

CLASSIFICATION OF MALIGNANCY IN LUNG USING DEEP LEARNING ALGORITHMS

Submitted in partial fulfilment of the requirements for the award of Bachelor of
Engineering Degree in **Biomedical Engineering**

by

POOJASAKTHI S (38240024)



**DEPARTMENT OF BIOMEDICAL ENGINEERING
SCHOOL OF BIO AND CHEMICAL ENGINEERING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY - 2022



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF BIOMEDICAL ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **S. PoojaSakthi (38240024)** who carried out the project entitled "**CLASSIFICATION OF MALIGNANCY IN LUNG USING DEEP LEARNING ALGORITHMS**" under our supervision from **September 2021 to March 2022**.

Dr. S. KRISHNAKUMAR, M.Sc., Ph.D.,

Guide

Dr. T. SUDHAKAR, M.Sc., Ph.D.,

Head of the Department

Submitted for Viva-voce Examination held on 02.05.2022

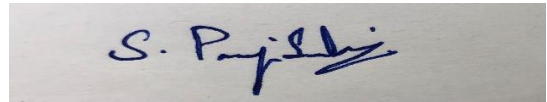
Internal Examiner

Dr. BETHANNEY JANNEY J

External Examiner

DECLARATION

I **PoojaSakthi S (38240024)** hereby declare that the Project Report entitled “**Classification Of Malignancy In Lung Using Deep Learning Algorithms**” done by us under the guidance of Dr. **S.Krishnakumar**, M.Sc., Ph.D., Department of Biomedical Engineering is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Biomedical Engineering.

A rectangular box containing a handwritten signature in blue ink. The signature appears to be 'S. Pooja Sakthi' with a stylized flourish at the end.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

We are pleased to acknowledge my sincere thanks to the **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 my thanks to Dr. **T. Sudhakar**, M.Sc., Ph.D., Head of the Department, Department of Biomedical Engineering for providing me necessary support and details at the right time during the progressive reviews.

We would like to express my sincere and deep sense of gratitude to my Project Guide Dr. **S. Krishnakumar**, M.Sc., Ph.D., for his/her valuable guidance, suggestions, and constant encouragement that paved way for the successful completion of my project work.

We wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Biomedical Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

The disease of the lung is one of the most common that cannot be ignored and can lead to death if not treated. CT can currently assist doctors in detecting lung diseases in the early stages. As mentioned, in many cases, the diagnosis of lung diseases is based on the experience of doctors, which may cause some problems for some patients. In this paper, a Deep Learning-based methodology is presented for detecting abnormal lung tissue growth. The objective is to have a tool that can detect with a greater probability and with great accuracy. Manually interpreting results in misdiagnosis. In this study, full-fledged Computer Tomography images of lungs of humans have been compared with normal and malignant state images. Using the distinctive texture features of the images, a network can be built to classify the normal images, distinguishing them from malignant ones. To identify the normal images among the images in this study, a deep learning network has been proposed. Using a confusion matrix, the proposed network's capability has been quantified in terms of classification accuracy.

CHAPTER NO	TABLE OF CONTENTS	PAGE NO
	TITLE	I
	ABSTRACT	v
	LIST OF ABBREVIATIONS	ix
	LIST OF FIGURES	x
1	INTRODUCTION	01
1.1	INTRODUCTION	
1.2	PROJECT OVERVIEW	02
1.1.1	APPLICATION OF LUNG CANCER DETECTION	02
1.1.2	ELEMENT OF RESEARCH	02
1.3	PROJECT GOALS	02
1.4	PROJECT CHALLENGES	03
1.4.1	NEURAL NETWORK ARCHITECTURE DESIGN CHOICE	03
1.4.2	UNDERSTANDING DEEP LEARNING CONCEPTS	03
1.4.3	DEEP LEARNING INTEGRATION	03
1.5	LUNG CANCER	04
1.5.1	ADENOCARCINOMA	04
1.5.2	SQUAMOUS CELL CARCINOMA	04
1.5.3	LARGE CELL CARCINOMA	04
2	LITERATURE SURVEY	06
2.1	SURVEY OF PREVIOUSLY USED ALGORITHMS	06
3	AIM AND SCOPE	09
3.1	AIM	09

	3.2	SCOPE	09
4		METHODS AND ALGORITHMS USED	11
	4.1	ARTIFICIAL NEURAL NETWORK	11
	4.1.1	FORWARD PROPAGATION	11
	4.1.2	BACKWARD PROPAGATION	12
	4.1.3	ACTIVATION FUNCTIONS	12
	4.2	CT ACQUISITION AND PICTURE PROCESSING	13
	4.2.1	PREPROCESSING	14
	4.2.2	FEATURE MASTERING vs CHARACTERISTIC CHOICE	14
	4.2.3	HIERARCHICAL FUNCTION	15
	4.2.4	END-TO-END LEARNING	15
	4.3	CONVOLUTION NEURAL NETWORK	17
	4.3.1	OVERVIEW	17
	4.3.2	ARCHITECTURE	17
	4.3.3	CONVOLUTION LAYER	18
	4.3.4	POOLING LAYER	19
	4.4	GOOGLE NET	20
	4.4.1	AUXILIARY CLASSIFIERS	21
	4.4.2	CHARACTERISTICS AND FEATURES OF GOOGLNET CONFIGURATION	22
	4.5	VGG16	23
	4.5.1	TRAINING IMAGE SIZE	26
	4.5.2	CONFUSION MATRIX	28
	4.6	BLOCK DIAGRAM OF CNN	32
	4.6.1	PREPROCESSING	32
	4.6.2	CLASSIFICATION	33
	4.6.3	TRAINING THE NETWORK	33
	4.7	BLOCK DIAGRAM OF GOOGLNET	35
	4.7.1	PREPROCESSING	36
	4.7.2	CLASSIFICATION	36

	4.7.3	TRAINING THE NETWORK	36
4.8		BLOCK DIAGRAM OF VGG16	37
	4.8.1	PREPROCESSING	37
	4.8.2	CLASSIFICATION	38
	4.8.3	TRAINING THE NETWORK	38
5		RESULTS AND DISCUSSION	39
	5.1	RESULTS	39
	5.1.1	RESULT FOR CNN	39
	5.1.2	OUTPUT	39
	5.1.3	RESULT FOR GOOGLNET	41
	5.1.4	OUTPUT	41
	5.1.5	RESULT FOR VGG 16	42
	5.1.6	OUTPUT	42
	5.2	DISCUSSION	42
6		SUMMARY AND CONCLUSIONS	43
	6.1	SUMMARY	43
	6.2	CONCLUSIONS	43
		REFERENCES	45

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
AUC	AREA UNDER THE ROC CURVE
ANN	ARTIFICIAL NEURAL NETWORK
CT	COMPUTED TOMOGRAPHY
CNN	CONVOLUTION NEURAL NETWORK
DNN	DEEP NEURAL NETWORK
DSB	DETECTION INFORMATION TECHNOLOGY BOWEL
FPSO	FUZZY PARTICLE SWARM OPTIMIZATION
FC	FULLY CONNECTED
GIF	GRAPHICS INTERCHANGE FORMAT
GPU	GRAPHICS PROCESSING UNIT
HOG	HISTOGRAM OF ORIENTED GRADIENTS
ILSCVRC14	IMAGE NET LARGE SCALE VISUAL RECOGNITION EXPERTS CHALLENGE,2014
ICO	INITIAL COIN OFFERING
JPEG	JOINT PHOTOGRAPHIC EXPERTS GROUP
LCLC	LARGE CELL LUNG CARCINOMA
LUSC	LUNG SQUAMOUS CELL CARCINOMA
LID-IDRI	LUNG IMAGE DATABASE CONSORTIUM
IC	IMAGE COLLECTION
LBP	LOCAL BINARY PATTERN
MATLAB	MATrix LABoratory
MRI	MAGNETIC RESONANCE IMAGING
NSCLC	NON-SMALL CELL LUNG CANCER
PET	POSITRON EMISSION TOMOGRAPHY
PGF	PORTABLE GRAPHICS FORMAT
PSD	PHOTOSHOP DOCUMENT
ReLU	RECTIFIED LINEAR UNITS
SAE	STACKED AUTOENCODER
SCC	SQUAMOUS CELL CARCINOMA
TIFF	TAGGED IMAGE FILE FORMAT
VGG	VISUAL GEOMETRY GROUP

LIST OF FIGURES

FIGURE NO	NAME OF THE FIGURE	PAGE NO
4.1	AN ARTIFICIAL NEURAL NETWORK	11
4.2	FORWARD AND BACK PROPAGATION IN CNN	12
4.3	ACTIVATION FUNCTION	13
4.4	CONVOLUTION NEURAL NETWORK EXAMPLES	17
4.5	CONVOLUTION NEURAL NETWORK ARCHITECTURE EXAMPLE	18
4.6	STRIDE CONVOLUTION EXAMPLE	19
4.7	MAX AND AVERAGE POOLING OPERATION	20
4.8	CONVOLUTION AND POOLING LAYER OF GOOGLNET	21
4.9	LAYERS OG VGG16	25
4.10	TRAINING THE DATA	27
4.11	CONFUSION MATRIX	28
4.12	ACCURACY CALCULATION WITH EXAMPLE	30
4.13	PRECISION CALCULATION WITH EXAMPLE	30
4.14	RECALL CALCULATION WITH EXAMPLE	31
4.15	BLOCK DIAGRAM OF CNN	32
4.16	BLOCK DIAGRAM OF GOOGLNET	35
4.17	BLOCK DIAGRAM OF VGG16	37
5.1	CONFUSION MAATRIX FOR CNN NETWORK	39
5.2	TRAINING THE DATA IN INITIAL STAGE	40
5.3	TRAINING THE DATA IN MIDDLE STAGE	40
5.4	TRAINING THE DATA IN FINAL STAGE	40
5.5	CONFUSION MATRIX FOR GOOGLNET	41
5.6	CONFUSION MATRIX FOR VGG16	42

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Lung cancer is one of the most serious diseases in the world today, and it is the leading cause of mortality in the previous several decades. It also kills more people each year than breast, prostate, and colon cancer put together. The addiction to cigarettes is one of the leading causes of lung cancer. Furthermore, carcinogenic surroundings such as radioactive gas and air pollution contribute to the spread of this disease. In addition, genetic factors also have a major contribution to lung cancer. Uncontrolled magnification of tissue creates lung cancer. Primary originate from cells within secondary (metastatic) cancer begin in another part of the body and therefore spread to lungs. Lung cancer can be cancerous (malignant) or non-cancerous (benign). They are evaluated on the basis of the cells in which they appear, as well as a number ranging from to. Lower-grade cancers are classified as Grades I and II. In some cases, cancer grades III and IV are regarded to be of higher severity. In the human body, there are two types of cells. Normal cells are small and confined, whereas cancer-affected cells are rapidly forming and can be easily spotted. These cells appear to be aberrant and dissimilar to regular cells. This type of cell grows quickly and is more prone to spread. Poorly differentiated or high grade are terms used to describe them. Lung cancer can be life-threatening; thus, a patient's exact diagnosis and treatment are critical. Tumour tissue analysis is performed in a pathological laboratory. Microscopic investigation, such as biopsy, and electronic modalities, such as CT, Ultrasound, and others are used to examine tumour tissue. CT Scan is the most commonly utilised pathological test, and it is very popular for diagnosis. It uses high-resolution, high-contrast pictures of the lung in various positions to provide a three-dimensional assessment of the lesion. An automated strategy for cancer diagnosis was developed in this research work, which used CT Scan Gray-scale images.

1.2 PROJECT OVERVIEW

1.2.1 Application for lung cancer detection

A piece of internet software has been developed to demonstrate proof of concept. A consumer must add a CT Scan to the software. After that, the software processes the document and displays the images to the customer. The customer then selects which experiment she or he wishes to participate in, and the software performs a pre-CT Scan and compares the image to the predicted version. The version's output is then displayed to the customer. The customer can choose to view the images in a carousel format.

1.2.2 Element of research

This file contains a number of research that aid in the deep understanding of cap potential for locating lung tumours in CT scans. The information utilised to educate the version was gathered from a comprehensive understanding of opposition known as LUNA 2016, which consisted of CT scans that were interpreted by radiologists. The radiologists determined whether a nodule was lung cancer or not, and the location was plotted in three-dimensional coordinates. Deep Learning research has also been carried out to ensure that the best framework for teaching the model can be utilised.

