

CLASSIFICATION OF MICROGLIAL CELLS USING DEEP LEARNING TECHNIQUES

Submitted in partial fulfillment of the requirements for the award of

Bachelor of Engineering Degree in

Electronics and Communication Engineering

By

ROHAN S (REG NO: 39130377)

NANDHA KISHORE CS (REG NO: 39130316)



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

SCHOOL OF ELECTRONICS

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,

CHENNAI – 600119, TAMILNADU

APRIL - 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

Jeppiar Nagar, Rajiv Gandhi Salai, Chennai -600119

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of
ROHAN S (REG NO: 39130377) and NANDHA KISHORE CS
(REG NO: 39130316), who carried out the project entitled
"CLASSIFICATION OF MICROGLIAL CELLS USING DEEP LEARNING
TECHNIQUES" under my supervision from November 2022 to April 2023.

Internal Guide

Dr.P.CHITRA M.E, Ph.D.

Head of the Department

Dr.T.RAVI M.E, Ph.D.

Submitted for Viva voce Examination held on 19-04-2023

Internal Examiner
External Examiner

DECLARATION

We, ROHAN S (REG NO: 39130377) and NANDHA KISHORE CS (REG NO: 39130316), hereby declare that the Project Report on "CLASSIFICATION OF MICROGLIAL CELLS USING DEEP LEARNING TECHNIQUES" done by us under the guidance of Dr.P.CHITRA M.E, Ph.D. at Sathyabama Institute of Science and technology is submitted in partial fulfilment of the requirements of the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

SIGNATURE OF THE CANDIDATES

DATE: 19-04-2023

1. Rohan S

PLACE: CHENNAI

2. Nandha Kishore CS

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of management of SATHYABAMA for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

We convey our thanks to Dr.N.M. NANDITHA, M.E, and Ph.D., Dean of the Department, Electronics and Communication Engineering, for providing us the necessary support and details at the right time during the progressive reviews.

We convey our thanks to Dr T.RAVI, M.E, Ph.D., Head of the Department, Department of Electronics and Communication Engineering, for providing us the necessary support and details at the right time during the progressive reviews.

I would like to express our sincere and deep sense of gratitude to our Project Guide Dr.P.CHITRA, M.E, Ph.D., for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of ELECTRONICS AND COMMUNICATION ENGINEERING who were helpful in many ways for the completion of the project.

ABSTRACT

Microglia are immune cells exclusive to the central nervous system (CNS) and about 1.5 trillion of them reside in the brain and spinal cord. In response to a variety of signals, microglia show a range of phenotypes, from protective to detrimental associated with motility and morphological changes. In the healthy brain, microglia constantly survey the surrounding tissue with extended processes, clear debris from dead cells, and prune and maintain brain synapses. They are also essential to learning and memory, protect neurons from damage, and mediate pain. In response to an injury or infection, microglia initiate an early, protective response by moving towards the site of injury, where they release a cascade of chemicals leading to repair of the damaged area. Microglial activation is a hallmark of chronic neuro-inflammation, which is believed to play an important role in a range of brain disorders, which has yet to be fully understood, including stroke, multiple sclerosis, Parkinson's, Huntington's and Alzheimer's disease, and can also reflect a neuroprotective behaviour in these chronic conditions

The aim is to use deep learning techniques to classify microglial cells based on images of their morphology. Various pre-defined CNN models will be used to classify the dataset and these models will be compared against each other in order to choose the most effective model.

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	IV
	LIST OF TABLES	VII
	LIST OF FIGURES	VIII
	LIST OF SYMBOLS AND ABBREVIATIONS	IX
1	INTRODUCTION	1
	1.1 INTRODUCTION	1
	1.2 IMAGE RECOGNITION	2
	1.3 DEEP NEURAL NETWORK	3
	1.4 CONVOLUTIONAL NEURAL NETWORK	4
	1.4.1 ARCHITECTURE	5
	1.5 DATA SET	6
	1.6 PROBLEM STATEMENT	7
	1.7 OBJECTIVE	7
2	LITERATURE SURVEY	8
	2.1 LITERATURE SURVEY	8
3	EXISTING SYSTEM AND PROPOSED SYSTEM	15
	3.1 EXISTING SYSTEM	15
	3.1.1 CLASSIFICATION USING NEURAL NETWORKS	15
	3.1.2 CLASSIFICATION ALGORITHMS	16
	3.1.2.1 METHODOLOGIES	17

	3.2 AIM	19
	3.3 SCOPE OF THE INVESTIGATION	20
4	METHODOLOGIES USED	21
	4.1 DATA PRE-PROCCESSING	21
	4.2 DATA AUGMENTATION	22
	4.3 FEATURE EXTRACTION	23
	4.4 CNN	25
	4.5 CNN MODELS	28
	4.6 HYPERPARAMETER TUNING	36
	4.7 PRECISION	37
	4.8 RECALL	37
	4.9 F1 SCORE	38
5	RESULTS AND DISCUSSIONS	39
6	SUMMARY AND CONCLUSION	46
	REFERENCES	48

LIST OF TABLES

TABLE NO.	TITLE	PAGE NO.
5.1	Comparing performances of all networks	42

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
1.1	The architecture of a neural network	4
1.2	Abnormal N9 Microglial cells of Murine	6
1.3	Normal N9 Microglial cells of Murine	6
4.1	Data Augmentation	26
4.2	CNN Architecture	30
4.3	ResNet	31
4.4	ResNet50 Architecture	32
4.5	InceptionV3 Architecture	33
4.6	VGG16 Architecture	35
5.1	Validation graph IncV3	38
5.2	Validation graph Vgg16	39
5.3	Validation graph Vgg19	39
5.4	Validation graph ResNet50	40
5.5	Validation graph Xception	40
5.6	Training loss and training accuracy graph	41
5.7	Validation loss and validation accuracy graph	41
5.8	Confusion matrix for IncV3	43
5.9	Confusion matrix for Resnet50	43
5.10	Confusion matrix for Vgg16	44
5.11	Confusion matrix for Vgg19	44
5.12	Confusion matrix for Xception	44
5.13	Sample Prediction	45

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
DNN	Deep Neural Network
SVM	Support Vector Machine
ROI	Region of Interest
ReLU	Rectified Linear Unit
CNS	Central Nervous System
ANN	Artificial Neural Network
KNN	K Nearest Neighbor
SIANN	Space Invariant Artificial Neural Network

CHAPTER 1

INTRODUCTION

Microglia are a type of neuroglia (glial cell) located throughout the brain and spinal cord. Microglia account for 10–15% of all cells found within the brain. As the resident macrophage cells, they act as the first and main form of active immune defence in the central nervous system (CNS). Microglia (and other neuroglia including astrocytes) are distributed in large non-overlapping regions throughout the CNS. Microglia are key cells in overall brain maintenance—they are constantly scavenging the CNS for plaques, damaged or unnecessary neurons and synapses, and infectious agents. Since these processes must be efficient to prevent potentially fatal damage, microglia are extremely sensitive to even small pathological changes in the CNS. This sensitivity is achieved in part by the presence of unique potassium channels that respond to even small changes in extracellular potassium. Recent evidence shows that microglia are also key players in the sustainment of normal brain functions under healthy conditions. Microglia also constantly monitor neuronal functions through direct somatic contacts and exert neuroprotective effects when needed.

The brain and spinal cord, which make up the CNS, are not usually accessed directly by pathogenic factors in the body's circulation due to a series of endothelial cells known as the blood–brain barrier, or BBB. The BBB prevents most infections from reaching the vulnerable nervous tissue. In the case where infectious agents are directly introduced to the brain or cross the blood–brain barrier, microglial cells must react quickly to decrease inflammation and destroy the infectious agents before they damage the sensitive neural tissue. Due to the lack of antibodies from the rest of the body (few antibodies are small enough to cross the blood–brain barrier), microglia must be able to recognize foreign bodies, swallow them, and act as antigen-presenting cells activating T-cells.

In the past few decades, Deep Learning has proved to be a very powerful tool because of its ability to handle large amounts of data. The interest to use hidden layers has surpassed traditional techniques, especially in pattern recognition. One of the most popular deep neural networks is Convolutional Neural Networks.

1.2 IMAGE RECOGNITION:

Image recognition is a mechanism used to identify an object within an image and to classify it in a specific category, based on the way human people recognize objects within different sets of images.

The goal of image recognition is to identify, label and classify objects which are detected into different categories. Object or image recognition is a whole process that involves various traditional computer vision tasks:

- Image classification: labelling an image and creating categories.
- Object localization: identifying the location of an object in an image, by surrounding it with a bounding box.
- Object Detection: determining the presence of objects with the help of bounding boxes and categorizing it within the class it belongs to.
- Object Segmentation: distinguishing the various elements. Identify and locate each and every item of the picture. Segmentation doesn't use bounding boxes but highlights the contour of the object in the image.

For the past few years, this computer vision task has achieved big successes, mainly thanks to machine learning applications.

First of all, the machine has to know exactly what it has to look for. Thus, it is necessary to give it the parameters you decide to work on. Defining the dimensions of bounding boxes and what elements are inside is crucial. To do so, the machine has to be provided with some references, which can be pictures, videos or photographs, etc. These elements will allow it to be more efficient when analysing future data. This will create a sort of data library that will then be used by the Neural Network to distinguish the various objects. A Neural Network is composed of

multiple artificial neurons. These neurons are meant to imitate the human brain. It works with a set of various algorithms also inspired by the way the brain functions. If we want the image recognition model to analyse and categorize different races of dogs, the model will need to have a database of the various races in order to recognize them.

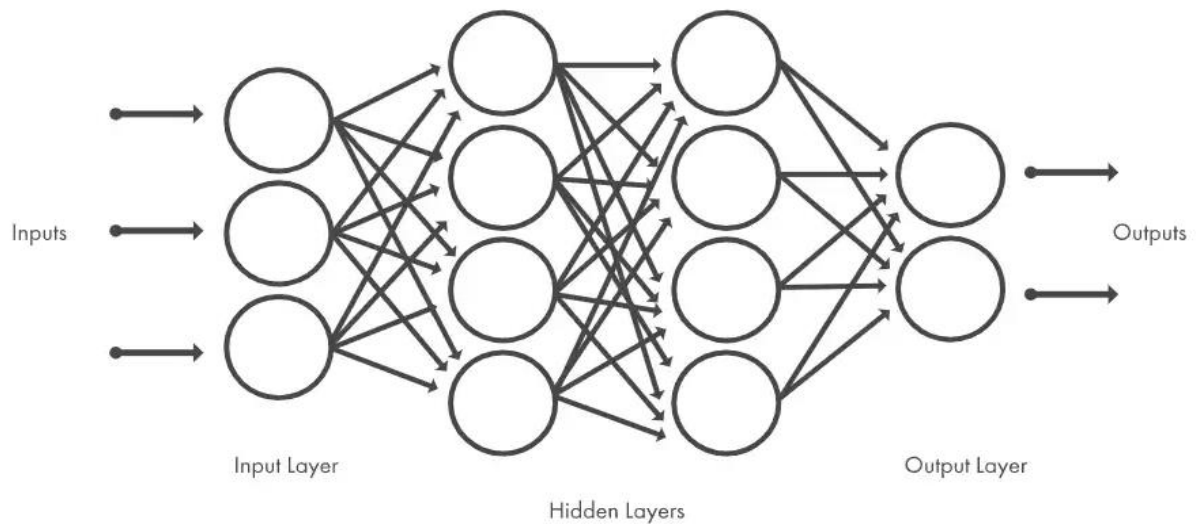
Second, the model needs to go on a training phase. The dataset needs to be entered within a program in order to function properly. And this phase is only meant to train the Convolutional Neural Network (CNN) to identify specific objects and organize them accurately in the correspondent classes.

Before an AI model is being used, it needs to be thoroughly tested. To do so, it is necessary to propose images that were not part of the training phase. Based on whether or not the program has been able to identify all the items and on the accuracy of classification, the model will be approved or not.

1.3 DEEP NEURAL NETWORK:

Image recognition is one of the tasks in which deep neural networks (DNNs) excel. Neural networks are computing systems designed to recognize patterns. Their architecture is inspired by the human brain structure, hence the name. They consist of three types of layers: input, hidden layers, and output. The input layer receives a signal, the hidden layer processes it, and the output layer makes a decision or a forecast about the input data. Each network layer consists of interconnected nodes (artificial neurons) that do the computation.

What makes a neural network deep? The number of hidden layers: While traditional neural networks have up to three hidden layers, deep networks may contain hundreds of them.



***Fig.1.1 The architecture of a neural network, each layer consists of nodes.
The number of hidden layers is optional.***

1.4 CONVOLUTIONAL NEURAL NETWORK:

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of artificial neural network (ANN), most commonly applied to analyse visual imagery. CNNs are also known as Shift Invariant or Space Invariant Artificial Neural Networks (SIANN), based on the shared-weight architecture of the convolution kernels or filters that slide along input features and provide translation-equivariant responses known as feature maps. Counter-intuitively, most convolutional neural networks are not invariant to translation, due to the down sampling operation they apply to the input. They have applications in image and video recognition, recommender systems, image classification, image segmentation, medical image analysis, natural language processing, brain–computer interfaces, and financial time series.

