Comparative study of Neural Learning methods in the context of Brain Tumor Classification

CSCI-B 565 - Data Mining - Fall 2021

Submitted in partial fulfillment of the requirements of
Master of Science Data Science

By

Immanuel Savio Donbosco and Sushmitha Keerthy Anekal Sathyakeerthy
ID No. 2000919040 - 2000913990

Under the supervision of:

Dr. Yuzen Ye

IN

Indiana University Bloomington
February 2023

Indiana University Bloomington

Abstract

Comparative study of Neural Learning methods in the context of Brain Tumor Classification

by Immanuel Savio Donbosco and Sushmitha Keerthy Anekal Sathyakeerthy

The goal of the project is to compare different neural network models namely one Convolutional Neural Network model, and 3 different transfer learning models - VGG16, InceptionV3, ResNet50. We attempt to perform the following experimental comparison on the dataset consisting of Magnetic resonance images (MRI) of the human brain. The dataset is a coalition of two different datasets of 3460 and 435 MRI scans of the brain. They are divided into 4 classes - Glioma Tumor, Meningioma Tumor, Pituitary Tumor and one class of normal tumor free brains. These 3895 images are used to train and test on the models. We perform data mining tasks such as exploratory data analysis (EDA) and data pre-processing to efficiently extract information from the available dataset. Using loss and accuracy as our mertic to compare our models we attempt to realise the best Neural Network model to classify brain MR Image dataset.

Contents

List of Figures

Chapter 1

Introduction

1.1 Deep Learning

Deep Learning is a sub-field of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. Deep learning (also known as deep structured learning or hierarchical learning) is part of a broader family of machine learning methods based on artificial neural networks. Learning can be supervised, semi-supervised or unsupervised.

Deep learning architectures such as deep neural networks, deep belief networks, recurrent neural networks and convolutional neural networks have been applied to fields including computer vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, bioinformatics, drug design, medical image analysis, material inspection and board game programs, where they have produced results comparable to and in some cases superior to human experts.

1.1.1 Deep Neural Network

A deep neural network is an artificial neural network which has multiple hidden layers, apart from the input and output layers. The essence of a deep neural network lies in its depth. A deep NN can be a significant tool in representing complex, non-linear relationships.

Deep neural networks are good at discovering correlation structures in data in an unsupervised fashion. Therefore it is widely used in speech analysis, natural language processing and in computer vision. This information of the structure of the data is stored in a distributed fashion. i.e. Information about the model is distributed across different layers in a neural network and in each layer, model information (weights) are distributed in different neurons. There are a lot of ways to combine the information in a layer spread across different neurons and there are lot of ways to combine layers in order to minimize a loss function (which is a proxy for how well the neural network is doing in terms of achieving its goals).

1.2 Convolutional Neural Networks

Convolutional Neural Networks are the new class of neural network architecture that are considered a breakthrough in the fiels of image recognition. They are usually used in the field of image processing and vision applications like recognition, classification, segmentation etc. A CNN, which is very similar to an ANN is made up (not limited to) of 4 types of layers:

1. Convolutional Layers
2. ReLU Layers
3. Pooling Layers
4. Fully Connected Layer

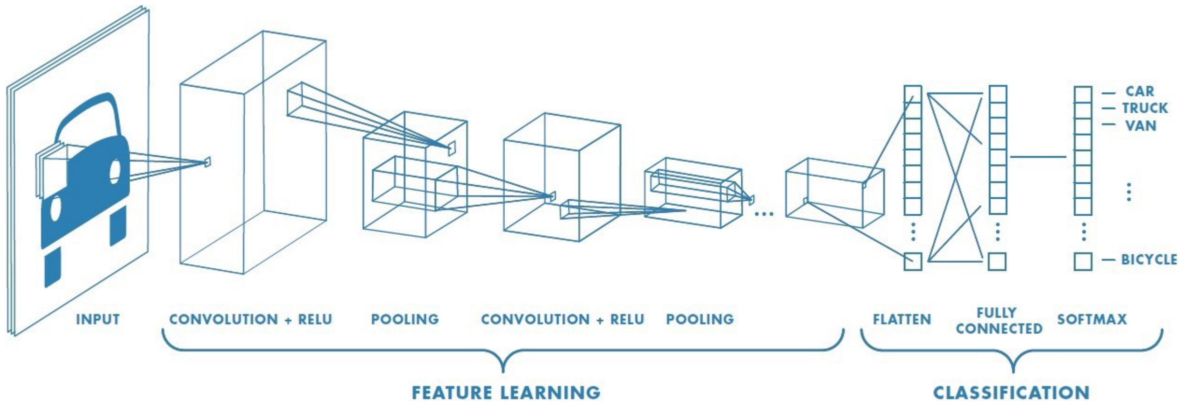

Figure 1.1: Convolutional Neural Network for Image Classification

1.3 Transfer Learning

Transfer Learning is a new technique which is considered the next big thing in deep learning. The basic idea in the implementation of transfer learning is that when a model is trained on a set of data and expected to perform well on some unseeen data, similarly we use a model which was trained on a set of data, retrain some part of the network on the new data and hope it would "Transfer" the knowledge of the network to apply on the new set of data.

We are trying to levarage the power of transfer learning in our use case and to see how better or worse do they perform in comparison to traditional Convolutional Neural Networks.

Chapter 2

Problem Statement and Data Description

2.1 Problem Definition

Each year, approximately 70,000 - 170,000 cancer patients are diagnosed with brain tumors, while approximately 100,000 will die every year as a result of brain metastases. Malignant brain tumors cause an average of 20 Years of Potential Life Lost. Brain tumors also represent the highest per-patient initial cost of care for any cancer group, with an annualized mean net cost of care approaching $150,000. Detection and classification of such cancers using automated classification techniques help us understand their science and evolution better. It also helps us come up with novel ways to treat such conditions with higher accuracy than manual classification. Hence, proposing a system performing detection and classification by using Deep Learning Algorithms using Convolution Neural Network (CNN), and Transfer Learning (TL) would prove helpful to doctors around the world.

2.2 Literature Survey

Paper 1 : Kaldera, H. N. T. K., Shanaka Ramesh Gunasekara, and Maheshi B. Dissanayake. "Brain tumor classification and segmentation using faster R-CNN." In 2019 Advances in Science and Engineering Technology International Conferences (ASET), pp. 1-6. IEEE, 2019.

Findings: This paper speaks about how to use the CNN model to classify brain tumors.

Relevance to the Project: This paper was useful in helping us design the CNN model and in understanding why the CNN model is the go to model for such tasks

Paper 2 : Karen Simonyan, Andrew Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", 4 Sep 2014,revised 10 Apr 2015, 1409.1556v6

Findings: This paper gave us a detailed overview of the VGG16 architecture and how its layers are laid out.

Relevance to the Project: The information from this paper was pretty useful on helping us design the first transfer learning model for the experiment using the VGG16 network.

Paper 3 : Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818-2826. 2016.

Findings: This paper spoke about how the InceptionV3 was designed from the previous Inception network models and how it is better than AlexNet architecture but with lesser neurons and layers but a better performance.

Relevance to the Project: This paper helped us design the second Transfer Learning Model using InceptionV3 and how it was supposed to better than the VGG Model in classification

Paper 4 : He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016

Findings: This is the main paper that introduced ResNet50 architecture and speaks about its structure and features in detail.

Relevance to the Project: This literature is from where we used to design the final transfer learning model using ResNet50 architecture for the study

2.3 Why CNN vs Transfer Learning

Nowadays AI and more especially deep learning is one of the most active areas of research with a plethora of practical applications in every industry.

In typical applications of deep learning we have to deal with:

1. Big Data: ImageNet dataset contains a few TB of data, in industry, even more! As an example, Facebook users upload 800M images per day. These images are all processed by deep learning models to learn more about the users.
2. Parameters Storage: Models are made of hundreds of layers which represents a few hundred to 2Bn parameters. It represents 0.1–8GiB just to store the model (which resides in memory

when in use, not on the hard drive). Note that the training phase of the model usually takes more memory so it becomes harder to fit the model on a single mainstream PC.

3. Computation Power: With the increasing complexity of tasks, labels tend to get huge too. Typical applications of image recognition try to classify an object in one category among 10–22k but face recognition for instance requires 1 label per people. Such models with this large number of categories take weeks to train.

This tells us that Deep learning is a Super Computing problem. We need high computational power along with efficient data processing and storage.