1.3 PROJECT GOAL

The completed challenge must meet the following objectives:

1. Deep Learning Self-Education A large part of this task is numerous self-education; for example, the writer knew little or nothing about deep learning, and as part of this challenge, the writer must have a strong grasp of deep learning ideas.
2. Evaluation and User Research The challenge must have a layout that is focused on the customer. This implies that the device must be created in such a way that it can support specific customers. The client in this challenge could be medical professionals who specialise in lung cancer diagnosis.
3. Upload CT scans to the system The device must be capable of collecting CT scans from users in order to be utilised by Deep Learning.
4. Lung Cancer Detection System The device must be capable of detecting lung tumours within the CT experiment images that clients have supplied.

5. System: Show the Results The device must be capable of delivering information that our customers can understand and benefit from.

1.4 PROJECT CHALLENGES

This phase discusses the demanding situations that had been triumph over to finish this challenge. Each of the demanding situations are in short delivered and particular in later sections of the record. This phase may be beneficial for individuals who purpose to do a deep gaining knowledge of challenge with huge photo datasets.

1.4.1 Neural Network Architecture Design Choices

Neural networks are designs that are built in and of themselves, and there are many different types of architectures accessible to handle certain challenges. One of the writer's main responsibilities is to use a structure that can discover malignant tumour types inside the information. To overcome this, the author must conduct extensive research in order to implement the best version layout prior to teaching. Multi-layer perceptron's, convolutional networks, and collection styles are examples of neural community architectures.

1.4.2 Understanding Deep Learning Concepts

Deep learning is an intriguing, though novel, concept for the author. To complete this assignment, the author conducted research and performed an examination in order to fulfil the challenge objectives. Stanford professor Andrew Ng's deep learning of route on Coursera has been a good valuable resource that has been used. This path allows newbies to benefit from a foundation for more in-depth learning.

1.4.3 Deep Learning Integration

In comparison to typical system learning methods, Python is unusual in its ability to integrate deep learning fashions into applications. A standard system for learning algorithms can be serialised as a document and loaded into a programme. TensorFlow is a framework for deep learning about fashions. Integrating the version necessitates additional code to ensure that the version executes in the same thread as the software. Within the implementation chapter, this is also defined.

1.5 LUNG CANCER

1.5.1 Adenocarcinoma

In the United States, lung adenocarcinoma is the most frequent kind of primary lung cancer. It is classified as non-small cell lung cancer (NSCLC) and has a significant link to past smoking. Although the incidence and mortality rates have decreased, it is still the most common cause of cancer death in the United States. Lung adenocarcinoma mainly develops from the mucosal glands and accounts for roughly 40% of all lung malignancies. It is the most prevalent subtype seen among non-smokers. Lung adenocarcinoma normally develops on the periphery of the lung, and it can be seen in scars or areas of persistent inflammation in many cases.

1.5.2 Squamous cell carcinoma

Squamous cell carcinoma of the lung (SCC of the lung), often known as squamous cell lung cancer, is a non-small cell lung cancer (NSCLC). Squamous cell lung tumours are most commonly seen in the centre of the lung or in the major airway, such as the left or right bronchus. Tobacco smoke is the primary cause of cellular change. This activity examines the diagnosis and treatment of squamous cell carcinoma of the lung, with a focus on the role of the interprofessional team in patient care.

Squamous cell carcinoma of the lung (SCC of the lung), often known as squamous cell lung cancer, is a non-small cell lung cancer (NSCLC). The most prevalent kind of NSCLC is adenocarcinoma, which is followed by squamous cell carcinoma of the lung, which is more frequent in women. This is ascribed to a shift in cigarette smoking habits, however there is no conclusive proof.

1.5.3 Large cell carcinoma

Non-small cell lung cancer comes in a variety of types, including large cell lung carcinoma (LCLC) (NSCLC). LCLC is a kind of lung cancer that starts in the outer areas of the lungs and grows and spreads more quickly than other types of lung cancer. Shortness of breath and exhaustion are the most common early signs of LCLC. NSCLC accounts for over 85% of all lung cancers, with LCLC accounting for about 10%. When a tumour is examined under a microscope, big cell lung carcinomas, also known as large cell lung cancers, are named for the huge size of the cancer cells that can be observed (as opposed to the tumour size, which also tends to be quite large). Hormone-like compounds secreted by LCLC cells can produce paraneoplastic disorders. The drugs cause the immune system to target

certain bodily cells. Specific gene mutations, including as those in the RUNX1, ERBB4, BRCA1, and EPHA3 genes, have been linked to an increased risk of certain kinds of LCLC, according to research.

CHAPTER 2

LITERATURE SURVEY

This Chapter provides an overview of previous research carried out on Lung Cancer Image Classification. All researchers have aim to develop such a system which predict and detect the cancer in its early stages. Also tried to improve the accuracy of the Early Prediction and Detection system by preprocessing, segmentation feature extraction and classification techniques of extracted database. The major contributions of the research are summarized below.

2.1 SURVEY OF PREVIOUSLY USED ALGORITHMS

Ali Imadad, et. al., (2020) conducted study on cancer lungs detection using artificial neural network backpropagation based grey level occurrence matrices characteristics on CT scan images. The perspective of the lung pictures is one of the key advantages of employing CT scan images in this technique. The results demonstrate that the system can recognise CT images of normal and cancerous lungs with an accuracy of 80%. Hopefully, it will be used to assist medical workers and researchers in detecting lung cancer.

Asuntha (2020) focused on the effect of input size on the categorization of lung nodules using convolutional neural networks. The goal of this study was to employ convolutional neural networks to evaluate CT lung screenings in order to decrease false positives and demonstrate their power and influence on overall accuracy. Various functions were applied.

Bhandary, et. al., (2019) conducted a study focused entirely on the identification of lung cancer medical images using deep neural networks. The goal of this study was to see if there was any evidence of cancer in a patient's lungs. To aid clinicians in visual diagnostics by training deep neural networks to detect lung cancer. The key benefit is that doctors will have more assistance in detecting and treating lung cancer in its early stages.

Chenwei Deng, et. al., (2012) worked in using LUAD and LUSC for Classification and Mutation Prediction from Non-Small Cell Lung Cancer Histopathology Images.

Coccia and mario, (2019) wrote a paper titled "Optimization of features using artificial neural networks for categorization of type of lung cancer." The overall goal is to decrease human error in CT scans of lung cancer patients and to diagnose them.

Coudray, et. al., (2018) studied and featured the use of convolutional neural networks in Deep Learning for the detection and classification of lung cancer nodules in CT scans.

Gemma paella (2020) showed how they used Convolution neural networks, Autoencoders, and Deep Belief Networks for Lung NoduleClassification on Computed Tomography Images Using Deep Learning.

Ginneken (2020) completed their research and study focused on lung cancer detection using 3D convolutional deep neural networks. Using CAD is one of the key benefits. The convolutional neural network is one such technique, which best defines a series of deep learning models with filters that can be built with local pooling processes alternating on input CT images to build an array of hierarchical complicated features.

Han et al (2020) released their study on the notion of Deep Learning for lung cancer diagnosis and classification. They employed the Fuzzy Particle Swarm Optimization (FPSO) technique and the Histogram of Oriented Gradients (HoG), Local Binary Pattern (LBP), Scale Invariant Feature Multimedia Tools and Applications Transform (SIFT), and Zernike Moment.

Kanavati (2020) Lung Diseases screening using low-dose CT scans using a deep learning methodology was applied. Computed tomography scans are used for identifying lung diseases because they offer a thorough view of the tumour in the body and follow its progress. Minho N DO (2003) explained drawbacks include that they can't help with disease detection at an early stage. Naik (2020) proposed algorithm for Lung Diseases Detection Using Co-Learning from Chest CT Pictures and Clinical Demographics is a technique for detecting lung diseases in CT scan images that uses an automated approach.

Pandian, et. al., (2016) used an algorithm for diagnosing lung illnesses is being developed using various methods. The following are some of the restrictions: Allows the radiologist to spend more time assessing the patient. Implementing neural network systems is complex.

Pandian (2015) proposed a computed tomography (CT) scan algorithm. Image processing technologies are used to analyse the images. The death rate from lung illnesses is increasing day by day in both young and old individuals, and its limits are becoming more obvious when compared to other diseases. The reaction time is really slow. Examine the data in further detail. It's a technique for extracting higher-level information from a photograph, such as shading, shape, and texture. It is one of the most significant features for recognising objects or a photographic region of interest. Surface is an example of data or an exhibit of the structure set up in an image, and it is a big component of many picture formats.

Requelme (2020) implemented surface, from a broad viewpoint, alludes to surface characteristics and the existence of Texture highlights can be erased in a variety of ways, including using factual, auxiliary, model-based, and changing data, with the Gray Level Co event Matrix being the most popular (GLCM). The second-request factual data regarding spatial connection of pixels in a picture is contained in GLCM.

Lakshmana prabu (2018) implemented to ensure the accuracy of any arrangement framework, it is usually relied on the selection of the most suited qualities. As a result, finding a good arrangement of highlights, which is a quantifiable approach that leverages the spatial connection of pixels, is critical.

In this chapter, we have discussed about the reference papers which are helpful in doing our proposed method. Here, I have taken classification of Lung Cancer Image as the base paper of our project which uses Deep Learning algorithm CNN, GoogLeNet and Vgg 16. In this proposed method, I have developed Algorithm for Lung Cancer Image Classification.

CHAPTER 3

AIM AND SCOPE

3.1 AIM

Lung diseases are the most frequent diseases that should not be disregarded since they might lead to death if not treated promptly. Currently, CT can be utilised to aid clinicians in the early detection of lung diseases. In many situations, doctors' diagnoses of lung diseases are based on their experience, which can lead to certain patients being overlooked and causing issues. In several fields of medical imaging diagnostics, deep learning has shown to be a popular and effective strategy. Three types of deep neural networks (e.g., CNN, VGG, ANN, DNN, and SAE) are used to model lung disease calcification in this article. With modest modifications for benign and malignant lung nodules, these networks are used to the CT image classification job. The LIDC-IDRI database was used to assess these networks.

3.2 SCOPE

In this project, classification of lung cancer such as Adenocarcinoma, Large Cell Carcinoma, Squamous Cell Carcinoma has been differentiated from the normal lung images through deep learning techniques.

Two different pre-trained neural networks namely GoogleNet and Vgg16 network are applied to the dataset and its performance such as accuracy, sensitivity and precision value are determined and these values are compared with the CNN network with two block of layers consists of convolution layer, normalization layer followed by the pooling layer.

To train and test the performance of the network, 100 samples images are taken from each class. Out of which 70 images are used for training and the remaining 30 are used for validation purpose. It is observed the accuracy achieved by GoogleNet and Vgg16 Network was attained with just two blocks of CNN network. The reason why GoogleNet and Vgg16 Network are selected is that out of existing pre-trained Neural Network, the smallest and heaviest Networks are GoogleNet and Vgg16 Network respectively. GoogleNet has depth of 28 layers with Memory Size of 27MB and 7.0 million parameters which accepts the image with the size of

224x224 pixels. On the other hand, Vgg16 has 16 layers, 535 MB memory size with 144 million parameters which accepts the image with the size of 224x224 pixels.

CHAPTER 4

METHODS AND ALGORITHMS USED

4.1 ARTIFICIAL NEURON

A synthetic neuron is included in neural networks, much as a mind neuron. A synthetic neuron has a weight to determine the strength of a connection, a linear characteristic to compute, and an activation characteristic to compute the weighted sum of the linear characteristic, which is then compared to a threshold value.

