages BWT techniques for precise and quick segmentation with little human involvement. The method helps neurosurgeons identify patients quickly and accurately, with 98.5% accuracy compared to cutting-edge technologies. In the future, accuracy can be increased by combining segmentation and extraction methods with real-time photos and clinical cases [12]. The output is accurate and clear because each step has been processed. The suggested system detects brain tumors using neighborhood pixels and a conventional neural network approach. The retrieved output is the best and most efficient when used on numerous photos [13]. For automatic segmentation and categorization of brain tumors in their early stages, the method uses 440 MRI scans. 80 HGG and 80 LGG photos are used in the second stage, while 100 normal and 180 aberrant tumorous images are used in the first stage. The approach reaches 99% accuracy, sensitivity, and specificity, and it may be used to diagnose additional brain conditions and other medical pictures [14]. The method clusters characteristics in MR images using hybrid K-means and SVM algorithms to find anomalies in the brain. With less training needed, this technology finds tumors more quickly and accurately. It delivers quicker tumor identification and requires less training, thanks to Python programming [15]. In order to increase the accuracy of brain image MRI, the research introduces a hybridized feature selection scheme. Using Principal Component Analysis to minimize dimensionality, a combined collection

of local and global features was chosen, and a Grey Level Covariance Matrix was then calculated. The classifier model was tested on test picture samples using 26 features, including 13 local and 13 global features. A maximum accuracy of 95% is indicated by the experimental findings. To increase the model's accuracy, future improvements will investigate various feature extraction methods and use hybrid features with various classifiers [16].

Chapter 3 Methodology

3.1 Dataset

The many TransferLearning (TL)-based algorithms as well as our model used in this work were trained, validated, and tested using the publicly available recent MRI dataset which was uploaded three years ago from the Kaggle repository. A medical imaging method called magnetic resonance imaging (MRI) is used in radiology to create excellent two dimensional- or three-dimensional brain and brainstem images. T1-weighted, contrast-enhanced pictures in two dimensions are included in the database. Three separate views are included in the dataset: axial, coronal, and sagittal views. This dataset includes 3264 various types of grayscale and JPG-formatted MRI scans of the human brain. The dataset shows four classifications of brain tumors which are Glioma (with 826 images for training as well as 100 for testing), Meningioma (with 822 images for training as well as 115 for testing), No-tumor (with 395 images for training as well as 105 for testing) and Pituitary (827 images for training as well as 110 for testing). Eighty percent of the images were used for the training task, while twenty percent were used for validation. To increase the number of photos to be used, however, and to give a sufficient input size for each distinct model, Resizing as well as Data Augmentation were performed to the dataset during the preprocessing stage. 4836 images were produced as a result, with the training dataset being 70% of the total, and the testing dataset comprising the remaining 30%. The table below presents the information related to the dataset.