In case of Transfer Learning, the problem starts with selecting a network is a pretrained network on some other dataset. This itself reduces one of the problem which we mentioned in the CNN part, Computational Power. Computational power requirement is reduced exponentially as the network is already trained on a bigger dataset and has some of the weights and biases saved within it.

Data availability is also tackled when we use Transfer Learning methods. The networks used in this project are VGG16, ResNet50 and InceptionV3 are trained on the ImageNet dataset which is one of the biggest datasets available on the internet. By training on such a heavy dataset, it learns to recognize a lot of intricate details which might not be possible if trained on a smaller dataset. By using the network trained on this dataset, we can use a smaller dataset to train the network to apply to a specific use case with higher accuracy.

2.4 The Dataset

The dataset consists of raw magnetic resonance images (MRI) of the human brain. It consists of images of the brain with and without tumors. The complete dataset is made by merging two datasets of which one consists of only axial view and the other consists of all sagittal, coronal and axial views.

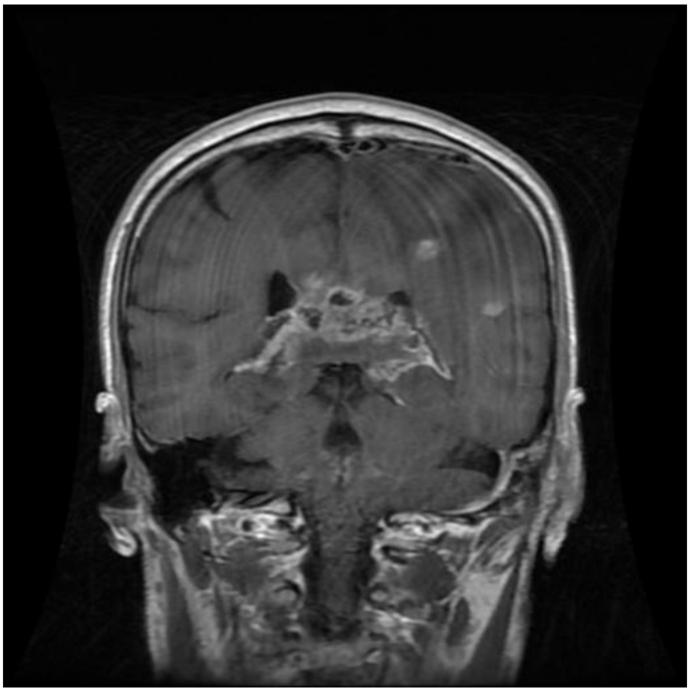

Figure 2.1: Raw MRI Images

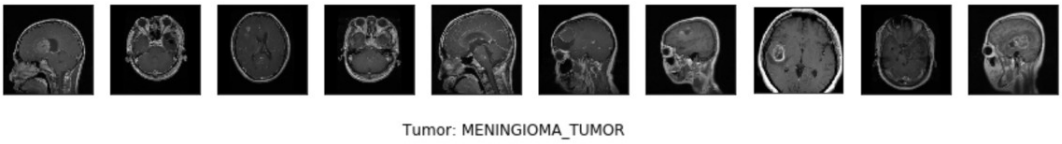

Figure 2.2: Grid plot

2.4.1 Data Description

The dataset is a coalition of two different datasets. The training set consisting of 28,825 images, and a test set of 3,391 MRI scans of the brain respectively. The dataset is further divided into 4 classes - Glioma Tumor, Meningioma Tumor, Pituitary Tumor and one class of normal tumor free brains. These 3895 images are used to train and test on the model.

Chapter 3

Data Preprocessing and Exploratory Data Analysis (EDA)

3.1 Data Collection

Data for the project consists of 2 different datasets consisting of 3264 images, and 435 MRI scans of the brain respectively. Once they were collected (already seperated into the four types of tumors) we had to perform data visualizations and some EDA to see how well the data suited our use case. Next we also had to perform some preprocessing on them to make them usable and easy to process for the networks we planned to use.

3.2 Exploratory Data Analysis

The first step to any data science problem is to perform some basic EDA with the data in hand. This would give us an idea of what type of data we are dealing with, what are the classes, imbalances or missing data etc.

3.2.1 Data Plotting - Grid Plots

Grid plots of the data allow us to visualize how the data looks like. This was the first step in the EDA process. We plotted a grid plot of the data and randomly selected 10 scans from each class of tumors and also from the non tumor brain to see how they look. The images are shown below

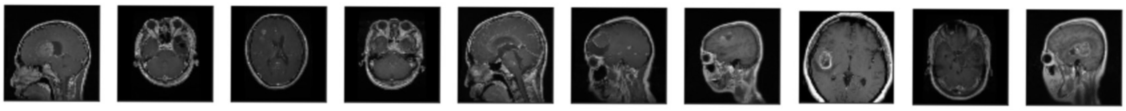

Figure 3.1: Glioma Tumor Grid Plot

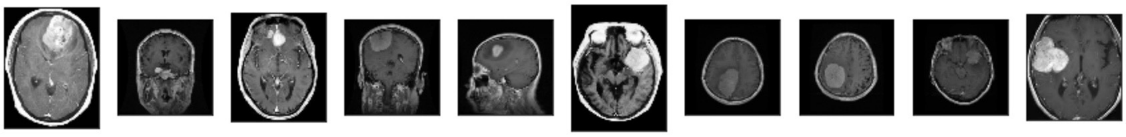

Figure 3.2: Meningioma Tumor Grid Plot

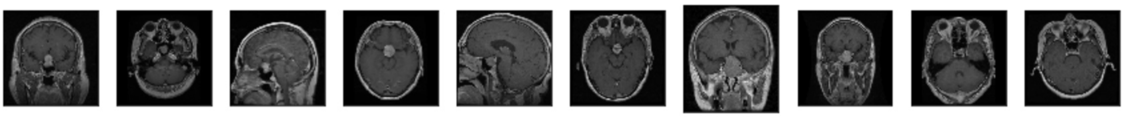

Figure 3.3: Pituitary Tumor Grid Plot

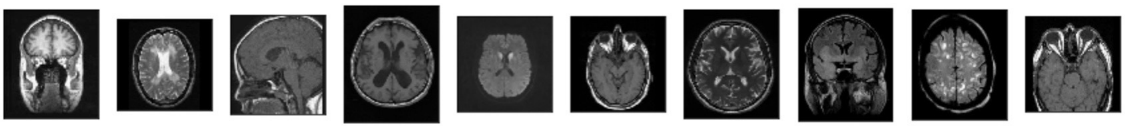

Figure 3.4: No Tumor Grid Plot

3.2.2 Class Ratio and Imbalance

It is very important that the dataset used in an ML or DL problem is balanced. Imbalance of any kind (too much data or too less data) of any one of the classes could lead to imperfect training and reduction in accuracy of the model. This is why we performed a class ratio comparison which gives us an idea of how much data is present of each class. We performed a bar plot of a random sample of data to see how much data of each class is selected and a pie plot of the whole dataset to see how much percentage of each of the class is present.