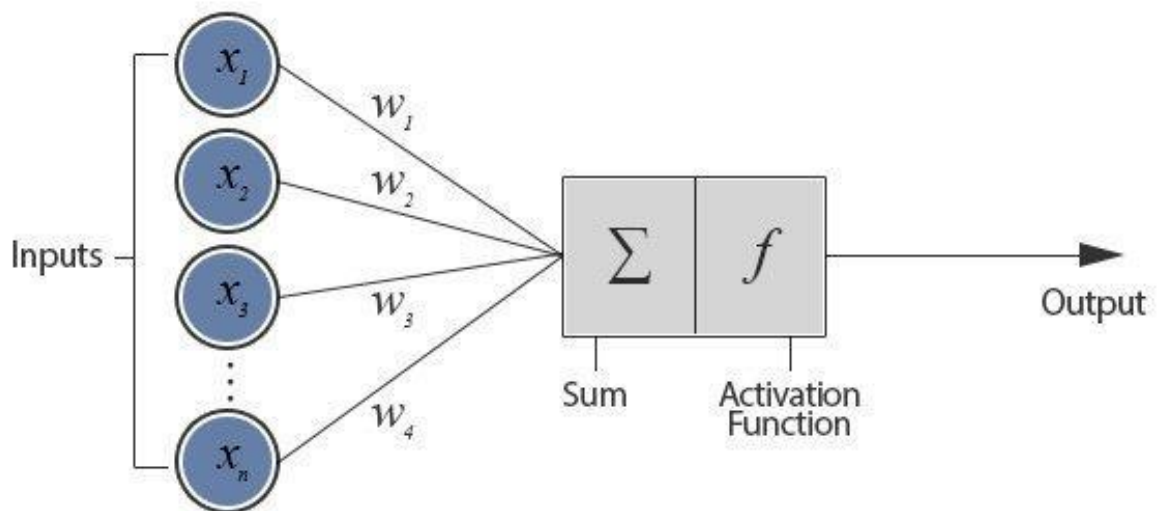


Fig :4.1: An Artificial Neuron additionally referred to as a Perceptron

It's also worth noting that synthetic neurons, exist. Three variables can be adjusted to include exceptional linear characteristic techniques like as Gradient Descent, Logistic Regression, and others, as well as exceptional activation features such as ReLU (Rectified Linear Unit) or TanH features, depending on the difficulty to solve.

4.1.1 Forward Propagation

Forward propagation is a technique that successively records movement from the corresponding entrance layer, hidden layers, and output layer. A hidden layer typically contains a weight that is used in an optimization set of rules such as Gradient Descent, an activation characteristic such as Sigmoid, TanH, Leaky ReLU, and many others, and a Loss Function that is used to calculate the loss or mistakes of the characteristic that we use to lower back propagate to change the weights.

4.1.2 Back Propagation

Because it feeds the community backwards, reverse propagation is an alternative to forward propagation. After the errors have been determined using forward propagation techniques, back propagation is utilised to change the load of the neural community. Gradient Descent is used in lower back propagation to change the weights of the neural network in subsequent portions of this chapter.

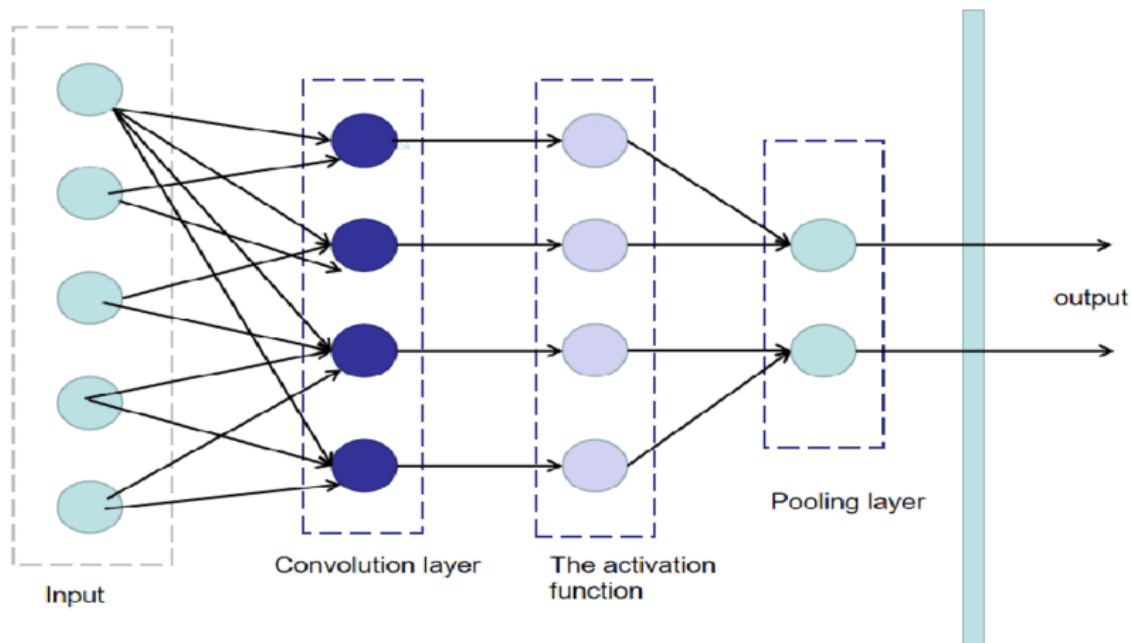


Fig :4.2 : Forward propagation and back propagation in CNN Algorithm

4.1.3 Activation Functions

A neuronal community's activation properties are critical. They make it possible for neural networks to develop nonlinear features in order to solve challenges. Sigmoid / Logistic, Tanh, ReLU, Leaky ReLU, Parametric ReLU, Exponential Linear Units (ELUs), Scaled Exponential Linear Unit (SELU) and so on. Along these Sigmoid, TanH, and ReLU are the three most often employed activation characteristics. Activation characteristics allow or inhibit neurons from firing onto the next layer by comparing them to activation characteristic thresholds. Activation features are used in both forward and backward propagation. In forward propagation, an activation characteristic is used to calculate the loss when the output of a characteristic is compared to a real range, and in backward

propagation, they're used to replace the neural community's parameters. In a neuronal community, figure depicts commonly utilised activation properties.

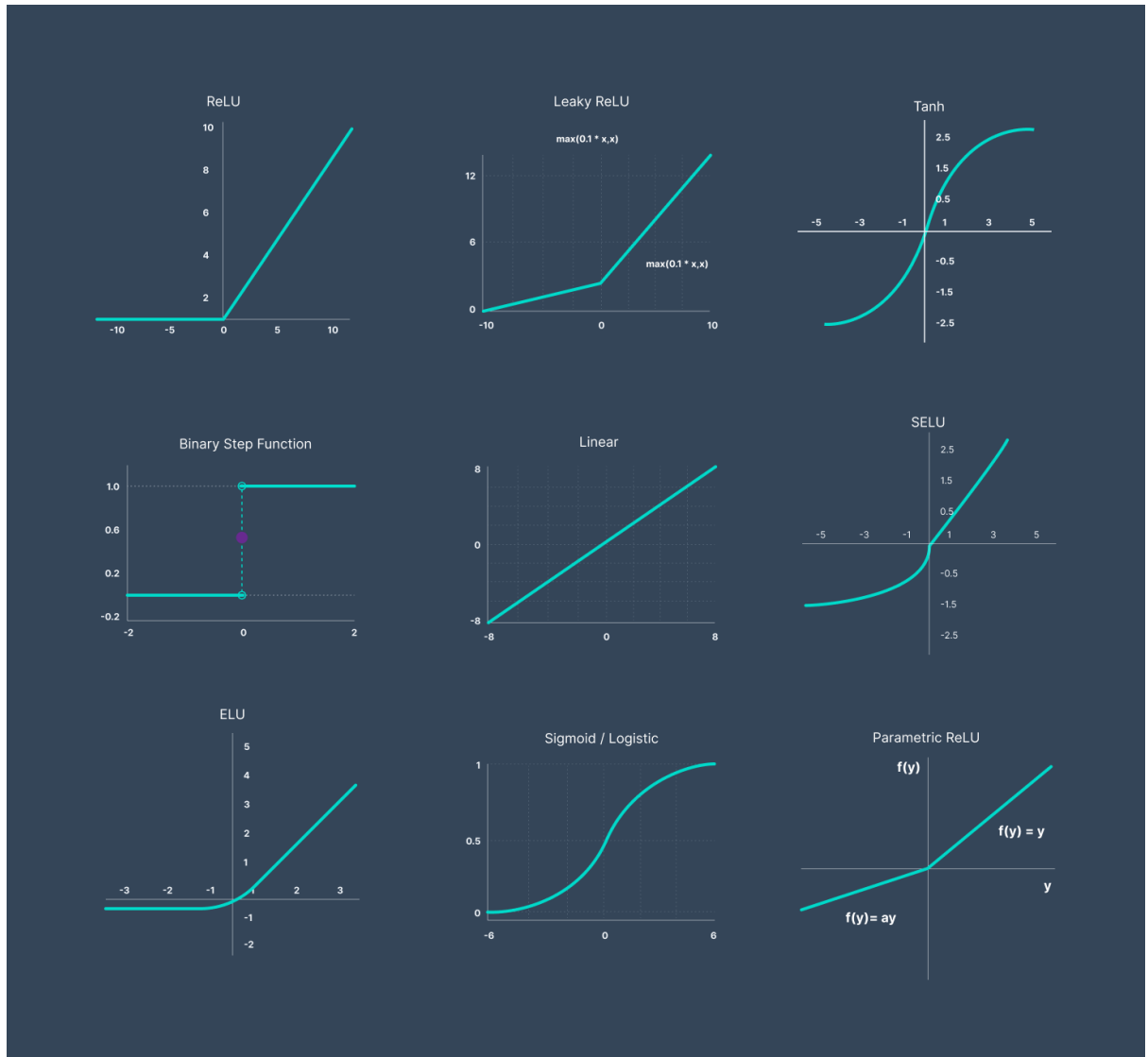


Fig:4.3: Activation Functions

4.2 CT ACQUISITION AND PICTURE PREPROCESSING

CTs were collected using a GE "Light speed" CT scanner (GE Medical System) for remedy, pre-treatment, and follow-up scans, as per our institution's defined scanning protocols. A portion of the snap photos came from PET–CT acquisitions, and the follow-up scans had varying axial spacing. For pre-treatment and 1, 3, and 6-month follow-up CT scans following definitive radiation therapy, the entry of the tumour image site is specified in the middle of the identified seed factor.

In 3-D Slicer 4.8.1, the seed factors had to be stated manually. The CT voxels were interpolated to 1x1x1 mm³ using linear and neighbour interpolation because

to the wide range of slice thicknesses and in-flight resolution. It is necessary to interpolate the imaging data to a homogenous resolution in order to have a strong entry for the suggested design.

4.2.1 Preprocessing

In a simple neural network, a CNN, Vgg16 and Gogglenet will increase in the number of hidden nodes. The neural community can be utilised to keep track of more difficult entrance calculations. Because if the activation characteristic is linear, in comparison to a single hidden layer neural community, the intensity of the hidden layer of the community no longer decorates the ability to express, the nonlinear must be utilised for each hidden layer.

The CNN, Vgg16 and Gogglenet utilised to deconstruct the processing component of the pulmonary nodule so that exclusive community layers may be used to get attributes of pulmonary nodules of different sizes. Within the CNN, there are additional issues with adjacent extremum and gradient diffusion.

Convolutional Neural Networks (CNN) have grown to dominate sample detection, identification, segmentation, and classification systems in both clinical and non-clinical sectors. Indeed, in a world where adequate schooling data is available, CNNs have largely replaced the previous period of radiomic/texture evaluation methodologies described above. In our own research, after we had accumulated and selected sufficient large education units, our CNN-based totally techniques began to beat the previous brand new texture and Vgg-based totally strategy.

While an in-depth examination of such tactics is beyond the scope of this essay, it is good worth familiarizing yourself with the primary differences from previous strategies and their advantages.

4.2.2 Feature mastering vs. Characteristic choice

Unlike Radiomic/texture assessment procedures, CNN strategies build functions from start rather than selecting from a palette of manufactured or pre-determined sets that rely on the set of rules developer's contextual experience.

4.2.3 Hierarchical function

The initial few levels of a CNN often contain many layers of functions, allowing the community to evaluate the correlations between functions at a far higher level of sophistication than is possible with a single characteristic extraction degree. Consider the following example: spiculations extending into the parenchyma may

be detected using a textural characteristic, which includes adjacent entropy of the joint histogram. However, a CNN can examine this, as well as the fact that spiculations encompass the whole perimeter of the nodule, indicating that it is a malignant nodule.

4.2.4 End-to-end learning

CNNs are often taught that the entire community is taught to optimum the bother of interest, i.e., all of the community's parameters are altered till the height class price is achieved.

In conclusion, every level of the LungX texture-based approach that we developed needed to be built and optimised in my perspective, and there was no guarantee that the entire pipeline would be ideal. Segmentation-free Because segmentation is handled with implicitly inside the set of rules, the CNN approach can work without the nodule segmentation phase. We identified enormous sensitivity of the prediction rating to the segmentation stage in our next examination of our LungX set of rules.