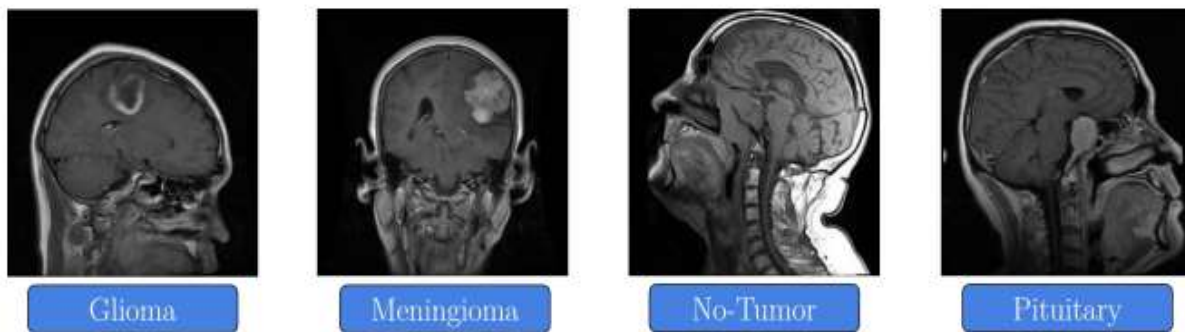


Figure1: Sample of MRI images in the dataset

3.2 Data preprocessing

Prior to feeding data to neural networks, we attempt to uniformize it at the pre-processing stage. The following pre-processing methods for the dataset are suggested by our proposed research: Resizing, Labeling, and also Data Augmentation. We reduced the size of our high-resolution photos to 224x224x3, which preserves all important information for the networks in smaller file sizes. In the MRIs, the center image of the brain was surrounded by a black background. If this black background were supplied or fed to neural networks, it would be useless information for categorization. The pictures were therefore cropped to include the primary contour. Here, the largest contour has been marked and chosen. Then, we locate the contour's extreme points and crop the picture to those endpoints. Thus, the majority of the undesired background as well as some noise from the original image were eliminated. For each picture in the dataset, this procedure is carried out. Contrarily, there have been instances when the contour mapping as well as cropping algorithm failed to accurately identify the correct contour and incorrectly cropped the image. Following the 'augmentation' step, these photos were manually eliminated because they produced distorted images. The models might not fit because of the extremely small amount of data that was collected. In order to expand the amount of data, we would employ the brilliant data augmentation technique. This method uses rotations, flips, changes in exposure, etc. to produce comparable images.⁷

