Detection of Intracranial Hemorrhage on CT Scan Images using Convolutional Neural Network

by

Afridi Ibn Rahman
17201107
Subhi Bhuiyan
17201116
Ziad Hasan Reza
17201076
Jasarat Zaheen
17201100
Tasin Al Nahian Khan
17201085

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
September, 2021

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Afri. Rahman

Afridi Ibn Rahman
17201107

Subhi Bhuiyan
17201116

Ziad Hasan Reza

Ziad Hasan Reza
17201076

Jasarat

Jasarat Zaheen
17201100

Nahian

Tasin Al Nahian Khan
17201085

Approval

The thesis titled "Detection of Intracranial Hemorrhage on CT Scan Images using Convolutional Neural Network" submitted by

1. Afridi Ibn Rahman (17201107)
2. Subhi Bhuiyan (17201116)
3. Ziad Hasan Reza (17201076)
4. Jasarat Zaheen (17201100)
5. Tasin Al Nahian Khan (17201085)

Of Summer, 2021 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on October 02, 2021.

Examining Committee:

Supervisor:
(Member)

Zavid Parvez
Digitally signed by Zavid Parvez
DN: cn=Zavid Parvez, o=Brac University, ou=CSE, email=zavid.parvez@bracu.ac.bd, c=US
Date: 2021.10.05 16:48:30 +11'00'

Dr. Mohammad Zavid Parvez
Assistant Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Ziad

Dewan Ziaul Karim
Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Intracranial hemorrhage is an acute bleeding within the skull which can damage the brain tissue and can eventually lead to disability or even death. It is a serious medical condition that occurs when blood is built up within the skull after a blood vessel is ruptured. Brain damage can be minimized if intracranial hemorrhage is diagnosed immediately, and the patient may regain mobility. Deploying applications of Artificial Intelligence (AI) in clinical medicine to accelerate the accuracy of intracranial hemorrhage diagnosis aims to minimize the severity of the condition, therefore, enhancing medical care. Adequate analysis of the Computed Tomography (CT) scan imaging is integral for diagnosis and management. Deep Learning, which is a subset of AI, is widely used in interpreting medical images and has shown promising advancements in diagnosing brain hemorrhage. With time playing a crucial factor, automatic lesion identification is one of the most important factors in precision medicine dealing with huge datasets of neuroimaging compared to manual lesion segmentation. This paper proposes a Deep Learning method called Convolutional Neural Network (CNN) on neuroimaging with transfer learning techniques to assist in the diagnosis of intracranial hemorrhage on CT scan images. We used six pretrained CNN models (EfficientNetB6, DenseNet121, ResNet50, InceptionResNetV2, InceptionV3, VGG16) and also present a traditional 11-layer CNN model for binary classification and detection of intracranial hemorrhage using brain CT scan images. The paper depicts a comparative analysis on the performance between the proposed traditional and pre-trained CNN models in terms of accuracy, precision, recall, F1 score, and AUC curve on the existing dataset. The EfficientNetB6 model yields an accuracy of 95.99%, which is higher than any of the experimental results of the CNN models used in this dataset.

Keywords: Deep Learning; Convolutional Neural Network; CT Scan Images; Intracranial Hemorrhage; EfficientNetB6; DenseNet121; ResNet50; InceptionResNetV2; InceptionV3; VGG16.

Dedication

We would like to dedicate our research to our beloved family members. Without their unwavering support, encouragement and faith in us, we would not be able to do our task successfully. We would also like to dedicate our humble efforts to all our well-respected teachers.

Acknowledgement

First and foremost, we are grateful to the Almighty Allah for providing us with the opportunity, direction and guidance to complete this research within time. Secondly, we wish to express our sincerest gratitude to our honorable thesis supervisor Dr. Mohammad Zavid Parvez and Co-supervisor Mr. Dewan Ziaul Karim who relentlessly supported, mentored and guided us through a challenging topic. We were able to overcome the obstacles because of their unwavering support and constant feedback. Despite an ongoing pandemic, they managed to spare time for us and offer constructive insights to improve our work and we will forever be grateful for that. Thirdly, we would like to take this chance to thank all of the faculty members for the help and support they have provided in our time in Brac University. Lastly, we would like to express our gratitude towards our beloved parents for their continued prayers, encouragement, and support.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AI	Artificial Intelligence
ANN	Artificial Neural Network
AUC	Area Under the Curve
CNN	Convolutional Neural Network
CT	Computed Tomography
FNN	Feedforward Neural Network
MRI	Magnetic Resonance Imaging
SVM	Support Vector Machine

Chapter 1

Introduction

1.1 Background

With the advent of technology surging through all spheres of life, AI has undoubtedly revolutionized many domains, with its largest impact in the healthcare industry. Integrating AI in the healthcare ecosystem has paved the way for continual growth and improvement in the medicine field and an increased revenue potential. The medical industry continues to evolve as the application of AI becomes further comprehensive and allows for a myriad of benefits, from analyzing big data sets of patients to automating tasks so as to deliver better healthcare faster. Deep learning is a sub-discipline of Artificial Intelligence's further subset called 'Machine Learning', that has shown potential and prospects to deliver data-driven clinical decision support by using data along with algorithms based on artificial neural networks with representation learning to confer invaluable automated insights to physicians and healthcare specialists.

The brain is the most complex organ in the human body. The human brain is very powerful and it posses nearly 100 billion neurons with nearly a quadrillion connections that is capable of generating the highest level of consciousness and the mental processes with which humans perceive, act, learn, and remember. Higashida and Chair (2003) classified the typology of stroke into two categories: Ischemic and Hemorrhagic [1]. The ischemic stroke is the most frequently occurring and it elucidates for 80% of all the strokes whereas hemorrhagic stroke accounts for about 20% [1]. An ischemic stroke develops when there is a lack of blood flow in the major arteries that lead to the brain. The aftermath of this stroke can result in a temporary or permanent loss of the body's normal functions [1]. On the other hand, a hemorrhagic stroke is caused by a bleeding in the brain [1]. Intracranial brain hemorrhage is a medical emergency and it is a serious type of hemorrhagic stroke that develops when a brain's blood vessel ruptures and this causes blood to build up within the skull as the oxygen supply is restricted [2]. Prompt treatment is vital to diminish the damage and save lives. The two modalities routinely used for mapping lesion in the brain are: computed tomography (CT) and magnetic resonance imaging (MRI). The preferred procedure is CT scan as the first step to assess a stroke patient as it has proven to be an efficient technique in determining if the individual is experiencing a stroke. A study suggests that more than 92% accuracy is achieved in identifying hemorrhage strokes by CT scans [3]. Therefore, with time playing a crucial factor, CT is the preferred approach with the advantages

of speed, expense and reduced exclusion measure corresponding to MRI [4].
In recent years, deep learning methods have been extensively used for the delineation of stroke on CT scans as these models are capable of outperforming humans in image classification. The medicine and healthcare industry has substantially profited from the application of Convolutional Neural Networks (CNN) which saves time and produces accurate results. CNN is effectual for the recognition of patterns and image-processing as this algorithm constructs a model which processes the images that are taken as inputs so as to extract the features from it as well as to discern a pattern [5]. CNN recognizes the similarities of a new input quite precisely by using the pattern and it is in demand because of its simple architecture, minimized training-parameters, pliancy, and it also reduces a network model's complexity.

1.2 Motivation

As today's medical industry continues to rapidly evolve, the concept of computer-based clinical decision support has accelerated as a prevalent topic in research so as to enhance the quality of decision-making in the medicine and healthcare domain. AI possess potential to optimize personalized care by facilitating diagnosis and therapeutic decisions. Hence, this research aims to find a way to develop an understanding of how incorporating the applications of AI accelerates the diagnosis of intracranial brain hemorrhage and create a solution that leverages on data accuracy for improved decision-making. The goal is to develop a solution using the concepts of Image Processing with Convolutional Neural Network models that will assist healthcare specialists to make an effective and improved diagnosis.

The motivations of this research are:

- To thoroughly understand the applications of Artificial Intelligence and how it can be incorporated in our interested domain.
- To study and examine deep learning models for the automatic lesion delineation and binary classification of intracranial brain hemorrhage on CT scan images.
- To employ traditional and transfer learning approach of Convolutional Neural Network and contrast their performance.
- To examine and evaluate the best performing model for our intended research.

1.3 Thesis Orientation

This segment gives an overall overview of what has been discussed in each chapter of this thesis paper. After discussing the motivation behind this study and what we intend to do and plan to accomplish in this chapter, the remainder of the paper is assembled in the consequential manner:

- Chapter 2 includes the Literature Review where we summarize related works and information collected from various scholarly articles relevant to this thesis
- Chapter 3 presents the research objectives in details
- Chapter 4 includes the methodology suggested for the whole study's workflow
- Chapter 5 describes the architecture for the traditional 11-layer CNN model proposed for the study
- Chapter 6 discusses the pre-trained CNN models used for Transfer Learning Approach
- Chapter 7 evaluates the performance of the CNN models
- Chapter 8 outlines the experimental results and analysis
- Chapter 9 concludes this thesis with subsequent plans

Chapter 2

Literature Review

As of today, stroke is a prominent cause of permanent disability in adults [6]. About 16 million first-ever strokes occur in the world, that causes a total of 5.7 million deaths annually [7]. With a globally ageing population estimated to triple by the year 2050, neurophysiological investigation of patients suffering from stroke with advances of AI in cognitive neuroscience will evolve the comprehension of the most complex human organ [6]. The stroke typology is broadly classified into two categories: Ischemic and Hemorrhagic. Ischemic stroke is the most frequent type which is caused by a blood clot that blocks the brain's blood vessels and nearly 87% of all strokes are ischemic stroke [8]. Hemorrhagic stroke is another major type of stroke in which the blood vessel of a brain ruptures and causes bleeding and it makes up about 13% of all strokes. This type of stroke is directly caused by Intracranial hemorrhages [2]. Intracerebral hemorrhage, which is a subtype of intracranial hemorrhage, occurs at a rate of 24.6 per million people globally.

The two modalities routinely used for mapping lesion in the brain are: computed tomography (CT) and magnetic resonance imaging (MRI). According to Baird et al. (2009), CT Scan is considered to be the standard convention for the precise exclusion of brain hemorrhage [9]. Over the years, notable progress was made in the scanner hardware, and the latest CT units can scan the entire human brain in a matter of a few seconds [10]. With time playing a crucial factor, CT is the preferred approach compared to MRI [4] . Using CT scan images, early detection of ischemic strokes was done by Rajni and Bhavani (2013) with segmentation, midline shift and image feature characteristics. In their research, they obtained accuracy scores of 98% with Support Vector Machine (SVM), 97% with k-NN, 96% with Artificial Neural Network (ANN) and 92% with decision tree [11]. The current method for lesion identification is still manual and that puts forward a number of disadvantages [12]. Even though hemorrhagic stroke appears more clearly on a CT scan image, lesion identification of the more common ischemic stroke takes nearly over a day using the manual delineation approach. Ischemic stroke is strenuous to observe in CT scans, particularly during the first few hours after the stroke occurs, which is the duration when treatment decisions are utmost vital. Therefore, by the time the region of the abnormal brain tissue is localized, delay of treatment propels the brain damage thus likely to worsen the individual's chances to regain mobility [12]. CT scan merges a procession of X-ray images taken from varied angles and employs computer imaging to construct cross-section images of blood vessels, soft tissues and bones within the body. Initially, the CT scan images are in the transverse anatomical plane. State-of-the-art scanners scan data to be formatted as images on alternative planes. Digital geometry analysis can produce a 3D image of an object within the body from a collection of 2D radiographic images taken by rotating around a fixed object. The features of the lesion of a stroke differ depending on the imaging modality type and so these features must be cautiously extracted from the input images to establish an accurate procedure for stroke delineation in order to analyze the various deep architectures used for stroke diagnosis and segmentation depending on the underlying imaging modality. It can further propose other potential deep architectures that can be suggested to improve outcomes in the identification of stroke lesions and the emerging developments in hemorrhage delineation have also been detailed in this evaluation. In recent years, deep learning methods have been extensively used for the detection of hemorrhage on CT scans. Deep learning Artificial Intelligence models are capable of outperforming humans in image classification. This also helps to explain the importance of two deep learning models to the medical image processing, namely the Convolutional Neural Network (CNN) and the fully Convolutional Network (FCN).

The detection of 'stroke' using CNN is a prevalent and extensive research domain. In 2018, Grewal et al. utilized DenseNet along with a bi-directional long short-term memory (Bi-LSTM) layer aimed at hemorrhage diagnosis [13]. In their research, they used a dataset of 77 brain CT scans on which the LSTM layer was incorporated for combining dependencies between slices and they named this model Recurrent Attention DenseNet (RADnet) which achieved 81.82% hemorrhage prediction accuracy, 88.64% sensitivity and 81.25% precision that can be comparable to radiologists for CT scan images. Although the types of intracranial hemorrhage examined were not mentioned in the paper [13]. In the same year, Chilamkurthy et al. employed a modified version of ResNet18 in order to delineate intracranial hemorrhage on a dataset of 21,095 CT scans and obtained an AUC score of 91.94% [14]. In another research, Arbabshirani et al. suggested a simple CNN that consisted of two fully connected layers and five convolutional layers on a dataset of 37,074 CT scan images on which they attained an AUC score of 84.6%, sensitivity score of 73%, and specificity score of 80% [15]. However, their model does not specify the location of the hemorrhage in the brain.