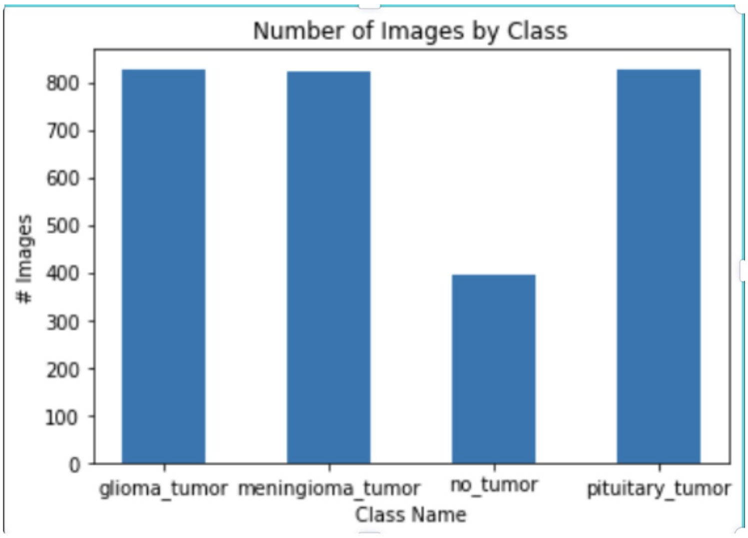

FIGURE 3.5: Class Ratio Bar Plot

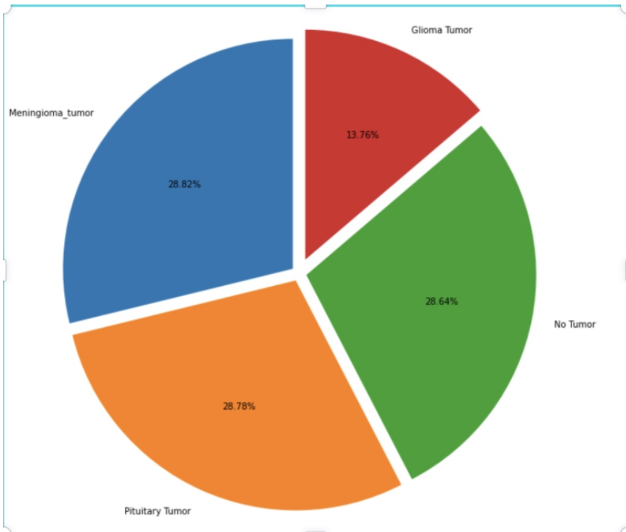

FIGURE 3.6: Pie Plot of Image Percentage by Class

3.2.2.1 Image Ratio

Image size is a very important factor when we use images as input. As we are using transfer learning models like VGG which take the input image in the size of (224, 244) it is best if all the images are of the same size and more preferably of the unit ratio (1:1). We plotted a graph to see how many of the images were of the same size ratio.

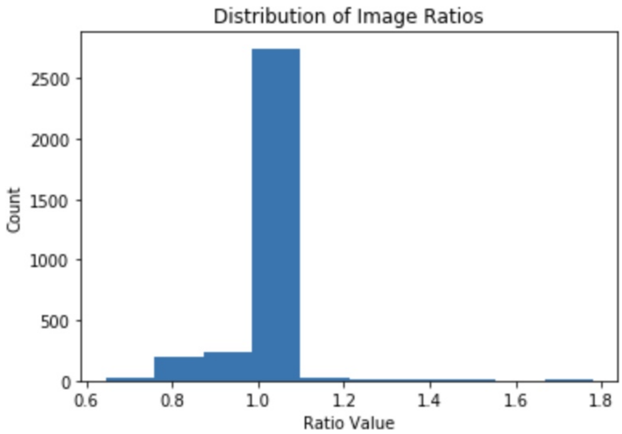

FIGURE 3.7: Image Ratio Plot

3.3 Image Preprocessing

As mentioned above we need to perform some preprocessing in order for the model to perform efficiently.

3.3.1 Cropping and Normalization

First step was to crop the images to have exactly the area of interest and discarding any unwanted border areas. This normalization process was performed by finding out the extreme points of the image and then cropping out the rectangular part of it. This will help us to incease the % of the image that we are interested in. One example plot of the 4 steps is shown below

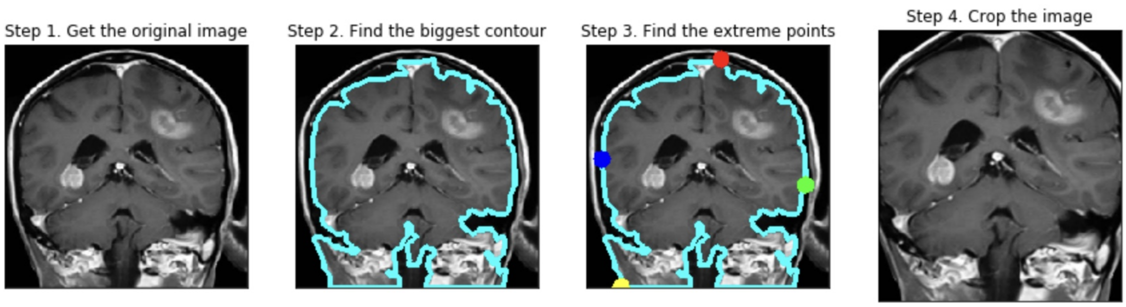

FIGURE 3.8: Image Normalization

3.3.2 Image Augmentation

In real life the dataset can be of any direction or orientation. This is why we decided to augment images in all possible directions and angles. This also involved flipping and mirroring the images to get more data to work with as well. One image was selected at random to visualize the augmentation process which is shown below.

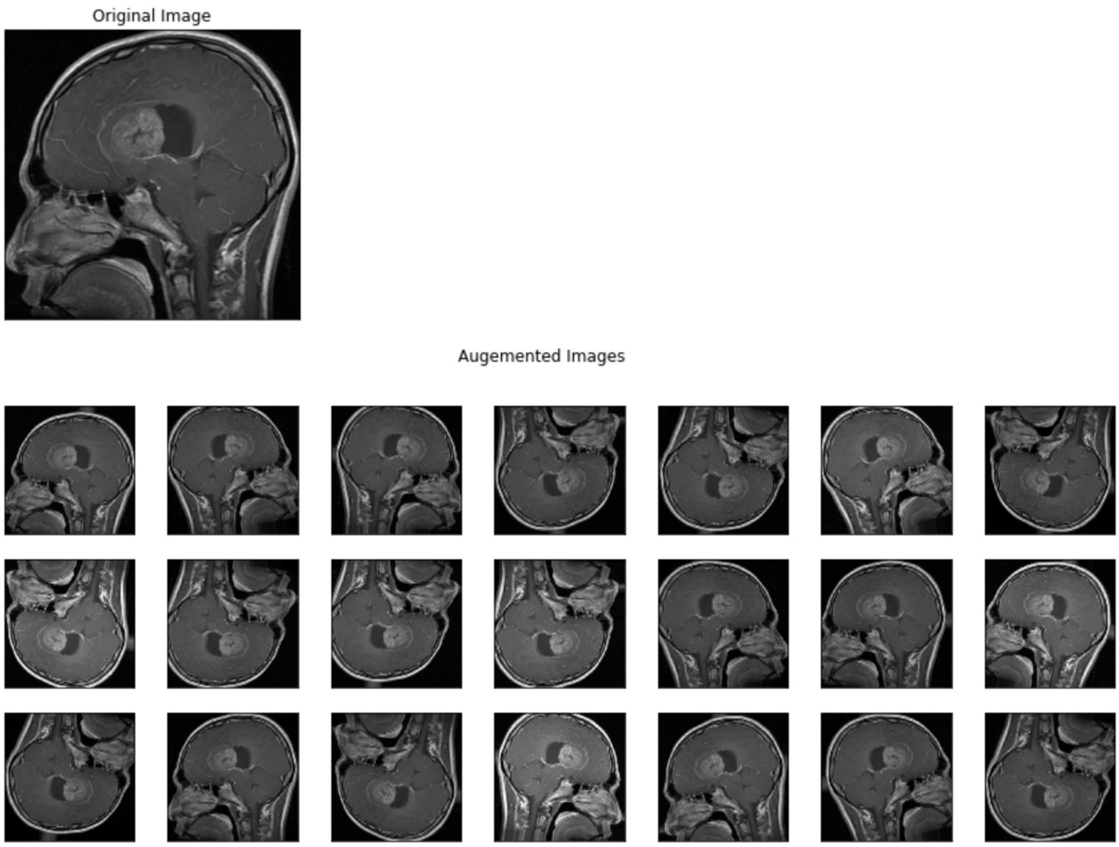

FIGURE 3.9: Image Augmentation

3.3.3 Image Interpolation

Image interpolation occurs when you resize or distort your image from one pixel grid to another. Image resizing is necessary when you need to increase or decrease the total number of pixels, whereas remapping can occur when you are correcting for lens distortion or rotating an image. Zooming refers to increase the quantity of pixels, so that when you zoom an image, you will see more detail.