This -degree approach was developed to avoid competition by inferring the check set labels with a large number of entries. In comparison to the LungX competition, this one required the creation of a fully automated pipeline that took a CT image and generated a risk of most malignancies.

The impacts were assessed using the log-loss characteristic, which is well-known in Kaggle tournaments. Unlike AUC, the log-loss characteristic penalises more certain, but incorrect, outputs more than fewer certain ones. All of the top three submissions used Deep Learning-trained CNNs and scored within a few decimal places of each other, with scores of 0.39755, 0.40117, and 0.40127, respectively, where a log-lack of 0 corresponds to a perfect rating. The pinnacle-groups were found to have AUC-ROC effects of 0.85 and 0.87, respectively.

The winning entry employed a three-D Convolutional Neural Network training using a combination of DSB training data and the publicly available dataset inside the LUNA16 nodule identification competition, which was obtained from the LIDC-IDRI dataset. Because no nodules are found in the validation and check datasets, an accurate automated nodule identification step is required for effective classification. In reality, based only on the following write-ups from the winning teams, a significant amount of effort was put into this stage rather than the next class step.

The triumphing crew made an interesting observation about the distribution of nodule sizes. The LUNA16 dataset included a lot of larger tiny nodules (mean = eight mm), but the DSB datasets had a lot of big lesions (mean =14 mm), therefore the team had to change the education set of rules to compensate. Furthermore, the size distribution of nodules among most malignancies and benign patients was claimed to be quite exclusive; the malignant nodules were large, while the benign nodules were small. As a result, forecasting the analysis just on the basis of length is unlikely to yield precise results.

The problem of length bias in education and check units is a serious issue that we've looked into in depth. The most significant predictor of a nodule's malignancy, imaged at one point in time, is its length, whether expressed as its lengthy axis, a mean of the long and brief axes, or as a volume, according to the chance version literature. The reason for this is simple: benign nodules form as a result of methods that are self-limiting in length, such as infected lymph nodes, whereas malignant tumours do not, and are limited by a variety of factors such as the length of growth, the molecular replication time, the tumour's ability to invade adjoining structures, and its vascular and oxygen supply. As a result, it's possible that nodule length will be covered as part of any nodule MATLAB device, either implicitly or explicitly.

However, due of choice bias in data collection, there may be further variances within the length distribution of benign and malignant nodules. For instance, a naive approach to gathering instances of malignant lesions would most likely be to pick all retrospective CTs for patients diagnosed with lung cancer and all retrospective CTs for patients diagnosed with benign nodules. However, outside of a screening programme, the majority of patients diagnosed with lung cancer present with symptoms prior to diagnostic imaging and are therefore at an advanced stage, with nodules that are larger than benign nodules. When applied to, for example, a screening utility in which the distribution of malignant nodule sizes is higher than benign nodule sizes, a system mastering a set of rules taught on such information may perform extremely badly. We looked into this issue in a similar approach, testing the overall performance of a MATLAB device trained on length-matched and length-unrivalled data.

The first (A), which included 640 strong nodules, was built to eliminate length as a distinguishing factor between benign and malignant strong nodules; all malignant

strong nodules with a diameter of four to twenty millimetres were chosen, and for each, a benign strong nodule with a diameter that most closely matched it in diameter was chosen. Any malignant nodule for which no equivalently sized benign nodule could be found within 0. Eight mm was ruled out. Automatic volumetric segmentation was used to measure the sizes. The second dataset (B), which also included 640 participants, included all malignant nodules in A, but benign nodules were chosen at random based on the empirical length distribution of the entire NLST dataset.

4.3 CONVOLUTIONAL NEURAL NETWORKS

4.3.1 Overview

The usage of Deep Learning with Computer Vision is all about Convolutional Neural Networks. Reflecting on a Neural Network Architecture and the way it's miles carried out to observable duties, like as images and video, is a great method to gain intuition about that. Convolutional Neural Networks have been used to develop facial recognition, object recognition

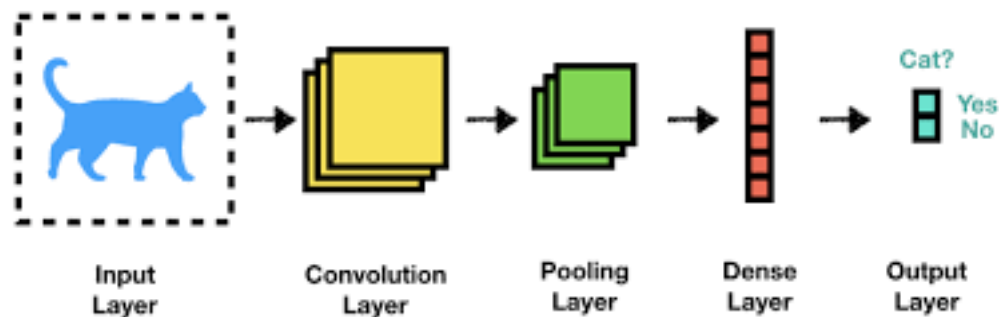


Fig: 4.4 : Convolutional Neural Networks Examples

4.3.2 Architecture

A conventional Convolutional Neural Network, like a Neural Network, has several hidden layers termed as Convolutional Layers, each of which computes strided convolutions over an image to extract capabilities. It also has a pooling layer that computes all other characteristics, such as Max Pool or Average Pool, to reduce the size of the image within the neuron and speed up the computation. It accomplishes this by extracting the capabilities of the neuron image and disregarding the rest, resulting in a more resilient network. There is also an entirely connected layer, which is a type of hidden layer in a neural network in which the total of each layer's outputs is flattened and every cost is an input to the next layer

seen via an activation characteristic and an output [3]. See Figure 4.7 for further information.

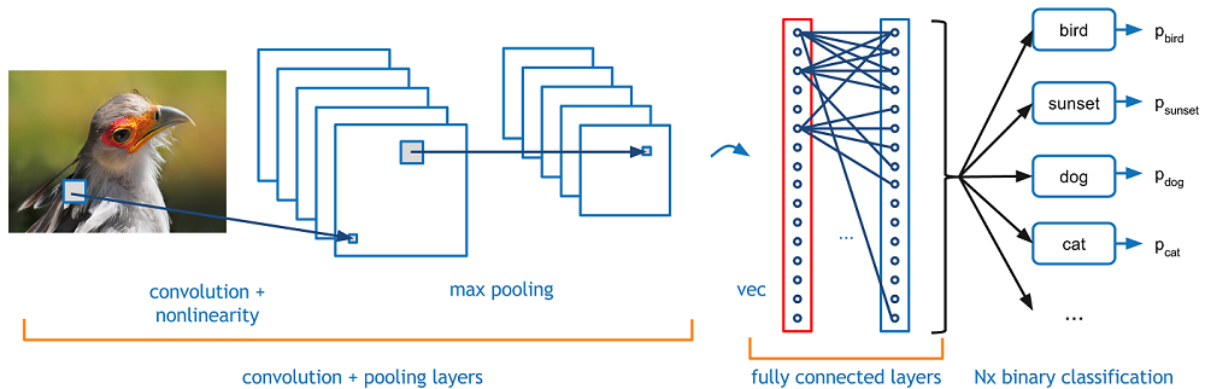


Fig:4.5 : A Convolutional Neural Network Architecture Example

4.3.3 Convolutional Layer

A convolutional layer is a linear characteristic that is utilised in a Convolutional Neural Network. Through the use of image processing characteristic detectors, each node inside the buried layer extracts unique capabilities. For example, inside the first layer, the main node may also extract the horizontal edges of an image, the second node may also extract vertical edges, and so on. A kernel is used to extract these capabilities. Figure 3.11 shows how strided convolutions appear on an image. The backside is a one-of-a-kind snapshot, and the summit is the convolutions' product. It's also worth mentioning that the convolutions' output reduces the size of the original shot.

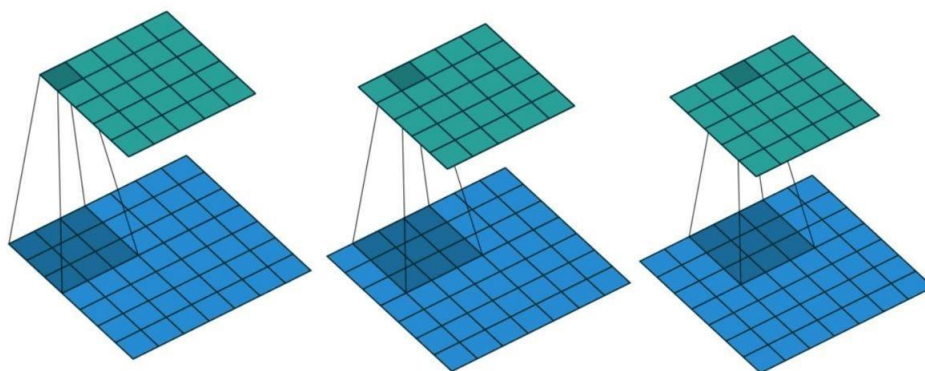


Fig:4.6: Stride Convolution Example

4.3.4 Pooling Layer

The convolutional layer is usually calculated first, followed by the pooling layer. Pooling is done in order to reduce the size of the convolutional layer and simply extract the capabilities in order to make the model more resilient. There are two types of pooling completed: maximum pooling and standard pooling. Common pooling estimates the common pixel cost that has to be retrieved, whereas max pooling derives the best pixel cost from a feature. The third diagram. The max and common pooling operations are shown below.

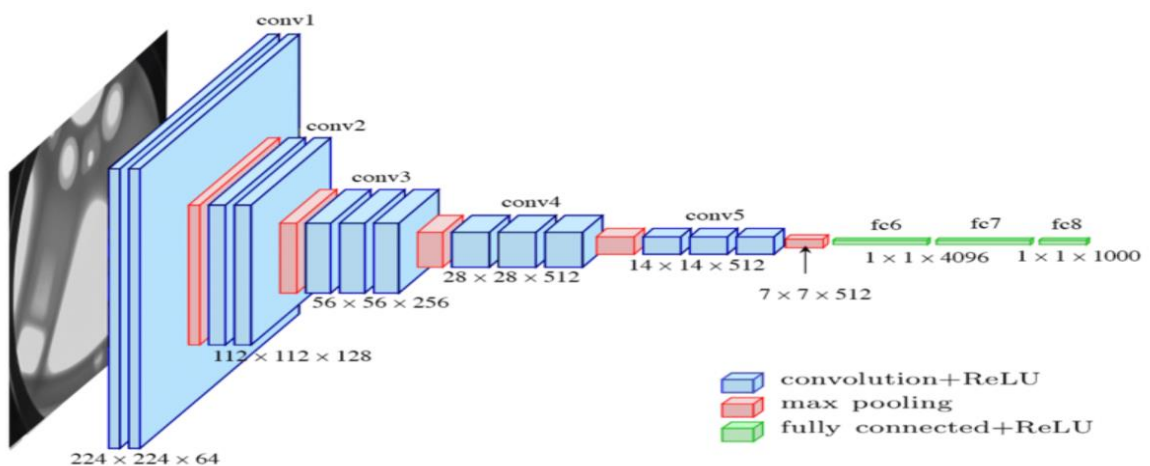
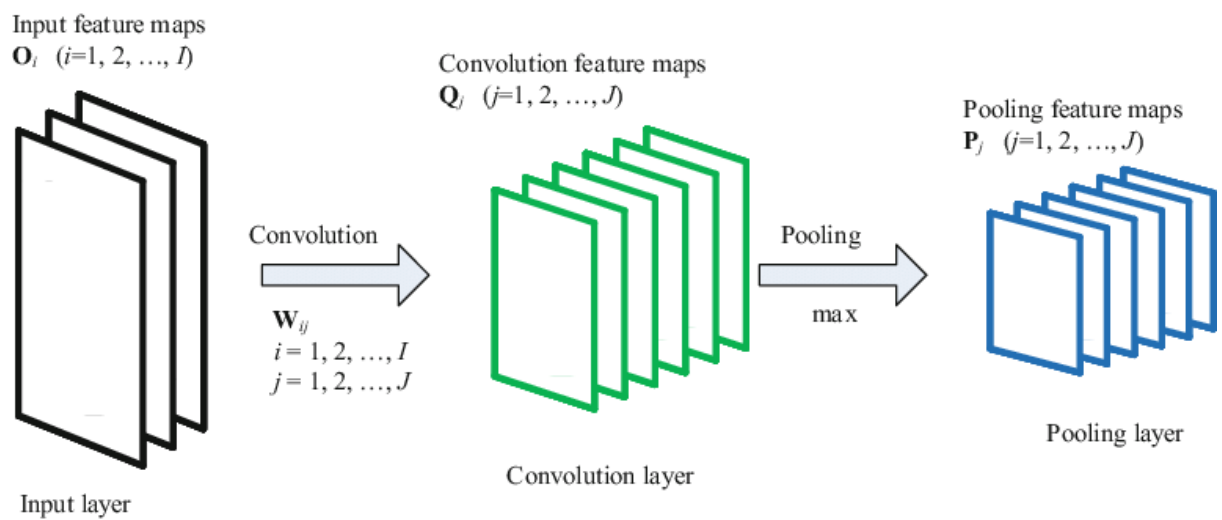


Fig:4.7 : Max and Average Pooling Operation

4.4 GOOGLNET

GoogLeNet is a 22-layer deep convolutional neural network developed by Google researchers as a variation of the Inception Network, a Deep Convolutional Neural Network.