CNNs are regularized versions of multilayer perceptrons. Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. The "full connectivity" of these networks make them prone to overfitting data. Typical ways of regularization, or preventing overfitting, include: penalizing parameters during training (such as weight decay) or trimming connectivity (skipped connections, dropout, etc.) CNNs take a different

approach towards regularization: they take advantage of the hierarchical pattern in data and assemble patterns of increasing complexity using smaller and simpler patterns embossed in their filters. Therefore, on a scale of connectivity and complexity, CNNs are on the lower extreme.

Convolutional networks were inspired by biological processes in that the connectivity pattern between neurons resembles the organization of the animal visual cortex. Individual cortical neurons respond to stimuli only in a restricted region of the visual field known as the receptive field. The receptive fields of different neurons partially overlap such that they cover the entire visual field.

CNNs use relatively little pre-processing compared to other image classification algorithms. This means that the network learns to optimize the filters (or kernels) through automated learning, whereas in traditional algorithms these filters are hand-engineered. This independence from prior knowledge and human intervention in feature extraction is a major advantage.

1.4.1 ARCHITECTURE:

Convolutional neural networks are a specialized type of artificial neural networks that use a mathematical operation called convolution in place of general matrix multiplication in at least one of their layers. They are specifically designed to process pixel data and are used in image recognition and processing.

A convolutional neural network consists of an input layer, hidden layers and an output layer. In any feed-forward neural network, any middle layers are called hidden because their inputs and outputs are masked by the activation function and final convolution. In a convolutional neural network, the hidden layers include layers that perform convolutions. Typically this includes a layer that performs a dot product of the convolution kernel with the layer's input matrix. This product is usually the Frobenius inner product, and its activation function is commonly ReLU. As the convolution kernel slides along the input matrix for the layer, the convolution operation generates a feature map, which in turn contributes to the input of the next

layer. This is followed by other layers such as pooling layers, fully connected layers, and normalization layers.

1.5 DATA SET:

A dataset in computer vision is a curated set of digital photographs that developers use to test, train and evaluate the performance of their algorithms. The algorithm is said to learn from the examples contained in the dataset. What learning means in this context has been described by Alan Turing (1950): “it is best to provide the machine with the best sense organs that money can buy, and then teach it to understand and speak English. This process could follow the normal teaching of a child. Things would be pointed out and named, etc.” A dataset in computer vision therefore assembles a collection of images that are labelled and used as references for objects in the world, to ‘point things out’ and name them.

The dataset which will be used for training and testing this model are cultured from Murine by researchers. Below are some samples of what a cultured N9 microglial cells of murine looks like:

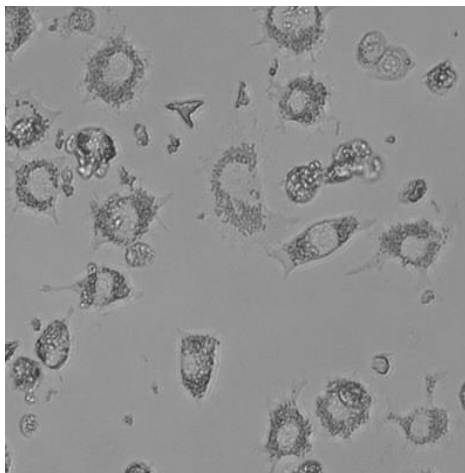


Fig. 1.2 Abnormal N9 Microglial cells of Murine

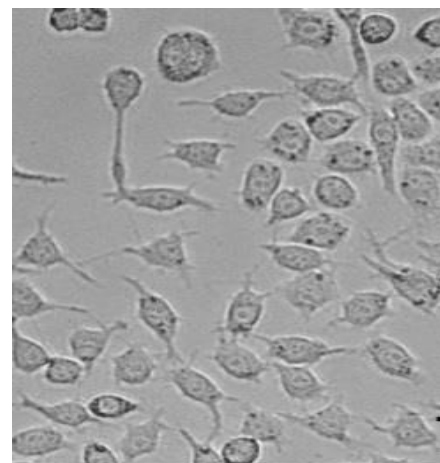


Fig. 1.3 Normal N9 Microglial cells of Murine

1.6 PROBLEM STATEMENT:

Microglia have been extensively studied for their harmful roles in neurodegenerative diseases, such as Alzheimer's disease, Parkinson's disease, multiple sclerosis, as well as cardiac diseases, glaucoma, and viral and bacterial infections. Developing an ideal deep learning model will help pathologists in diagnosing the disease in advance by classifying microglia activation states in histological images. The model should also provide key information for understanding diseases of the CNS and is essential for the validation of in vivo biomarkers that allow the stratification and monitoring of patients and populations at risk.

1.7 OBJECTIVE:

The objective is to use deep learning techniques to classify microglial cells based on images of their morphology. Various pre-defined CNN models will be used to classify the dataset and these models will be compared against each other in order to choose the most effective model.

CHAPTER 2

LITERATURE SURVEY

2.1 LITERATURE SURVEY:

Massengill, et.al (2021) propose a novel segmentation routine, implemented within FIJI-ImageJ, to perform automated segmentation and cell counting of retinal microglia. This paper shows a routine which could perform cell counts with accuracy similar to manual observers using the I307N Rho model. Using feature selection with Chi2 and principal component analysis, the authors resolved cells along a morphological gradient, demonstrating that extracted features were sufficiently descriptive to capture subtle morphological changes within cell populations in I307N Rho and Balb/c TLR2-/- retinal degeneration models [16].

Brackhan and Pahnke (2021) developed an automatic image analysis model for detailed pathology analysis. Automatic image analysis can help to increase objectivity and reduce time. They present the evaluation of the DeePathology STUDIOTM for automatic analysis of histological whole-slide images using machine learning/artificial intelligence [14].

Albawi, et.al (2017) reviews Convolutional neural network in this paper. It take this name from mathematical linear operation between matrixes called convolution. CNN have multiple layers; including convolutional layer, non-linearity layer, pooling layer and fully-connected layer. The convolutional and fully-connected layers have parameters but pooling and non-linearity layers don't have parameters. The CNN has an excellent performance in machine learning problems. Especially the applications that deal with image data, such as largest image classification data set

(Image Net), computer vision, and in natural language processing (NLP) and the results achieved were very amazing. This paper explains and defines all the elements and important issues related to CNN, and how these elements work and it will also state the parameters that effect CNN efficiency [20].

Colonna and Butovsky (2017) proposed a paper on Microglial cells and its functions. Microglia, type of neuronal support cell (neuroglia) occurring in the central nervous system of invertebrates and vertebrates that functions primarily as an immune cell. As the name microglia suggests, these cells are small—the smallest of all the neuroglia. Microglia nuclei are typically oval-shaped, and projecting out from their cell bodies are slender elongated processes that enable the cells to move via chemotaxis (movement along a chemical gradient). For many years the function of microglia was unclear. However, today it is known that these cells mediate immune responses in the central nervous system by acting as macrophages, clearing cellular debris and dead neurons from nervous tissue through the process of phagocytosis (cell eating). Microglia account for approximately 10% of cells and are the most abundant mononuclear phagocytes in the central nervous system (CNS) [1].

Jeong Ji, et.al (2013) in their paper mentioned that microglia are key players in brain injury and disease. Microglia can respond to different types of stimuli, including exogenous infectious pathogens as well as injurious self-proteins, and initiate a neuro-inflammatory response. Therefore to understand the patterns and roles of brain inflammation in the injured brain, it is necessary to analyse the behaviour of all cell types in the injured brain immediately after the onset of injury. Based on their experience with both in vitro and in vivo experimental models of brain inflammation, they had concluded that not only microglia, but also astrocytes, blood inflammatory cells, and even neurons participate and/or regulate brain inflammation in the injured brain. Furthermore, brain inflammation played by these cells protects neurons and repairs damaged microenvironment but not induces neuronal damage. [2].

Akiyama and Barger (2013) proposed a paper on Inflammation and Alzheimer's disease. As part of their response microglial cells secrete cytokines, chemokines, prostaglandins, and reactive oxygen species, which help to direct the immune response. Additionally, they are instrumental in the resolution of the inflammatory response, through the production of anti-inflammatory cytokines. Microglia have also been extensively studied for their harmful roles in neurodegenerative diseases, such as Alzheimer's disease, Parkinson's disease, multiple sclerosis, as well as cardiac diseases, glaucoma, and viral and bacterial infections. Altered gene expression leads to the production of numerous potentially neurotoxic mediators. These mediators are important in the normal functions of microglia and their production is usually decreased once their task is complete. In chronic neuro inflammation, microglia remain activated for an extended period during which the production of mediators is sustained longer than usual. This increase in mediators contributes to neuronal death [3].

Stanley, et.al (2019) reported some attempts to understand the dynamics of complex physiologic fluctuations by adapting and extending concepts and methods developed very recently in statistical physics. This paper reviews progress using two analysis methods – detrended fluctuation analysis and wavelets – sufficient for quantifying mono-fractal structures and then describes recent work that quantifies multifractal features of interbeat interval series, and the discovery that the multifractal structure of healthy subjects is different than that of diseased subjects [7].

Valous, et.al (2013) developed a multistage workflow for segmenting and counting murine microglia from histopathological bright field images, in a permanent focal cerebral ischemia model. Automated counts are useful, since for the assessment of inflammatory mechanisms in ischemic stroke there is a need to quantify the brain's responses to post-ischemia, which primarily is the rapid activation of microglial cells. Microglia cells were segmented in seven sequential steps: (i) contrast boosting

using quaternion operations, (ii) intensity outlier normalization, (iii) nonlocal total variation de-noising, (iv) histogram specification and contrast stretching, (v) homomorphic filtering, (vi) global thresholding, and (vii) morphological filtering [8].

Withey, et al (2010) proposed a detailed review on medical image segmentation. Methods for segmentation of medical images are divided into three generations, where each generation adds an additional level of algorithmic complexity. The first generation is composed of the simplest forms of image analysis such as the use of intensity thresholds and region growing. The second generation is characterized by the application of uncertainty models and optimization methods, and the third generation incorporates knowledge into the segmentation process [9].

Yuheng and Hao [10] reviews Image segmentation in this paper. The technology of image segmentation is widely used in medical image processing, face recognition pedestrian detection, etc. The current image segmentation techniques include region-based segmentation, edge detection segmentation, segmentation based on clustering, segmentation based on weakly-supervised learning in CNN, etc. This paper analyses and summarizes these algorithms of image segmentation, and compares the advantages and disadvantages of different algorithms.

Jelinek, et.al (2018) proposed how microglia activation states can be differentiated with the help of fractal analysis. Microglia affect and are sensitive to events in the central nervous system, changing in morphology and function as they respond to and resolve disruptions. Monitoring microglia is, therefore, an important goal of neuroscience. We investigated morphological changes in cultured mouse microglia, using the box counting fractal dimension (DB), lacunarity, and other measures. The DB and lacunarity corresponded well to visually applied classification systems of these cells. Complementing such systems, which depend on grossly visible differences between cells, the DB also differentiated between visually indistinguishable microglia in different functional states (i.e., deramifying versus

reramifying). The results suggest that fractal analysis may help determine if a microglial cell is resting, rousing itself for action, acting subtly, or returning to a resting state. Because of the awesome potential microglia have to affect events in the central nervous system, the implications of this for the study of human health and disease are profound [11].

Karperien, et.al (2013) proposed a method where microglia can be classified with the help of fractal analysis. It is well established that microglial form and function are inextricably linked. From a quantitative perspective, it is problematic to measure the variability of microglia activation states using traditional methods, but one way of quantitating such detail is through fractal analysis. The techniques of fractal analysis have been used for quantitating microglial morphology, to categorize gross differences but also to differentiate subtle differences (e.g., amongst ramified cells). Multifractal analysis in particular is one technique of fractal analysis that may be useful for identifying intermediate forms [12].

Kamalahasan (2017) reviews various classification algorithms used in image processing. Classification algorithms play a major role in image processing techniques. It is used to classify the features that are extracted from the image into various classes based on different characteristics. This paper proposes the survey on numerous classification techniques which are widely used in image processing. The classification techniques are discussed on the applications in different fields. The most optimal algorithms for different domains are also analysed [15].

Esgiar, et.al proposed a study which investigates the value of fractal dimension in separating normal and cancerous images, and to examine the relationship between fractal dimension and traditional texture analysis features. A "leave-one-out" analysis approach was used to classify the samples into each group. With fractal analysis there was a highly significant difference between groups ($p < 0.0001$). Correlation and entropy features showed greater differences between the groups (p

< 0.0001). Nevertheless, the addition of fractal analysis to the feature analysis improved the sensitivity from 90% to 95% and specificity from 86% to 93% [17].

Erickson, et.al (2015) proposed a paper on how machine learning is crucial in the field of medicine and medical imaging. Machine learning is a technique for recognizing patterns that can be applied to medical images. Although it is a powerful tool that can help in rendering medical diagnoses, it can be misapplied. Machine learning typically begins with the machine learning algorithm system computing the image features that are believed to be of importance in making the prediction or diagnosis of interest. The machine learning algorithm system then identifies the best combination of these image features for classifying the image or computing some metric for the given image region. There are several methods that can be used, each with different strengths and weaknesses. Several metrics for measuring the performance of an algorithm exist; however, one must be aware of the possible associated pitfalls that can result in misleading metrics [18].

Shen Wu and Suk (2017) reviews the role of Deep learning in medical image analysis. Deep learning is rapidly proving to be the state-of-the-art foundation, achieving enhanced performances in various medical applications. In this paper, the authors introduce the fundamentals of deep learning methods; review their successes to image registration, anatomical/cell structures detection, tissue segmentation, computer-aided disease diagnosis or prognosis, and so on [19].