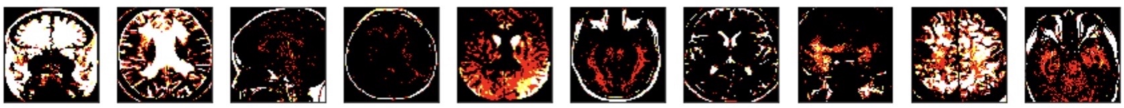

FIGURE 3.10: Glioma Tumor Interpolation Plot

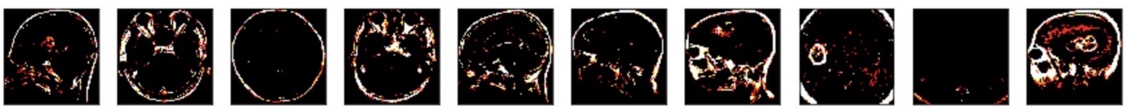

Figure 3.11: Meningioma Tumor Interpolation Plot

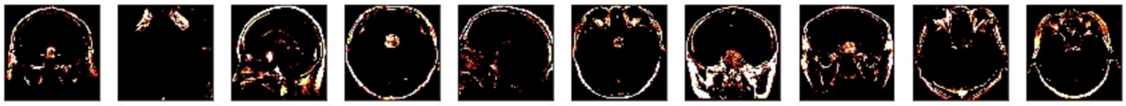

Figure 3.12: Pituitary Tumor Interpolation Plot

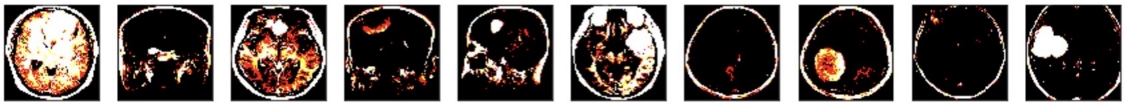

Figure 3.13: No Tumor Interpolation Plot

Chapter 4

Model 1 - Convolutional Neural Network

4.1 CNNs - How do they work?

Convolutional Neural Networks or CNNs are a class of neural networks that take in an image as input and perform complex mathematical operations on the data (pixel), update weights and biases to find out different features of the image such as corners, curves, edges etc. This will be used to differentiate the image form other images of that dataset. One advantage of a Convolutional Neural Network is that the preprocessing required is much lesser than other neural and traditional learning methods.

4.1.1 Architecture Overview

For the sake of this experiment, performed on Kaggle and the limited time constraint, we take a CNN with 15 layers (architecture diagram shown below) which consists of one input and one output layer. It also contains 3 convolutional layers, 3 max pooling layers, and 3 dense layers which will perform the classification.

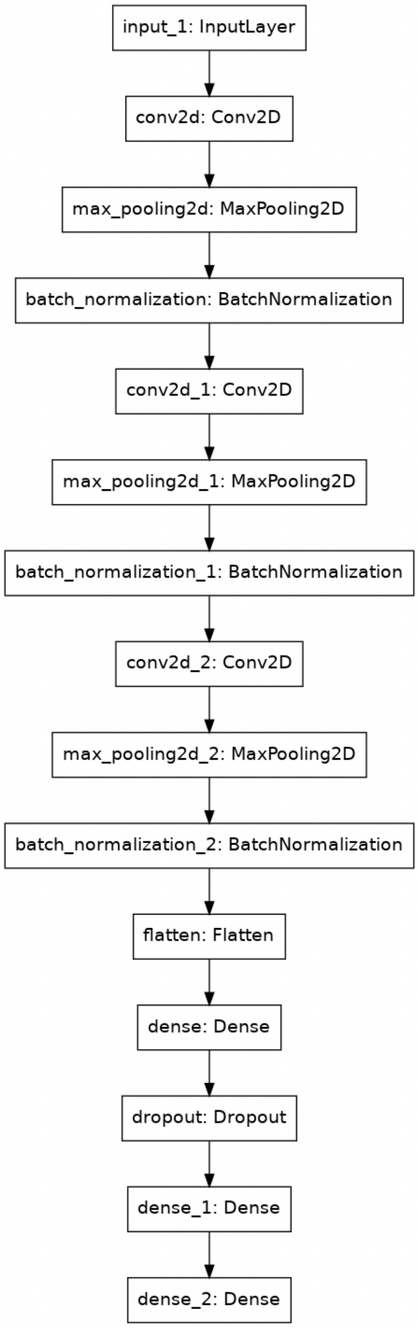

FIGURE 4.1: CNN Model Architecture

The input layer takes in the input image in the (224,244) pixel size and sends it to the convolutonal layer and furthers the processing.

4.1.2 Convolutional Layer

The convolutional layer is where most of the feature extraction from the image. The reason the layer is called the convolution layer is because the layer convolves through the image. We define a filter (e.g. 3x3) filter, this goes from line to line and layer to layer in the image to reduce the dimension using the filter. This concentrates the required features of the image so we can use it to process it further in the further layers.

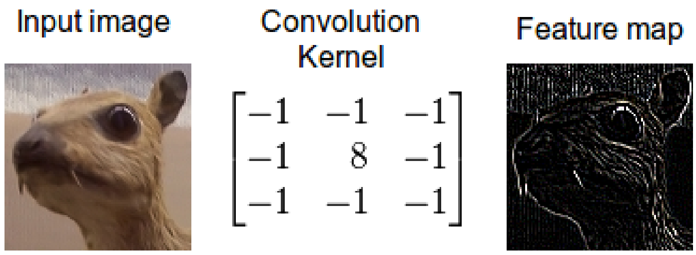

FIGURE 4.2: Convolution of an image

In our network the first convolution layer has 128 neurons, the second has 64 neurons, and the third convolution layer has 32 neurons. Reduction in the size of the neural layers as we progress further in the network helps in concentrating the features.

4.1.3 Max Pooling Layer

Pooling layers are responsible for downsizing the feature maps by summarizing the features of the image which a present as patches across the image into a localized form. Max pooling is one of the method where the max of the required values is taken according to the size of the layer. This will help the next layers of the network perform better as the features are show better.

We have 3 max pooling layers each of size 2x2 i.e. it will select every 2x2 submatrix in the feature map and use the max value from that matrix.

4.1.4 Dense Fully Connected Layers

The last 3 layers of the network are the fully connected layers or the dense layers. This set of layers are the ones that are responsible to perform the classification which will use the features collected in the previous layers to segregate them into one of the 4 classes.

4.2 Metrics

The CNN gave us a loss of 0.534 and a training accuracy of 0.9745. During the validation phase the network gave us a validation loss of 0.654 and a validation accuracy of 0.9243

Chapter 5

Model 2 - Transfer Learning - VGG16

5.1 What is VGG16?

The VGG16 network is a type of network proposed in the paper titled "Very Deep Convolutional Networks for Large-Scale Image Recognition" which achieved 92.7% in the imagenet dataset classification competetion, which is a dataset of over 14 million images with over 1000 classes. Below is the flow diagram of how the network and its layers are designed.

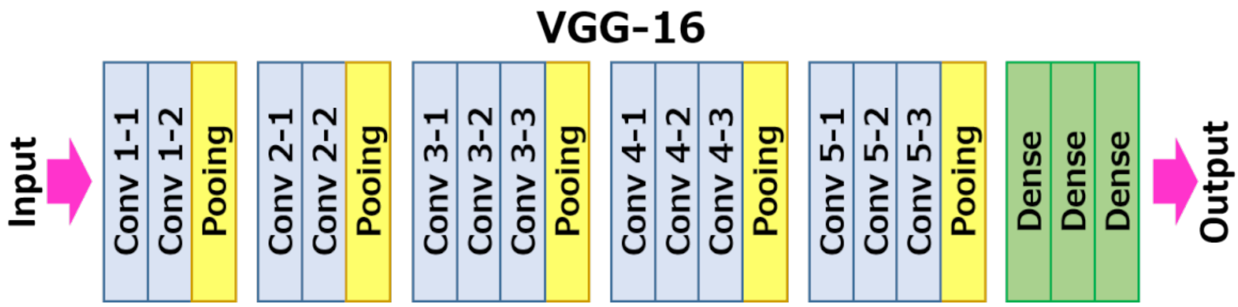

FIGURE 5.1: VGG16 Layers

5.2 Why VGG16?

This will be our first network in the transfer learning phase. As shown the VGG16 has the following architecture. Also the weights used in the network will be the same as the weights of the network when trained on the ImageNet dataset. This is because as it showed a 92% accuracy in classifying the data, we can expect better feature extraction if use them.