The dataset used in this thesis was collected by Hssayeni et al. (2020) [16]. In their paper, they employed U-Net to detect intracranial hemorrhage and achieved an accuracy of 87.00%. Their model had a high rate of false positives that influenced towards a lower dice score (0.31). In 2020, Anupama et al. developed a GC-SDL model(GrabCut-based segmentation with synergic deep learning) that can identify intracranial brain hemorrhage images in wearable networks [17]. Their proposed method used Gabor filtering to improve the image quality by noise removal and attained a precision of 95.79% and an accuracy of 95.73%. In the same year, Chen et al. presented an IoT-based implementation for hemorrhage classification using machine learning algorithms [18]. In their paper, the accuracy obtained was 80.67% for Support Vector Machine (SVM) and 86.70% for Feedforward Neural Network (FNN).

Li et al. also used this dataset for their research [19]. They proposed a UNet++ model which achieved a detection accuracy of 0.9859 along with a Dice score of 0.8033 on one of their datasets. However, they could not get satisfactory performance for their second dataset, which is the one being used for this paper. Majumdar et al.

used a dataset of 134 CT cases containing 4,300 images [20]. They created a modified version of the U-Net model for detecting intracranial hemorrhages which worked on 2D slices. Their model consisted of 9 convolutional layers and they found that using data augmentation improved the accuracy of the model. For augmentation, they used random flipping and random rotation of 10 degrees in either direction. Their model attained a sensitivity score of 81% per lesion and 98% specificity per case. However, for training their model, they only used cases with hemorrhages while keeping the normal cases only for testing, which might distort the results. Prevedello et al. [21] proposed two models that utilized CNNs. The former focuses on predicting intracranial hemorrhage, mass effect, and hydrocephalus (HMH) localization whilst the latter is utilized to find suspected acute infarct (SAI). These two models combine elements of CNN with recurrent neural networks (RNN) to detect hemorrhage. They used a dataset of 50 scans and found an AUC score of 0.91 on HMH and 0.81 on SAI. Chang et al. created a hybrid CNN model using slice slabs [22]. In their study, a dataset containing 10,159 Computed Tomography scans was utilized for training. They gained an accuracy of 97.5%. However, their dataset had a small quantity of cases that were positive, only 82 out of the 983 cases were positive and they also did not categorize the subtypes of ICH [22]. For detecting ICH, Jnawalia et al. suggested an ensemble of three distinct CNN algorithms [23]. These models are based on the AlexNet and GoogleNet structures, which were stretched to a 3D method by absorbing all slices. Their dataset consisted of 40,367 3D head CT scan images and they achieved an AUC score of 0.87 on the ensemble model.

For detecting and categorizing ICH regions, Ye et al. proposed a three-dimensional convolutional and recurrent (CNN-RNN) model [24]. The CNN model used was VGG16, and the RNN technique used was the bidirectional gated recurrent unit (GRU). Their dataset contained 2836 cases among which 1836 had ICH and 1000 were normal and the total number of images were 76,621. They gained AUC scores of > 0.8 AUC all subtypes. However, the amount of cases with intracranial hemorrhage was 65% which is significantly exorbitant in comparison to an actual clinical environment [24]. In our research, we investigate and examine the different approaches used and then suggest an improved solution influenced from the knowledge of the previous workings.

Chapter 3

Research Objectives

Some of the papers mentioned in Chapter 2 have attained significant outcomes on the specific dataset we are working with. Nonetheless, we found some constraints in the existing research reports which are noted as follows: (1) Some only focused on one single pre-trained CNN model in their paper and did not outline any comparison of its performance with other existing transfer learning methods [13], [14], (2) others did not achieve satisfactory results on the dataset we have used in this paper [18]. Hence, in this thesis, we suggest the implementation of transfer learning approach of six different pre-trained CNN models (EfficientNetB6, DenseNet121, ResNet50, InceptionResNetV2, InceptionV3, VGG16) and a simple traditional CNN model based on 11-layer architecture and make a comparative analysis of their performance to address the limitations mentioned as above. We employ the models for the binary classification and the automated detection of intracranial brain hemorrhage on brain CT scan images using the transfer learning approach of CNN. The significance of it is an attempt to assist the healthcare experts by reducing the time required for the detection of this particular medical condition and thus improving clinical diagnosis. The principal contributions of this paper are mentioned as follows:

1. This research employs more than one transfer learning CNN model. We have used six pre-trained CNN models and a traditional 11-layer CNN model that can automatically identify infarct and bleeding in Intracranial Hemorrhage CT scans.

2. Some of the pre-trained models' performance have yielded satisfactory results which is remarkably enhanced than existing research works done on this specific dataset.

3. This paper outlines a comparative analysis on the performance of the CNN models employed in this thesis.

4. A real-world application of this study is implemented by deploying a simple web application.

Chapter 4

Methodology

4.1 Workflow of the Methodology

In this segment, the methodology suggested for this thesis is demonstrated. We begin the workflow with the collection and organization of the dataset followed by applying pre-processing techniques to it. The workflow comprises of a traditional CNN model of 11-layered architecture and the transfer learning approach of six pre-trained CNN models (EfficientNetB6, DenseNet121, ResNet50, InceptionResNetV2, InceptionV3, VGG16) and then compare their performance based on the metrics of accuracy, precision, recall, F1 score, Confusion Matrix and AUC curve to eventually evaluate the best model for the detection and binary classification of intracranial hemorrhage on brain CT scan images. Figure 4.1 represents a detailed block diagram of the methodology's workflow. The methodology is sequentially outlined in the given steps as shown below:

Step 1: Data Collection
Step 2: Data Pre-processing
Step 3: Traditional CNN model
Step 4: Pre-trained CNN models for Transfer Learning
Step 5: Evaluate the CNN models' performance

4.2 Description of the Data

4.2.1 Data Collection

The dataset is acquired from the physionet repository [16]. It contains head CT scan images in JPG format. The images are from a study sanctioned by the Research and Ethics Board in Iraqi Ministry of Health-Babil Office, which was conducted in Al Hilla Teaching Hospital-Iraq in 2018 [25]. Their study comprises a total of 82 patients of which 46 are male subjects and 36 are female subjects. Only 31 out of the 82 patients are diagnosed with intracranial hemorrhage. Hence, for every patient on average 30 images are available. Table 4.1 represents the subject demographics. From the dataset, a total of 5001 images are accumulated among which 2501 images are of brain CT scans and the rest 2500 images are that of bones. We discard the bone images since we only need the brain CT images to train the CNN models for

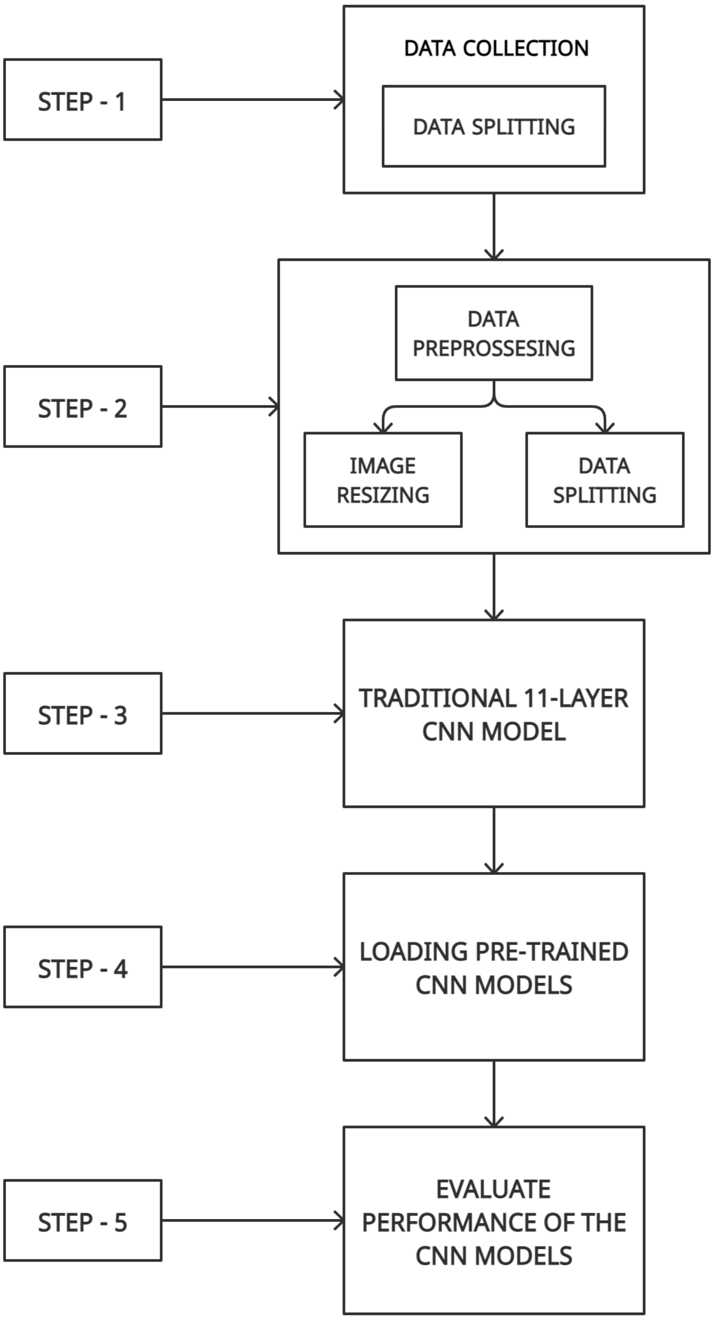


Figure 4.1: Full Workflow of the Methodology.

this research. For the purpose of binary classification, these 2501 brain images are then assembled into two groups - Hemorrhage and Normal. That is, 1104 images of hemorrhage and 1397 normal brain images are used in the training of these deep learning CNN models. Figure 4.2 shows a visual representation of this ratio of hemorrhage and normal brain images. The improvised dataset is further divided into a ratio of 9:1. That means 90% of this dataset are employed for 'training' and the spare 10% are utilized for the purpose of 'testing'. Then 20% out of this training dataset are kept for validation which is used to dispense an unbiased evaluation of a model fit on the training dataset. So, the dataset splitting is of the following ratio 70:20:10 for training, validation, and test respectively which is shown in Fig 4.3.

Table 4.1: Subject Demographics

Total Number of Subjects	82	Sex	46 Male
			36 Female
Age	27.8 ± 19.5	Age Group	1 day - 72 years
Quantity of Normal Images	46	Number of subjects with ICH	36

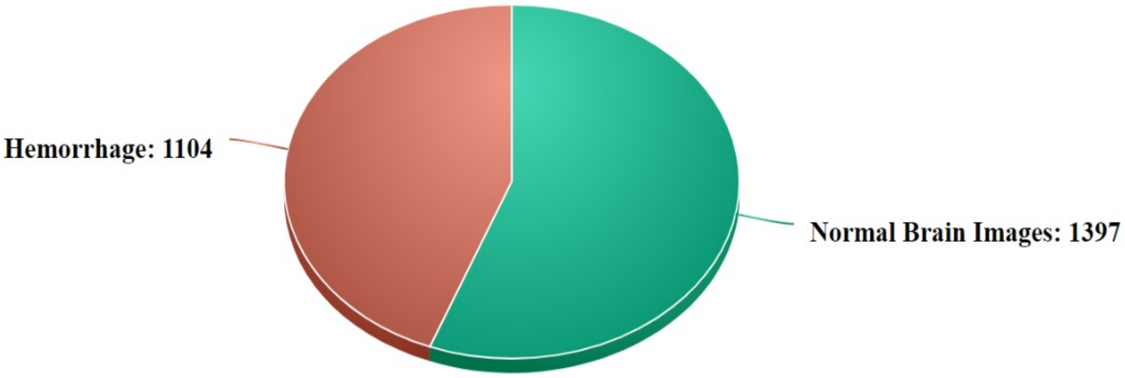


Figure 4.2: Ratio of Hemorrhage and Normal Brain Images

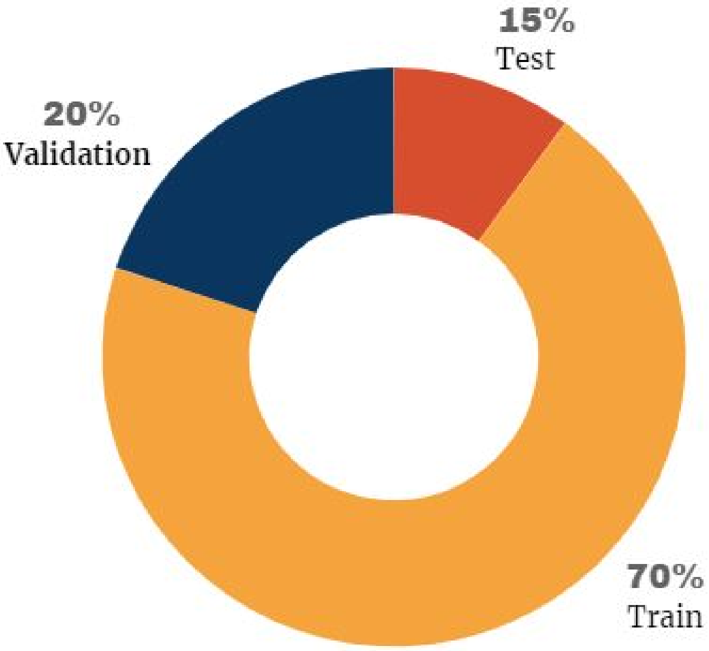


Figure 4.3: Dataset Splitting

4.2.2 Data Pre-processing

Data pre-processing is a method which is designed to remove unnecessary variables that do not contribute to improving the accuracy of the CNN models. Moreover, it makes all the necessary transformations on the raw data that can help in improving the CNN models' performance to give better results and accuracy [26]. Our data pre-processing is segmented into two parts: a) Image Resizing, and b) Data Augmentation. The dataset consists of images of various sizes which can impact and affect the architecture to give low accuracy. Therefore, these images are resized using the open-source Keras library. The input shape is taken as 224x224 pixels for all the models as most architectures downsample the input size of the images to 224x224. This resizing helps to considerably reduce the training time of the neural network models. We set the batch size to 32. Furthermore, data augmentation is applied on our training dataset to handle the problem of sparse data. It is used to artificially increase the dimensions of the training dataset by generating modified copies of the dataset's images [27]. The following geometric transformations were applied:

- RandomFlip
- RandomZoom
- RandomCrop
- RandomRotation

Figure 4.4 shows a visual representation of a brain CT scan image before and after data augmentation is performed on it.