Foskey, et.al (2008) generalized SVM to work with data objects that are naturally understood to be lying on curved manifolds, and not in the usual d -dimensional Euclidean space. The authors use geodesic distances, defined on the manifold to formulate our methodology. This approach addresses the important issue of analysing the change that accompanies the difference between groups by implicitly defining the notions of separating surface and separating direction on the manifold.

The methods are applied in shape analysis with target data being m-reps of 3 dimensional medical images [21].

Milošević, et.al (2013) in their paper outlines how fractal analysis can be used for further quantitative classification of microglial cell types using box-counting and multifractal analysis. The fractal dimension, besides its ability to discriminate among different cell types, can work as a reliable parameter in cell classification. A qualitative analysis of the morphology of neurons and glia cell types involves a detailed description of the structure and features of cells, and accordingly, their classification into defined classes and types [30].

CHAPTER 3

AIM AND SCOPE OF PRESENT INVESTIGATION

3.1 EXISTING SYSTEM:

Microglia are immune cells of the central nervous system capable of distinct phenotypic changes and migration in response to injury. These changes most notably include the retraction of fine dendritic structures and adoption of a globular, phagocytic morphology. Due to their characteristic responses, microglia frequently act as histological indicators of injury progression. While algorithms seeking to automate microglia counts and morphological analysis are becoming increasingly popular, few exist that are adequate for use within the retina and manual analysis remains prevalent. Common image segmentation methods are not suitable for inhomogeneous histology image analysis and accurate classification of microglial activation states has remained a challenge. In their paper, Yuchun, et.al [13] introduce an automated image analysis framework capable of efficiently segmenting microglial cells from histology images and analysing their morphology. The framework makes use of variational methods and the fast-split Bregman algorithm for image denoising and segmentation, and of multifractal analysis for feature extraction to classify microglia by their activation states. Experiments show that the proposed framework is accurate and scalable to large datasets and provides a useful tool for the study of microglial biology.

3.1.1 CLASSIFICATION USING NEURAL NETWORKS:

Image classification using deep neural networks is a popular application of machine learning and artificial intelligence. The goal of image classification is to automatically categorize images into different classes or categories based on their content. Deep neural networks are a type of artificial neural network that have multiple hidden layers and can learn complex patterns and features from images.

To perform image classification using deep neural networks, the following steps are typically involved:

1. Collect and prepare a dataset of images, which includes a set of labelled images for training the neural network and a set of unlabelled images for testing and evaluation.
2. Pre-process the images to ensure that they are in a suitable format for input to the neural network. This may involve tasks such as resizing, cropping, and normalization.
3. Design and train a deep neural network model using a training dataset. This typically involves choosing an appropriate architecture for the neural network, initializing the weights, and using a loss function to optimize the network's parameters.
4. Validate the trained model using a separate validation dataset and adjust the model's hyperparameters as needed to improve its performance.
5. Test the final model on an independent test dataset to evaluate its accuracy and performance.

Overall, image classification using deep neural networks is a powerful technique for automatically categorizing images and has many practical applications in fields such as computer vision, robotics, and healthcare.

3.1.2 CLASSIFICATION ALGORITHMS:

The image processing contains different techniques like feature extraction, feature selection, and classification. The images are captured from the respected experimental setup and then features are extracted. The features are the essential and relevant information or data that are obtained from the digital image. The process of obtaining the features is referred to as Feature Extraction. The appropriate features are selected on the required training set. The way in which the subset of features is obtained from the large set of features that are extracted from the image is known as Feature Selection. It reduces the large set of features by eliminating the redundant feature, and the features are selected which are most

relevant. The subset of features contains only the admissible features that are reduced by various properties. The features are then classified into non-identical classes. The classes are built based on numerous parameters which are defined by the end users or the features may be combined into different classes automatically. The subset of features is given as input to different classifiers. There are several classification algorithms available for real world applications such as Naive- Bayes, Support Vector Machine, k-Nearest Neighbour algorithms, Shadow algorithms, Minimum Mean Distance (MMD), Neural Networks, Decision Trees, Hidden Markov Model, K-means clustering algorithms, Machine Learning. These algorithms are suitable for distinct applications and the selection of classifier is based on the performance of the algorithm in a different kind of application domain.

3.1.1.1 Methodologies:

A. K - Nearest Neighbour Algorithm (KNN):

KNN is one of the simplest statistical classification algorithms among various machine learning algorithms. The objects are clustered with a maximum number of votes given by the neighbours. The objects are assigned with most similar neighbours in the feature space. The objects are assigned to single the closest neighbour if and only if $k=1$. The KNN algorithm uses the closest feature subsets for classification and provides effective performance for automatic facial recognition. It uses K-means, K- medoids, and CLARANS to find the subset of samples from the training set. The k-nearest clusters are grouped to form particular classes that can be differentiated into larger and smaller clusters.

B. Support Machine Vector (SVM):

The classification and analysis of the data sets are frequently performed by the SVM Classifier. It is also one of the supervised learning models that are non-probabilistic. It consists of points on the sample space and is divided into several regions mapped together. They are widely used in image segmentation, text and handwritten manuscripts and classifying the images in the field of medicine to identify various

parametric features. This helps in achieving better performance and accuracy rate. It normally includes the following stages:

- Pre-processing of the images in the database
- Separation of the database in training and test sets
- Choice of the representation of the input data
- Choice of the way of training, which includes:
 - ◆ Method of multi-class training
 - ◆ Value of the penalty term C
 - ◆ Choice of the kernel
- Training
- Test and evaluation of the performance.

C. Genetic Algorithm:

Genetic Algorithm (GA) is an optimization-based method that uses population for solving optimization problems. It is been effectively applied in image segmentation and enhancement of images. It mainly helps in improving the quality of the images and gives the better visual perception to the images to produce better outcomes. It is one of the evolutionary problems. GA works best on practical problems. It has two parameters such as crossover and mutation and works well than common image enhancement techniques. A typical genetic algorithm requires:

1. A genetic representation of the solution domain.
2. A fitness function to evaluate the solution domain.

Genetic Algorithm has many advantages in acquiring the optimized solution. It is one of the most powerful optimization technique in a larger space. The quality of image segmentation can be improved by selection of the parameters in an optimized manner.

D. Maximum Likelihood Classifier:

It is one of the most commonly used methods of classification in remote sensing. It is a method of estimating the parameters by finding the parameter values of a statistical model that maximize the likelihood from the observed parameters.

E. Naive-Bayes Classifier:

Naive-Bayes classifier is a kind of machine learning technique which is based on the Bayes theorem with strong inter-relationship among features. Naive-Bayes is one type of classifier that models around detailed features of the problem. Each feature is independent of the other features. It is trained by the number of classification rules such as maximum posteriori to determine the Bayesian statistics. It is most commonly used classifier in real time problems like Facial recognition. To determine the expression, one has to first choose the structure of the network that possesses the dependencies of variables and then the features are distributed. The problem with Naive–Bayes approach is that the independence of features applied to facial expressions can be too strong to showcase the emotions of humans.

3.2 AIM:

N9 Microglial cells are cultured from murine. The histological microglial images obtained from the culturing process will then be used for analysis. Image analysis is performed on the obtained images using the help of Deep learning algorithms used for image classification. In the proposed method the CNN models take care of segmentation, feature extraction and classification of the data obtained from the data base. Five major CNN models will be used for classification of the microglial cell images.

In the era of digital medicine, a vast number of medical images are produced every day. There is a great demand for intelligent equipment for adjuvant diagnosis to assist medical doctors with different disciplines. With the development of artificial intelligence, the algorithms of convolutional neural network (CNN) progressed

rapidly. CNN and its extension algorithms play important roles on medical imaging classification, object detection, and semantic segmentation.

3.3 SCOPE OF THE INVESTIGATION:

Pre-defined CNN models have proven to be highly effective for medical image classification tasks. CNNs, or convolutional neural networks, are a type of deep learning model that is specifically designed to process image data. They are highly effective at identifying patterns and features in images that are difficult for humans or traditional computer vision algorithms to detect.

Pre-defined CNN models, such as VGG, ResNet, Inception, and DenseNet, have been pre-trained on large datasets of general images, such as ImageNet, which makes them highly effective at extracting features from medical images as well. By using transfer learning, these pre-trained models can be fine-tuned on smaller medical image datasets to improve their performance on specific medical classification tasks.

Overall, pre-defined CNN models are a powerful tool for medical image classification, as they can help to automate the process of diagnosing and treating medical conditions. However, it's important to note that medical image classification is a complex task that requires careful consideration of many factors, including the quality of the imaging data, the choice of CNN model, and the training process. As with any medical application of AI, it's crucial to involve healthcare professionals in the development and deployment of these models to ensure that they are safe, effective, and aligned with clinical best practices.

CHAPTER 4

EXPERIMENTAL METHODS AND ALGORITHMS USED

4.1 DATA PRE-PROCESSING:

Data pre-processing is an essential step in any machine learning project, as it involves preparing the data in a way that makes it suitable for analysis and modelling. The goal of data pre-processing is to clean, transform, and organize the data so that it can be used effectively by the machine learning algorithms.

Here are some common steps involved in data pre-processing:

1. **Data Cleaning:** This involves identifying and correcting errors or inconsistencies in the data. This includes handling missing values, dealing with outliers and noise, and removing duplicates.
2. **Data Transformation:** This involves converting the data into a format that can be used by machine learning algorithms. This includes scaling the data, encoding categorical variables, and reducing the dimensionality of the data through feature selection or extraction.
3. **Data Normalization:** This involves rescaling the features so that they have a similar scale or distribution. This helps the machine learning algorithms to converge faster and improve the accuracy of the models.
4. **Data Splitting:** This involves dividing the data into training, validation, and testing sets. This ensures that the machine learning models are trained on one set of data, validated on another set, and tested on a completely different set of data.
5. **Data Augmentation:** This involves generating additional training examples by applying transformations such as rotations, translations, and scaling to the existing data. This can help to increase the size of the training set and improve the generalization of the models.
6. **Data Balancing:** This involves adjusting the class distribution of the data to address any imbalance in the classes. This can be done through over-sampling, under-sampling, or a combination of both.

Data pre-processing is a crucial step in machine learning projects, as it can have a significant impact on the performance of the models. By following best practices for data pre-processing, such as those listed above, you can improve the quality of your data and build more accurate and robust machine learning models.

4.2 DATA AUGMENTATION:

Data augmentation is a technique used in machine learning to increase the size and diversity of a training dataset by applying various transformations to the existing data. This technique can be particularly useful when the training dataset is small or when the data is highly imbalanced.

Data augmentation involves generating additional training examples by applying transformations to the existing data. These transformations can include:

1. Rotation: Rotating the image by a certain angle to simulate different camera angles.
2. Translation: Moving the image horizontally or vertically to simulate different camera positions.
3. Scaling: Resizing the image to simulate different object sizes.
4. Shearing: Slanting the image to simulate different viewing angles.
5. Flipping: Mirroring the image horizontally or vertically to simulate different object orientations.
6. Adding Noise: Adding random noise to the image to simulate variations in lighting conditions.

These transformations can be applied in different combinations to generate multiple versions of the same image. This can help to increase the size of the training dataset and improve the generalization of the machine learning models.

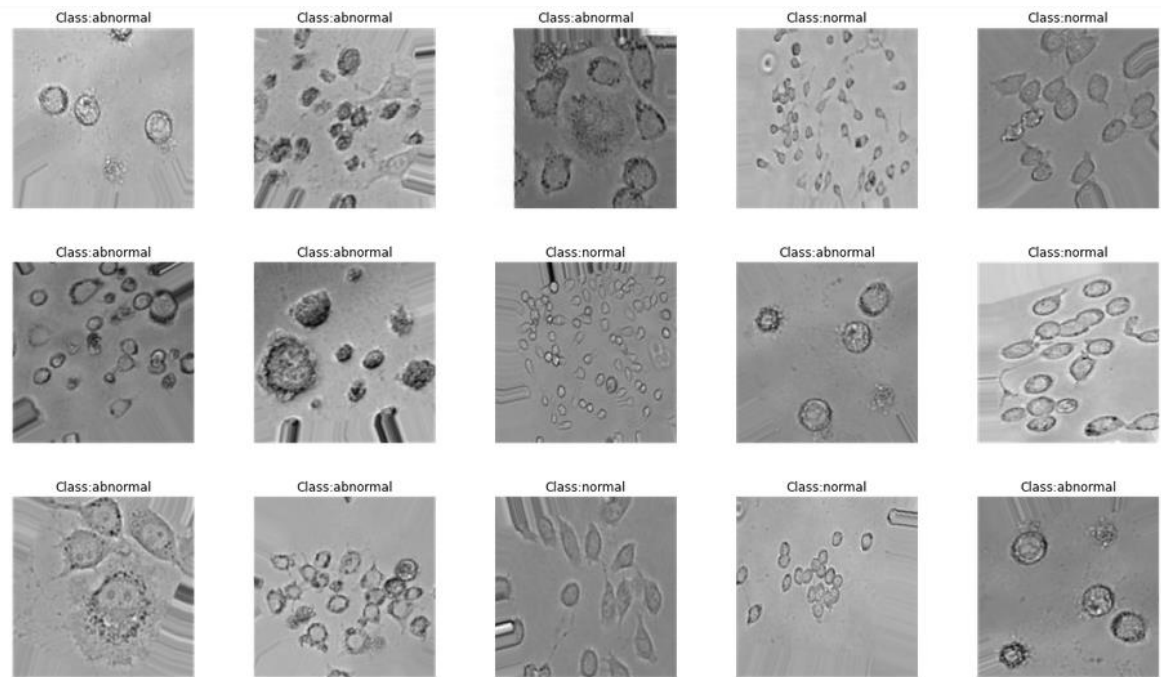


Fig: 4.1: Data Augmentation

Data augmentation can be implemented using various libraries and frameworks, such as Keras and PyTorch. These libraries provide built-in functions for applying different transformations to the images in the dataset.