The GoogLeNet architecture presented at the ImageNet Large-Scale Visual Recognition Challenge 2014(ILSVRC14) addressed computer vision tasks such as picture classification and object detection; see the conclusion part of this article for details on how well it did. GoogLeNet is now utilised for a variety of computer vision tasks, including face detection and identification, adversarial training, and so on. The GoogLeNet architecture was established with the goal of being a powerhouse with more computational efficiency than some of its predecessors or similar networks at the time. One of the ways the GoogLeNet achieves efficiency is by reducing the input image while maintaining key spatial information.

The dropout strategy reduces the number of linking neurons in a neural network at random. Each neuron has a risk of being left out, or rather, dropped out of the collected contribution from related neurons, at each training stage. The linear layer is made up of 1000 hidden units, which correspond to the 1000 different classes in the Imagenet dataset. The softmax layer is the last layer, and it employs the softmax function, which is an activation function for calculating the probability distribution of a collection of integers within an input vector.

A softmax activation function produces a vector whose collection of values indicates the likelihood of a class or event occurring. All of the values in the vector add up to one.

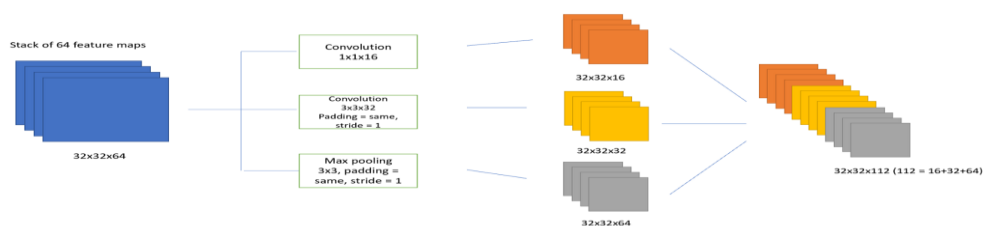


Fig: 4.8 : Convolution and pooling layer of GoogLeNet

4.4.1 Auxiliary Classifiers (Auxiliary Classifiers)

Before we wrap up our look at the GoogLeNet design, there's one more feature that the network's architects included to make it more regular and avoid overfitting. An Auxiliary Classifier is the name for this extra component. One of the key issues with a large network is that it suffers from diminishing gradient descent. Because of the modest gradient value, the update to the weights resulting from backpropagation is negligible inside the lowest layers, resulting in vanishing gradient descent. Simply put, during training, the network stops learning. Auxiliary Classifiers are introduced to the architecture's intermediate layers, especially the third (Inception 4[a]) and sixth (Inception 4[b]) (Inception4[d]). Auxiliary Classifiers are used exclusively during training and are not used during inference. An auxiliary classifier's job is to execute classification based on the inputs inside the network's middle and contribute the loss estimated during training to the network's overall loss.

Auxiliary classifier losses are weighted, which means the computed loss is compounded by 0.3.

The construction of an auxiliary classifier is seen below.

An auxiliary classifier is made up of an average pool layer, a conv layer, two fully connected layers, a 70% dropout layer, and a linear layer with a softmax activation function. The activations from the preceding inception modules are sent into each of the included auxiliary classifiers. The GoogLeNet architecture is depicted in the figure below, replete with all of its bells and whistles. Without any external data training, the GoogLeNet architecture scored well on the classification tasks in the ILSVRC 2014 classification challenge, coming in first with a top 5 error of 6.67 percent. Understanding the GoogLeNet architecture is essential for any deep learning practitioner interested in understanding about the evolution of deep conv networks in the field.

Revisiting research efforts from several years ago can yield a wealth of information. Most popular machine learning frameworks, such as TensorFlow, Keras, and PyTorch, now provide GoogLeNet as a standard architecture. You may also utilize a GoogLeNet network trained on imagenet using transfer learning instead of constructing or training the network yourself.

4.4.2 Characteristics and Features of GoogLeNet Configuration

The input layer of the GoogLeNet architecture takes in an image of the dimension 244 x 224

Type: This refers to the name of the current layer of the component within the architecture

Patch Size: Refers to the size of the sweeping window utilised across conv and pooling layers. Sweeping windows have equal height and width.

Stride: Defines the amount of shift the filter/sliding window takes over the input image.

Output Size: The resulting output dimensions(height, width, number of feature maps) of the current architecture component after the input is passed through the layer.

Depth: Refer to the number of levels/layers within an architecture component.

#1x1 #3x3 #5x5: Refers to the various convolutions filters used within the inception module.

#3X3 reduce #5x5 reduce: Refers to the numbers of 1x1 filters used before the convolutions.

Pool Proj: This is the number of 1x1 filters used after pooling within an inception module.

Params: Refers to the number of weights within the current architecture component.

Ops: Refers to the number of mathematical operations carried out within the component.

4.5 VGG16

ImageNet, a massive visual database project utilised in visual object identification software research, uses VGG16, a basic and widely used Convolutional Neural Network (CNN) Architecture. Karen Simonyan and Andrew Zisserman of the University of Oxford created and launched the VGG16 Architecture in their essay "Very Deep Convolutional Networks for Large-Scale Image Recognition" in 2014. The term 'VGG' stands for Visual Geometry Group, a group of scholars at the University of Oxford who designed this architecture, and the number '16' indicates that there are 16 levels in this architecture

16 layers of VGG16

- 1.Convolution using 64 filters
- 2.Convolution using 64 filters + Max pooling
- 3.Convolution using 128 filters
4. Convolution using 128 filters + Max pooling
5. Convolution using 256 filters
6. Convolution using 256 filters
7. Convolution using 256 filters + Max pooling
8. Convolution using 512 filters
9. Convolution using 512 filters
10. Convolution using 512 filters+Max pooling
11. Convolution using 512 filters
12. Convolution using 512 filters
13. Convolution using 512 filters+Max pooling
14. Fully connected with 4096 nodes
15. Fully connected with 4096 nodes
16. Output layer with SoftMax activation with 1000 nodes.

In ImageNet, a dataset of over 14 million pictures belonging to 1000 classes, the VGG16 model obtained 92.7 percent top-5 test accuracy. In the year 2014, it was one of the well-known models submitted to the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). It outperformed AlexNet by substituting huge kernel-sized filters (11 and 5 in the first and second convolutional layers, respectively) with numerous three-three kernel-sized filters one after the other. NVIDIA Titan Black GPUs were used to train VGG16 for weeks. Many deep learning image classification approaches employ VGG16, which is popular owing to its ease of use. Because of its advantages, VGG16 is often utilised in learning applications. The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2014 was won by VGG16, a CNN architecture. It is still considered to be one of the greatest vision architectures ever created. The convnets are fed a fixed-size 224 by 224 RGB picture during training. The only pre-processing done here is subtracting the mean RGB value derived on the training set from each pixel. The picture is processed through a stack of convolutional (conv.) layers, where filters with a very narrow receptive field are utilised, such as 3 3 (which is the lowest size

to capture the notions of left/right, up/down, and centre and has the same effective receptive field as one 7×7). It's more complex, with more non-linearities and fewer parameters. 1 1 convolution filters, which may be thought of as a linear modification of the input channels (followed by non-linearity), are also used in one of the configurations. For 3×3 convolutional layers, the convolution stride and spatial padding of the conv. layer input are both set to 1 pixel, ensuring that the spatial resolution is kept after convolution. Spatial pooling is aided by five max-pooling layers that follow parts of the convolutional layers. Max-pooling is done with stride 2 across a 22 pixel frame.

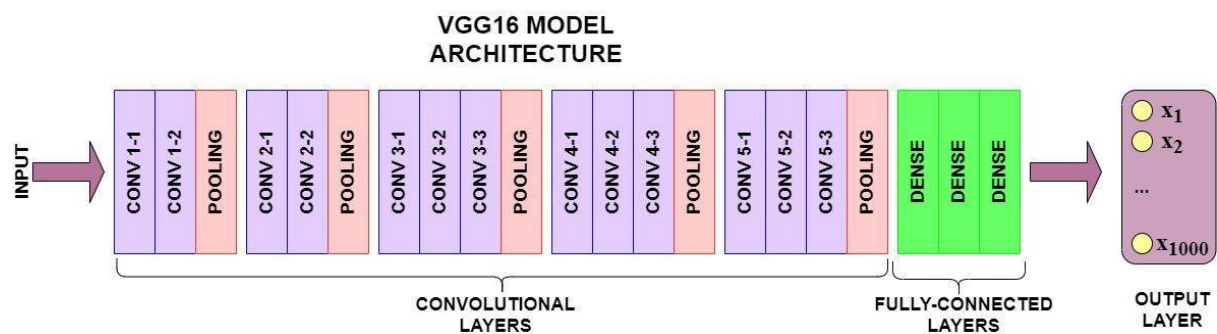


Fig:4.9 : Layers of Vgg16

Following a stack of convolutional layers (which have varying depths in different designs), there are three Fully-Connected (FC) layers: the first two have 4096 channels apiece, while the third performs 1000-way ILSVRC classification and so comprises 1000 channels (one for each class). The soft-max layer is the last layer. In all networks, the completely linked levels are configured in the same way.

As previously indicated, the 16 layer VGG design performed best, with a top-5 error rate of 7.3 percent (92.7 percent accuracy) in the ILSVRC — 2014. VGG16 surpassed the previous generation of models in the ILSVRC 2012 and ILSVRC 2013 contests by a wide margin.

The construction of the VGG16 is represented below:

The ConvNet training technique is often carried out by employing mini-batch gradient descent (based on back-propagation) with momentum to optimise the multinomial logistic regression goal. The batch size and momentum were both set to 256 and 0.9, respectively. For the first two fully-connected layers, the training was

regularized via weight decay (the L2 penalty multiplier was set to $5 \cdot 10^4$) and dropout

regularization (dropout ratio set to 0.5). When the validation set accuracy stopped improving, the learning rate was reduced by a factor of 10 and the learning rate was

set to 10^{-2} . The learning rate was reduced three times in total, and after 370,000 iterations, the learning was terminated (74 epochs). Because of (a) implicit regularization enforced by the increased depth and lower conv. filter sizes; and (b) pre-initialization of some layers, despite the larger number of parameters and greater depth of convolutional networks, the nets required fewer epochs to converge. The initialization of the network weights is critical, as poor initialization might cause learning to halt owing to the gradient's instability in deep nets. To get around this, we started training configuration which is shallow enough to be learned with random initialization. The first four convolutional layers and the last three fullyconnected layers with the layers of net A (the intermediate layers were randomly started) are then initialised for training further architectures.

4.5.1 Training image size

Let S be the smallest side of an isotropically rescaled training picture used to crop the ConvNet input (we also refer to S as the training scale). While the crop size is fixed at 224×224 , S can theoretically be any value less than 224: The crop will collect whole-image statistics, totally spanning the smallest side of a training picture, for $S = 224$; for $S < 224$, the crop will correspond to a small section of the image, including a small item or an object component.