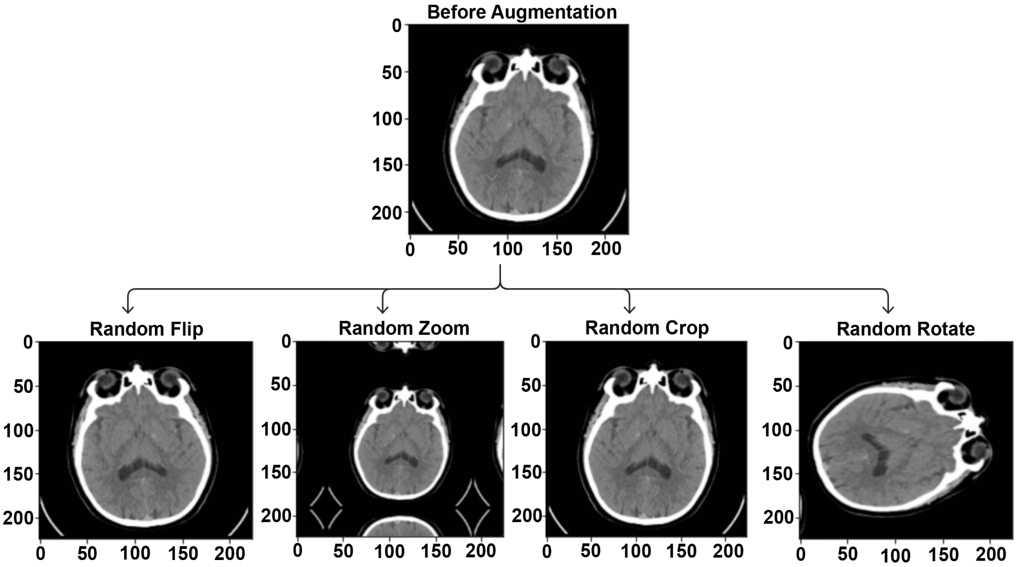


Figure 4.4: Before and After Augmentation.

Chapter 5

Proposed Traditional CNN Model

The architecture of the traditional 11-layer CNN model is discussed in this section. Table 5.1 depicts a summary of this proposed traditional 11-layer CNN model. A traditional CNN is primarily constructed with convolutional layer, pooling layer and fully-connected layer. Additionally, the other two principal parameters used are: Activation function and Dropout layer. Along with that regularization type L1 and L2 with a dropout of 40% was added to insure that the model do not get overfitted.Figure 5.1 shows a visual representation of a traditional CNN model. A description of the layers that are selected in the traditional 11-layer CNN model are given below:

1. **Convolutional Layer:** A total of four Conv2D layers have been selected in this layer.
2. **Pooling Layer:** For this layer, Maxpooling2D is used for every Conv2D layer. So the number of layers sum up to four in the Pooling layer.
3. **Fully Connected Layer:** Weights, biases and the neurons are incorporated in this layer and it is composed of :
   (a) Flatten Layer: This layer is applied after selecting the pooling layer so that it can flatten the whole network.
   (b) Dense Layer: Two dense layers are employed after the flatten layer in the model so as to feed all the outputs from the previous layer to all its neurons.
   (c) Activation Function: The ‘Sigmoid’ Activation Function is implemented on the dense layer. The equation of this function is specified in Eq 5.1.

$$f(x) = \frac{1}{1+e^{-x}} \tag{5.1}$$

In Eq 5.1, x is a real number and a trivial constant. The ’sigmoid’ activation function is an S-shaped curve and the value of this ranges from 0 to 1, which means it can easily predict the probability. Hence, the sigmoid function has been selected as the preferred choice for this type of binary image classification.

Table 5.1: A SUMMARY OF THE TRADITIONAL 11-LAYER CNN MODEL

Type of Layer	Output Shape	Parameter #
conv2d (Conv2D)	(None, 254, 254, 32)	896
max pooling2d (MaxPooling2D)	(None, 127, 127, 32)	0
conv2d-1 (Conv2D)	(None, 125, 125, 64)	18496
max-pooling2d-1 (MaxPooling2D)	(None, 62, 62, 64)	0
conv2d-2 (Conv2D)	(None, 60, 60, 128)	73856
max pooling2d-2 (MaxPooling2D)	(None, 30, 30, 128)	0
conv2d-3 (Conv2D)	(None, 28, 28, 128)	147584
max pooling2d-3 (MaxPooling2D)	(None, 14, 14, 128)	0
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 512)	12845568
dense -1 (Dense)	(None, 1)	513

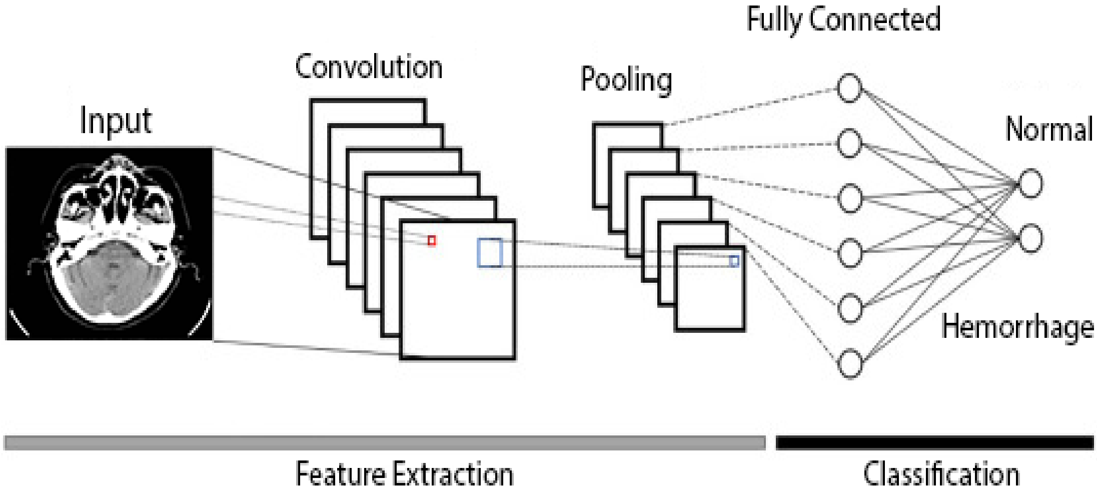


Figure 5.1: Architecture of Traditional CNN Model

Chapter 6

Pre-trained CNN Models for Transfer Learning

The pre-trained CNN models employed in this paper are EfficientNetB6, DenseNet121, ResNet50, VGG16, InceptionV3, and InceptionResNetV2. Since the traditional 11-layer CNN model performed poorly and did not yield a satisfactory result because the dataset used is small with only about 2500 images. So in order to optimize the performance of these neural network models, we migrated to transfer learning approach. We have also added 40% dropout which implies that 40% features are set to 0 during training. But while testing, all the neurons are used and so the model will become more robust. Moreover, we also used regularization techniques of type L1 and L2 to prevent the models from overfitting. A summary of each of the pre-trained models' architecture used in this paper are mentioned as follows:

6.1 EfficientNetB6

Mingxing Tan and Quoc V. Le (2019) proposed the EfficientNet model [28]. Figure 6.1 shows a visual representation of the EfficientNetB6 Architecture. They examined at model scaling and determined that balancing the depth, width, and resolution of the network can result in a good result, performance can be improved. Therefore, they developed a new scaling method for neural networks which can uniformly scale the network's depth, width, and resolution. Then they scaled it up to build the EfficientNets family of deep learning models, which outperformed the previous models in terms of accuracy and efficiency. [28]. The compound scaling method was used at first in order to scale the dimensions of the network. A grid search approach was utilized to discover the relationship between the network's various scaling dimensions. In this way, they discovered the scaling coefficients necessary for scaling up each dimension [28]. In the architecture of the EfficientNet model, they used Neural Architecture Search to automate the creation of the design. In this way, both the efficiency and the accuracy can be optimized on the basis of FLOPS or floating-point operations per second. MBConv or Mobile Inverted Bottleneck Convolution is used in the EfficientNet architecture [28].

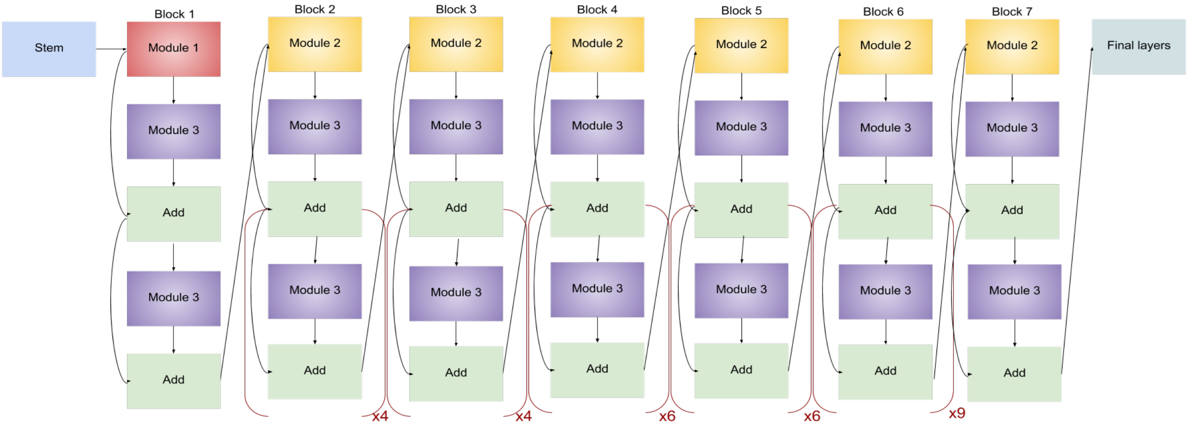


Figure 6.1: EfficientNetB6 Architecture [28].

6.2 DenseNet121

DenseNet (Dense Convolutional Network) is a network architecture designed to increase the depth of deep learning networks while making training easier and more efficient by employing shorter layered connections.DenseNet model is basically a convolutional neural network where each of its layer is connected to all the other layers further deep in the network; for example, the first layer is connected to the second, then to third, then to fourth so on and so forth [29]. As a result, the model requires fewer parameters than a typical CNN because no redundant feature mappings must be learned [30]. Therefore it is such an efficient and widely used model. Table 6.1 shows a summary of the DenseNet121 model.

Table 6.1: Summary of DenseNet121 Model

Layers	Output Size	DenseNet121
Convolution	112×112	
Pooling	56×56	
Dense Block-1	56×56	$[\ 1 \times 1\ ] \times 6$, $[\ 3 \times 3\ ] \times 6$
2-Transition Layer (1)	56×56	
	28×28	
Dense Block-2	28×28	$[\ 1 \times 1\ ] \times 12$, $[\ 3 \times 3\ ] \times 12$
2-Transition Layer (2)	28×28	
	14×14	
Dense Block-3	14×14	$[\ 1 \times 1\ ] \times 24$, $[\ 3 \times 3\ ] \times 24$
2-Transition Layer (3)	14×14	
	7×7	
Dense Block-4	7×7	$[\ 1 \times 1\ ] \times 16$, $[\ 3 \times 3\ ] \times 16$
2-Classification Layer	1×1	

Every dense block includes a different number of layers by repeating the same layers along with two convolutions each. A 1x1 sized kernel gets added as the bottleneck layer and a 3x3 kernel gets added in order to perform the convolution function, as shown in table 6.1. Again, each transition layer contains a 1x1 convolutional layer and a 2x2 average pooling layer as well with a stride of 2. Hence DenseNet-121 gets the following layers in its architecture - one 7x7 Convolution, fifty-eight 3x3 Convolution, sixty-one 1x1 Convolution, four AvgPool and one Fully Connected Layer. Building up to 120 Convolutions and 4 AvgPool. A visual representation of the DenseNet121 Architecture is presented in Figure 6.2.