It is important to note that data augmentation should be used with caution, as some transformations may not be appropriate for certain types of data. For example, flipping an image of a handwritten letter would result in a different letter, so this transformation would not be appropriate for a handwriting recognition task. Additionally, the transformations should be chosen carefully to ensure that they are relevant to the task at hand and do not introduce bias or noise into the data.

4.3 FEATURE EXTRACTION:

Feature extraction is the process of extracting relevant features from raw data to create a new representation of the data that is more informative and useful for a particular task. In machine learning, feature extraction is an essential step in building

models that can learn from complex and high-dimensional data, such as images, audio, and text.

The goal of feature extraction is to identify the most informative and discriminative features that capture the underlying patterns in the data. These features should be selected in a way that maximizes the performance of the machine learning models for the task at hand.

Here are some common techniques used for feature extraction:

1. **Dimensionality Reduction:** This involves reducing the number of features in the data by selecting the most important features or combining multiple features into a smaller set of features that capture the most relevant information.
2. **Feature Selection:** This involves selecting a subset of the original features that are most relevant to the task at hand. This can be done using statistical methods, such as correlation analysis, or using machine learning algorithms that can rank the features based on their importance.
3. **Feature Extraction:** This involves creating new features from the original features using mathematical transformations or other techniques. This can be done using methods such as Principal Component Analysis (PCA), which can transform the data into a new set of features that capture the most variance in the data.
4. **Transfer Learning:** This involves using pre-trained models to extract features from the data. For example, a pre-trained image classification model can be used to extract features from images that can be used for other tasks, such as object detection or image segmentation.

Feature extraction is a crucial step in machine learning, as it can have a significant impact on the performance of the models. By identifying the most informative and relevant features, machine learning models can learn more efficiently and accurately from complex and high-dimensional data.

4.4 CNN:

Deep Learning – which has emerged as an effective tool for analysing big data – uses complex algorithms and artificial neural networks to train machines/computers so that they can learn from experience, classify and recognize data/images just like a human brain does. Within Deep Learning, a Convolutional Neural Network or CNN is a type of artificial neural network, which is widely used for image/object recognition and classification. Deep Learning thus recognizes objects in an image by using a CNN. CNNs are playing a major role in diverse tasks/functions like image processing problems, computer vision tasks like localization and segmentation, video analysis, to recognize obstacles in self-driving cars, as well as speech recognition in natural language processing. As CNNs are playing a significant role in these fast-growing and emerging areas, they are very popular in Deep Learning.

A typical neural network will have an input layer, hidden layers, and an output layer. CNNs are inspired by the architecture of the brain. Just like a neuron in the brain processes and transmits information throughout the body, artificial neurons or nodes in CNNs take inputs, processes them and sends the result as output. The image is fed as input. The input layer accepts the image pixels as input in the form of arrays. In CNNs, there could be multiple hidden layers, which perform feature extraction from the image by doing calculations. This could include convolution, pooling, rectified linear units, and fully connected layers. Convolution is the first layer that does feature extraction from an input image. The fully connected layer classifies the object and identifies it in the output layer. “CNNs are feedforward networks in that information flow takes place in one direction only, from their inputs to their outputs. Just as artificial neural networks (ANN) are biologically inspired, so are CNNs. The visual cortex in the brain, which consists of alternating layers of simple and complex cells, motivates their architecture. CNN architectures come in several variations; however, in general, they consist of convolutional and pooling (or subsampling) layers, which are grouped into modules. Either one or more fully connected layers, as in a standard feedforward neural network.

CNNs have fundamentally changed our approach towards image recognition as they can detect patterns and make sense of them. They are considered the most

effective architecture for image classification, retrieval and detection tasks as the accuracy of their results is very high.

They have broad applications in real-world tests, where they produce high-quality results and can do a good job of localizing and identifying where in an image a person/car/bird, etc., are. This aspect has made them the go-to method for predictions involving any image as an input.

A critical feature of CNNs is their ability to achieve 'spatial invariance', which implies that they can learn to recognize and extract image features anywhere in the image. There is no need for manual extraction as CNNs learn features by themselves from the image/data and perform extraction directly from images. This makes CNNs a potent tool within Deep Learning for getting accurate results.

According to the paper published in 'Neural Computation', "the purpose of the pooling layers is to reduce the spatial resolution of the feature maps and thus achieve spatial invariance to input distortions and translations." As the pooling layer brings down the number of parameters needed to process the image, processing becomes faster even as it reduces memory requirement and computational cost.

While image analysis has been the most widespread use of CNNs, they can also be used for other data analysis and classification problems. Therefore, they can be applied across a diverse range of sectors to get precise results, covering critical aspects like face recognition, video classification, and street/traffic sign recognition, classification of galaxy and interpretation and diagnosis/analysis of medical images, among others.

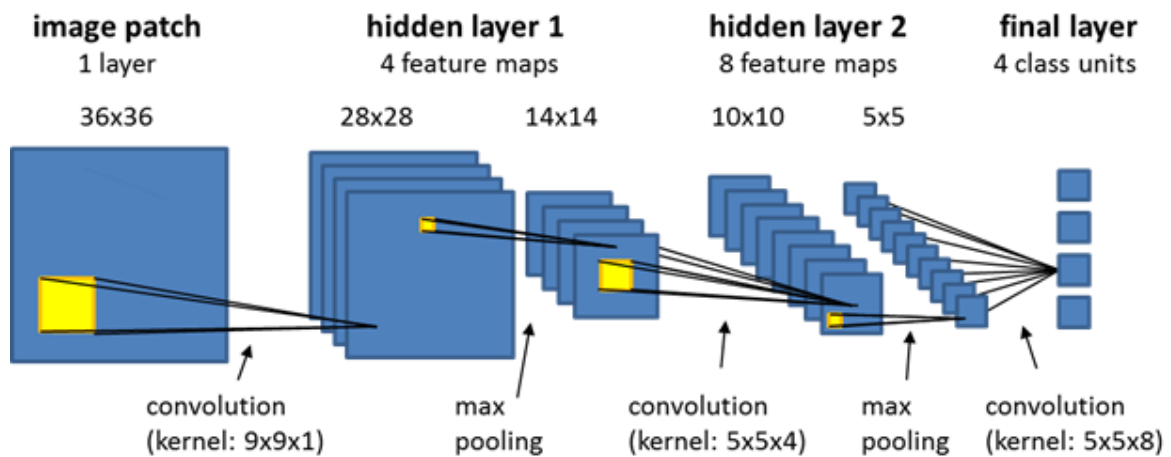


Fig: 4.2: CNN Architecture

4.4.1 LAYER TYPES:

There are many types of layers used to build Convolutional Neural Networks, but the ones you are most likely to encounter include:

- Convolutional (CONV)
- Activation (ACT or RELU, where we use the same or the actual activation function)
- Pooling (POOL)
- Fully connected (FC)
- Batch normalization (BN)
- Dropout (DO)

Stacking a series of these layers in a specific manner yields a CNN. We often use simple text diagrams to describe a CNN: INPUT => CONV => RELU => FC => SOFTMAX.

Here, we define a simple CNN that accepts an input, applies a convolution layer, then an activation layer, then a fully connected layer, and, finally, a softmax classifier to obtain the output classification probabilities. The SOFTMAX activation layer is

often omitted from the network diagram as it is assumed it directly follows the final FC.

Of these layer types, CONV and FC (and to a lesser extent, BN) are the only layers that contain parameters that are learned during the training process. Activation and dropout layers are not considered true “layers” themselves but are often included in network diagrams to make the architecture explicitly clear. Pooling layers (POOL), of equal importance as CONV and FC, are also included in network diagrams as they have a substantial impact on the spatial dimensions of an image as it moves through a CNN.

CONV, POOL, RELU, and FC are the most important when defining your actual network architecture. That’s not to say that the other layers are not critical, but take a backseat to this critical set of four as they define the actual architecture itself.

4.5 CNN MODELS:

4.5.1 RESNET50:

ResNets are made up of what's known as a residual block. This is built on the concept of "skip-connections" and uses a lot of batch-normalization to let it train hundreds of layers successfully without sacrificing speed over time.

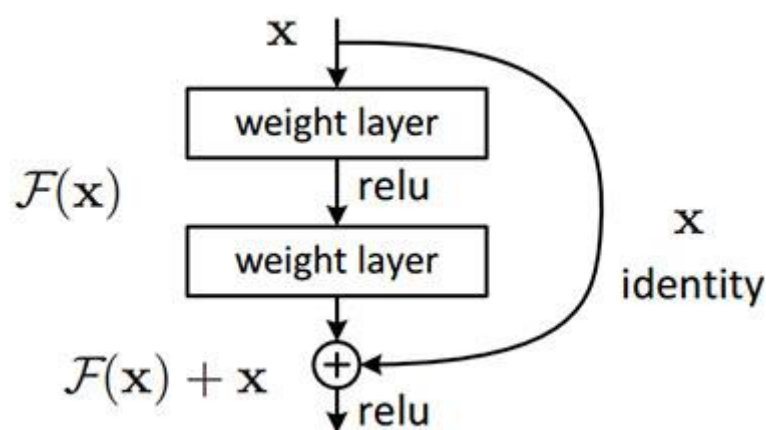


Fig: 4.3: Resnet layer

The first thing we note in the above diagram is that there is a direct link that skips several of the model's levels. The 'skip connection,' as it is known, lies at the core of residual blocks. Because of the skip connection, the output is not the same. Without the skip connection, input 'X' is multiplied by the layer's weights, then adding of a bias term.

The Architecture design is inspired on VGG-19 and has a 34-layer plain network to which shortcut and skip connections are added. As seen in the diagram below, these skip connections or residual blocks change the design into a residual network.

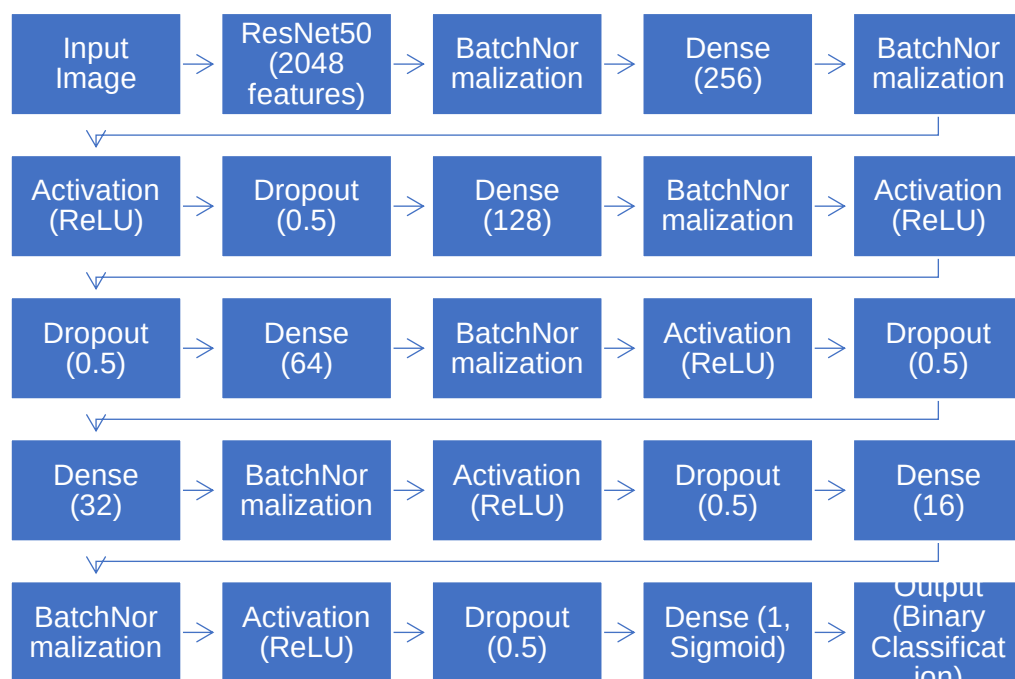


Fig: 4.4: Resnet50 Architecture

The idea was that the deeper layers shouldn't have any more training mistakes than their shallower equivalents. To put this notion into action, skip-connections were created. The creators of this network used a pre-activation variation of the residual block in which gradients can flow through the shortcut link to the earlier layers, minimising the problem of "vanishing gradients."

4.5.2 INCEPTIONV3:

The Inception V3 is a deep learning model based on Convolutional Neural Networks, which is used for image classification. The inception V3 is a superior version of the basic model Inception V1 which was introduced as GoogLeNet in 2014. As the name suggests it was developed by a team at Google.

The inception V3 is just the advanced and optimized version of the inception V1 model. The Inception V3 model used several techniques for optimizing the network for better model adaptation.