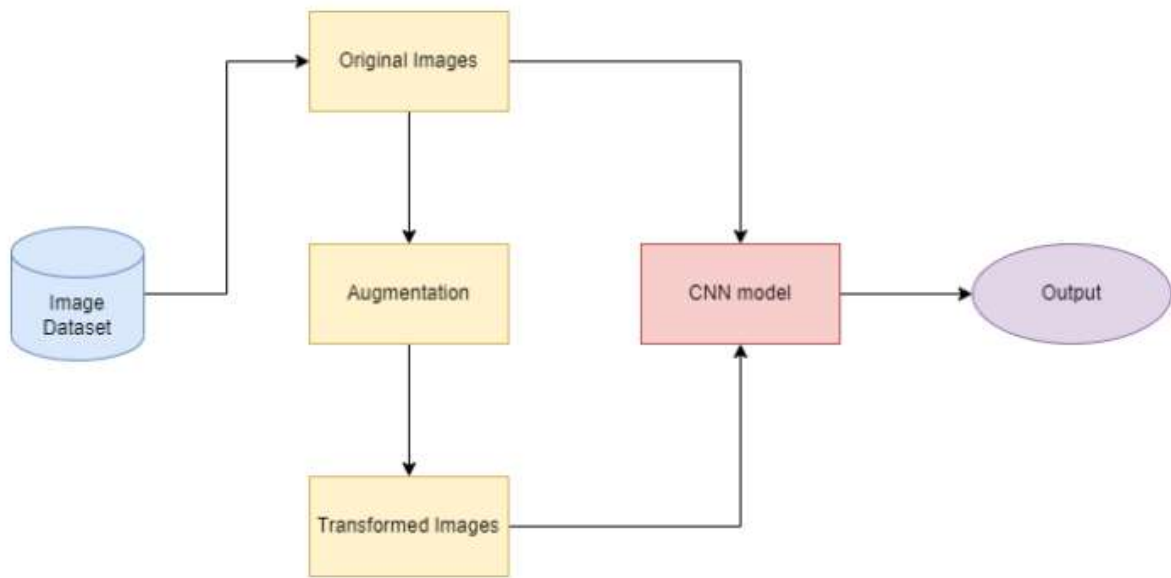


Figure 2: Classification pipeline

3.3 Data Augmentation

One of the most crucial data methodologies for tackling the problems of unequal distribution as well as data sparsity is data augmentation. This approach, which integrates geometric transformation operations including rotation, zoom or scale, brightness, has been employed in numerous research investigations on the classification of brain tumors. Here, to train and test a convolutional neural network, we have a limited number of images. The volume of data can be increased provided to a network to enable it to learn to a larger level. Data Augmentation was the method used to do this. To produce comparable photos, this technique makes use of rotations, flips, exposure changes, zooming, etc. Every picture of the dataset was subjected to the augmentation method. As a result, the number of photos rose by a factor of 21. Thus, the likelihood of the network being under-fitted can be decreased with such a large amount of data. The pictures were divided and jumbled in a 3:1 ratio. wherein 75% of the data were used for training and the remaining 25% for testing and validation. In the augmentation and pre-processing step, few photos were deformed. Because the contours were improperly cropped, these photos were deformed. As a result of their noise and obscurity, these photos weren't included in the final data set.

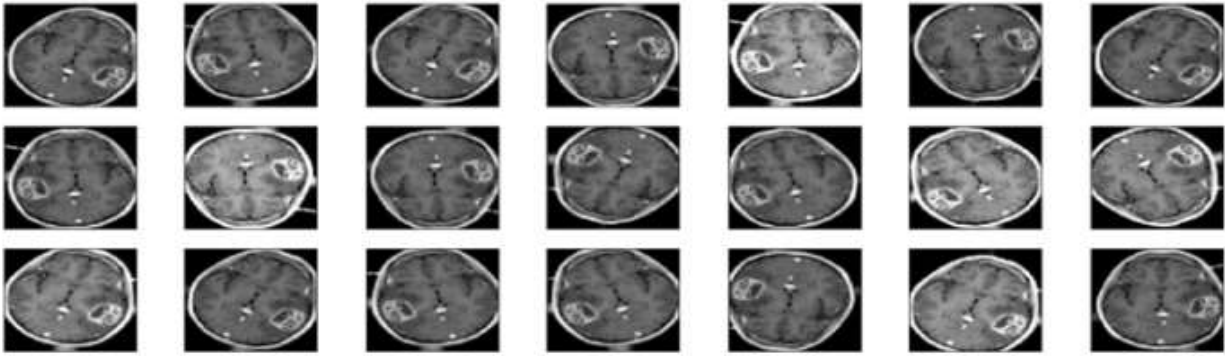


Figure 3: Augmented images

3.4 Convolutional Neural Network

One of the neural network variations that has demonstrated extraordinary accuracy in the field of image categorization is Convolutional Neural Networks (CNN). Convolutional layers, fully connected layers (dense layers), pooling layers, and normalizing layers are frequently seen in a CNN's hidden layers. A feature map is produced after a convolution layer scans the input image as well as applies several filters to it. These filters are all initialized at random and serve as our parameters that the network will later learn through backpropagation. By using these filters, the appropriate and pertinent features can be extracted from the incoming data. CNN extracts the spatial characteristics from an image. The arrangement of the pixels and their interactions within an image are referred to as spatial features. They aid in precisely identifying the object, pinpointing its location, and determining how it relates to other things in an image. Following that, a ReLU function is used to the feature map to help boost non-linearity. Then, each feature map is applied with a pooling layer, which gradually shrinks the spatial size of representation to lessen the amount of computation and parameters in the network. The pooled images are then flattened into a single, lengthy vector, which serves as the input for a fully linked artificial neural network. Additionally, the network operates just like an ANN, with output created on the top layer and error transmitted back through the network. This error then aids the network in modifying the weights to reduce the error in following epochs.