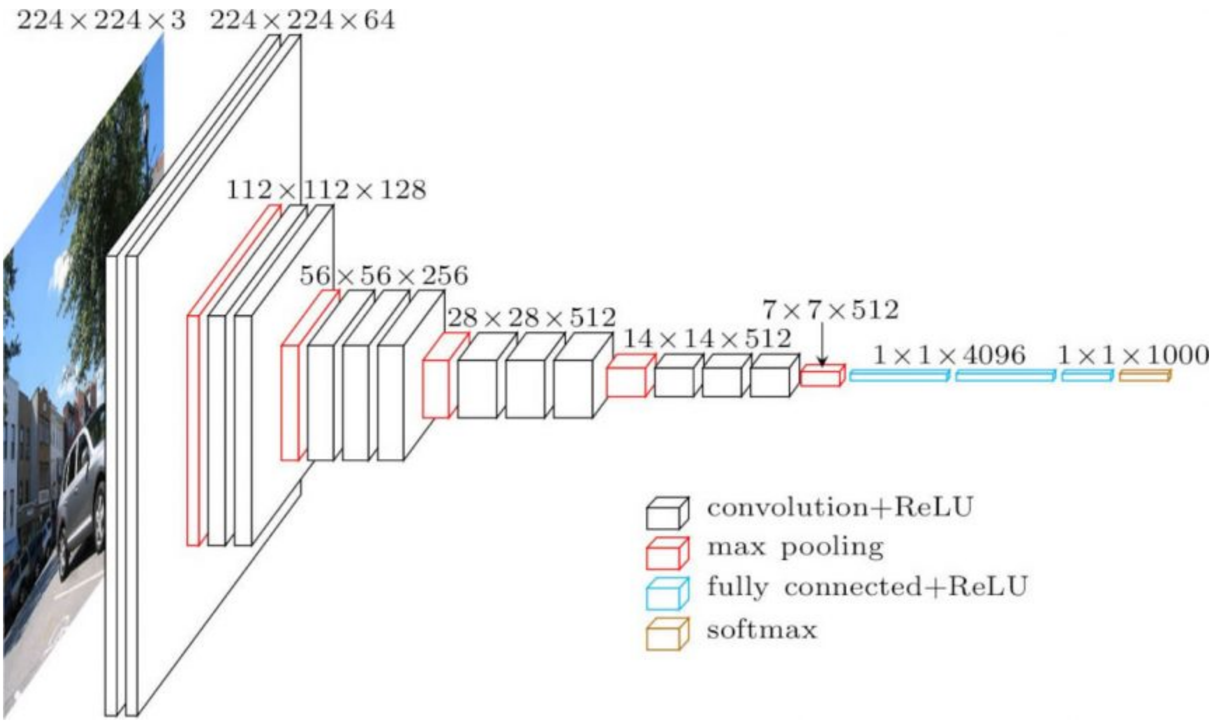

FIGURE 5.2: VGG16 Model Architecture

5.3 Training Methodology

Similar to any transfer learning methodology, we use this network to classify our brain tumor images. We train only the fully connected layers in this model by freezing the rest of the convolutional and pooling layers. This is done by setting the Layers.Trainable to False. This way none of the weights in those layers will not be edited.

Once the images are sent into the VGG network as input, we used matplotlib.pyplot to visualize how they were changing. This will give us a deepler look into how the network learns.

5.4 Metrics

VGG16 gave us a loss of 0.3027 and a training accuracy of 0.9724. During the validation phase the network gave us a validation loss of 1.3712 and a validation accuracy of 0.9134

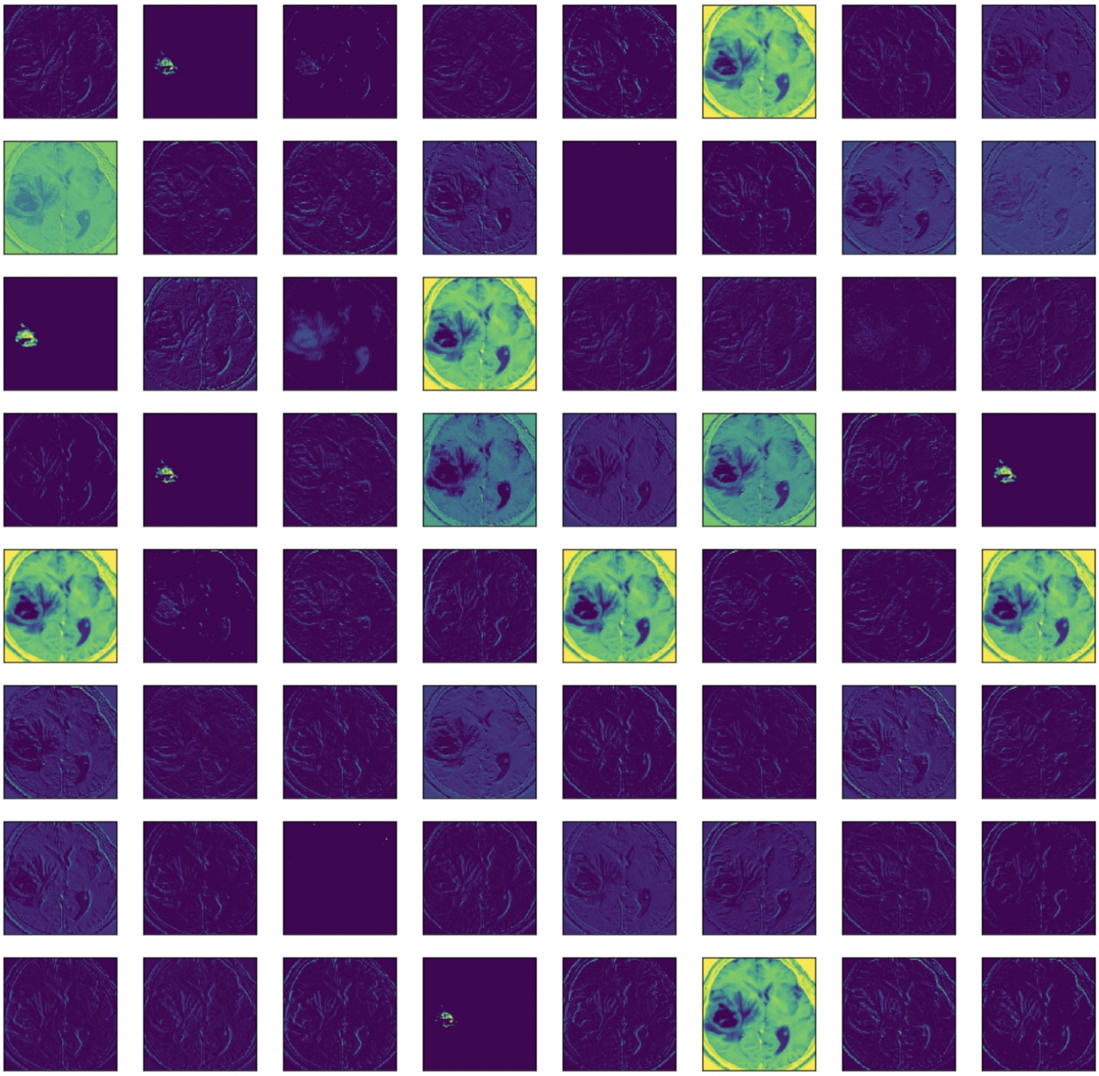

Figure 5.3: Training transitions of the VGG16 Network

Chapter 6

Model 3 - Transfer Learning - InceptionV3

6.1 What is InceptionV3?

InceptionV3 is a deep convolutional neural network designed by Google as part of the Googlenet project. It is similar to the VGG network as it was introduced during the ImageNet challenge as well. It focuses on concentrating the parameters to 25 million when compared to the 60 million in AlexNet. The InceptionV3 architecture is shown below. It has 148 layers in compared to VGG which has 21 layers. Inception is also known to perform better on the ImageNet dataset compared to VGG.