- It has higher efficiency
- It has a deeper network compared to the Inception V1 and V2 models, but its speed isn't compromised.
- It is computationally less expensive.
- It uses auxiliary Classifiers as regularizes.

The inception v3 model was released in the year 2015, it has a total of 42 layers and a lower error rate than its predecessors. Let's look at what are the different optimizations that make the inception V3 model better.

The major modifications done on the Inception V3 model are

- Factorization into Smaller Convolutions
- Spatial Factorization into Asymmetric Convolutions
- Utility of Auxiliary Classifiers
- Efficient Grid Size Reduction

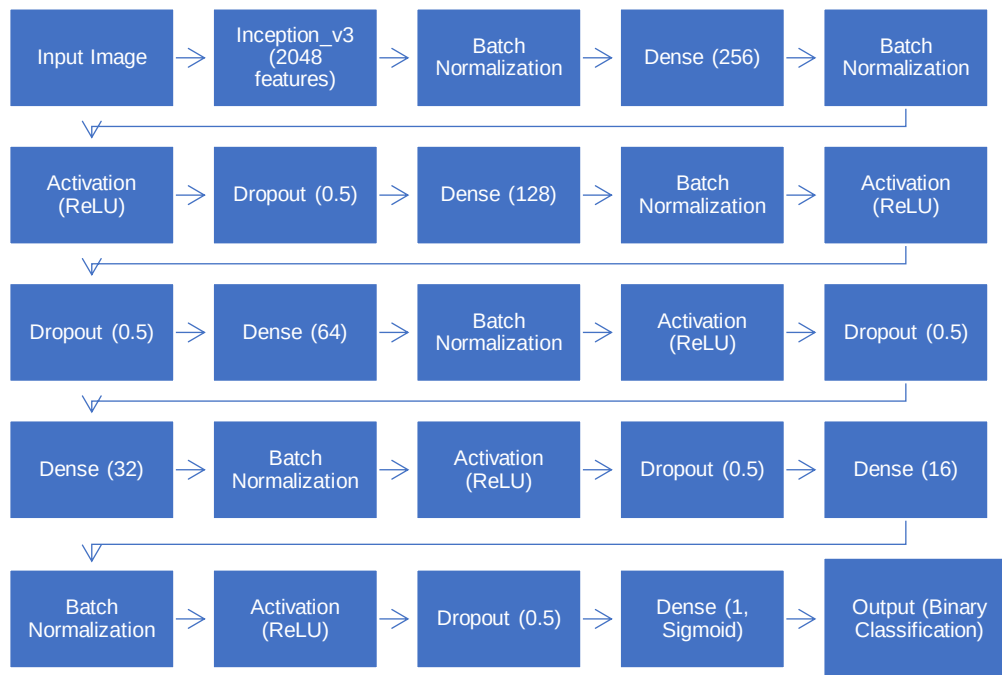


Fig: 4.5: InceptionV3 Architecture

In total, the inception V3 model is made up of 42 layers which is a bit higher than the previous inception V1 and V2 models. But the efficiency of this model is really impressive.

4.5.3 VGG16:

VGG16 (Visual Geometry Group 16) is a convolutional neural network (CNN) architecture that was developed by the Visual Geometry Group at the University of Oxford in 2014. It is a deep neural network that consists of 16 convolutional layers and three fully connected layers.

The VGG16 network is known for its simplicity and effectiveness in image classification tasks. It achieved state-of-the-art results on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2014 and has since been used as a base architecture for many other computer vision tasks.

The architecture of VGG16 consists of five blocks of convolutional layers, each followed by a max pooling layer. The first two blocks have two convolutional layers

each, while the last three blocks have three convolutional layers each. The fully connected layers at the end of the network use dropout to prevent overfitting.

VGG16 has over 138 million parameters, making it a very large network. It is typically used for image classification tasks such as object recognition, scene classification, and image segmentation.

The architecture of VGG16 consists of 16 layers, which are grouped into five blocks, as follows:

1. Input layer: This layer takes an input image of size $224 \times 224 \times 3$.
2. Block 1: This block consists of two convolutional layers, each with 64 filters of size 3×3 , followed by a max pooling layer of size 2×2 .
3. Block 2: This block consists of two convolutional layers, each with 128 filters of size 3×3 , followed by a max pooling layer of size 2×2 .
4. Block 3: This block consists of three convolutional layers, each with 256 filters of size 3×3 , followed by a max pooling layer of size 2×2 .
5. Block 4: This block consists of three convolutional layers, each with 512 filters of size 3×3 , followed by a max pooling layer of size 2×2 .
6. Block 5: This block consists of three convolutional layers, each with 512 filters of size 3×3 , followed by a max pooling layer of size 2×2 .
7. Fully Connected Layers: After the five convolutional blocks, the feature maps are flattened and passed through three fully connected layers of size 4096, 4096, and 1000, respectively. The final layer has 1000 units, corresponding to the number of classes in the ImageNet dataset.
8. Output Layer: The output layer uses a softmax activation function to output the probability of each class for the input image.

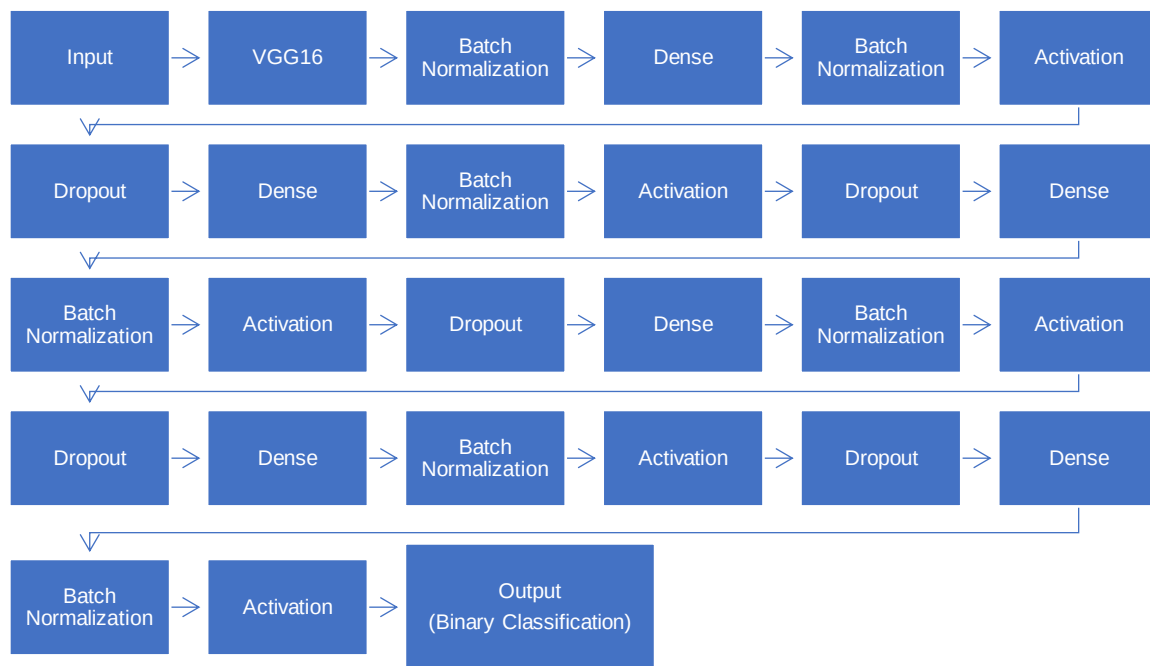


Fig: 4.6:Vgg16 Architecture

Overall, the VGG16 architecture has a total of 138 million parameters, which makes it one of the largest and most powerful convolutional neural networks for image classification tasks.

4.5.4 VGG19:

VGG19 is an extension of the VGG16 architecture, which was also developed by the Visual Geometry Group at the University of Oxford in 2014. VGG19 is a deep convolutional neural network that has 19 layers, including 16 convolutional layers and three fully connected layers.

The architecture of VGG19 is similar to that of VGG16, with the addition of more convolutional layers in each block. Specifically, VGG19 has four blocks of convolutional layers, each followed by a max pooling layer, and three fully connected layers at the end.

The architecture of VGG19 can be summarized as follows:

1. Input layer: This layer takes an input image of size 224 x 224 x 3.

2. Block 1: This block consists of two convolutional layers, each with 64 filters of size 3x3, followed by a max pooling layer of size 2x2.
3. Block 2: This block consists of two convolutional layers, each with 128 filters of size 3x3, followed by a max pooling layer of size 2x2.
4. Block 3: This block consists of four convolutional layers, each with 256 filters of size 3x3, followed by a max pooling layer of size 2x2.
5. Block 4: This block consists of four convolutional layers, each with 512 filters of size 3x3, followed by a max pooling layer of size 2x2.
6. Fully Connected Layers: After the four convolutional blocks, the feature maps are flattened and passed through three fully connected layers of size 4096, 4096, and 1000, respectively. The final layer has 1000 units, corresponding to the number of classes in the ImageNet dataset.
7. Output Layer: The output layer uses a softmax activation function to output the probability of each class for the input image.

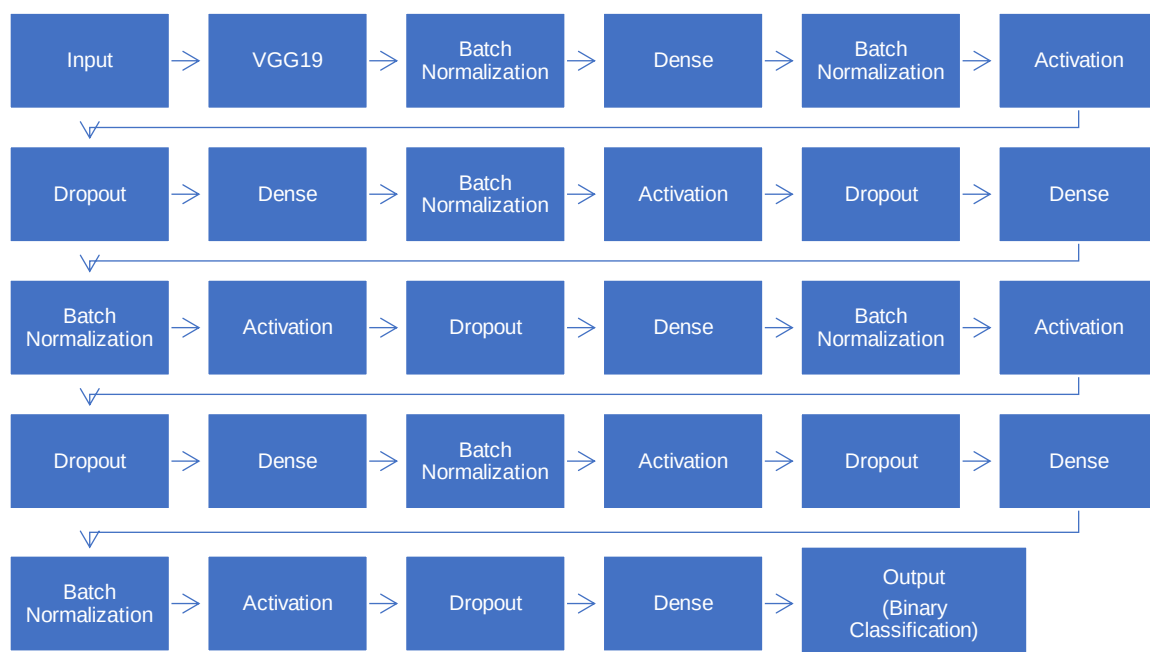


Fig: 4.7:Vgg19 Architecture

The VGG19 architecture has more parameters than VGG16, with a total of 143 million parameters. However, it also achieves higher accuracy on image

classification tasks, making it a popular choice for computer vision researchers and practitioners.

4.5.5 XCEPTION:

Xception (Extreme Inception) is a deep convolutional neural network architecture that was proposed by François Chollet in 2016. Xception is based on the Inception architecture, but with a novel approach to the way convolutions are performed.

In traditional convolutional neural networks, the convolution operation is performed by sliding a kernel across the input feature maps. In contrast, Xception separates the convolution operation into two separate steps: depth wise separable convolution and pointwise convolution.

The depth wise separable convolution performs a spatial convolution on each input channel separately, followed by a pointwise convolution that combines the output channels from the depth wise convolution. This approach reduces the number of parameters in the network, while maintaining or improving its accuracy.

The architecture of Xception can be summarized as follows:

1. Input layer: This layer takes an input image of size $299 \times 299 \times 3$.
2. Entry flow: This is the initial part of the network that consists of convolutional layers, batch normalization layers, and activation functions.
3. Middle flow: This is the main part of the network that consists of a repeated sequence of depth wise separable convolution layers, followed by batch normalization layers and activation functions.
4. Exit flow: This is the final part of the network that consists of a global average pooling layer, a fully connected layer, and a softmax activation function that outputs the probability of each class for the input image.