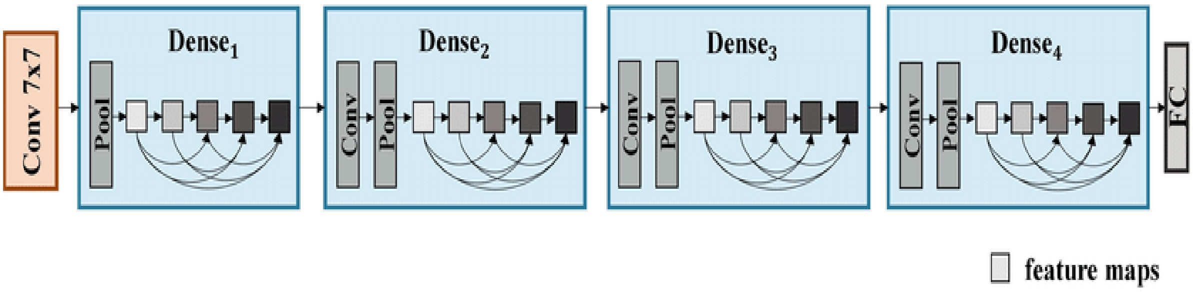


Figure 6.2: DenseNet121 Architecture [29].

6.3 ResNet50

ResNet or Residual Networks is a neural network architecture which is used for computer vision related tasks. ResNet won the 2015 ImageNet challenge. Using ResNet, training very deep neural networks containing over and above 150 layers had become possible, which was a major breakthrough. ResNet comes in a variety of flavors, each with a different number of layers but the same basic premise. The term ResNet50 refers to a variation that can work with up to 50 neural network layers. ResNet solves the vanishing gradient problem successfully. Because if the network is very deep, then the calculation of the gradients can get to zero when the chain rule is applied many times. This means that the weights are never updated and no learning occurs in the model [30]. Figure 6.3 shows a visual representation of the ResNet50 Architecture.

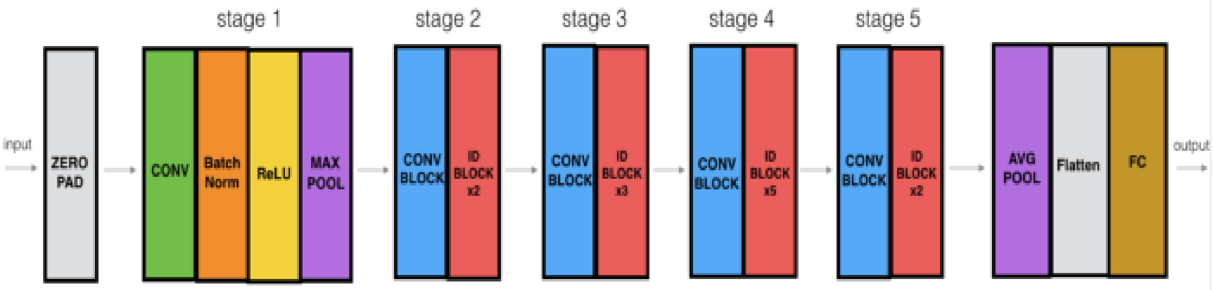


Figure 6.3: ResNet50 Architecture [30].

Table 6.2: Summary of ResNet50 Model

Layer Name	Output Size	ResNet50
Conv1	112 ×112	7 × 7, 64 stride 2
Conv2	56 × 56	3 × 3 max pool, stride 2
		[1 × 1, 64] × 3 [3 × 3, 64] × 3 [1 × 1, 256] × 3
Conv3	28 × 28	[1 × 1, 128] × 4 [3 × 3, 128] × 4 [1 × 1, 512] × 4
Conv4	14 × 14	[1 × 1, 256] × 6 [3 × 3, 256] × 6 [1 × 1, 1024] × 6
Conv5	7 × 7	[1 × 1, 512] × 3 [3 × 3, 512] × 3 [1 × 1, 2048] × 3

The ResNet 50 architecture, as shown in Table 6.2, is made up of a convolution layer with 7 × 7 sized kernel as well as 64 unique kernels, all of these kernels also have a stride size of 2. All of these together, is making up in a single layer. For this first convolution layer, a max pooling layer with a stride size of 2 has also been used.This model has 1 × 1, 64 kernel and a 3 × 3, 64 kernel, along with a 1 × 1, 256 kernel. These three layers has been repeated three times that gives the model nine levels. Afterwards, there is a kernel of 1 × 1, 128, again with with another kernel of 3 × 3, 128, along with another kernel of 1 × 1, 512. However, These layers are get repeated four times giving the model another 12 layers along with the first 9. Again,there is a 1 × 1, 256 kernel,a 3 × 3, 256 kernel and a 1 × 1, 1024 kernel has been added and repeated six times, resulting in adding another total of 18 layers. In the last parts of this model a 1 × 1, 512 kernel, two 3 × 3, 512 and 1 × 1, 2048 kernels get added too, which gives total of nine layers. Finally,an average pool is run and the model is completed with a fully connected layer with 1000 nodes and subsequently a softmax function gets attached. However, this activation functions as well as the max/average pooling layers do not get counted. All of the mentioned layers together actually makes the total number of layers for the ResNet50 architecture that eventually accumulates to 50 layers.

6.4 VGG16

Simonyan and Zisserman (2014) had suggested the VGG16 CNN model [31]. On ImageNet, a dataset with 14 million pictures divided into 1000 classes, VGG16 obtained a test accuracy of 92.7%. This model comprises 13 convolutional layers, 5 max-pooling layers, and 3 fully connected layers, and it was trained for weeks using NVIDIA Titan Black GPUs. As a result of the AlexNet ReLU tradition, there are 16 layers with customizable settings, 13 convolutional layers, and 3 fully linked layers. This is the reason why the model was named as VGG16 [31]. This network adds more layers to AlexNet and employs lower size filters (2x2) and (3x3) than AlexNet. It has 138 million parameters and takes up around 500 megabytes of storage [32]. Table 6.3 shows a summary of the VGG16 model. Here, the first block contains 64 filters, which are then doubled in the subsequent blocks until the total number of filters reaches 512. Two fully connected hidden layers and one output layer complete this model. The neuron numbers in the two fully connected layers are the same, at 4096. The output layer has 1000 neurons, which corresponds to the Imagenet dataset's number of categories [32]. Figure 6.4 shows a visual representation of the VGG16 Architecture.

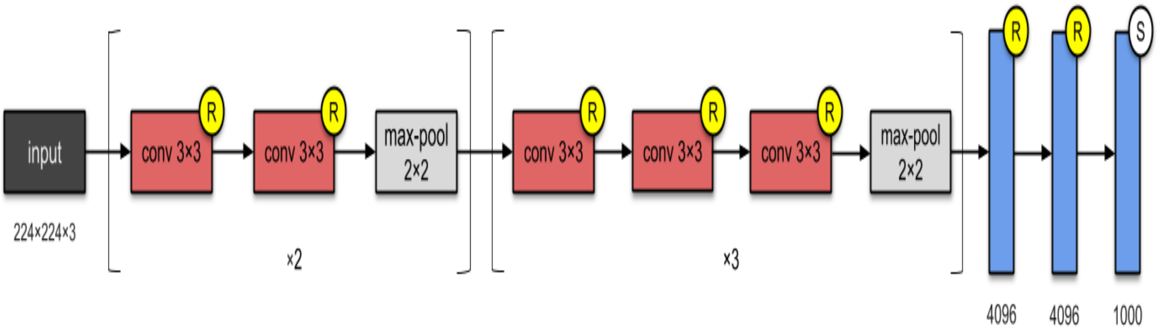


Figure 6.4: VGG16 Architecture [31].

Table 6.3: Summary of VGG16 Model

Type of Layer	Shape of the Output	Parameter #
conv2d (Conv2D)	(None, 224, 224, 64)	1792
conv2d-1 (Conv2D)	(None, 224, 224, 64)	36928
max_pooling2d (MaxPooling2D)	(None, 112, 112, 64)	0
conv2d-2 (Conv2D)	(None, 112, 112, 128)	73856
conv2d-3 (Conv2D)	(None, 112, 112, 128)	147584
max_pooling2d-1 (MaxPooling2D)	(None, 56, 56, 128)	0
conv2d-4 (Conv2D)	(None, 56, 56, 256)	295168
conv2d-5 (Conv2D)	(None, 56, 56, 256)	590080
conv2d-6 (Conv2D)	(None, 56, 56, 256)	590080
max_pooling2d-2 (MaxPooling2D)	(None, 28, 28, 256)	0
conv2d-7 (Conv2D)	(None, 28, 28, 512)	1180160
conv2d-8 (Conv2D)	(None, 28, 28, 512)	2359808
conv2d-9 (Conv2D)	(None, 28, 28, 512)	2359808
conmax_pooling2d-3 (MaxPooling2D)	(None, 14, 14, 512)	0
conv2d-10 (Conv2D)	(None, 14, 14, 512)	2359808
conv2d-11 (Conv2D)	(None, 14, 14, 512)	2359808
conv2d-12 (Conv2D)	(None, 14, 14, 512)	2359808
max_pooling2d-4 (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 4096)	102764544
dropout (Dropout)	(None, 4096)	0
dense-1 (Dense)	(None, 4096)	16781312
dropout-1 (Dropout)	(None, 4096)	0
dense-2 (Dense)	(None, 1000)	4097000

6.5 InceptionV3

By altering Inception designs, Inception V3 primarily focuses on using less computing power. Table 6.4 shows a summary of the InceptionV3 model. This sparsely linked architecture's key notion is the inception layer which comprises 11 convolutional layers with their output filter banks concatenated into a single output vector. This vector then performs the function as the input of the next stage [33]. Figure 6.5 shows a visual representation of the InceptionV3 Architecture.

Table 6.4: Summary of InceptionV3 Model

Type	Kernel Size / Stride	Input Size
Conv	3 × 3/2	299 × 299 × 3
Conv	3 × 3/1	149 × 149 × 3
Conv Padded	3 × 3/1	147 × 147 × 32
Pool	3 × 3/2	149 × 149 × 64
Conv	3 × 3/1	73 × 73 × 64
Conv	3 × 3/2	71 × 71 × 80
Conv	3 × 3/1	35 × 35 × 192
3 × Inception	1 × 1 and 3 × 3/1	35 × 35 × 288
5 × Inception	n × 1, 1 × n, and n × n/2	17 × 17 × 768
2 × Inception	1 × 1, 1 × 3, 3 × 1 and 3 × 3/2	8 × 8 × 1280
Pool	8 × 8	8 × 8 × 2048
Linear	Logits	1 × 1 × 2048
Softmax	Classifier	1 × 1 × 1000

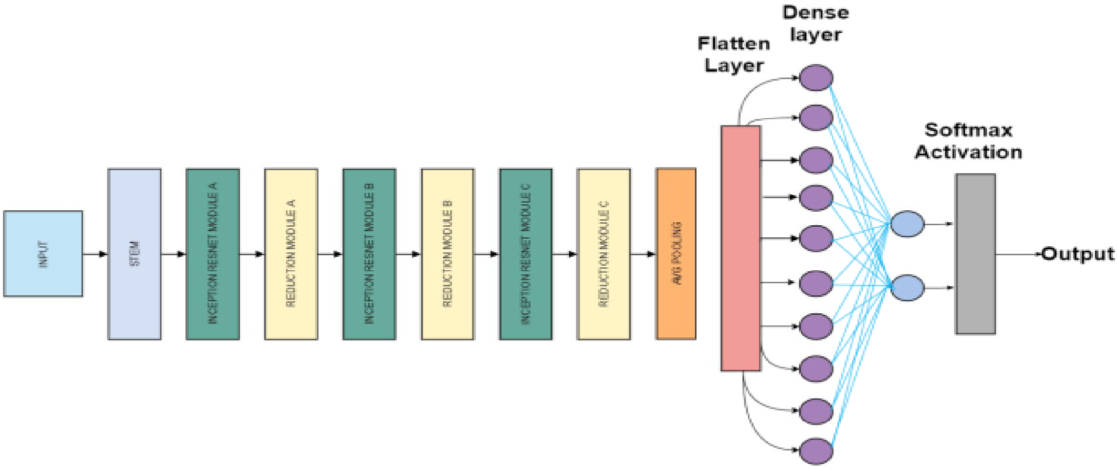


Figure 6.5: InceptionV3 Architecture [34].

6.6 InceptionResNetV2

InceptionResNetv2 is a CNN architecture that is an advanced form of the Inception family of architectures [35]. However, unlike regular Inception models, it includes residual connections instead of the architecture's filter concatenation method. The model is trained on the ImageNet database comprising more than a million images. The architecture has 164 layers. It has blocks called Inception-ResNet blocks where residual connections are implemented in synchronization with multiple convolutional filters [35]. The basic network architecture of InceptionResNetV2 is shown in Figure 6.6. On the traditional layers, batch normalization is used, but not on the summations of the residual connections. Residual connections are used as shortcuts in the model and which enables the model to have better performance [35].