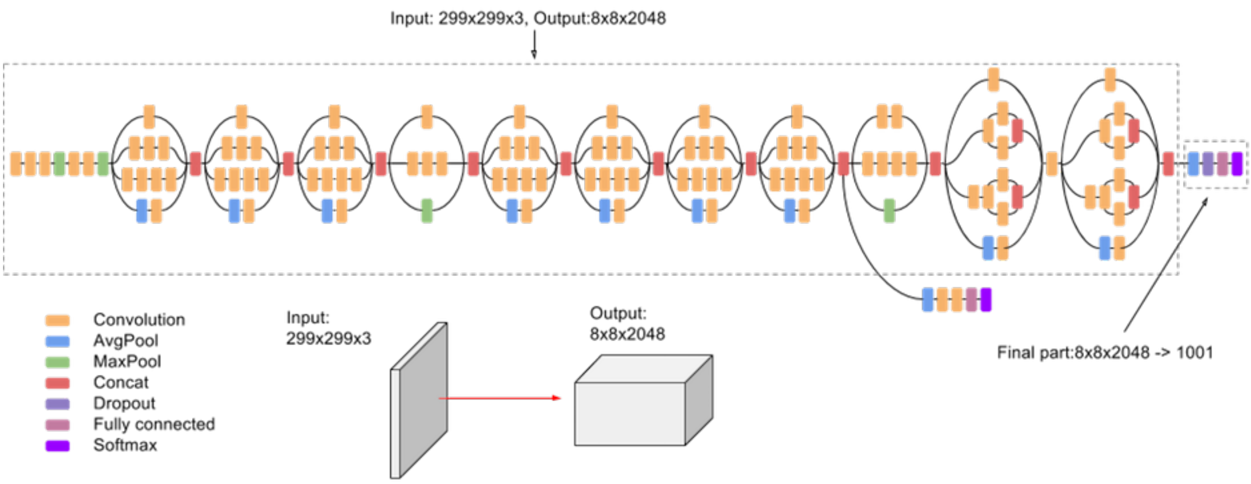

Figure 6.1: VGG16 Layers

6.2 Why VGG16?

This will be our second network in the transfer learning phase. The InceptionV3 has architecture as shown above. Also the weights used in the network will be the same as the weights of the network when trained on the ImageNet dataset.

We use InceptionV3 beacuse of the following reasons

1. Factorization into smaller convolutions
2. Spatial Factorization into Asymmetric Convolutions
3. Utility of Auxiliary Classifiers
4. Efficient Grid Size Reduction

6.3 Metrics

InceptionV3 gave us a loss of 0.523 and a training accuracy of 0.973. During the validation phase the network gave us a validation loss of 5.34 and a validation accuracy of 0.7563

Chapter 7

Model 4 - Transfer Learning - ResNet50

7.1 What is ResNet50?

ResNet50 is a deep convolutional neural network with 50 neural layers. It is similar to the VGG network as it was introduced during the ImageNet challenge as well. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224.

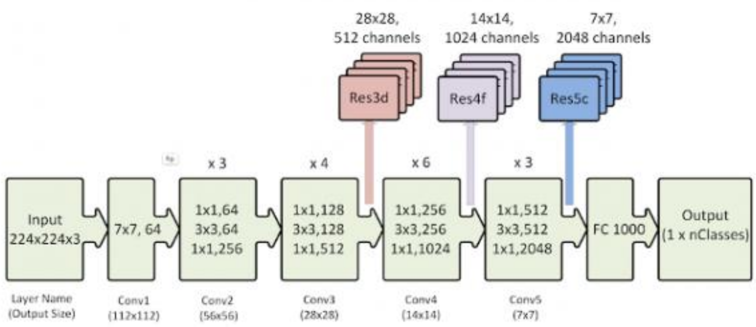

FIGURE 7.1: ResNet 50 Architecture

7.2 Metrics

ResNet50 gave us a loss of 0.574 and a training accuracy of 0.879. During the validation phase the network gave us a validation loss of 0.76 to 5 and a validation accuracy of 0.83

Chapter 8

Results and Observations

8.1 Observations

This report mainly focuses on devising a more efficient method to classify brain with tumor replacing the serial CNN method used all over the computer science industry. Big data generating huge datasets in the order of GBs and TBs makes serial training a huge issue. Below are the comparative results of the CNN and transfer learning based training.

8.2 Results

The first section will show the training trend in the CNN method and the second section will show the transfer learning training using three networks.

8.2.1 CNN Model

The CNN gave us a loss of 0.534 and a training accuracy of 0.9745. During the validation phase the network gave us a validation loss of 0.654 and a validation accuracy of 0.9243