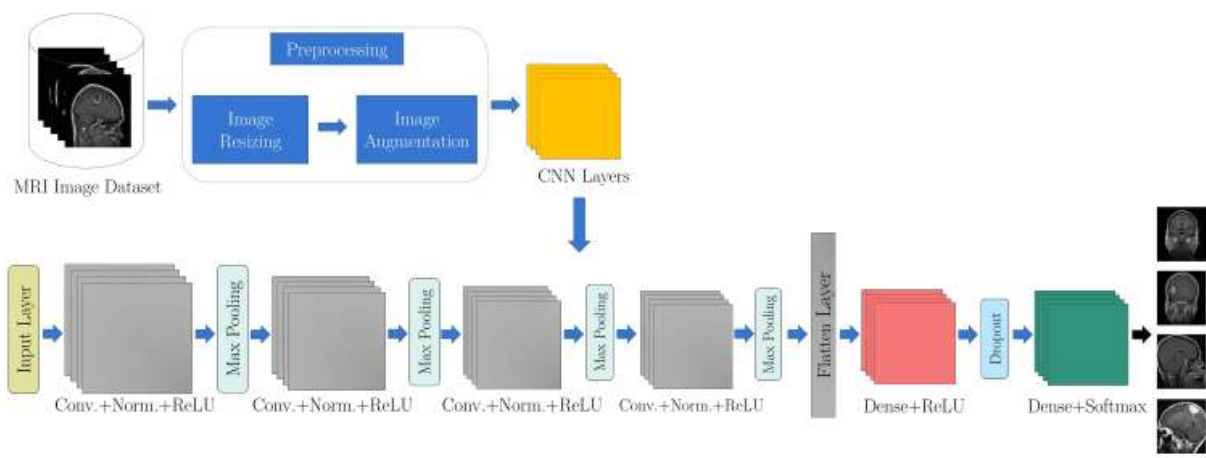


Figure 4: Generic CNN model architecture

3.5 Transfer Learning

Transfer learning is a method where a model that has been trained for one task is applied to another. For this paper, we trained a TL-model which is VGG16. University of Oxford professors K. Simonyan and A. Zisserman developed the CNN model known as VGG16. In ImageNet, a dataset of more than 14 million images divided into 1000 classes, this model achieves top-5 test accuracy of 92.7%. Images of (224,224,3) pixel dimensions are used as the model's input. Two of the several convolution layers receive these images initially. After that, we enter the second set of two convolution layers, which is followed by a ReLu as well as max-pooling. Once more, following ReLu and data pooling, add three sets of three layers of convolution. Next, there are three totally connected layers. When the output of the third fully connected layer is transferred to the softmax layer, the classification vectors are normalized.

Chapter 4 Tools and Libraries

Tools: We used Google Colab, and Google Drive for all our projects. We utilized Google Colab, to use the browser to develop and run any Python code.

Tools	Purpose
Google Colab	Training model and Code related work
Google drive	Storing and sharing data

Table1: Tools we used

Libraries:

Libraries	Purposes
Tensorflow	To compute tensor
cv2	To process image
tqdm	Progress bar visualizing.
ipywidgets	create interactive GUIs
Matplotlib	To plot graphs
Numpy	Math functions (Frame listing)
Keras	modeling in depth
Pandas	To ease data manipulation and analysis
Seaborn	Creating visually appealing statistical visuals

Table 2: Libraires we used in codes

Chapter 5 Investigation/Experiment, Result, Analysis and Discussion

5.1 Introduction

Using the `model.fit()` method, the code demonstrates that the model underwent 20 training epochs. The model's weights and biases were adjusted 20 times by the gradient descent algorithm and loss function throughout this process. Additionally, the code showcases the utilization of the `matplotlib` and `seaborn` libraries to plot the accuracy and loss of training and validation data for each epoch. Such visualizations are invaluable in assessing the model's efficiency and pinpointing any overfitting or underfitting.

5.2 Experiment

5.2.1 Epochs

Epochs refer to the number of times the model goes through the entire training dataset during the training process

The the model was trained for 20 epochs using the `model.fit()` method

This means that the model updated its weights and biases 20 times based on the gradient descent algorithm and the loss function

The code also shows how to plot the training and validation accuracy and loss for each epoch using `matplotlib` and `seaborn` libraries. These plots can help to monitor the model's performance and identify overfitting or underfitting issues

5.2.2 Parameters

Parameters refer to the variables that are learned by the model during the training process. These include the weights and biases of each layer in the CNN model

The code shows that the model has a total of 4,447,044 parameters, which can be seen in the model summary

The code also shows how to calculate the number of parameters for each layer using the formula:

For convolutional layers: $\text{parameters} = (\text{filter_height} * \text{filter_width} * \text{input_channels} + 1) * \text{output_channels}$

For dense layers: $\text{parameters} = (\text{input_units} + 1) * \text{output_units}$

For example, the first convolutional layer has 32 output channels and 3 input channels (RGB), and uses a 3x3 filter. Therefore, the number of parameters for this layer is $(3 * 3 * 3 + 1) * 32 = 896$

5.2.3 Hyperparameters

Hyperparameters in this code refer to the variables that are set before the training process and are not learned by the model. These include the learning rate, the batch size, the number of epochs, the dropout rate, and the filter size. The code shows that the following hyperparameters were used in this code:

Model	Epoch	Batch Size	Learning rate	Dropout rate	Filter size
CNN	20	32	0.0001	0.2	3x3

Table 3: Hyperparameter

5.2.4 Optimizer

Based on the loss function and the learning rate, the optimizer algorithm is what updates the model parameters.

The code efficiently utilizes the optimizer Adam, which is a renowned variant of gradient descent. Adam adapts the learning rate for each parameter and leverages momentum to expedite the convergence.

In the `model.compile()` method, the code specifies the optimizer and the learning rate by doing this:

```
model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=0.0001))
```

5.2.5 Learning rate

The learning rate of this page's code is the amount by which the model adjusts its parameters in each iteration based on the error gradient

The code shows that the learning rate used in this code is 0.0001, which is a relatively small value

The learning rate is specified as an argument in the Adam optimizer as follows:

```
loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True), metrics=['accuracy'])
```

5.3 Techniques

After using MRI Dataset, we got Noised and Denoised data. From these data we got Training and Testing set. And we got 89% and 95% from Noised and Denoised dataset consecutively.