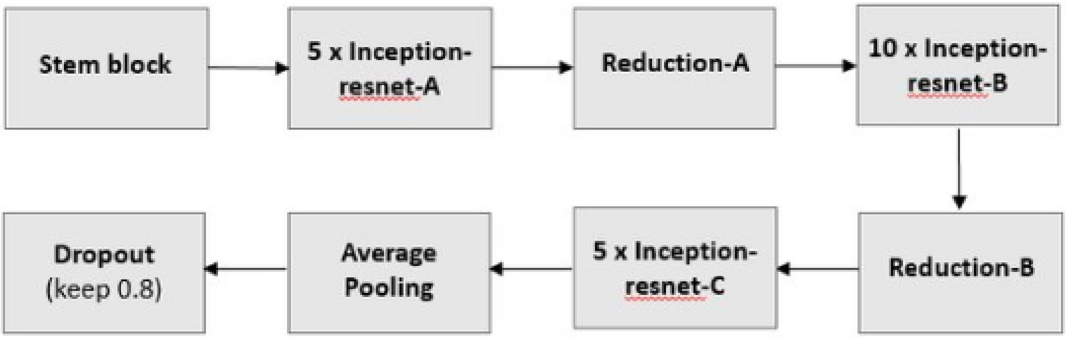


Figure 6.6: InceptionResNetV2 Architecture [35].

Chapter 7

Performance Evaluation of the CNN Models

With the aim of analysing the performances of the traditional and pre-trained CNN models so as to evaluate and outline a comparative analysis of the results; F1 score, precision, accuracy, recall, Confusion Matrix and AUC curve have been calculated for each of these models. In this chapter, we first present the equations of the performance metrics used for the study. Then we demonstrate the AUC Graphs and the Confusion Matrices of each of the Convolutional Neural Network models subsequently.

7.1 Performance Metrics

The equations of the performance metrics used are specified as follows:

- **Accuracy Formula [36] :**

$$\begin{aligned} Accuracy &= \frac{TP+TN}{P+N} \\ &= \frac{TP+TN}{TP+TN+FP+FN} \end{aligned} \tag{7.1}$$

- **Precision Formula [36] :**

$$\begin{aligned} PPV &= \frac{TP \times TPR}{TP+FP} \\ &= 1 - FDR \end{aligned} \tag{7.2}$$

- **Recall Formula [36] :**

$$\begin{aligned} PPV &= \frac{TP \times TPR}{TP+FN} \\ &= 1 - FDR \end{aligned} \tag{7.3}$$

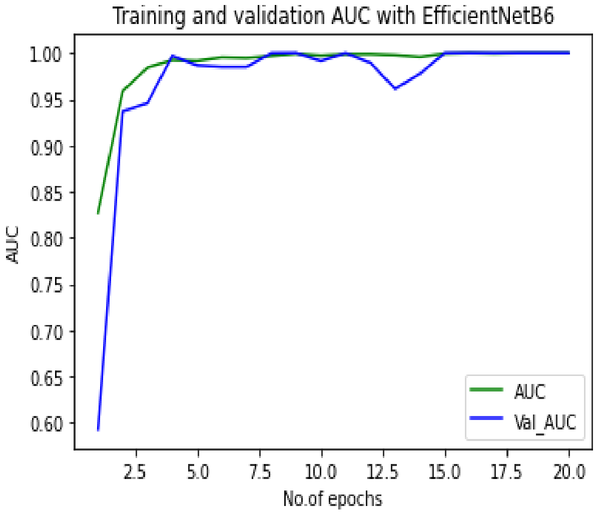
- **F1-Score Formula** [36] **:**

$$
\begin{aligned}
F1 &= 2 \times \frac{PPV \times TPR}{P + N} \\
&= \frac{2 \times TP}{(2 \times TP) + FP + FN}
\end{aligned}
\tag{7.4}
$$

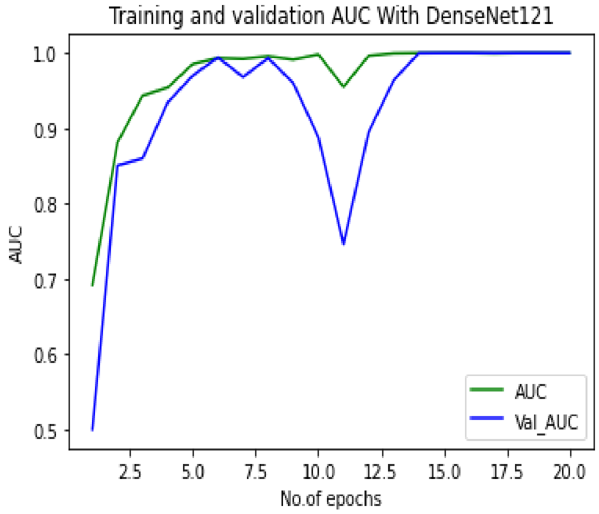
Here, the respective abbreviations are - TP = True Positive, TN = True Negative, P = Positive Case, N = Negative Case, FP = False Positive, FN = False Negative, PPV = Positive Predictive Value, TPR = True Positive Rate, FDR = False Discovery Rate

7.2 AUC Graphs

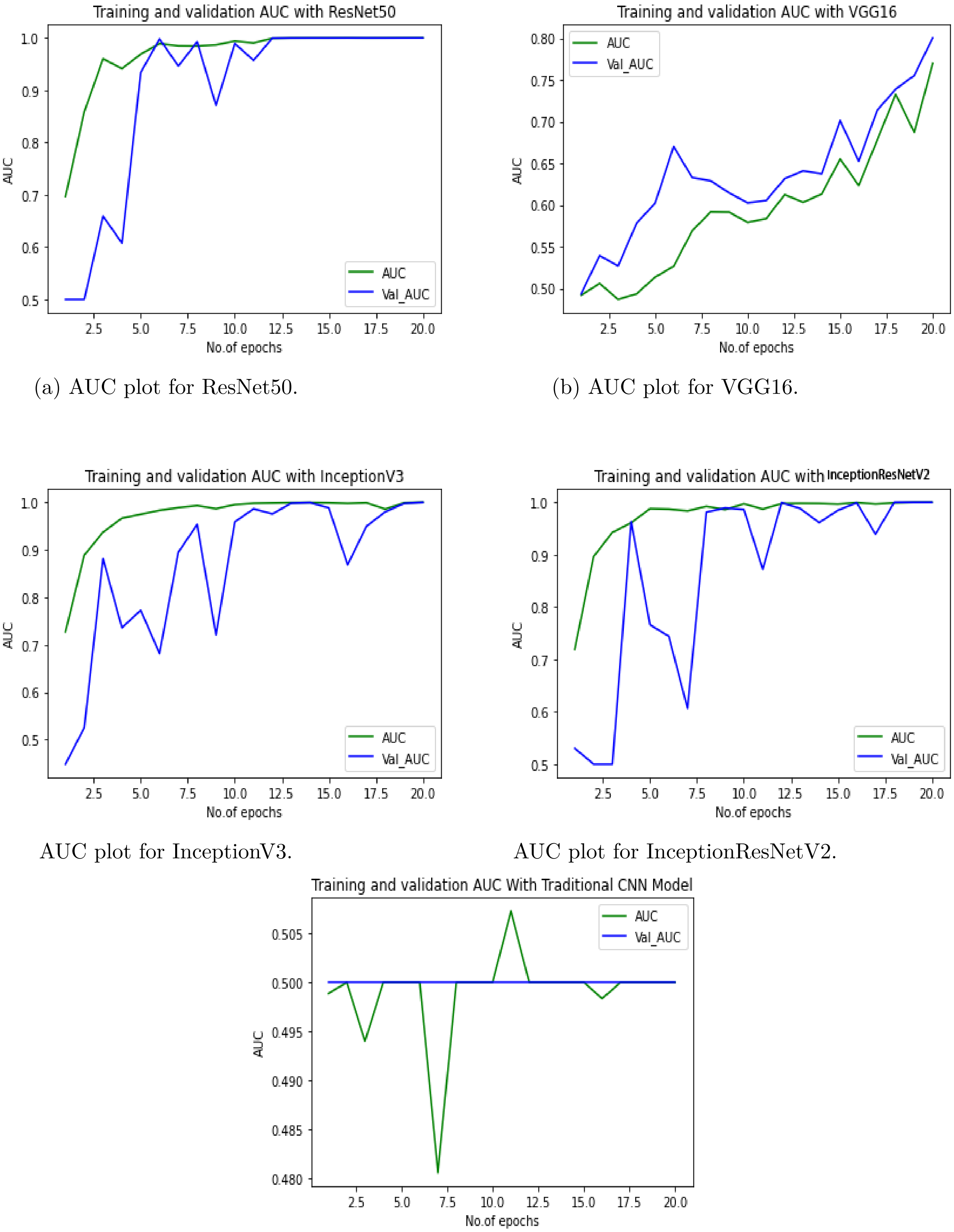
The Area Under the Curve (AUC) is the measurement of a classifier's ability to differentiate among the classes [37]. Because AUC is scale-invariant as well as classification-threshold-invariant, it not only measures the degree of separability but also offers an aggregated measure of performance over all conceivable classification thresholds. The higher the AUC score, the enhanced is the model's performance at distinguishing between the positive and negative classes. The AUC plots of all the CNN models used in this thesis are shown in Figure 7.2.



AUC plot for EfficientNetB6.



AUC plot for DenseNet121.



(a) AUC plot for ResNet50.

(b) AUC plot for VGG16.

AUC plot for InceptionV3.

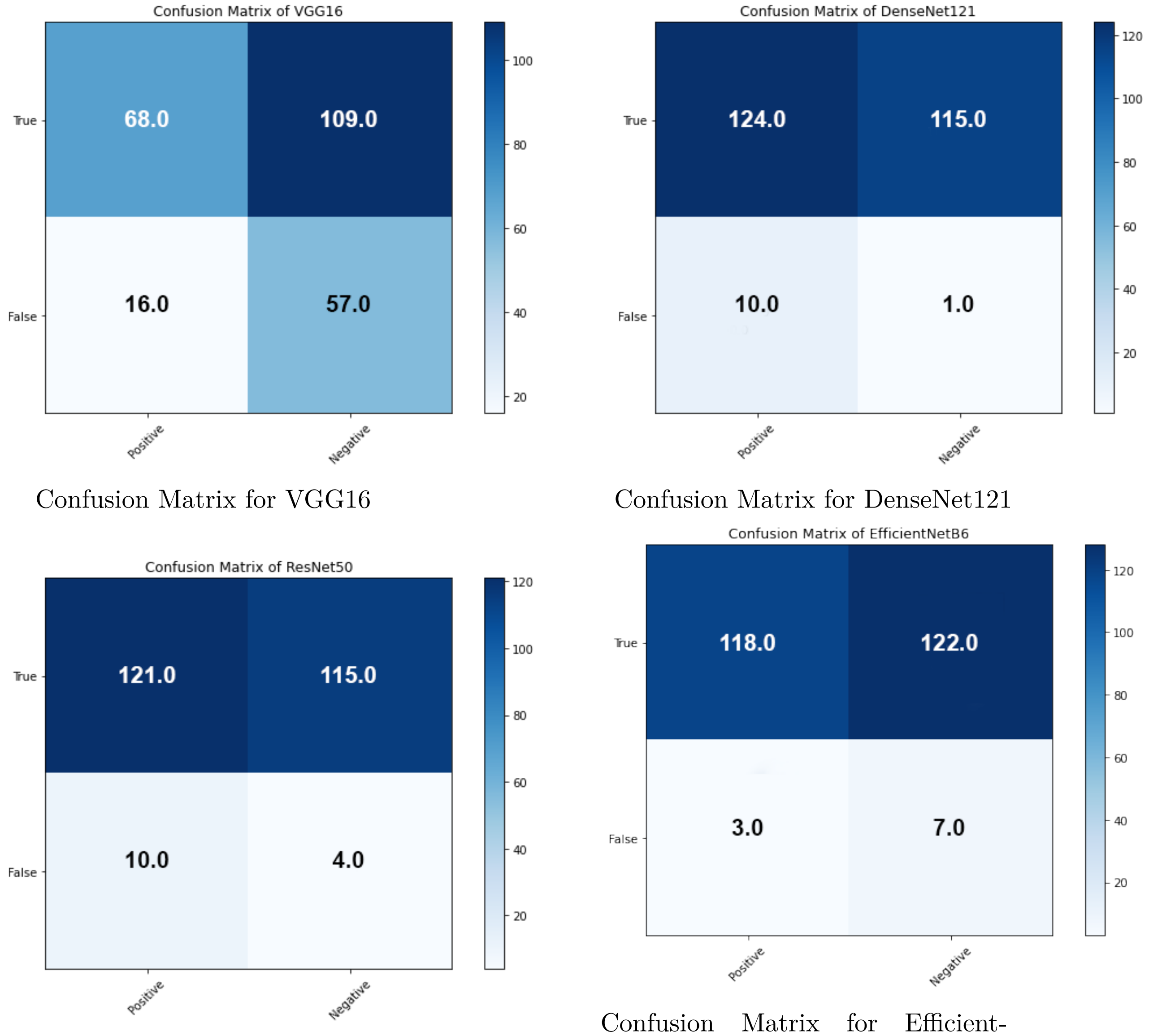
AUC plot for InceptionResNetV2.