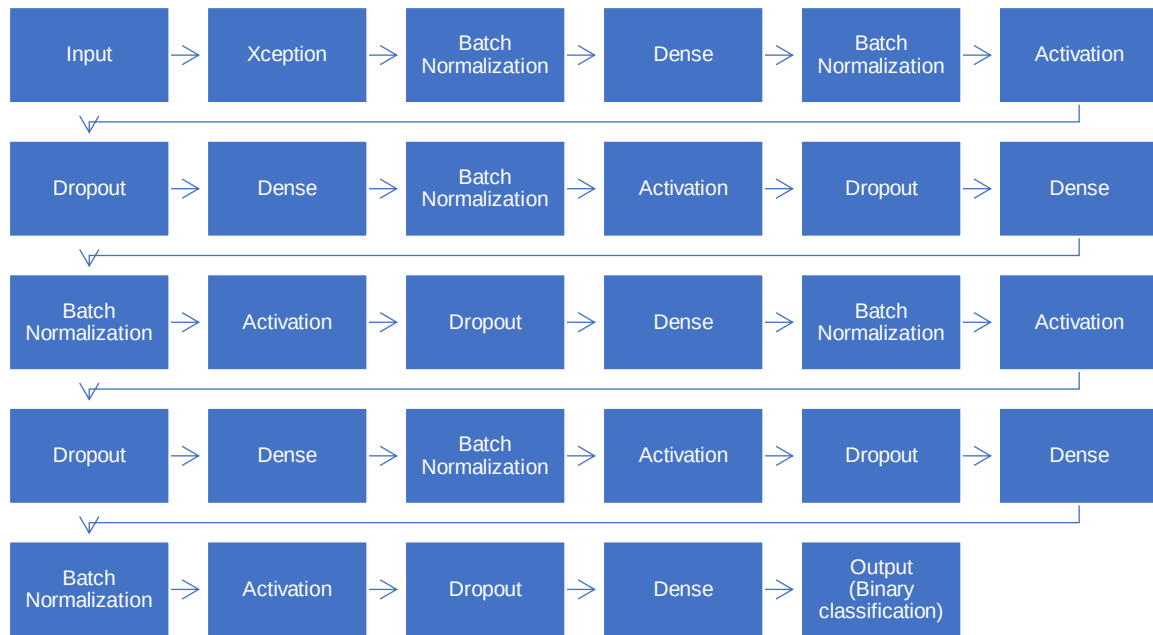


Fig: 4.8:Xception Architecture

The Xception architecture has 36 convolutional layers and a total of 22 million parameters, making it a relatively compact network compared to other deep neural networks. Xception has achieved state-of-the-art performance on various computer vision tasks, such as image classification, object detection, and semantic segmentation.

4.6 HYPERPARAMETER TUNING:

Hyperparameter tuning refers to the process of selecting the optimal values for the hyper parameters of a deep learning algorithm. Hyper parameters are parameters that are not learned during the training process, but are set by the user before training begins. Examples of hyper parameters include learning rate, batch size, number of hidden layers, and regularization strength.

Hyper parameter tuning is important because selecting the right hyper parameters can significantly improve the performance of a deep learning model. The optimal hyper parameters may vary depending on the dataset, the task, and the model architecture. Therefore, it is essential to experiment with different hyper parameter settings to find the best combination.

4.7 PRECISION:

Precision in CNN refers to the ability of the model to correctly predict the true positive instances out of all the instances it has predicted as positive. In other words, it is the ratio of true positive instances to the total number of instances predicted as positive. A high precision score indicates that the model is making very few false positive predictions.

Precision is an important metric in evaluating the performance of a CNN model, especially in applications where false positive predictions can be costly or harmful. For example, in medical diagnosis, a false positive prediction could result in unnecessary treatments or interventions.

Precision can be calculated as follows:

$$\text{Precision} = \text{True Positives} / (\text{True Positives} + \text{False Positives})$$

4.8 RECALL:

Recall in CNN refers to the ability of the model to correctly predict the true positive instances out of all the instances that are actually positive. In other words, it is the ratio of true positive instances to the total number of positive instances. A high recall score indicates that the model is able to detect a high proportion of positive instances.

Recall is an important metric in evaluating the performance of a CNN model, especially in applications where false negatives can be costly or harmful. For example, in medical diagnosis, a false negative prediction could result in missed diagnosis or delayed treatment.

Recall can be calculated as follows:

$$\text{Recall} = \text{True Positives} / (\text{True Positives} + \text{False Negatives})$$

4.9 F1 SCORE:

In machine learning, the F1 score is a measure of a model's accuracy that considers both precision and recall. It is the harmonic mean of precision and recall, where a score of 1 indicates perfect precision and recall, and a score of 0 indicates no precision or recall at all.

In the context of a convolutional neural network (CNN), the F1 score can be used to evaluate the performance of the model in a classification task. For example, if the CNN is trained to classify images as either "dog" or "cat", the F1 score can be calculated to determine how well the model is able to correctly identify images of each class.

To calculate the F1 score in a CNN, first, the precision and recall for each class must be calculated. Precision is the number of true positives (i.e., the number of images that are correctly classified as a certain class) divided by the total number of predicted positives (i.e., the number of images that are classified as a certain class, whether correctly or incorrectly). Recall is the number of true positives divided by the total number of actual positives (i.e., the total number of images that belong to a certain class).

Once the precision and recall for each class are known, the F1 score can be calculated as follows:

$$\text{F1 score} = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$$

This calculation takes into account both precision and recall for each class and provides a single score that represents the overall performance of the model in the classification task.

CHAPTER 5

RESULTS AND DISCUSSION

The size of the dataset, the hidden layers, and the hidden neurons per layer, the type of model, the experience environment, and other factors all affect how much time is needed. The approaches described in the methodology are examined in this section's analysis of the results. The outcomes of the Resnet50, vgg16, InceptionV3, Vgg19, and Xception methods are compared. Then, they are examined using the four factors of accuracy, F1 score, precision, and recall.