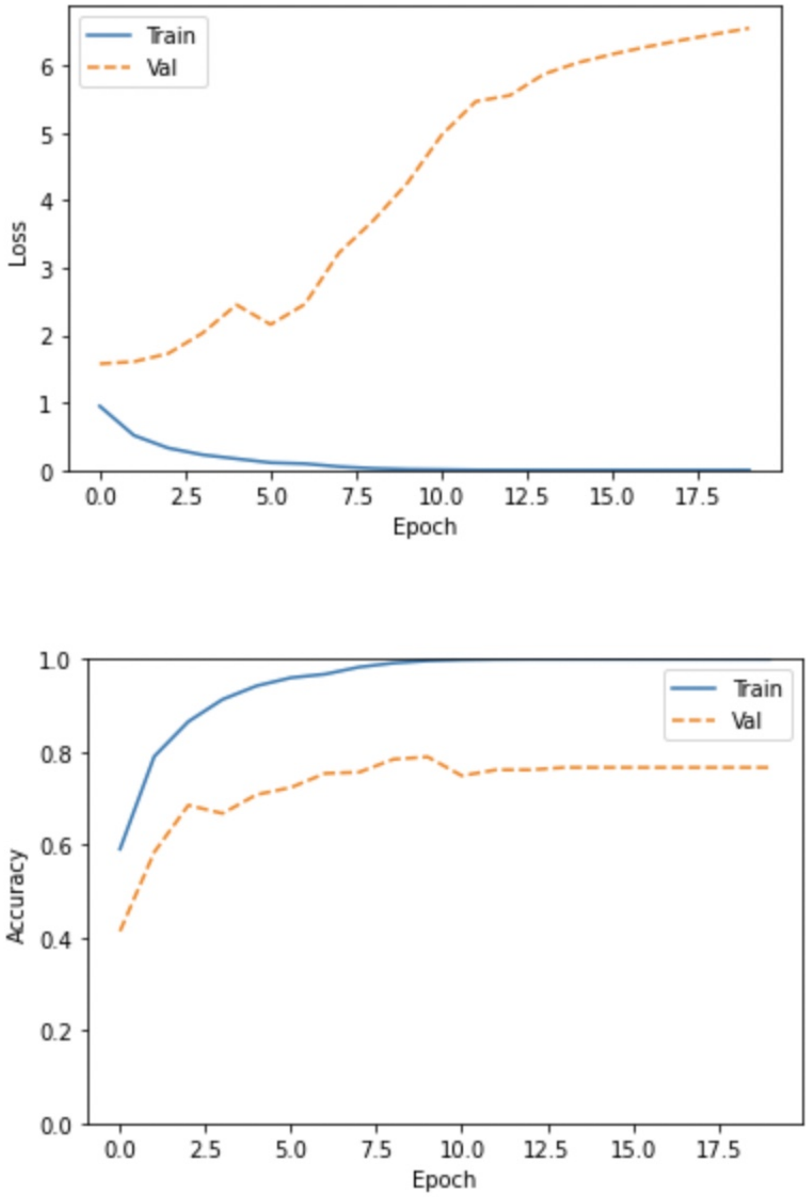

FIGURE 8.1: CNN Performance

The confusion matrix for the CNN is as shown below

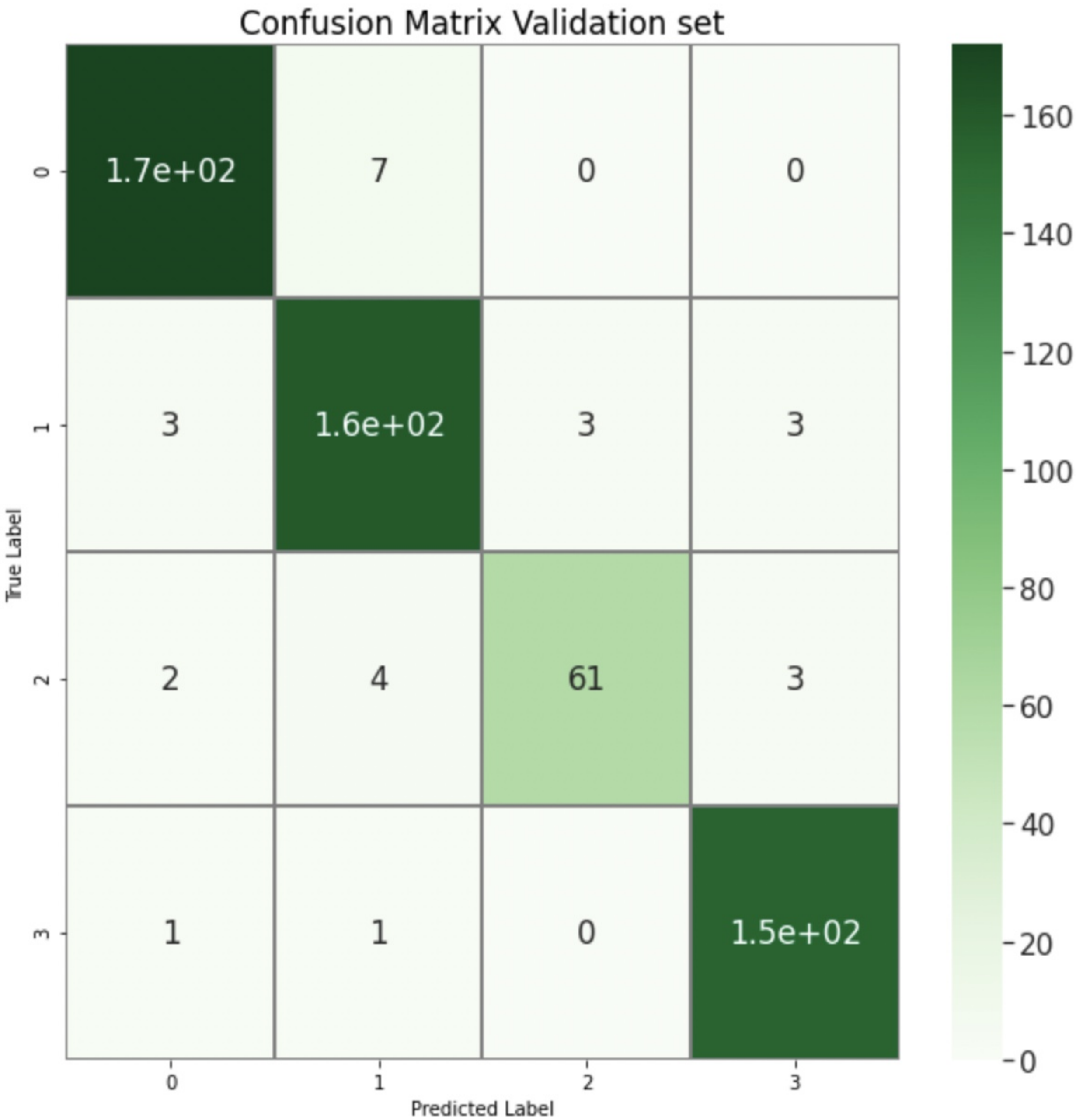

FIGURE 8.2: Confusion Matrix

8.2.2 Transfer Learning Results

This section will show the results of the 3 transfer learning models - VGG16, InceptionV3 and ResNet50.

8.2.2.1 Model 1 - VGG16

VGG16 gave us a loss of 0.3027 and a training accuracy of 0.9724. During the validation phase the network gave us a validation loss of 1.3712 and a validation accuracy of 0.9134

The Loss and Accuracy graph are shown below

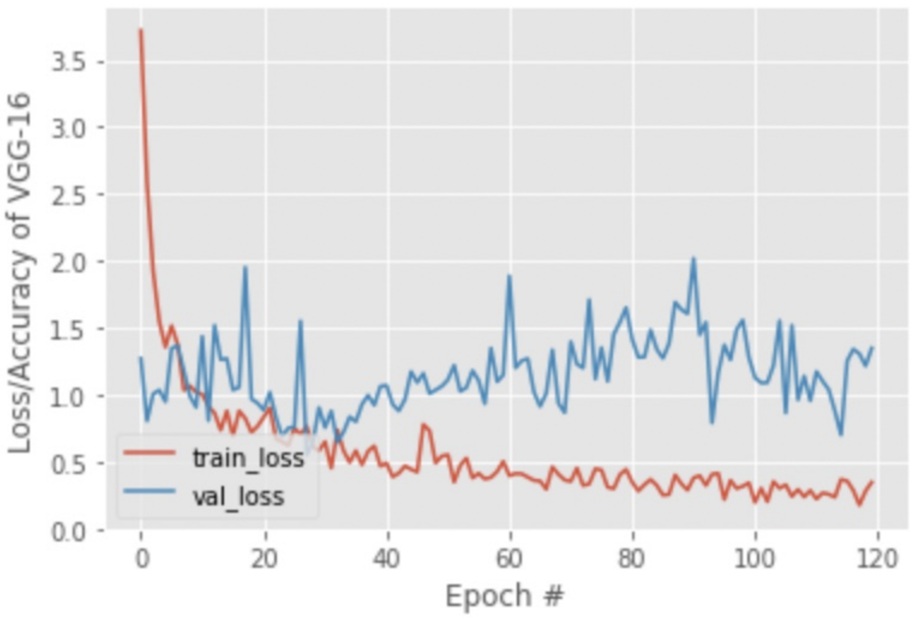

FIGURE 8.3: VGG16 Loss Graph

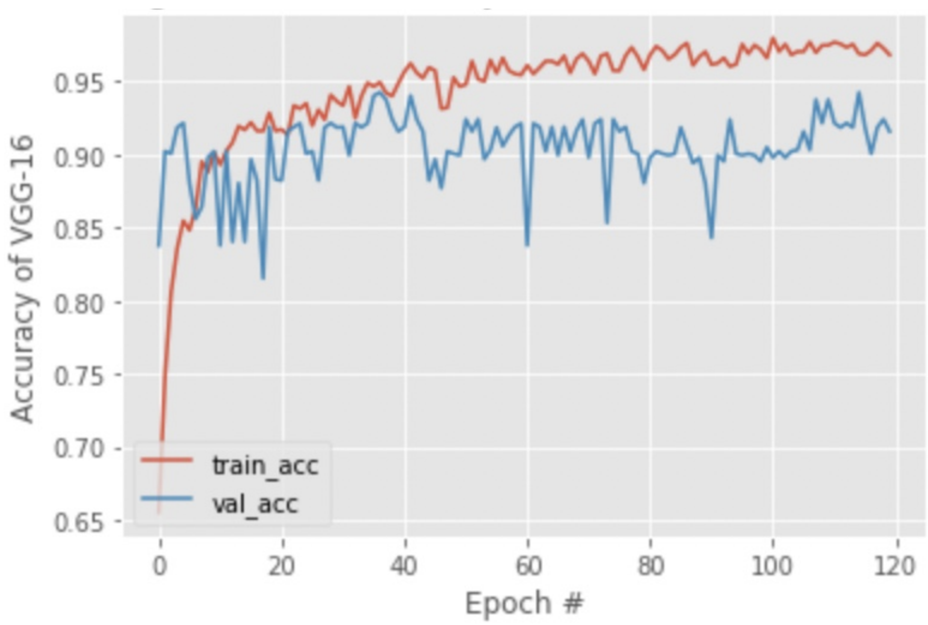

FIGURE 8.4: VGG16 Accuracy Graph

8.2.2.2 Model 2 - InceptionV3

InceptionV3 gave us a loss of 0.523 and a training accuracy of 0.973. During the validation phase the network gave us a validation loss of 5.34 and a validation accuracy of 0.7563