For determining the training scale S , two techniques are examined. Fixing S , which stands for single-scale training, is the initial step (note that image content within the sampled crops can still represent multi-scale image statistics). The evaluation of models is trained at two fixed scales: $S = 256$ (which has been widely used in the prior art (Krizhevsky et al., 2012; Zeiler & Fergus, 2013; Sermanet et al., 2014)) and $S = 384$ (which has been widely used in the prior art (Krizhevsky et al., 2012; Zeiler & Fergus, 2013; Sermanet et al. The network was originally trained with $S = 256$ after being given a ConvNet setup. To speed up the training of the $S = 384$ network, the weights were pre-trained with $S = 256$, and the initial learning rate was reduced to 103%.

Multi-scale training is the second method for determining S , in which each training picture is rescaled independently by randomly picking S from a given range [S_{min} , S_{max}] ($S_{min} = 256$ and $S_{max} = 512$ was utilised). Because objects in photos might be of various sizes, it is useful to consider this during training. This is also known as scale jittering, which is when a single model is taught to detect objects across a wide variety of sizes. Multi-scale models were trained by fine-tuning all layers of a single-scale model with the same setup, pre-trained with fixed $S = 384$, for speed considerations. Using a pre-trained convnet, A pre-trained network may be used in two ways: feature extraction (with and without data augmentation) and fine-tuning, both of which are addressed in depth in the following sections.

Keras is pre-installed on the VGG16 model. The `keras.applications` module may be used to import it. VGG16 is unusual in that it focuses on having 3×3 filter convolution layers with a stride 1 and always utilises the same padding and maxpool layer of 2×2 filter stride 2 instead of having a huge number of hyper-parameters. Throughout the design, the convolution and max pool layers are arranged in the same way. It has two fully connected (FC) layers at the end, followed by a softmax for output.

Creating the VGG16 model:

To create the VGG16 convolutional base, do the following:

```
import VGG16 conv base = VGG16(weights = 'imagenet',  
include_top = False,  
input_shape = (150, 150, 3)) from keras.applications
```

The function `Object() { [native code] }` receives three arguments:

The weight checkpoint from which the model is initialised is specified by `weights`. `include_top` specifies whether the densely connected classifier should be included (or not) on top of the network.

The densely connected classifier corresponds to the 1,000 ImageNet classes by default, however it is not necessary to add it because it is meant to utilise a native densely connected classifier (with two classes: cat and dog). The shape of the image tensors given to the network is `input_shape`. This is an optional parameter that, if left blank, allows the network to process inputs of any size.

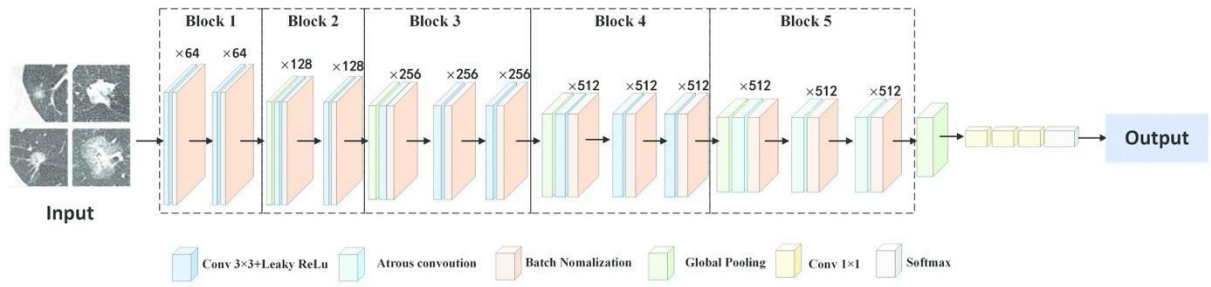


Fig: 4.10 :Training the Data

We have two options for moving forward from here.

The first is to run the convolutional base over the dataset, save the result to a Numpy array on disc, and then feed this data to a standalone, densely connected classifier. This technique is quick and inexpensive to implement since it only has to run the convolutional base once for each input picture, despite the fact that the convolutional base is the most expensive portion of the pipeline. However, this approach does not allow for data augmentation for the same reason.

Extending the present model (conv base) by layering Dense layers on top and running the entire thing on the input data from start to finish. Because every input picture passes through the convolutional base every time it is viewed by the network, data augmentation will be possible. However, as a result of this, this strategy is far more expensive than the first, and it contradicts what is said there.

Both strategies are described in detail below. Examining the code needed to build up the first one: recording conv base's output on your data and utilising those outputs as inputs to a new model.

4.5.2 Confusion matrix

		Predicted	
		Negative (N) -	Positive (P) +
Actual	Negative -	True Negative (TN)	False Positive (FP) Type I Error
	Positive +	False Negative (FN) Type II Error	True Positive (TP)

Fig: 4.11 : Confusion Matrix

binary classification problem has only two classes to classify, preferably a positive and a negative class. Now let's look at the metrics of the Confusion Matrix.

True Positive (TP): It refers to the number of predictions where the classifier correctly predicts the positive class as positive.

True Negative (TN): It refers to the number of predictions where the classifier correctly predicts the negative class as negative.

False Positive (FP): It refers to the number of predictions where the classifier incorrectly predicts the negative class as positive.

False Negative (FN): It refers to the number of predictions where the classifier incorrectly predicts the positive class as negative.

It's always better to use confusion matrix as your evaluation criteria for your machine learning model. It gives you a very simple, yet efficient performance measures for your model. Here are some of the most common performance measures you can use from the confusion matrix.

The rows correspond to the anticipated class (Output Class) and the columns to the true class on the confusion matrix diagram (Target Class). The diagonal cells relate to appropriately categorized observations. The off-diagonal cells relate to observations that were erroneously categorized. Each cell displays the number of observations as well as the percentage of the total number of observations.

The percentages of all the instances expected to belong to each class that are properly and wrongly categorized are shown in the column on the far right of the figure. The accuracy (or positive predictive value) and false discovery rate are two

often used measurements. The percentages of all the instances belonging to each class that are successfully and wrongly categorized are shown in the row at the bottom of the figure. The recall (or true positive rate) and false negative rate are two often used measures. The total accuracy is shown in the cell at the bottom right of the plot.

As we've seen, the confusion matrix provides four distinct and distinct metrics. Other metrics may be computed based on these four measures to provide more information about how the model behaves:

1. Accuracy
2. Precision
3. Recall

The percentage of correct predictions for the test data is known as **accuracy**. It's simple to figure out simply dividing the number of right guesses by the total number of forecasts.

$$\text{Accuracy} = \frac{\text{correct predictions}}{\text{all predictions}}$$

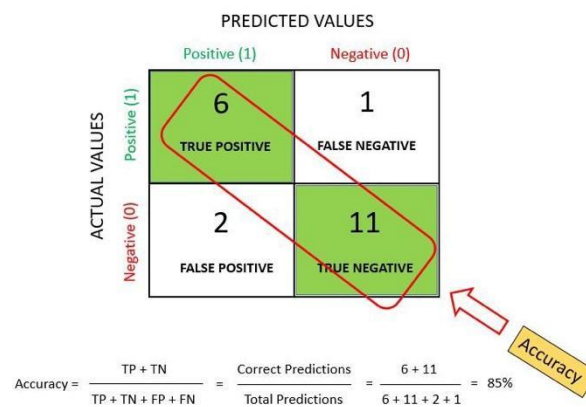


Fig:4.12 :Accuracy Calculation with Example

Precision is defined as the percentage of relevant instances (true positives) among

all the examples expected to belong to a certain class.

$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

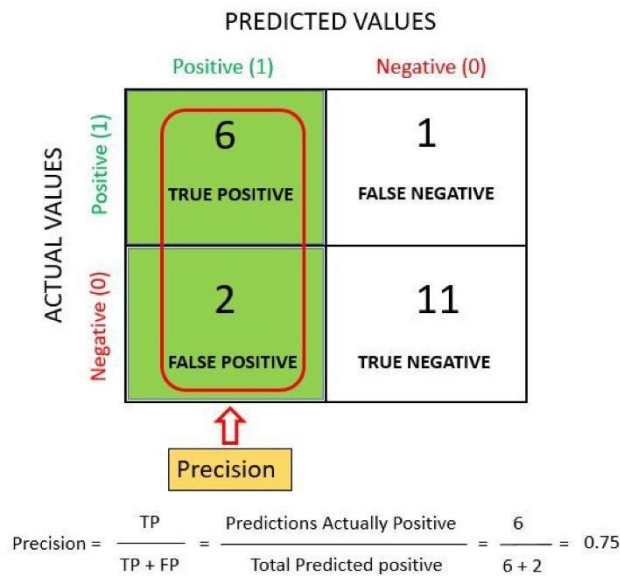


Fig:4.13 :Precision Calculation with Example

The proportion of instances predicted to belong to a class compared to all of the examples that genuinely belong in the class is known as **Recall**.

$$\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

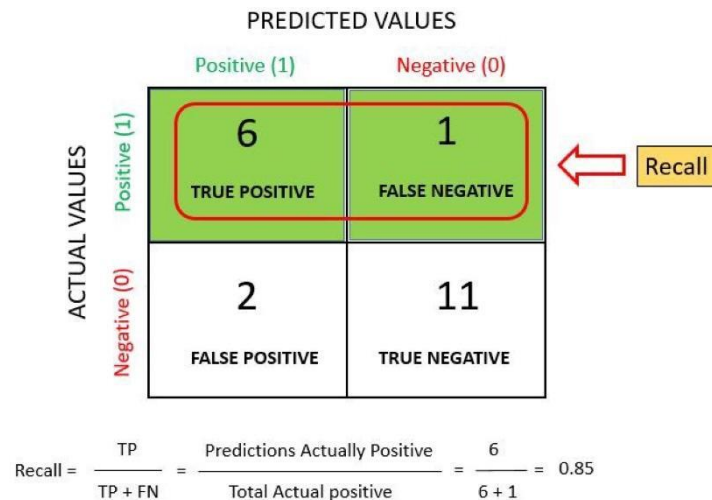


Fig:4.14 : Recall Calculation with Example

F1 score - F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as easy to understand as accuracy, but F1 is usually more useful than accuracy, especially if you have an uneven class distribution. Accuracy works best if false positives and false negatives have similar cost. If the cost of false positives and false negatives are very different, it's better to look at both Precision and Recall.

The F1 score is a number between 0 and 1 and is the harmonic mean of precision and recall. We use harmonic mean because it is not sensitive to extremely large values, unlike simple averages.

F1 score sort of maintains a balance between the precision and recall for your classifier. If your precision is low, the F1 is low and if the recall is low again your F1 score is low.

$$F1 = 2 \times \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.6 BLOCK DIAGRAM OF CNN

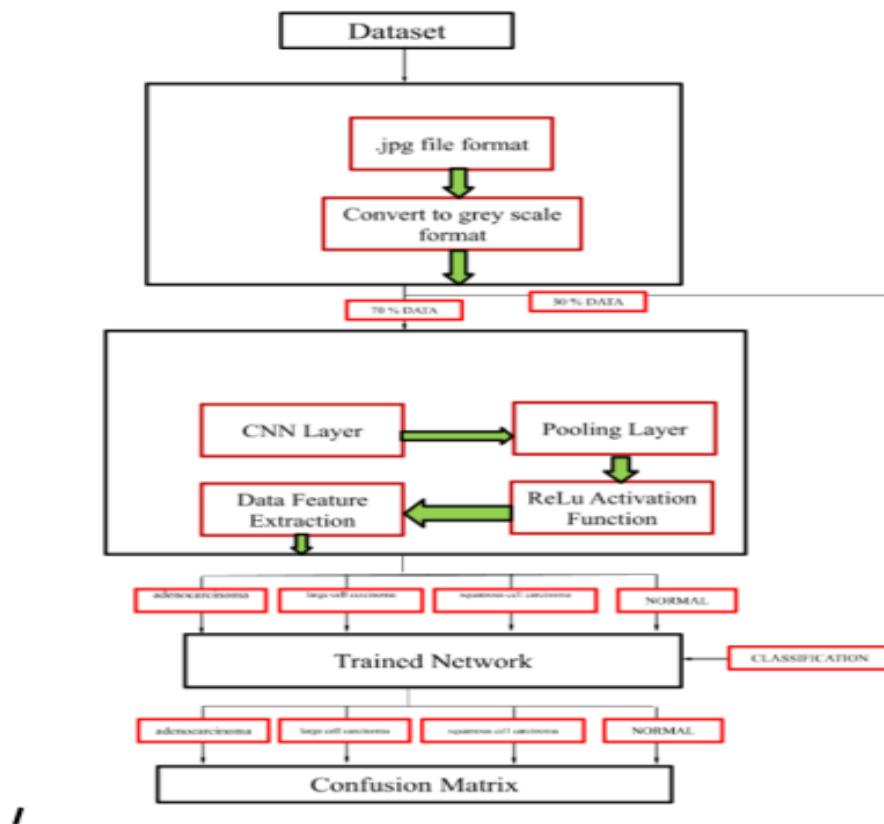


Fig:4.15:Block diagram of CNN

4.6.1 Preprocessing

To begin editing, follow the installation instructions and then run the software on your computer. To open the file finder window, click the Open button in the main toolbar. To bring the JPG image into PhotoPad to convert to black and white, find and select it on your computer, then click Open (Grayscale). Alternatively, you may drag & drop your JPG file into PhotoPad.