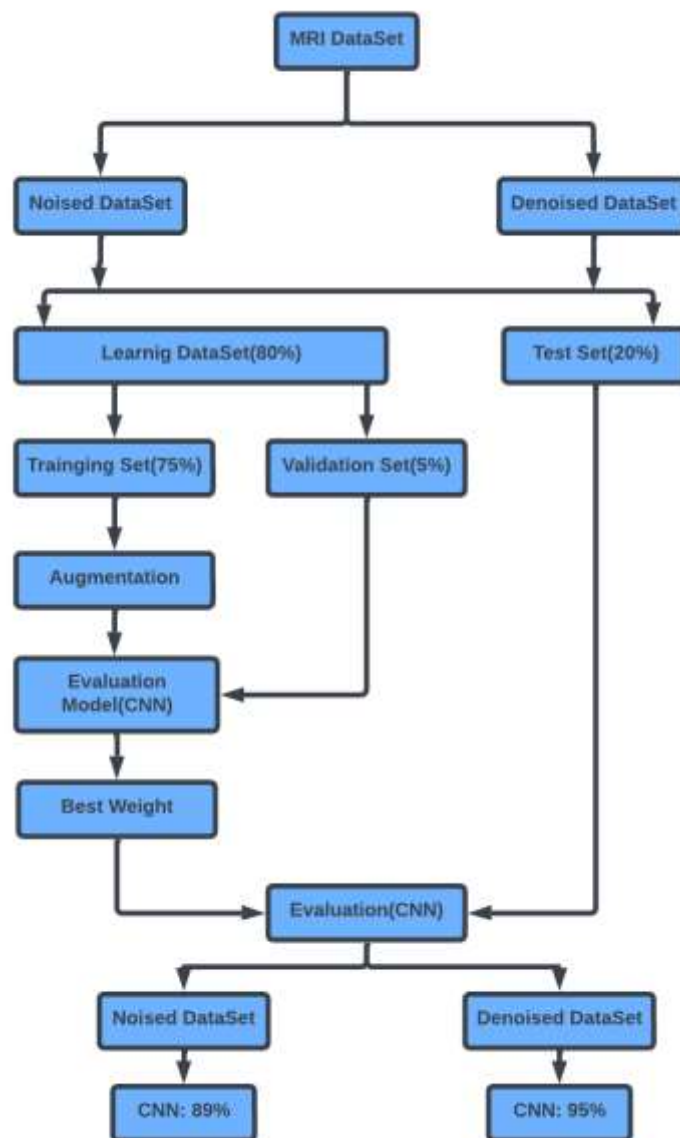


Figure 5: Techniques

5.3.1 Accuracy of the model

The accuracy of the model is the percentage of correct predictions on the test set. The model achieved an accuracy of 0.9519 on the training set and 0.8605 on the validation set after 20 epochs.

The accuracy on the test set can be calculated by using the `model.predict()` method on the `X_test` array and comparing the predicted classes with the true classes in the `y_test` array. It also shows how to plot a confusion matrix to visualize the performance of the model on each class.

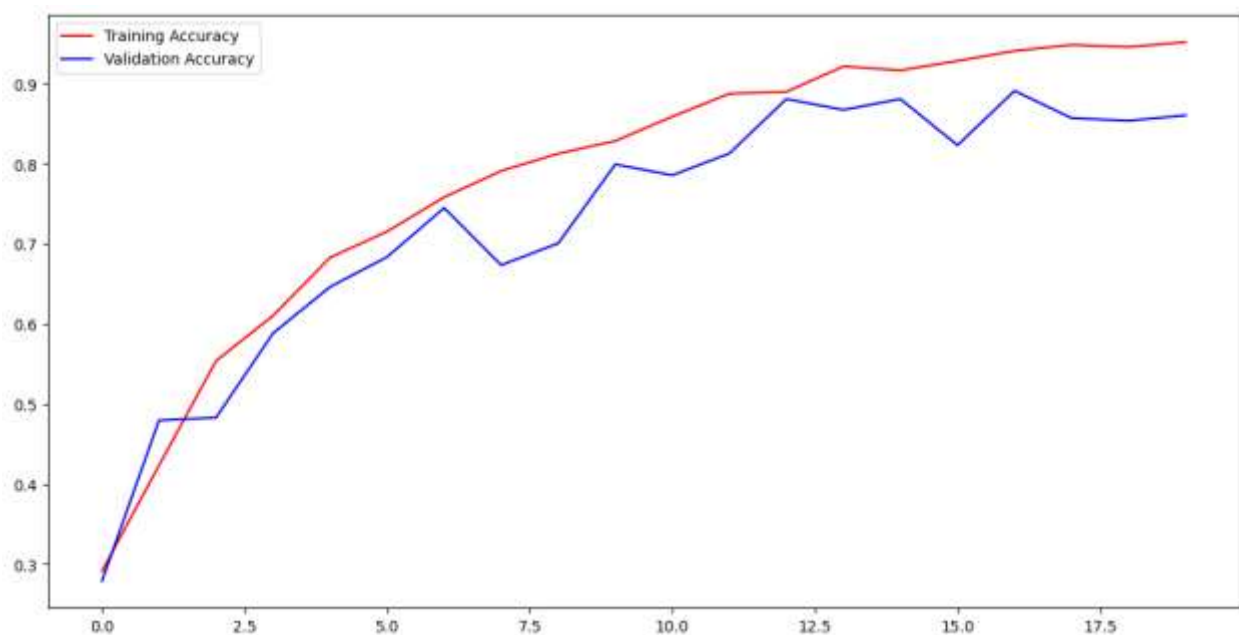


Figure 6: Accuracy of the model

5.3.2 Loss of the model

The loss of the model is the categorical crossentropy, which measures how well the model predicts the correct class for each image

The loss of the model on the training data and the validation data is plotted in the code as follows:

According to the plot, the loss of the model on the training data decreases from about 1.6 to 0.13 over 20 epochs, while the loss of the model on the validation data decreases from about 1.36 to 0.46 over 20 epochs

This means that the model is learning to fit the data well, but it may be overfitting, since the validation loss is higher than the training loss.