AUC plot for Traditional CNN Model.

Figure 7.3: Training and Validation AUC Graphs of all the Models

7.3 Confusion Matrix

A confusion matrix is used to evaluate and summarize the performance of a classification model. Hence, we have constructed confusion matrices of each of the CNN models employed in this thesis using the test data at the end of training. The Confusion Matrices of all the CNN models are shown in Figure 7.4.

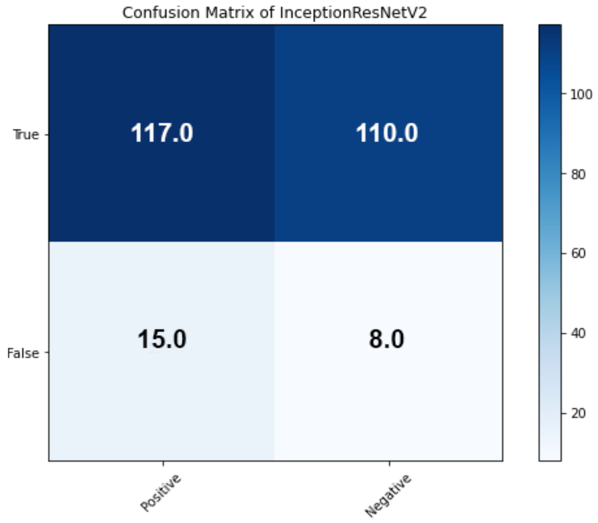


Confusion Matrix for VGG16

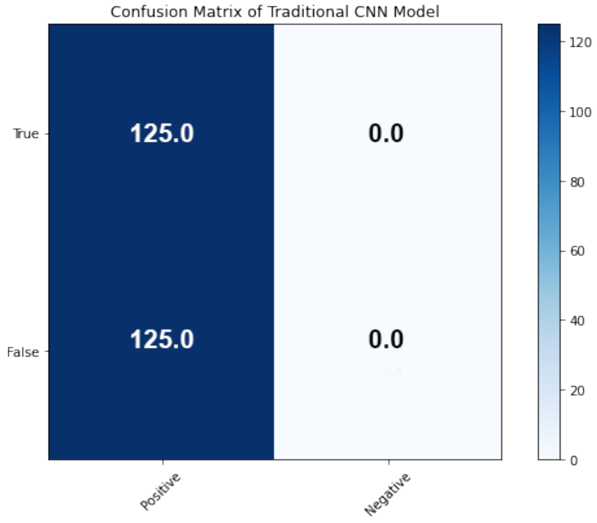
Confusion Matrix for DenseNet121

Confusion Matrix for ResNet50

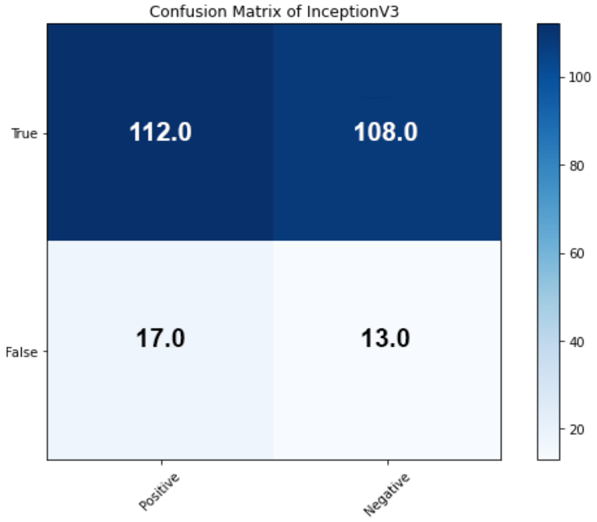
Confusion Matrix for EfficientNetB6



Confusion Matrix for InceptionResNetV2



Confusion Matrix for Traditional CNN Model



Confusion Matrix for InceptionV3

Figure 7.5: Confusion Matrices of all the Models

Chapter 8

Experimental Results and Analysis

8.1 Performance Analysis

In this section, we contrast the performance of the traditional CNN model of 11-layer architecture and six pre-trained CNN models, namely EfficientNetB6, DenseNet121, ResNet50, InceptionResNetV2, InceptionV3, and VGG16.

From the experimental results outlined in Table 8.1 and Table 8.2, we can deduce that the pre-trained models' performance are vastly superior than that of the traditional model. Figure 8.1 presents a bar graph on the accuracy of the CNN models. EfficientNetB6 has accomplished the highest accuracy of 95.99%. The accuracy attained by the other pre-trained CNN models are such that: DenseNet121 was 95.59% , ResNet50 was 94.40% , InceptionResNetV2 was 90.79% , InceptionV3 was 87.99% and VGG16 was 68%. However, the traditional CNN model has achieved the lowest accuracy of 50% and it performed poorly because the given dataset was small with only about 2500 images whereas a minimum of 10,000 images are required for any CNN model to get a decent accuracy. However, the use of transfer learning models on the same dataset shows how powerful pre-trained models can be because of their higher learning rate during training and faster convergence. We also observe that the results of the VGG16 and the traditional 11-layer CNN model shown in Table 8.1 and Table 8.2 are quite similar. It is noted that all the high performing models have a large number of layers in their architecture whereas the two underperforming models, namely VGG16 and the traditional 11-layer CNN model have comparatively the lowest number of layers in their architecture.

DenseNet121 attained the highest AUC Score of 99.40%, followed by ResNet50 with an AUC Score of 99%. For the AUC graph comparison for all the CNN models as shown in Figure 8.2, we noticed that the models with a large number of layers have shown much better AUC performance whereas VGG16 (77.74%) and the Traditional CNN Model (50%) have underperformed significantly because of their fewer number of layers. Therefore, it can be noted that the number of layers plays a crucial role in the classification accuracy. Hence, the greater the number of layers, the better is the accuracy and the overall performance of the models.

Table 8.1: PERFORMANCE COMPARISON IN TERMS OF ACCURACY,PRECISION,RECALL AND F1 SCORE

Models	Accuracy	Precision	Recall	F1
EfficientNetB6	95.99%	97.52%	94.40%	95.93%
DenseNet121	95.59%	92.53%	99.20%	95.75%
ResNet50	94.40%	92.37%	96.80%	94.53%
InceptionResNetV2	90.79%	88.64%	93.60%	91.05%
InceptionV3	87.99%	86.82%	89.60%	88.19%
VGG16	70.80%	80.95%	54.40%	65.07%
Traditional CNN model	50.00%	50.00%	100.00%	66.67%

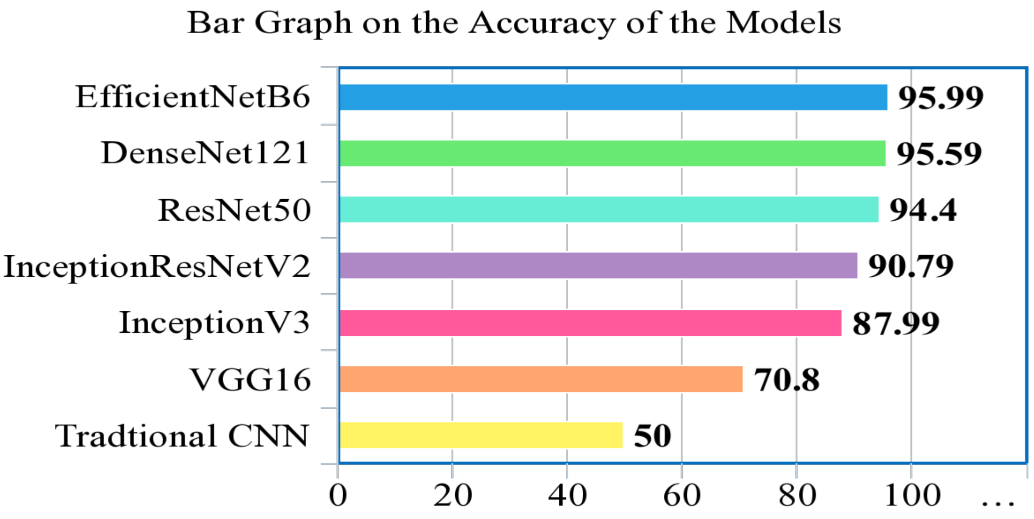


Figure 8.1: Bar Graph on the Accuracy of the Models

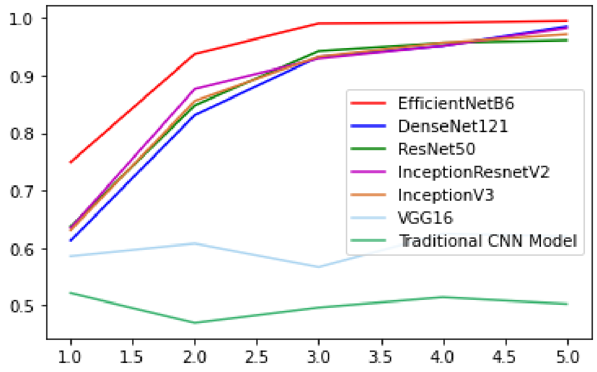


Figure 8.2: AUC Graph comparisons of all the Models.

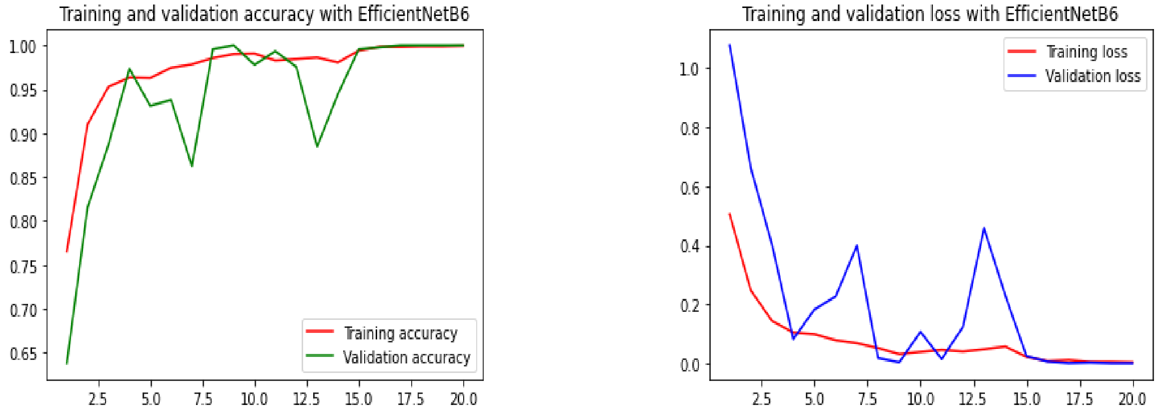
Table 8.2: PERFORMANCE COMPARISON IN TERMS OF AUC AND CONFUSION MATRIX

		Confusion Matrix			
Models	AUC Score	TP	TN	FP	FN
EfficientNetB6	98.40%	118	122	3	7
DenseNet121	99.40%	124	115	10	1
ResNet50	99.00%	121	115	10	4
InceptionResNetV2	97.20%	117	110	15	8
InceptionV3	95.40%	112	108	17	13
VGG16	77.74%	68	109	16	57
Traditional CNN model	50.00%	125	0	125	0

8.2 Training and Validation Accuracy and Loss

EfficientNetB6

Using EfficientNetB6 model, we attained a testing accuracy of 95.99%. Fig 8.3 (a),(b) represent the accuracy and loss graphs for EfficientNetB6 respectively. In the training and validation loss graph, we perceived that the loss decreases sufficiently over time, which means that there is no underfitting. We also observed that the training and validation curves are converging. Hence, we can say that there is no overfitting for EfficientNetB6. There are some fluctuations in the validation curve which are caused by the relatively small amount of data available.



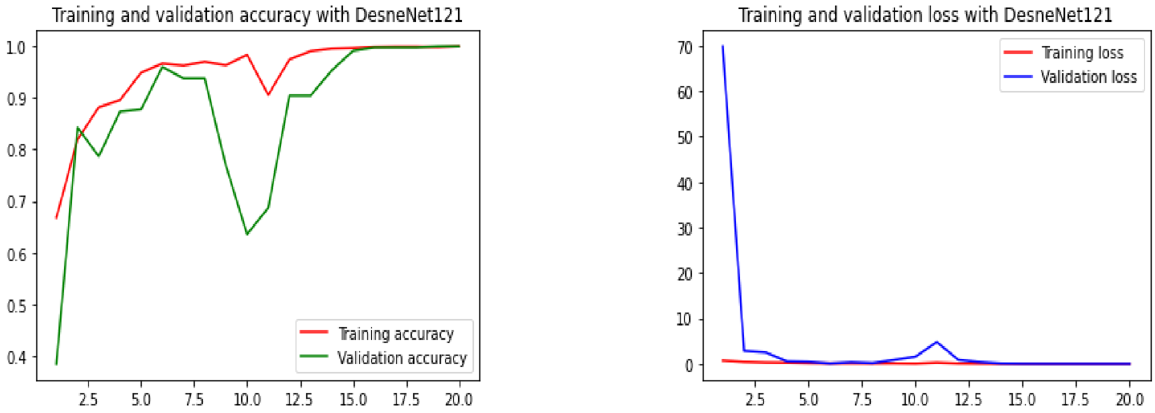
(a) Training and Validation Accuracy for EfficientNetB6

(b) Training & Validation Loss for EfficientNetB6

Figure 8.3: Training & Validation Accuracy & Loss with EfficientNetB6.