To convert your JPG image to grayscale automatically, go to the Color tab in the main toolbar, click the Filters icon, and pick Grayscale from the drop-down menu. You may also pick from a variety of Grayscale filters under the Color Filter area on the right-hand side, under the Effects and Layers column, including Red, Green, and Blue Filters. Smokestack, Panda, and a few more. The Strength slider may also be used to vary the grayscale filter's strength.

To save your grayscale image, click the Save button on the main toolbar. Your grayscale picture can be saved in JPG/JPEG, GIF, ICO, PNG, PSD, or TIF/TIFF format.

4.6.1 Classification

Classification with deep learning in medical image analysis commonly uses target lesions displayed in medical pictures, and these lesions are categorised into two or more classes.

Deep learning, for example, is commonly used to classify lung nodules on computed tomography (CT) scans as benign or malignant. For efficient classification using CNN, it is important to compile a large quantity of training data with associated labels, as illustrated.

CT scans of the lungs and their designations (adenocarcinoma, squamous cell carcinoma, big cell carcinoma, and normal lung) are utilised as training data for lung malignancy classification. 70% of the data is utilised to train the network, whereas 30% is used to classify the data

4.6.2 Training the network

CNNs are comparable to artificial neural networks, with the exception that the inputs are assumed to be pictures. We can embed some attributes into the CNN architecture thanks to this assumption. A typical CNN design is made up of numerous layers that allow it to learn hierarchical feature representations of images. The presence or absence of edges at specific orientations and positions in the picture is often represented by features in the first layer of representation. The second layer recognizes themes by recognizing edge configurations, regardless of minor differences in edge placements. The third layer may group motifs into bigger groups that represent elements of recognised things, and succeeding layers can recognise items as combinations of these parts.

CNNs are comparable to artificial neural networks, with the exception that the inputs are assumed to be pictures. We can embed some attributes into the CNN architecture thanks to this assumption. A typical CNN design is made up of numerous layers that allow it to learn hierarchical feature representations of images. The presence or absence of edges at specific orientations and positions in the picture is often represented by features in the first layer of representation. The second layer recognises themes by recognising edge configurations,

regardless of minor differences in edge placements. The third layer may group motifs into bigger groups that represent elements of recognised things, and succeeding layers can recognise items as combinations of these parts.

A typical image categorization convolutional neural network (CNN) design. The CNN architecture is made up of a series of layers that convert the picture volume into output class scores. Every layer uses a differentiable function to convert one volume of activations to another. Convolution (Conv), pooling (Pool), nonlinearity (rectified linear unit [ReLU]), and fully connected (FC) layers are the four types of layers that make up a CNN.

A CNN's essential building piece is the convolution (CONV) layer. A series of learnable filters make up the CONV layer's parameters. Each filter might be tiny in size but covers the whole depth of the input volume. Each filter is convolved over the width and height of the input volume during the forward pass, computing dot products between the filter's entries and the input at any location. As the filter moves through the input volume, a two-dimensional map is created, displaying the filter's responses at every spatial point. Each filter produces a distinct map, and each CONV layer comprises a collection of filters. A stack of these maps along the depth dimension makes up the output volume. Every item in the output volume may thus be read as the output of a neuron that examines a small part of the input and shares parameters with other neurons in the same activation map.

From input to output, a convolution is depicted. The use of weight sharing in convolutional neural networks (CNNs) distinguishes them from standard artificial neural networks. Images are enormous pixel matrices, with each pixel representing one input. It would be computationally intensive to use distinct weights for each pixel. The presence of recurring patterns is another way to characterise visuals. This is a trait that CNNs make advantage of. Each layer of the network is made up of tiny weight matrices, commonly known as filters. Each filter is gradually moved along the picture, resulting in a dot product between the pixels and the weights in each region. A neuron per pixel in the picture is created by the dot product of each filter and each unique area.

The spatial dimension is frequently down-sampled by the pooling layer. Between consecutive CONV layers, a pooling layer is frequently inserted. The pooling layer's job is to gradually shrink the spatial size of the representation in order to reduce

the number of parameters and computations in the network while also preventing overfitting.

The pooling layer works independently on each depth slice of the input and spatially resizes it. The MAX pooling function, which utilises the maximum value from each cluster of neurons from the previous layer to produce a new neuron in the following layer, is the most popular pooling function. Other functions, such as average pooling, are also useful. The nonlinearity layer uses a unique activation function to apply element-wise nonlinearity . As mentioned, the most common activation function is the ReLU function, which is simply thresholded at zero.

4.7 BLOCK DIAGRAM OF GoogLeNet



Fig:4.16 :Block diagram of GoogLeNet

4.7.1 Preprocessing

Follow the installation instructions, but to begin editing, simply run the software on your computer.

To launch the file finder window, click the Play button on the main toolbar. On your computer, locate and select the JPG image you wish to convert to black and white, then click Open (Grayscale). You may also use PhotoPad to drag and drop your JPG file.

To convert your JPG photo to grayscale automatically, go to the Color tab on the main toolbar, click the Filters icon, and pick Grayscale from the drop-down menu. Under the Color Filter section on the right-hand side, under the Effects and Layers column, you may also choose from a variety of Grayscale filters, such as Red, Green, and Blue Filters. Smokestack, Panda.

4.7.2 Classification

In medical image analysis, target lesions are frequently utilized in deep learning classification, and these lesions are sorted into two or more classes.

For example, deep learning is commonly used to classify lung lesions on computed tomography (CT) scans as benign or malignant. For efficient classification using Vgg16, it is important to create a large amount of training data with labels, as illustrated.

CT scans of the lungs and their classifications (adenocarcinoma, squamous cell carcinoma, big cell carcinoma, and normal lung) are utilised as training data for lung malignancy classification. The data is utilised to train the network 70% of the time, with the remaining 30% being used to categorise it.

4.7.3 Training the network

Classified 70% of the data is now being trained. For feature extraction, grayscale photos will be evaluated in Google Net. There are a lot of layers in GoogLeNet. In the training procedure, a classifier called the loss 3 classifier is utilised. The picture input layer is the first element of the network's Layers attribute. This layer requires 224-by-224-by-3 input pictures for a GoogLeNet network, wherein 3 is the number of colour channels. Other networks may require images of varying sizes as input. The coefficient of determination is a metric for determining how much variability in one component can be attributed to its connection with another. This relationship is referred to as the goodness of fit. The photos are categorised into the featured data when the fit coefficient is established. The photographs are categorised and forwarded on to the next step of the process after they have all of the necessary information.

4.8 BLOCK DIAGRAM OF Vgg16

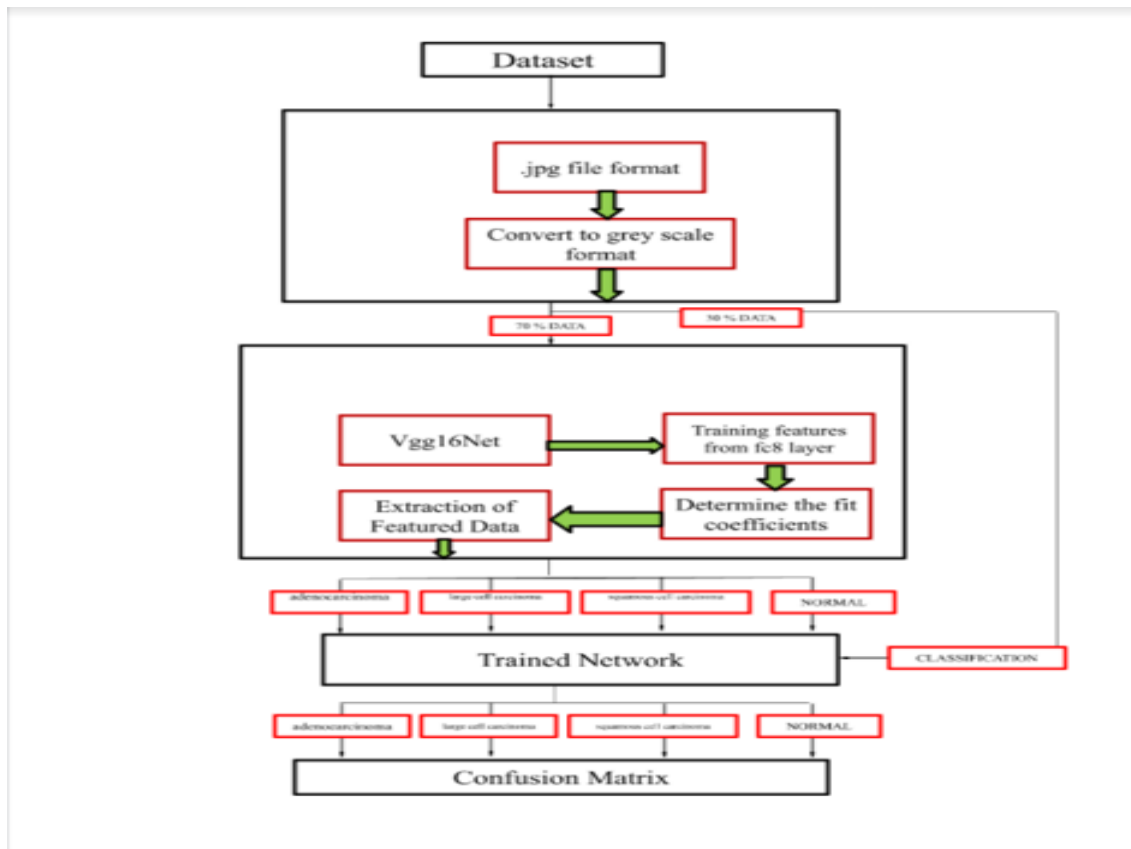


Fig:4.17 :Block diagram of Vgg16

4.8.1 Preprocessing

Follow the installation instructions but then just start the programme on your computer to begin editing.

Click the Launch button in the main toolbar to open the file finder window. Find and pick the JPG image you want to convert to black and white on your computer, then click Open (Grayscale). You may also drag and drop your JPG file into PhotoPad. Go to the Color tab on the main toolbar, click the Filters icon, and select Grayscale from the drop-down menu to convert your JPG picture to grayscale automatically. You may also choose from a selection of Grayscale filters, such as Red, Green, and Blue Filters, under the Color Filter section on the right-hand side, under the Effects and Layers column.

Panda, Smokestack. The grayscale filter's strength may also be adjusted using the Strength slider.

Click the Save button on the main toolbar to save your grayscale image. You may save your grayscale image as a JPG/JPEG, GIF, ICO, PNG, PSD, or TIF/TIFF file.

4.8.2 Classification

Target lesions exhibited in medical images are often used in deep learning classification in medical image analysis, and these lesions are classified into two or more classes.

Deep learning is frequently used to categorise lung nodules on computed tomography (CT) images as benign or cancerous, for example. It is critical to generate a substantial quantity of training data with accompanying labels for efficient classification using Vgg16, as shown.

As training data for lung malignancy classification, CT scans of the lungs and their classifications (adenocarcinoma, squamous cell carcinoma, large cell carcinoma, and normal lung) are used. 70% of the data is utilised to train the network, with the remaining 30% being used to categorise the data.

4.8.3 Training the network

Classified Currently, 70% of the data is being trained. Grayscale photographs will be examined in Google Net for feature extraction. In Vgg16, there are a lot of layers. A layer termed the fc8 layer is used in the training process. The picture input layer is the first element in the Layers attribute of the network. Other networks may demand different-sized photos as input. The coefficient of determination is a statistic for measuring how much of one component's variability may be attributable to its relationship with another. The quality of fit is the term used to describe this relationship. When the fit coefficient is determined, the photographs are categorised into the highlighted data. After they have all of the essential information, the images are categorise and moved on to the next phase of the process.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 RESULTS

In this chapter the results for the malignancy classification is described for CNN, GoogleNet and Vgg16 in the form of confusion matrix.