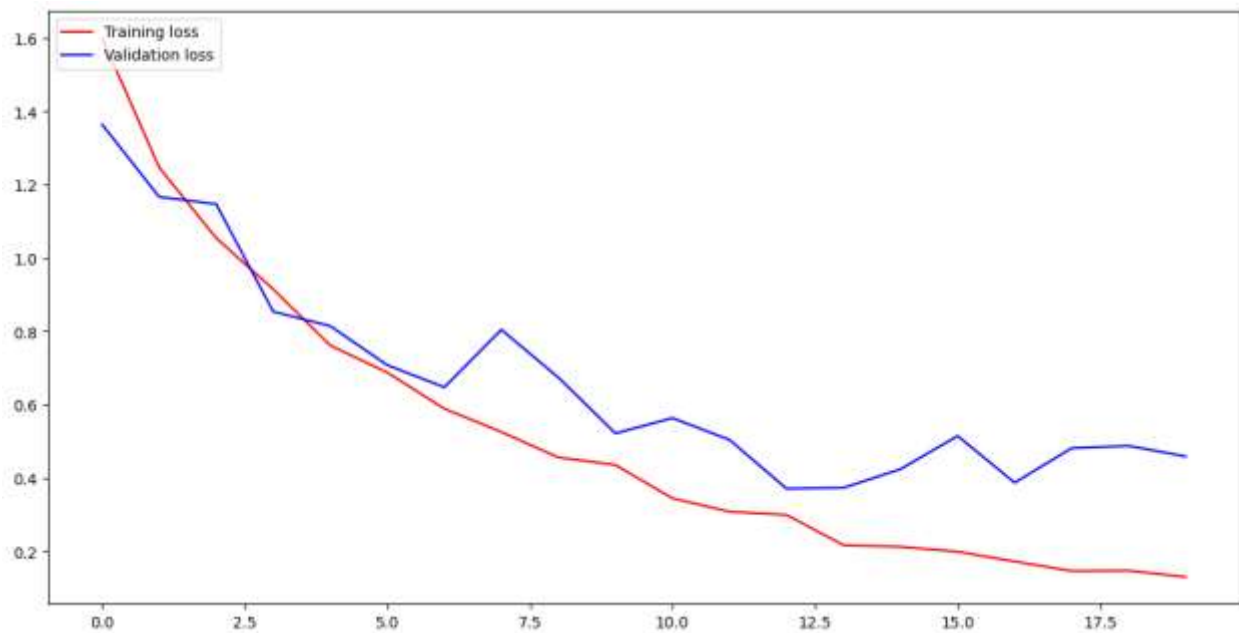


Figure 7: Loss of the model

5.4 Discussion and Future improvements

The model has become Overfitted. This means though the Training performs well Testing is not. In order to solve this issue we need to reduce the Epoch value. To do that we need to reduce the learning rate.

Chapter 6 Conclusions

6.1 Summary

Due to the intricacy of the pictures and the dearth of anatomical models that accurately represent the various deformations in each component, segmenting medical images is a difficult problem. In this paper, we proposed a method by which the detection procedure of the brain tumors can be reduced greatly. As brain tumors are a major cause behind cancer-related deaths, early detection of brain tumors is very much essential for improving the survival rates. We have used CNN and this model have been shown to be affective for brain tumor detection. The initial cluster size and cluster centers may be successfully managed using the suggested strategy. This discovery suggests a technique to divide the brain tissue that requires little any human intervention. The major objective of our suggested approach is to let human specialists or neurosurgeons quickly identify the patients. We believe that our method has the potential to be used in clinical settings to improve the early detection of brain tumors

6.2 Limitations

We can use more MRI data when training our model so that we can get more accurate training accuracy. One scenario can occur that is, our method was only tested on a relatively small dataset of MRI brain images. It is possible that the method would not perform as well on a larger dataset of images. The methods' computational time, system complexity, and memory space needs can all be further decreased.