5.1 VALIDATION GRAPHS OF THE MODELS USED:

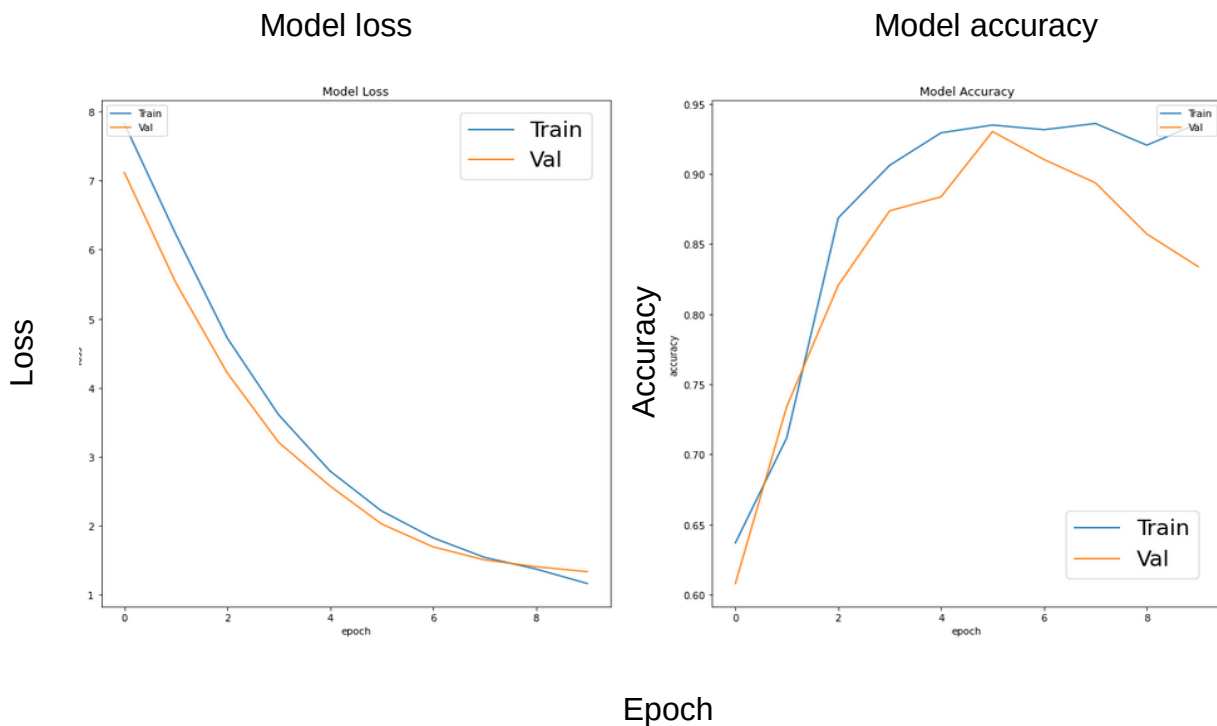


Fig: 5.1: Model loss and Model accuracy graphs for InceptionV3

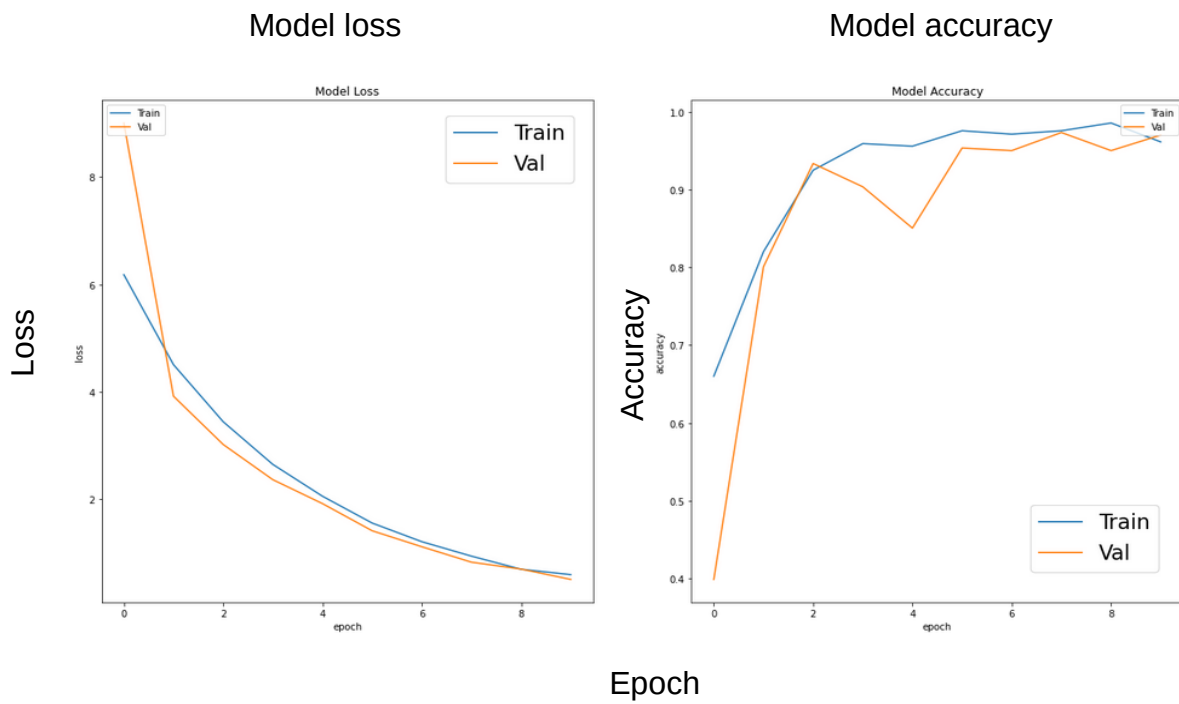


Fig: 5.2: Model loss and Model accuracy graphs for VGG16

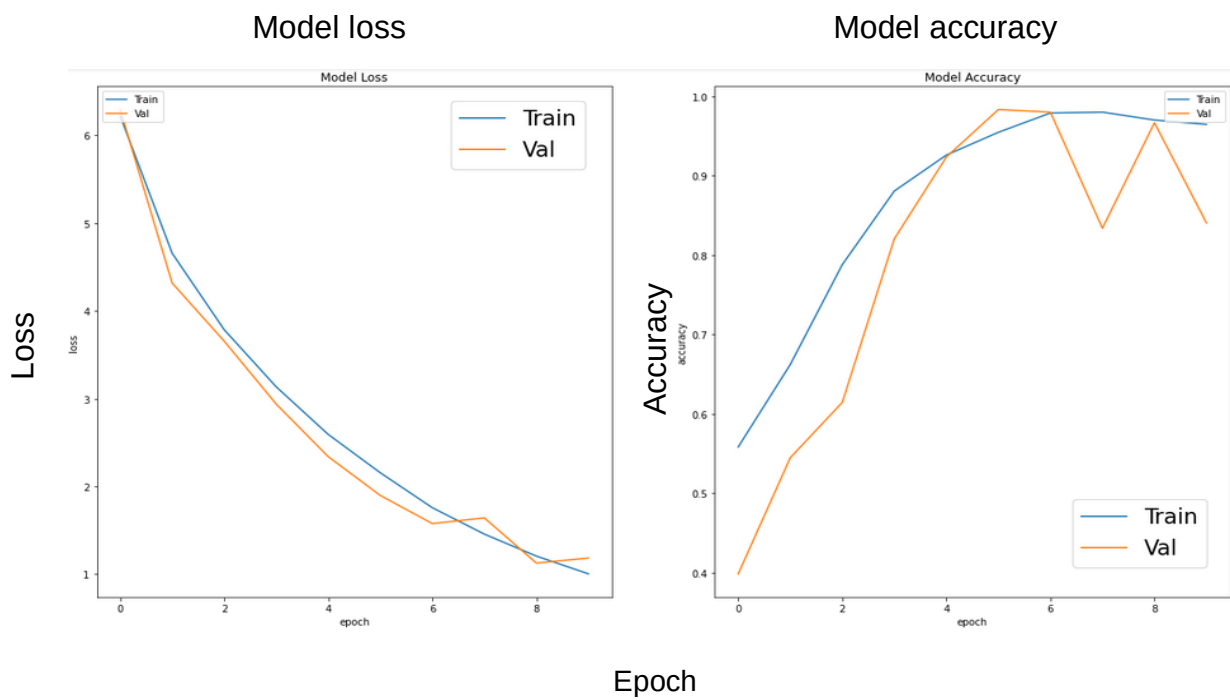


Fig: 5.3: Model loss and Model accuracy graphs for VGG19

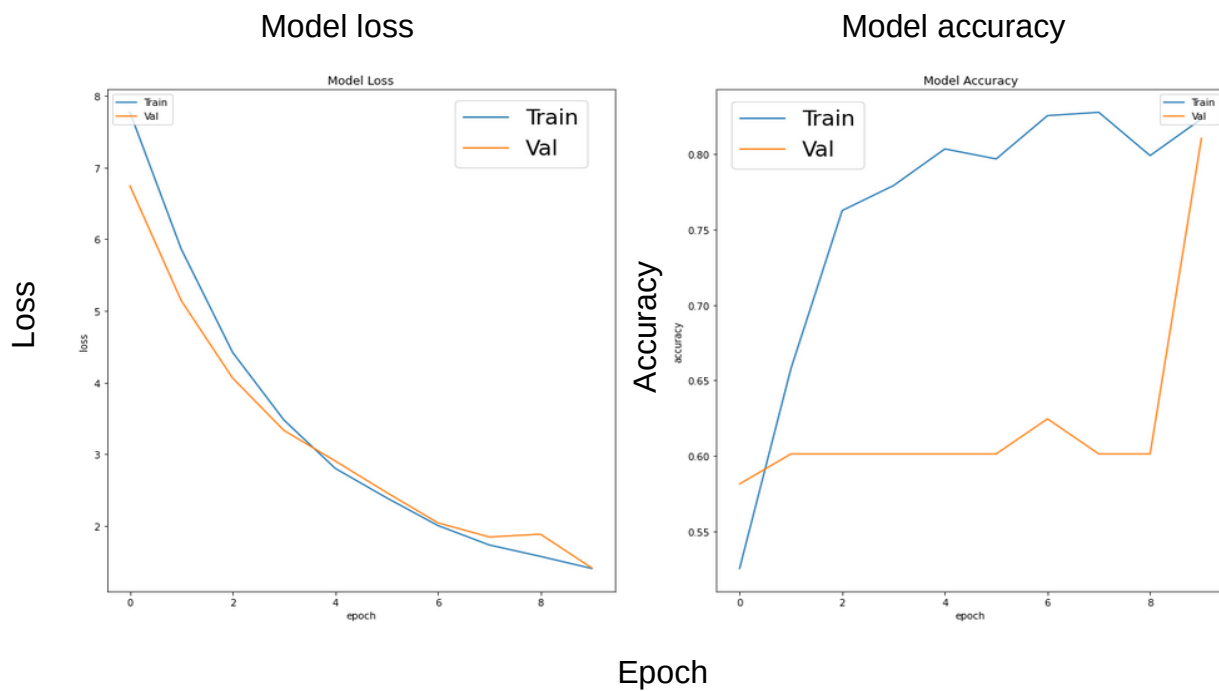


Fig: 5.4: Model loss and Model accuracy graphs for ResNet50

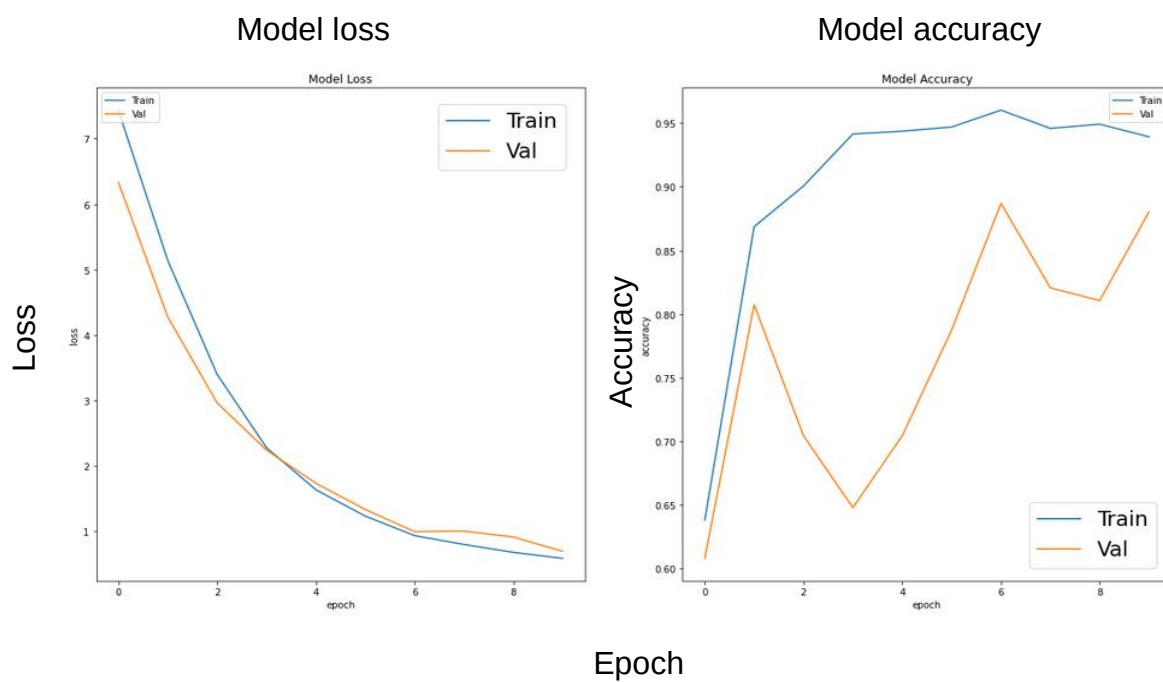


Fig: 5.5: Model loss and Model accuracy graphs for Xception

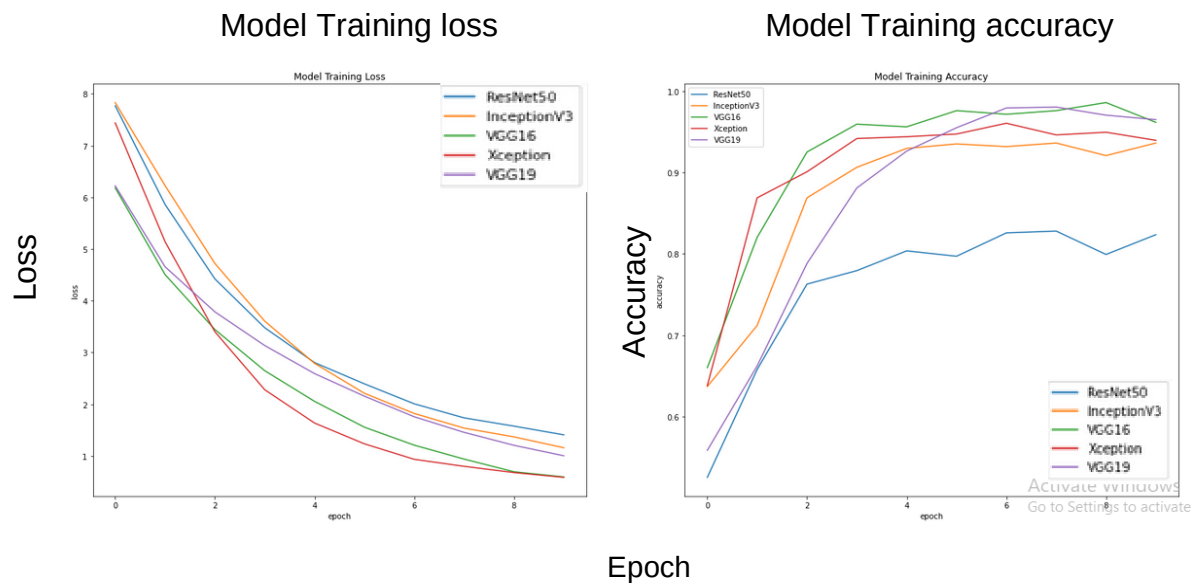


Fig: 5.6: Training loss and training accuracy graph for all the models used

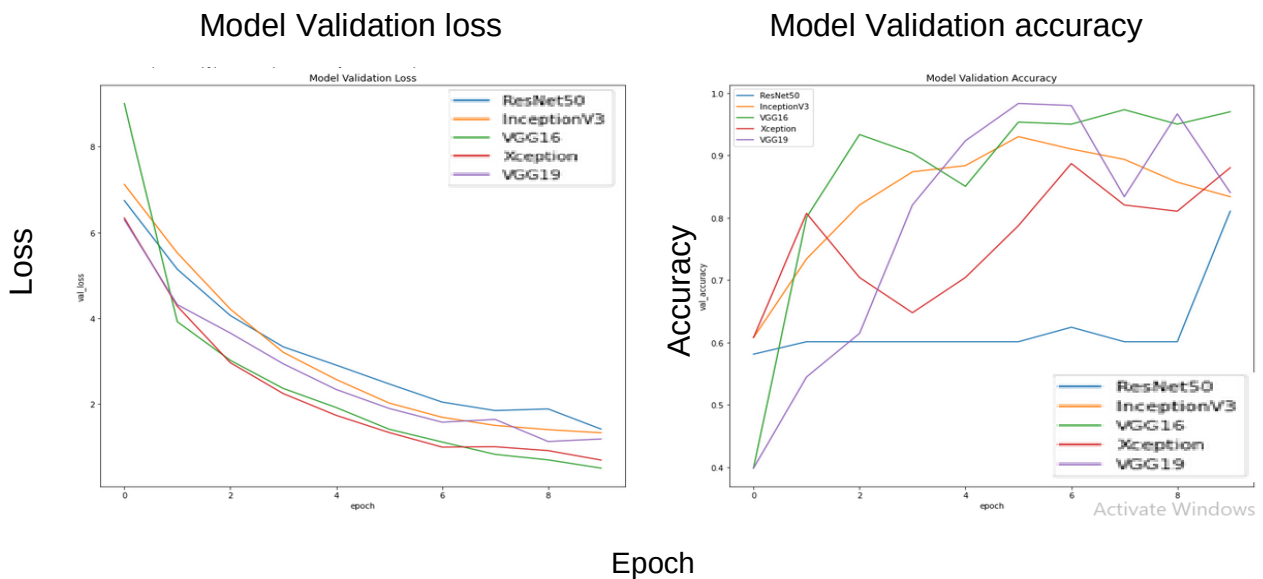


Fig: 5.7: Validation loss and validation accuracy graph for all the models used

The training loss and validation loss both decrease and stabilized at a specific point for all the five methods in the graphs used above.

In machine learning, a confusion matrix is a table that is often used to describe the performance of a classification model on a set of test data for which the true values

are known. A confusion matrix can help to evaluate the accuracy of a model by showing the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) produced by the classifier.

A confusion matrix can be useful in determining the overall performance of a classification model and in identifying areas for improvement. Other metrics such as precision, recall, and F1 score can be calculated using the values from the confusion matrix.