The Loss and Accuracy graph are shown below

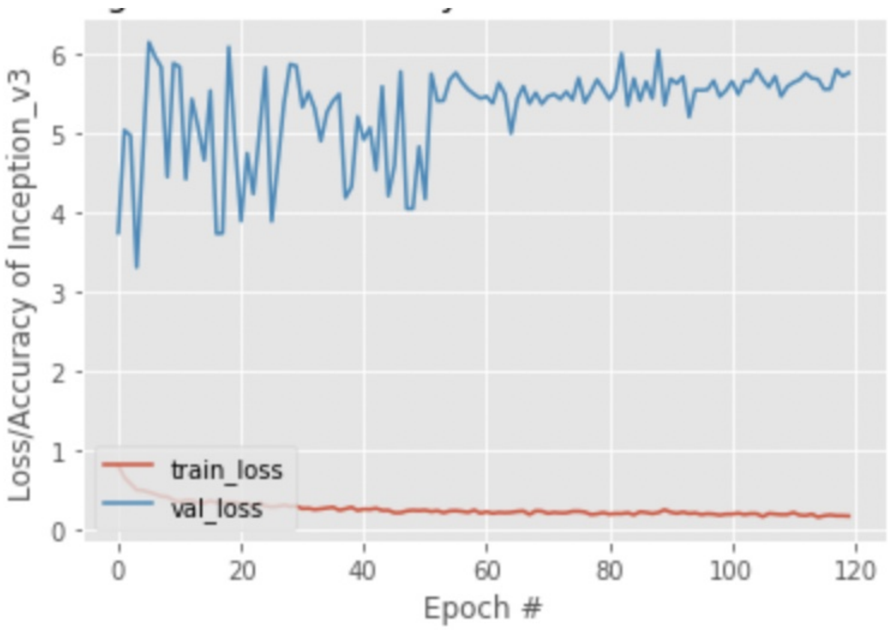

FIGURE 8.5: InceptionV3 Loss Graph

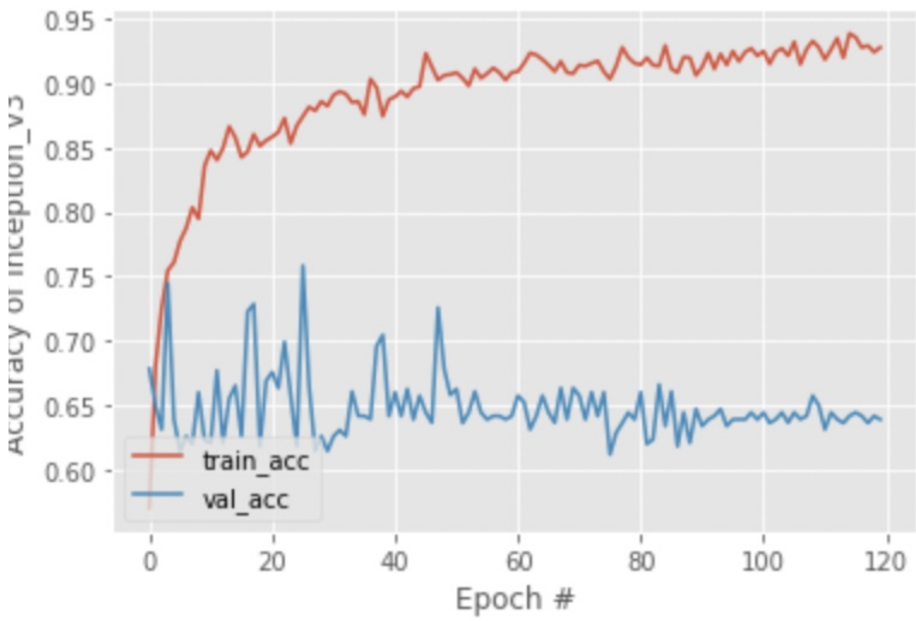

FIGURE 8.6: InceptionV3 Accuracy Graph

8.2.2.3 Model 3 - ResNet50

ResNet50 gave us a loss of 0.574 and a training accuracy of 0.879. During the validation phase the network gave us a validation loss of 0.76 to 5 and a validation accuracy of 0.83

The Loss and Accuracy graph are shown below

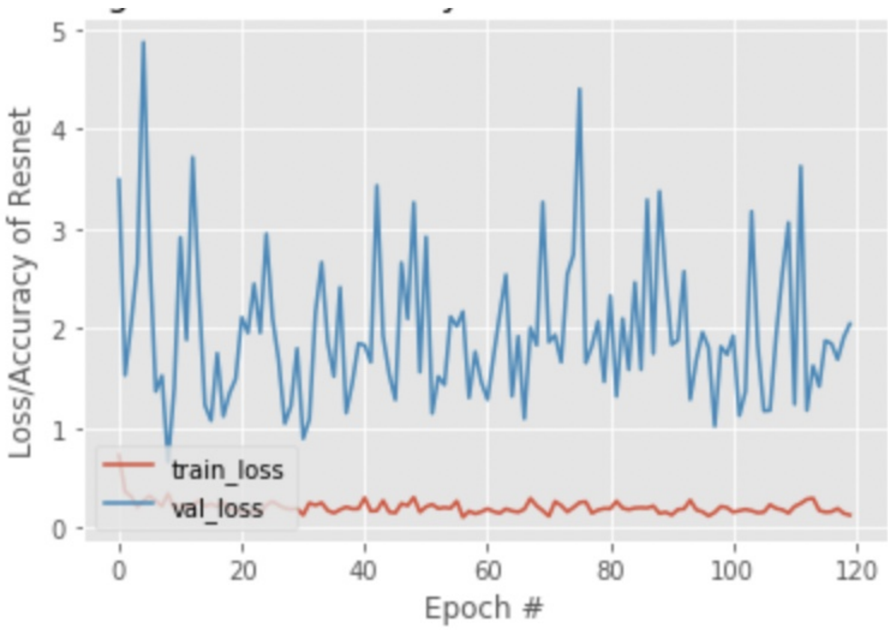

FIGURE 8.7: ResNet50 Loss Graph

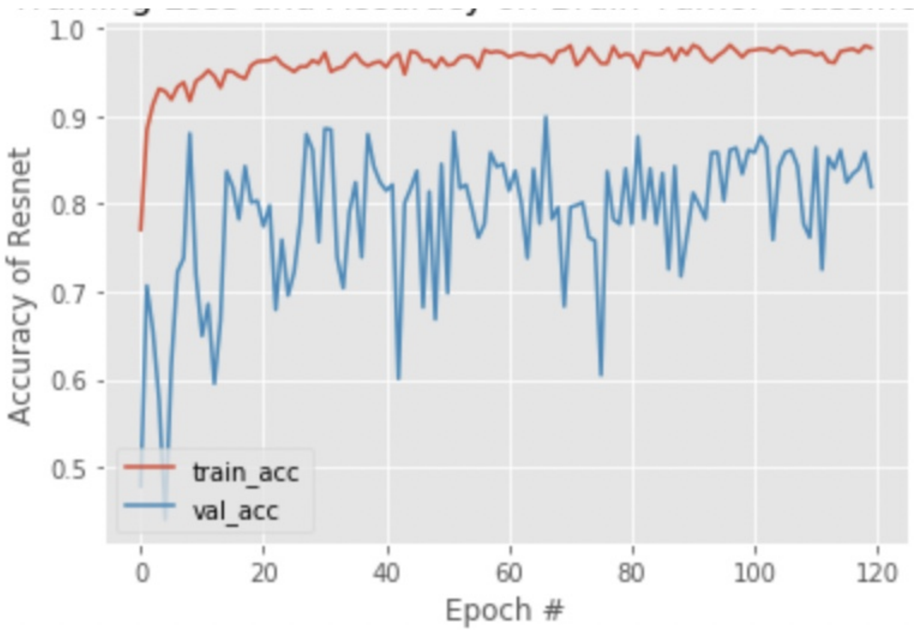

FIGURE 8.8: ResNet50 Accuracy Graph

8.3 Conclusion

Comparing the results of the CNN, and the three transfer learning models we can see that the CNN and the transfer learning model VGG16 give very similar results. In comparison to that InceptionV3 stands third and the ResNet50 has given the worst accuracy. This means that we can rule out Inception and Resnet from the race. Comparing CNN and VGG16 both have almost similar accuracies but the VGG16 model performed the best. The validation accuracy and test accuracy is higher than the other method. Time of computation and computation power is required lesser than traditional method as pretrained weights are used. This is conclusive evidence that VGG16 performed the best in the context of Brain Tumor Classification.

Chapter 9

References

[1] Karen Simonyan, Andrew Zisserman, ”Very Deep Convolutional Networks for Large-Scale Image Recognition”, 4 Sep 2014,revised 10 Apr 2015, 1409.1556v6.

[2]Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. ”Rethinking the inception architecture for computer vision.” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818-2826. 2016.

[3] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. ”Deep residual learning for image recognition.” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016.

[4]Kaldera, H. N. T. K., Shanaka Ramesh Gunasekara, and Maheshi B. Dissanayake. ”Brain tumor classification and segmentation using faster R-CNN.” In 2019 Advances in Science and Engineering Technology International Conferences (ASET), pp. 1-6. IEEE, 2019.

[5] https://www.qubole.com/blog/deep-learning-the-latest-trend-in-ai-and-ml/

[6] https://towardsdatascience.com/wtf-is-image-classification-8e78a8235acb

[7] https://www.topbots.com/transfer-learning-in-nlp/

[8] https://sisu.ut.ee/imageprocessing/book/3

[9] https://developer.nvidia.com/discover/convolution

[10] https://neurohive.io/en/popular-networks/vgg16/

[11] https://machinelearningmastery.com/what-is-deep-learning/

[12] https://en.wikipedia.org/wiki/Deep_ learning/

[13] https://www.eenewseurope.com/news/resnet-50-misleading-machine-learning-inference-benchmark-megapixel-images