6.3 Future Improvement

The same method may be used to identify and examine other disorders seen in other bodily organs, such as the kidney, liver, lungs, etc. Future research can use multiple classifiers that employ optimization approach to increase accuracy by combining more efficient segmentation and extraction algorithms with real-time pictures and clinical cases and a larger data set that includes a variety of scenarios.

References

1. J. R. McFaline-Figueroa and E. Q. Lee, “Brain tumors,” *Am. J. Med.*, vol. 131, no. 8, pp. 874–882, 2018.
2. E.-S. A. El-Dahshan, H. M. Mohsen, K. Revett, and A.-B. M. Salem, “Computer-aided diagnosis of human brain tumor through MRI: A survey and a new algorithm,” *Expert Syst. Appl.*, vol. 41, no. 11, pp. 5526–5545, 2014.
3. D. Shen, G. Wu, and H.-I. Suk, “Deep learning in medical image analysis,” *Annu. Rev. Biomed. Eng.*, vol. 19, no. 1, pp. 221–248, 2017.
4. K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, “Deep convolutional neural network for inverse problems in imaging,” *arXiv [cs.CV]*, 2016.
5. M. M. Badža and M. Č. Barjaktarović, “Classification of brain tumors from MRI images using a convolutional neural network,” *Appl. Sci. (Basel)*, vol. 10, no. 6, p. 1999, 2020.
6. S. S. R. Phaye, A. Sikka, A. Dhall, and D. Bathula, “Dense and Diverse Capsule Networks: Making the capsules learn better,” *arXiv [cs.CV]*, 2018.
7. A. Gumaei, M. M. Hassan, M. R. Hassan, A. Alelaiwi, and G. Fortino, “A hybrid feature extraction method with regularized extreme learning machine for brain tumor classification,” *IEEE Access*, vol. 7, pp. 36266–36273, 2019.
8. A. Pashaei, H. Sajedi, and N. Jazayeri, “Brain tumor classification via convolutional neural network and extreme learning machines,” in *2018 8th International Conference on Computer and Knowledge Engineering (ICCKE)*, 2018.
9. L. Pei, L. Vidyaratne, W.-W. Hsu, M. M. Rahman, and K. M. Iftekharuddin, “Brain tumor classification using 3D convolutional neural network,” in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, Cham: Springer International Publishing, 2020, pp. 335–342.

10. M. K. Abd-Ellah, A. I. Awad, A. A. M. Khalaf, and H. F. A. Hamed, "Two-phase multi-model automatic brain tumour diagnosis system from magnetic resonance images using convolutional neural networks," *EURASIP J. Image Video Process.*, vol. 2018, no. 1, 2018.
11. A. Rehman, S. Naz, M. I. Razzak, F. Akram, and M. Imran, "A deep learning-based framework for automatic brain tumors classification using transfer learning," *Circuits Systems Signal Process.*, vol. 39, no. 2, pp. 757–775, 2020.
12. M. Arif, F. Ajesh, S. Shamsudheen, O. Geman, D. Izdrui, and D. Vicoveanu, "Brain tumor detection and classification by MRI using biologically inspired orthogonal wavelet transform and deep learning techniques," *J. Healthc. Eng.*, vol. 2022, p. 2693621, 2022.
13. G. Hemanth, M. Janardhan, and L. Sujihelen, "Design and implementing brain tumor detection using machine learning approach," in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, 2019.
14. F. P. Polly, S. K. Shil, M. A. Hossain, A. Ayman, and Y. M. Jang, "Detection and classification of HGG and LGG brain tumor using machine learning," in *2018 International Conference on Information Networking (ICOIN)*, 2018.
15. M. Siar and M. Teshnehlal, "Brain tumor detection using deep neural network and machine learning algorithm," in *2019 9th International Conference on Computer and Knowledge Engineering (ICCKE)*, 2019, pp. 363–368.
16. S. Kumar C.K. and H. D. Phaneendra, "Categorization of brain tumors using SVM with hybridized local-global features," in *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, 2020, pp. 311–314.