DenseNet121

From the DenseNet121 model, we received a testing accuracy score of 95.59%. Fig 8.4 (a),(b) represent the accuracy and loss graphs for DenseNet121 respectively. In the training and validation loss graph, we observed that there is sufficient reduction in loss and the curves are also converging. Therefore, we deduced that there is no overfitting and no underfitting for the DenseNet121 model.



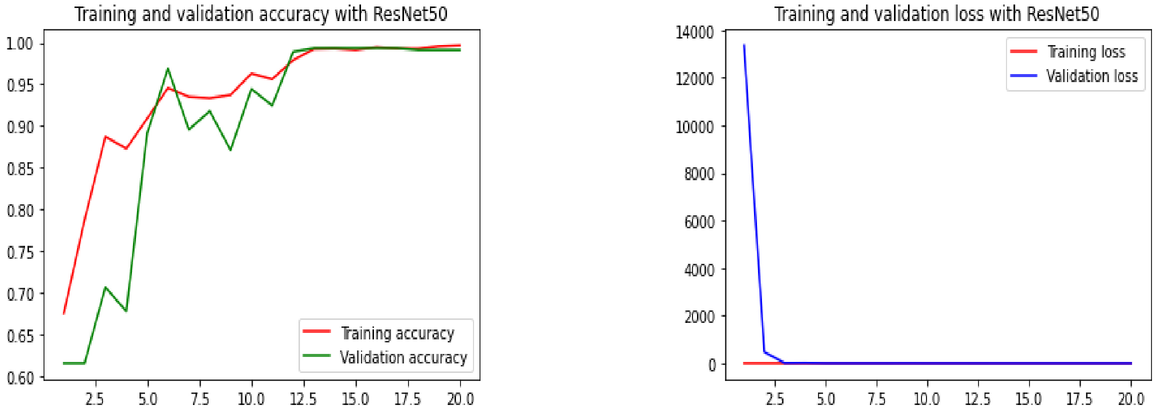
(a) Training and Validation Accuracy for DenseNet121

(b) Training & Validation Loss for DenseNet121

Figure 8.4: Training & Validation Accuracy & Loss with DenseNet121.

ResNet50

In the ResNet50 model, we achieved a testing accuracy of 94.40%. Fig 8.5 (a),(b) represent the accuracy graph and loss graph for ResNet50 respectively. The loss values in Figure 8.5(b) reach very close to zero. We also noticed that the curves converge with each other. So, this means that there is no overfitting and no underfitting for the ResNet50 model.



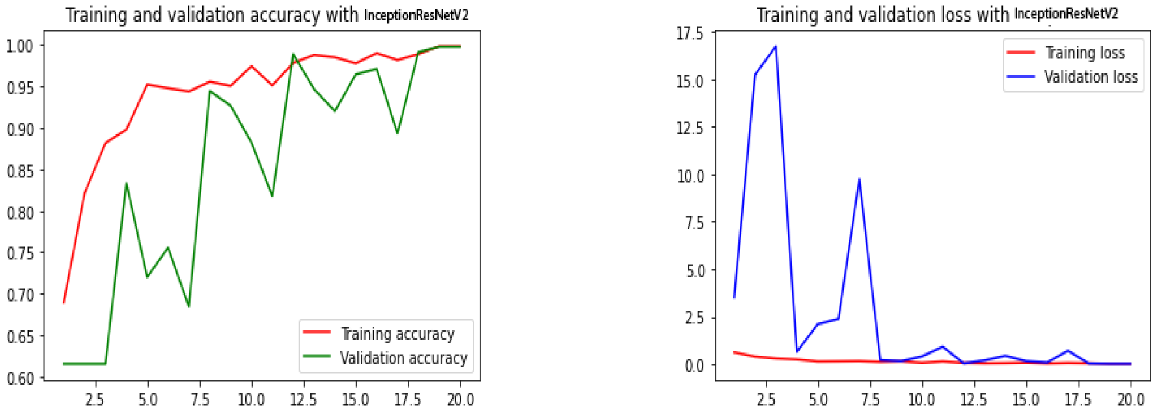
(a) Training & Validation Accuracy for ResNet50

(b) Training & Validation Loss for ResNet50

Figure 8.5: Training & Validation Accuracy & Loss with ResNet50.

InceptionResNetV2

The InceptionResNetV2 model achieved a testing accuracy of 90.79%. Fig 8.6 (a),(b) represent the accuracy graph and loss graph for InceptionResNetV2 respectively. There are some fluctuations in Figure 8.6(b) due to the low amount of data. However, the loss values decrease sufficiently and the curves converge with each other which means that there is no overfitting and no underfitting here.



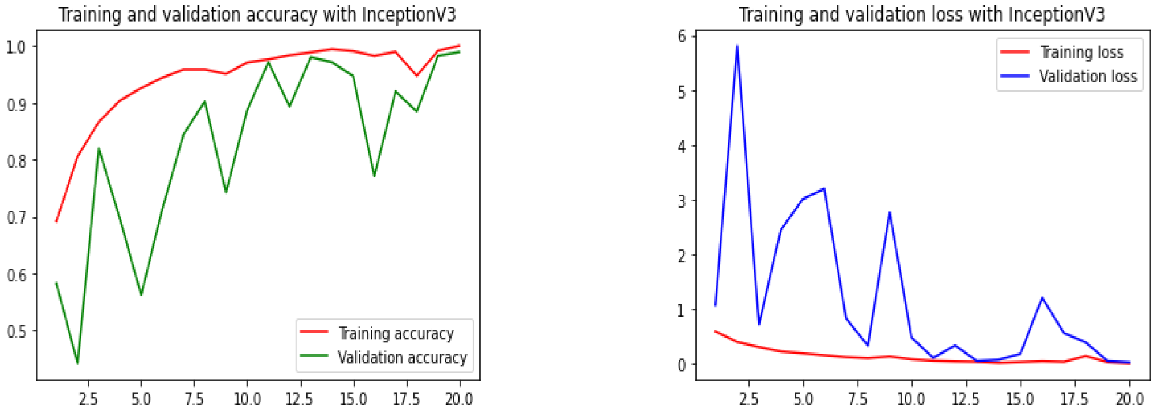
(a) Training and Validation Accuracy for InceptionResNetV2

(b) Training & Validation Loss for InceptionResNetV2

Figure 8.6: Training & Validation Accuracy & Loss with InceptionResNetV2.

InceptionV3

Using the InceptionV3 model, we found the testing accuracy to be 87.99%. Fig 8.7 (a),(b) represent the accuracy and loss graphs for InceptionV3 respectively. Although there are fluctuations in the training and validation curves, they converge with each other which suggests that there is no overfitting. The values of the training and validation loss also decrease enough to suggest that there is no underfitting for InceptionV3 model.



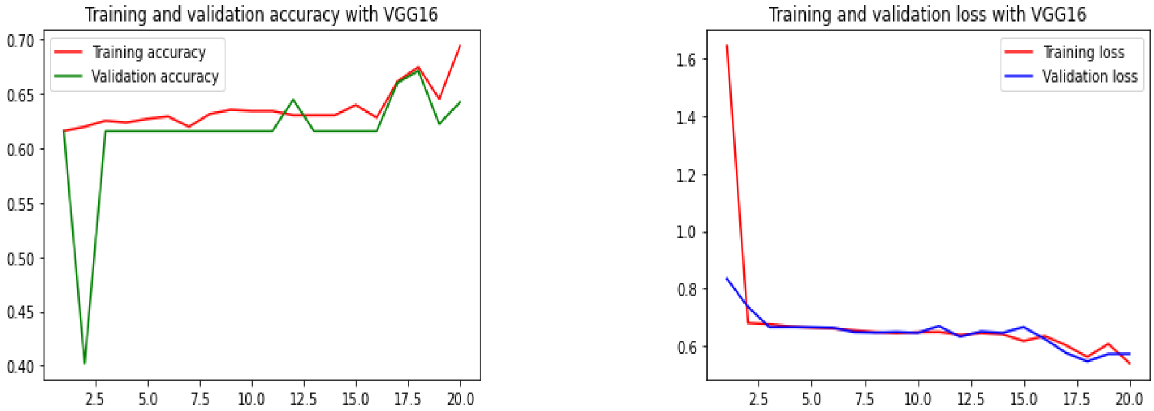
(a) Training and Validation Accuracy for InceptionV3

(b) Training & Validation Loss for InceptionV3

Figure 8.7: Training & Validation Accuracy & Loss with InceptionV3.

VGG16

The VGG16 model doesn't produce a decent result since the testing accuracy value is only 70.80%. Fig 8.8 (a),(b) represent the accuracy and loss graphs of VGG16 respectively. The training and validation loss functions decrease significantly which means that there is no underfitting for VGG16 model. The curves also converge meaning that there is no overfitting here. The VGG16 model has only 16 layers, which is quite low in comparison to the other pre-trained CNN models used in this thesis.



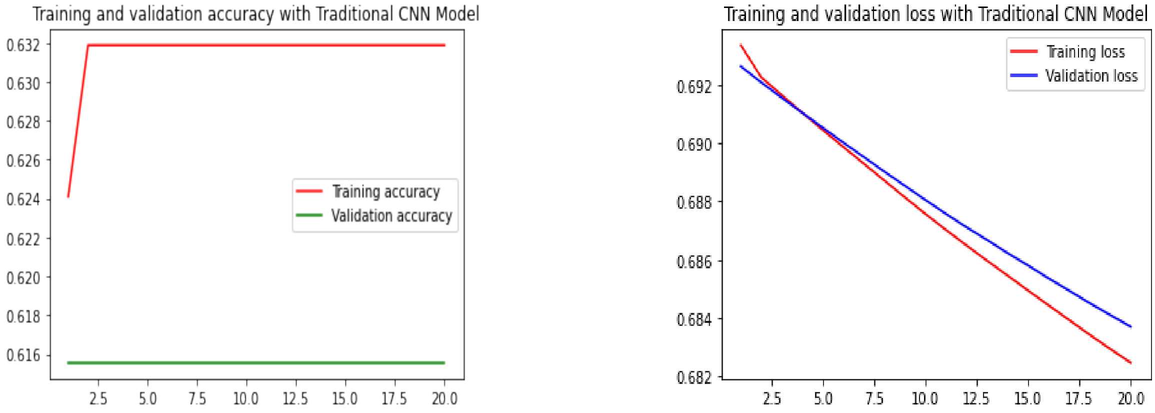
(a) Training& Validation Accuracy for VGG16

(b) Training & Validation Loss for VGG16

Figure 8.8: Training & Validation Accuracy & Loss with VGG16.

Traditional CNN Model

The 11-layer traditional CNN model performs very poorly as it reports a testing accuracy of only 50%. Fig 8.9 (a),(b) represent the accuracy and loss graphs for the Traditional CNN Model respectively.



(a) Training & Validation Accuracy for Traditional CNN Model

(b) Training & Validation Loss for Traditional CNN Model

Figure 8.9: Training & Validation Accuracy & Loss with Traditional CNN Model.

8.3 Accuracy Comparison on Related Works

In this section, a comparison on the accuracy attained by this thesis paper and related recent papers on the same dataset as mentioned in Chapter 2 Literature Review are established. The dataset used is collected from the physionet repository by Hssayeni et al. (2020) [16]. In their study, they employed U-Net to detect intracranial hemorrhage and achieved an accuracy of 87.00%. In 2020, Chen et al. presented an IoT-based implementation for hemorrhage classification where the accuracy obtained was 80.67% for Support Vector Machine (SVM) and 86.70% for Feedforward Neural Network [18]. In the same year, Anupama et al. developed a GC-SDL model that can detect intracranial hemorrhage images in wearable networks and their proposed method obtained an accuracy of 95.73% [17]. Table 8.3 demonstrates a summary of the accuracy review achieved by this paper and other research works recently done on this particular dataset that is being used in this paper. It is observed that the CNN model (EfficientNetB6) employed in this paper attained the highest accuracy of 95.99% compared to the other methods.
Figure 8.10 illustrates a bar graph on this accuracy review.