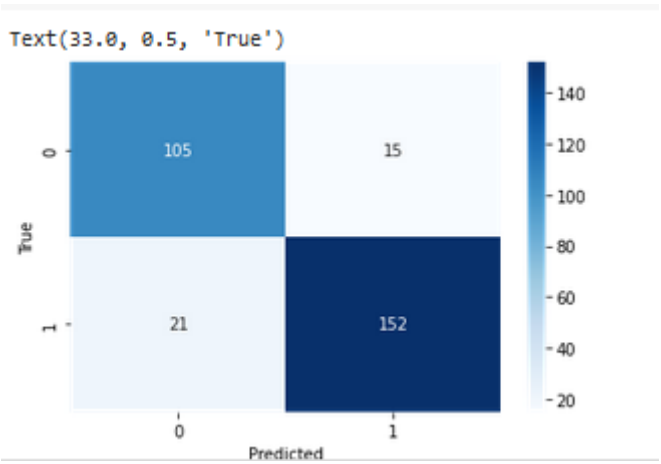


Fig: 5.8: Confusion matrix for InceptionV3

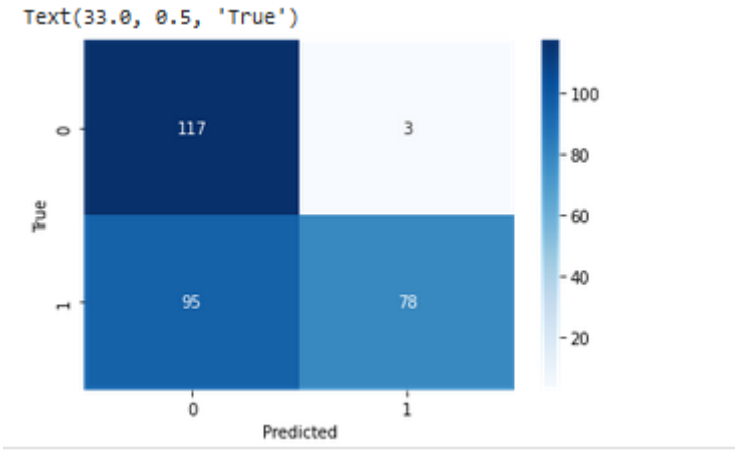


Fig: 5.9: Confusion matrix for Resnet50

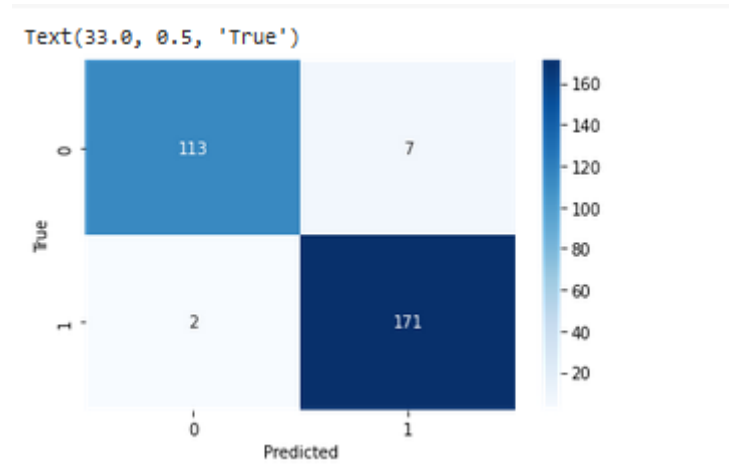


Fig: 5.10: Confusion matrix for VGG16

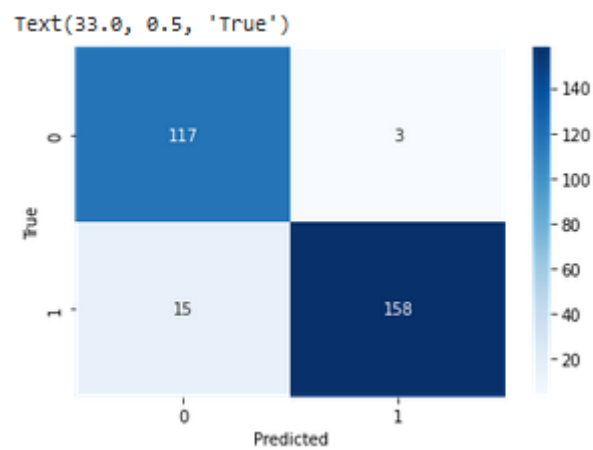


Fig: 5.11: Confusion matrix for VGG19

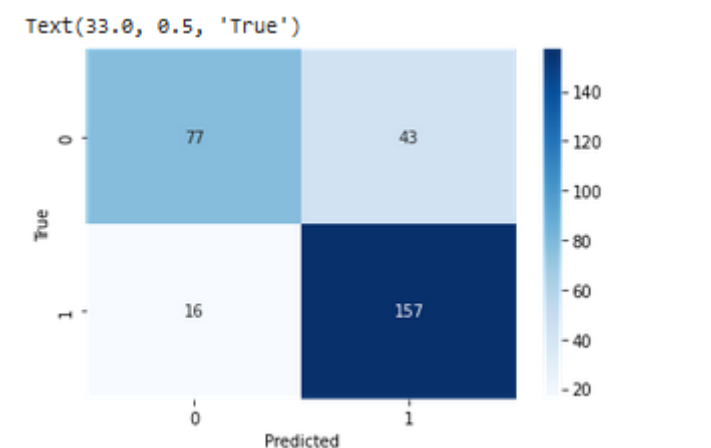


Fig: 5.12: Confusion matrix for Xception

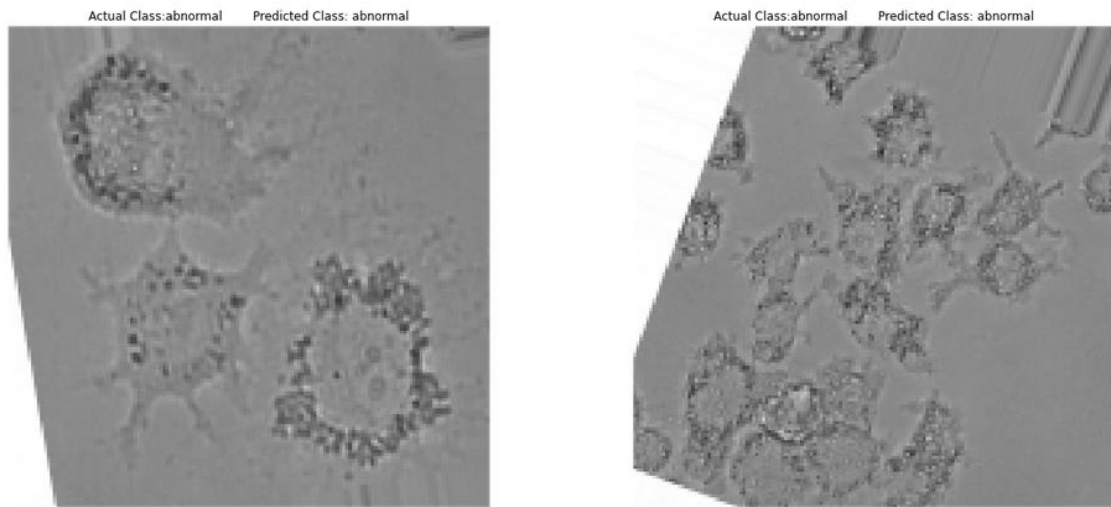


Fig: 5.13: Sample Prediction

Table 5.1: Comparing performances of all networks

Model	Class	Precision	Recall	F1-score	Accuracy
Resnet50	Ab(0)	0.76	0.89	0.82	0.81
	N (1)	0.88	0.75	0.81	
InceptionV3	Ab(0)	0.93	0.79	0.85	0.87
	N (1)	0.83	0.95	0.89	
Vgg16	Ab(0)	0.97	0.85	0.91	0.93
	N (1)	0.88	0.98	0.93	
Xception	Ab(0)	0.99	0.82	0.90	0.91
	N (1)	0.86	0.99	0.92	
Vgg19	Ab(0)	0.85	0.97	0.91	0.90
	N (1)	0.97	0.84	0.90	

From the above table, the model with high accuracy will be chosen and Hyper parameter tuning will be performed for that model. Here, VGG16 is the model which yields higher accuracy. So after performing hyper parameter tuning for that model, the accuracy of the model increases from 93% to 96% as we fine tune the learning rate of the model.

CHAPTER 6

SUMMARY AND CONCLUSIONS

The heterogeneity of microglial functions is in part linked to their shape and activation state, and much information can be obtained from their morphological characteristics. Microglial cell shape evolves from a resting fully ramified shape with extending processes and smaller soma, to the fully activated amoeboid shape with a larger soma and shorter processes. To date, microglia activation has been linked to three distinct functions: a classical pro-inflammatory activation state, an alternative activated anti-inflammatory state and a complementary deactivation state associated with an anti-inflammatory and functional repair phenotype. Classifying microglia activation states in histological images can help pathologists with disease diagnosis, provides key information for understanding diseases of the central nervous system and is essential for the validation of in vivo biomarkers that allow the stratification and monitoring of patients and populations at risk.

One of the major challenges in quantitative microglial analysis from immunohistological images is the development of automated microglial segmentation methods. Manual or semi-automated segmentation methods are time consuming and require user intervention with an element of subjectivity and inter-observer variability. Image analysis approaches commonly used for quantifying histology that rely on thresholding struggle with inhomogeneous immunohistochemistry images, where even staining over a single brain slice can be challenging. Another key challenge is the classification of distinct morphological states from resting ramified to activated amoeboid, particularly the intermediate states, which is intimately linked to the multiple phenotypes of microglia. Recent work has considered microglia as mono fractals, and used a single fractal dimension value to describe the entire microglial structure. However, fractal dimension alone is not enough to discriminate microglia activation states as it only measures the dimension of the space that the data fills but not how the data fills the space. Multifractal analysis methods have the potential to classify microglia morphology across the range of shapes from ramified

to activated microglia. However, a systematic review has indicated that there is little research in classification of microglia activation states.

In this work, the microglial cells images are classified with the help of deep learning techniques. Deep learning models which are used here takes care of segmentation, feature extraction and classification of the data set. In the existing work, for segmentation and feature extraction the authors used different methods in order to achieve the goal but in this work by introducing deep learning, the efficiency and accuracy of the prediction has also increased. For future work, the microglia classification framework defined here will be tested on large histology datasets of healthy and diseased brains, along with ground truth images, to validate its sensitivity to disease progression and pro-inflammatory states and therefore a viable tool for studying microglial biology.

REFERENCES

- [1] Oleg and Butovsky, "Microglia Function in the Central Nervous System During Health and Neurodegeneration," *The Annual Review of Immunology*, February 9, 2017.
- [2] Jeong, et.al "Brain Inflammation and Microglia: Facts and Misconceptions," *Experimental Neurobiology*, vol. 22, 2013.
- [3] Akiyama, et.al "Inflammation and Alzheimer's disease", *Neurobiology aging*, 2000.
- [4] Perry, et.al "Activation patterns of microglia and their identification in the human brain," *Neuropathology and Applied Neurobiology*, 2013.
- [5] Guang Luo, et.al "The changing phenotype of microglia from homeostasis to disease," *Transl Neurodegener*, vol. 1, 2012.
- [6] Streit, "Microglial response to brain injury: a brief synopsis," *Toxicologic pathology*, vol. 28, 2000.
- [7] Stanley, et.al "Statistical physics and physiology: monofractal and multifractal approaches," *Physica A: Statistical Mechanics and its Applications*, vol. 270, 1999.
- [8] Valous, et.al "Multistage histopathological image segmentation of iba1-stained murine microglia in a focal ischemia model: Methodological workflow and expert validation," *Journal of neuroscience methods*, vol. 213, 2013.
- [9] Koles and Withey, "Medical Image Segmentation: Methods and Software," International Conference on Functional Biomedical Imaging (NFSI-ICFBI), 2007.
- [10] Yan and Song, "Image Segmentation Algorithms Overview", *Computer Vision and Pattern Recognition*, 2017.

- [11] Jelinek, et.al "Differentiating grades of microglia activation with fractal analysis," *Complexity International*, 2008.
- [12] Karperien, et.al "Quantitating the subtleties of microglial morphology with fractal analysis," *Frontiers in cellular Neuroscience*, 2013.
- [13] Yuchun, et.al "Novel Methods for Microglia Segmentation, Feature Extraction and Classification," *IEEE/ACM Trans Comput Biology Bioinformation*, 2016.
- [14] Bascuñana, et.al "Machine Learning-Supported Analyses Improve Quantitative Histological Assessments of Amyloid- β Deposits and Activated Microglia," *Journal of Alzheimer's disease : JAD*, vol. 76, no. 2, 2021.
- [15] Kamalahasan, "REVIEW ON CLASSIFICATION ALGORITHMS IN IMAGE PROCESSING," *International Journal of Innovative Trends in Engineering and Research*, 2017.
- [16] Neil Ash, et.al "Automated segmentation and analysis of retinal microglia within ImageJ," *Experimental Eye Research*, 2021.
- [17] Esgiar, et.al "Fractal Analysis in the Detection of Colonic Cancer Images" *IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE*, MARCH 2002.
- [18] Erickson, et.al "Machine Learning for Medical Imaging", *Radiographics*, Feb 2017.
- [19] Shen Wu, and Suk, "Deep Learning in Medical Image Analysis", *Annual Review of Biomedical Engineering*, March 9, 2017.
- [20] ALBAWI, et.al "Understanding of a Convolutional Neural Network" *ICET*, 2017.

- [21] Foskey, et.al "SUPPORT VECTOR MACHINE FOR DATA ON MANIFOLDS: AN APPLICATION TO IMAGE ANALYSIS" IEEE International Symposium in Biomedical Imaging, 2008.
- [22] Shelhamer, and Darrell, "Fully Convolutional Networks for Semantic Segmentation", IEEE Transactions on Pattern Analysis and Machine Intelligence Vol.39 No.4, April 2017.
- [23] Watkins, et.al "Glial activation: a driving force for pathological pain" *TRENDS in Neurosciences* Vol.24 No.8 August 2001.
- [24] França, et.al "Fractal and Multifractal Properties of Electrographic Recordings of Human Brain Activity: Toward Its Use as a Signal Feature for Machine Learning in Clinical Applications" 10 December 2018.
- [25] Fernandez and Jelinek, "Use of Fractal Theory in Neuroscience: Methods, Advantages, and Potential Problems" *Methods*. August 2001.
- [26] Karperien, et.al "Differentiating grades of microglia activation with fractal analysis" *Complexity International*, October 2008.
- [27] SMITH JR, et.al "A FRACTAL ANALYSIS OF CULTURED RAT OPTIC NERVE GLIAL GROWTH AND DIFFERENTIATION", *Neuroscience*, 2011 .
- [28] Abdolhoseini, et.al "Segmentation, Tracing, and Quantification of Microglial Cells from 3D Image Stacks" *Scientific Reports*, 12 June 2019.
- [29] Lopez, et.al "An unsupervised machine learning method for discovering patient clusters based on genetic signatures" *Journal of Biomedical Informatics*, 2018.
- [30] Krstonošić, et.al "Box-Counting and Multifractal Analysis in Neuronal and Glial Classification", *Advances in Intelligent Systems and Computing*, Vol. 187, 2013.

[31] Morrison and Filosa, "A quantitative spatiotemporal analysis of microglia morphology during ischemic stroke and reperfusion," *Journal of Neuroinflammation* 2013.