Table 8.3: Accuracy Review on this Dataset

Approaches	Methods	Accuracy
This Paper	CNN	95.99%
Anupama et al. [17]	GC-SDL	95.73%
Hssayeni et al. [16]	U-Net	87%
Chen et al. [18]	Feedforward Neural Network	86.70%
Chen et al. [18]	SVM	80.67%

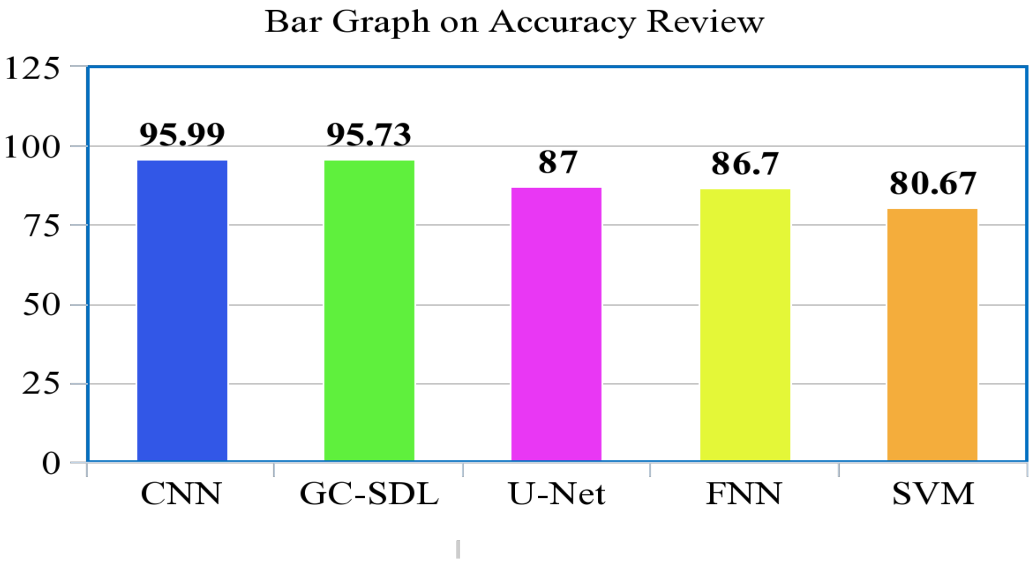


Figure 8.10: Bar Graph on Accuracy Review

8.4 Deployment Considerations

A simple web application based on the best performing model (EfficientNetB6) of this study has been deployed on hosted on Heroku to demonstrate a real-world application. The user is asked to select a head CT image and submit it to the website in order to determine if the CT scan shows any signs of intracranial hemorrhage. The site supports widely used image formats such as JPG, PNG and JPEG image formats. After analysing the characteristics of the input image, the application tells whether this image is classified as a hemorrhage or normal within a timeframe of 4 to 10 seconds. The website mainly runs on python 3.6, tensorflow 2, flask, Gevent. We furthermore two other libraries - Scikit Learn and Scikit Image to process the image inputs. The website is built with an intention to run it from any possible device that is capable of running a web browser. Therefore, the user interface of the website has been kept responsive and very lightweight with the help of Heroku's standard-1X dynos consisting of 512 MB of memory which allowed the web application to have the user not requiring any further processing power to maximise user convenience. Furthermore, the application does not record any user history and this in turn helps to secure patient confidentiality. The source code for the web application of our proposed model is accessible at [38]. Figure 8.11 illustrates successful classification of hemorrhage CT scan as well as normal CT scan images using this web application.

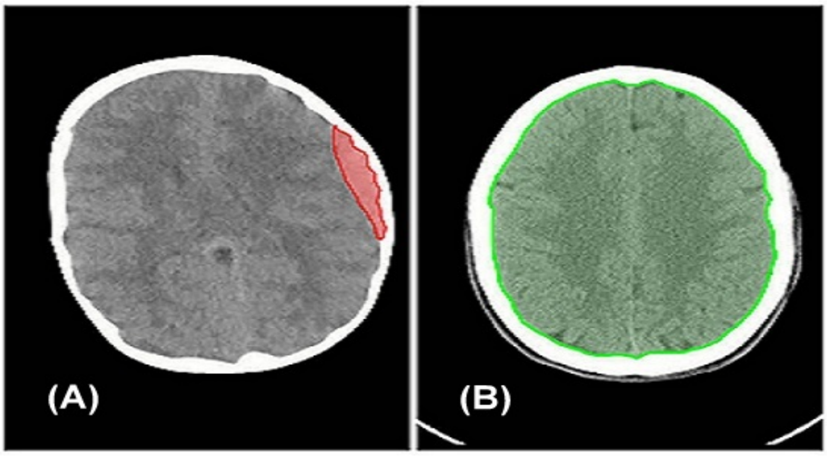


Figure 8.11: Successful Detection of (A) Hemorrhage and (B) Normal CT Scan.

Chapter 9

Conclusion

Health is the greatest asset and advances in technology have been profoundly shaping healthcare by opening up more avenues of extensive research and exploration to improve the quality of life. This research wanted to address the challenge of a leading life-threatening medical emergency and figure out how incorporating AI in this sphere could help to assist healthcare experts for an improved and effective clinical diagnosis. The primary intention was to provide a deep learning-based approach using CNN models that detects and narrows down the lesion delineation on CT scan images of the commonly occurring intracranial hemorrhage. In this paper, a comparative analysis is presented on the performance of seven CNN models: six pretrained CNN models (EfficientNetB6, DenseNet121, ResNet50, InceptionResNetV2, InceptionV3, VGG16) and one traditional CNN model of 11-layer architecture for the detection and binary classification of intracranial brain hemorrhage. EfficientNetB6 has accomplished the highest accuracy of 95.99%. By accomplishing our goal, not only will we be able to use technology to help individuals suffering from hemorrhage but also better our understanding of the most complex organ of the human body. In the future, our subsequent research plan is to implement multi-class classification and to use deep learning ensemble techniques to obtain enhanced predictive performance compared to just any of the constituent CNN models alone that are used on this published dataset.

Bibliography

[1] S. C. Johnston, R. T. Higashida, D. L. Barrow, L. R. Caplan, J. E. Dion, G. Hademenos, L. N. Hopkins, A. Molyneux, R. H. Rosenwasser, F. Vinuela, *et al.*, "Recommendations for the endovascular treatment of intracranial aneurysms: A statement for healthcare professionals from the committee on cerebrovascular imaging of the american heart association council on cardiovascular radiology," *Stroke*, vol. 33, no. 10, pp. 2536–2544, 2002.

[2] J. A. Caceres and J. N. Goldstein, "Intracranial hemorrhage," *Emergency medicine clinics*, vol. 30, no. 3, pp. 771–794, 2012.

[3] M. Al-Ayyoub, D. Alawad, K. Al-Darabsah, and I. Aljarrah, "Automatic detection and classification of brain hemorrhages," *WSEAS transactions on computers*, vol. 12, no. 10, pp. 395–405, 2013.

[4] C. R. Gillebert, G. W. Humphreys, and D. Mantini, "Automated delineation of stroke lesions using brain ct images," *NeuroImage: Clinical*, vol. 4, pp. 540–548, 2014.

[5] L. Chen, S. Wang, W. Fan, J. Sun, and S. Naoi, "Beyond human recognition: A cnn-based framework for handwritten character recognition," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, IEEE, 2015, pp. 695–699.

[6] A. K. Boehme, C. Esenwa, and M. S. Elkind, "Stroke risk factors, genetics, and prevention," *Circulation research*, vol. 120, no. 3, pp. 472–495, 2017.

[7] A. Di Carlo, *Human and economic burden of stroke*, 2009.

[8] J. Vymazal, A. M. Rulseh, J. Keller, and L. Janouskova, "Comparison of ct and mr imaging in ischemic stroke," *Insights into imaging*, vol. 3, no. 6, pp. 619–627, 2012.

[9] K.-O. Lövblad and A. E. Baird, "Computed tomography in acute ischemic stroke," *Neuroradiology*, vol. 52, no. 3, pp. 175–187, 2010.

[10] B. K. Menon, C. D. d'Esterre, E. M. Qazi, M. Almekhlafi, L. Hahn, A. M. Demchuk, and M. Goyal, "Multiphase ct angiography: A new tool for the imaging triage of patients with acute ischemic stroke," *Radiology*, vol. 275, no. 2, pp. 510–520, 2015.

[11] N. H. Rajini and R. Bhavani, "Computer aided detection of ischemic stroke using segmentation and texture features," *Measurement*, vol. 46, no. 6, pp. 1865–1874, 2013.

[12] K. S. Yew and E. Cheng, "Acute stroke diagnosis," *American family physician*, vol. 80, no. 1, p. 33, 2009.

[13] M. Grewal, M. M. Srivastava, P. Kumar, and S. Varadarajan, "Radnet: Radiologist level accuracy using deep learning for hemorrhage detection in ct scans," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, IEEE, 2018, pp. 281–284.

[14] S. Chilamkurthy, R. Ghosh, S. Tanamala, M. Biviji, N. G. Campeau, V. K. Venugopal, V. Mahajan, P. Rao, and P. Warier, "Development and validation of deep learning algorithms for detection of critical findings in head ct scans," *arXiv preprint arXiv:1803.05854*, 2018.

[15] M. R. Arbabshirani, B. K. Fornwalt, G. J. Mongelluzzo, J. D. Suever, B. D. Geise, A. A. Patel, and G. J. Moore, "Advanced machine learning in action: Identification of intracranial hemorrhage on computed tomography scans of the head with clinical workflow integration," *NPJ digital medicine*, vol. 1, no. 1, pp. 1–7, 2018.

[16] M. Hssayeni, "Computed tomography images for intracranial hemorrhage detection and segmentation," *PhysioNet.*, 2019.

[17] C. Anupama, M. Sivaram, E. L. Lydia, D. Gupta, and K. Shankar, "Synergic deep learning model–based automated detection and classification of brain intracranial hemorrhage images in wearable networks," *Personal and Ubiquitous Computing*, pp. 1–10, 2020.

[18] H. Chen, S. Khan, B. Kou, S. Nazir, W. Liu, and A. Hussain, "A smart machine learning model for the detection of brain hemorrhage diagnosis based internet of things in smart cities," *Complexity*, vol. 2020, 2020.

[19] L. Li, M. Wei, B. Liu, K. Atchaneeyasakul, F. Zhou, Z. Pan, S. A. Kumar, J. Y. Zhang, Y. Pu, D. S. Liebeskind, *et al.*, "Deep learning for hemorrhagic lesion detection and segmentation on brain ct images," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 5, pp. 1646–1659, 2020.

[20] A. Majumdar, L. Brattain, B. Telfer, C. Farris, and J. Scalera, "Detecting intracranial hemorrhage with deep learning," in *2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, IEEE, 2018, pp. 583–587.

[21] L. M. Prevedello, B. S. Erdal, J. L. Ryu, K. J. Little, M. Demirer, S. Qian, and R. D. White, "Automated critical test findings identification and online notification system using artificial intelligence in imaging," *Radiology*, vol. 285, no. 3, pp. 923–931, 2017.

[22] P. D. Chang, E. Kuoy, J. Grinband, B. D. Weinberg, M. Thompson, R. Homo, J. Chen, H. Abcede, M. Shafie, L. Sugrue, *et al.*, "Hybrid 3d/2d convolutional neural network for hemorrhage evaluation on head ct," *American Journal of Neuroradiology*, vol. 39, no. 9, pp. 1609–1616, 2018.

[23] K. Jnawali, M. R. Arbabshirani, N. Rao, and A. A. Patel, "Deep 3d convolution neural network for ct brain hemorrhage classification," in *Medical Imaging 2018: Computer-Aided Diagnosis*, International Society for Optics and Photonics, vol. 10575, 2018, p. 105751C.

[24] H. Ye, F. Gao, Y. Yin, D. Guo, P. Zhao, Y. Lu, X. Wang, J. Bai, K. Cao, Q. Song, *et al.*, "Precise diagnosis of intracranial hemorrhage and subtypes using a three-dimensional joint convolutional and recurrent neural network," *European radiology*, vol. 29, no. 11, pp. 6191–6201, 2019.

[25] M. D. Hssayeni, M. S. Croock, A. D. Salman, H. F. Al-khafaji, Z. A. Yahya, and B. Ghoraani, "Intracranial hemorrhage segmentation using a deep convolutional model," *Data*, vol. 5, no. 1, p. 14, 2020.

[26] S. Tang, S. Yuan, and Y. Zhu, "Data preprocessing techniques in convolutional neural network based on fault diagnosis towards rotating machinery," *IEEE Access*, vol. 8, pp. 149 487–149 496, 2020.

[27] L. Taylor and G. Nitschke, "Improving deep learning with generic data augmentation," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, 2018, pp. 1542–1547.

[28] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, PMLR, 2019, pp. 6105–6114.

[29] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[32] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[33] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[34] S. Giri and B. Joshi, "Transfer learning based image visualization using cnn," vol. Vol.11, pp. 41–49, Jul. 2019. DOI: 10.5121/ijaia.2019.11404.

[35] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

[36] E. Hussain, M. Hasan, S. Z. Hassan, T. H. Azmi, M. A. Rahman, and M. Z. Parvez, "Deep learning based binary classification for alzheimer's disease detection using brain mri images," in *2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, IEEE, 2020, pp. 1115–1120.

[37] S. Narkhede, "Understanding auc-roc curve," *Towards Data Science*, vol. 26, pp. 220–227, 2018.

[38] A. I. Rahman and S. Bhuiyan, *Ich detector webapp*, Sep. 2021. [Online]. Available: https://hemorrhage-detect.herokuapp.com/.