5.1.1 Result for CNN

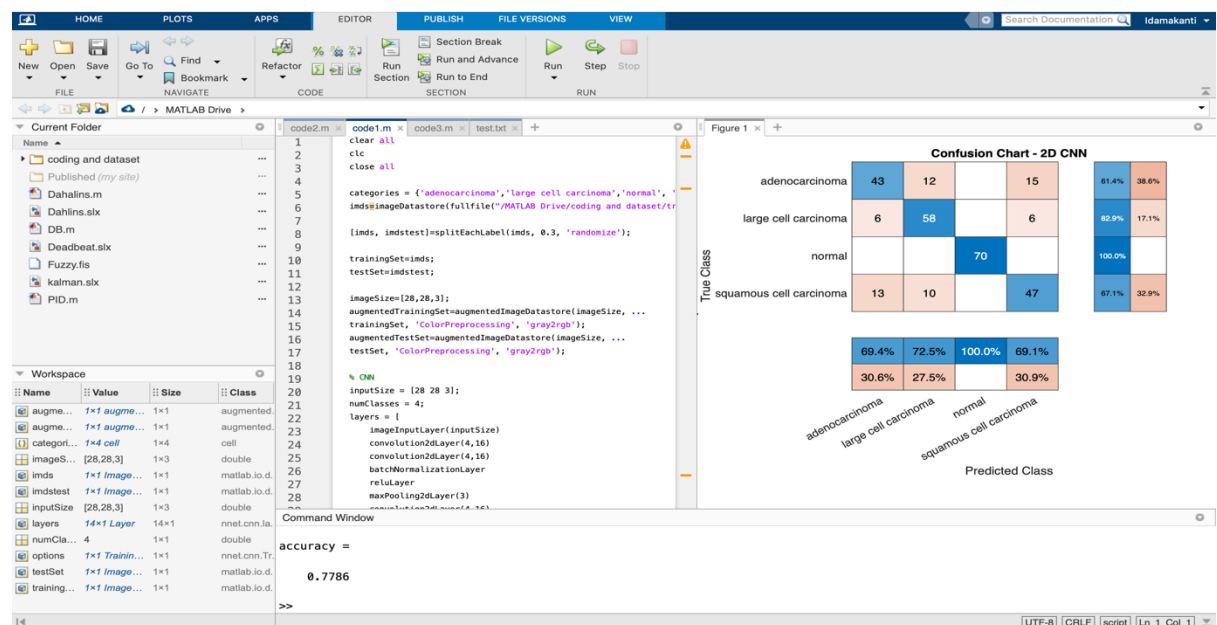


Fig:5.1: Confusion matrix for CNN network

While running 50 greyscale images in CNN network, the required classification is obtained. The output obtained is in the form of confusion matrix. In the confusion matrix the predicted class percentage for adenocarcinoma, squamous cell carcinoma, large cell carcinoma and normal lung is shown. The predicted class percentage is 69.4%, 72.5%, 100%, 69.1% for adenocarcinoma, large cell carcinoma, normal lung and squamous cell carcinoma respectively.

5.1.2 Output

Now the features are extracted based on adenocarcinoma, squamous cell carcinoma, large cell carcinoma and a normal lung. With these classified data, it is now compared with previously preserved 30 percent data.

Finally, it is compared and classified with respective features. Outcome of the project result is expressed in confusion matrix

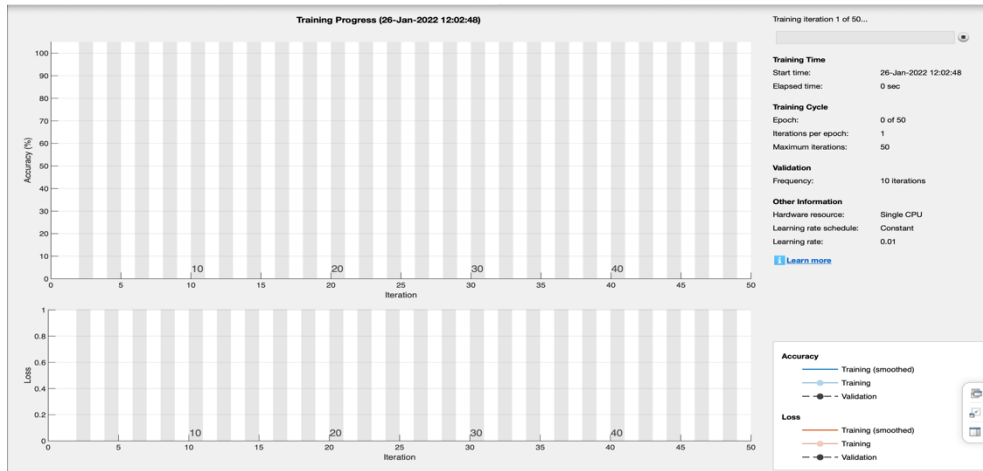


Fig:5.2: Training the Data in Initial Stage

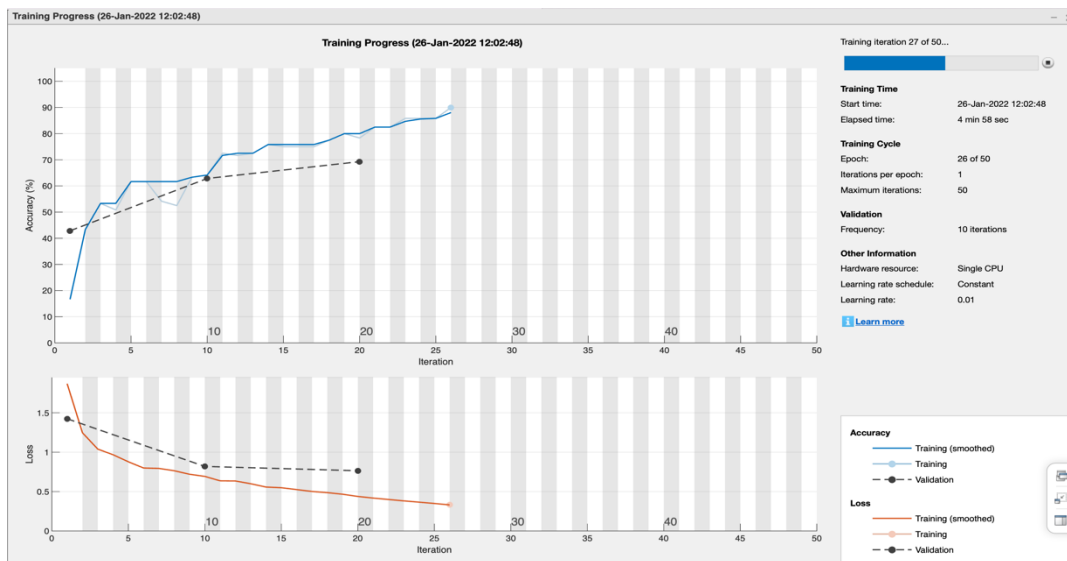


Fig:5.3: Training the Data in Middle Stage

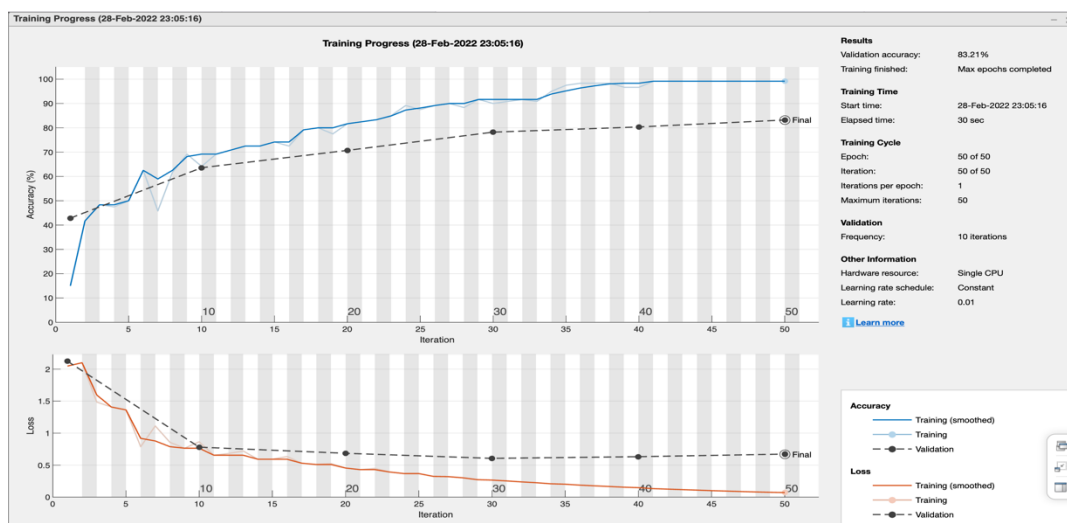


Fig:5.4 : Training the Data in Final Stage

5.1.3 Result for GoogLeNet

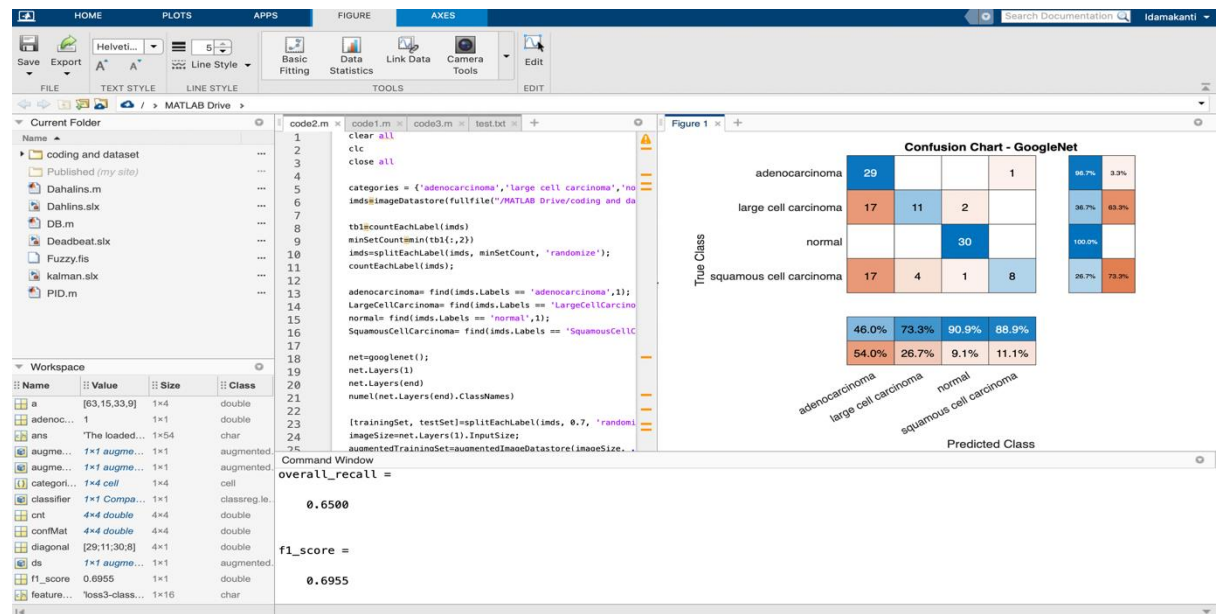


Fig.5.5 :Confusion matrix for GoogLeNet network

While running 50 greyscale images in GoogLeNet network, the required classification is obtained. The output obtained is in the form of confusion matrix. In the confusion matrix the predicted class percentage for adenocarcinoma, squamous cell carcinoma, large cell carcinoma and normal lung is shown. The predicted class percentage is 54.0%, 73.3%, 90%, 88.9% for adenocarcinoma, large cell carcinoma, normal lung and squamous cell carcinoma respectively.

5.1.4 Output

Characteristics are extracted from adenocarcinoma, squamous cell carcinoma, large cell carcinoma, and a healthy lung. It is now compared to the 30% of data that was previously saved utilising these classified data.

Finally, it is compared and classified based on its features.

A confusion matrix is used to depict the project's conclusion

5.1.5 Result for Vgg16

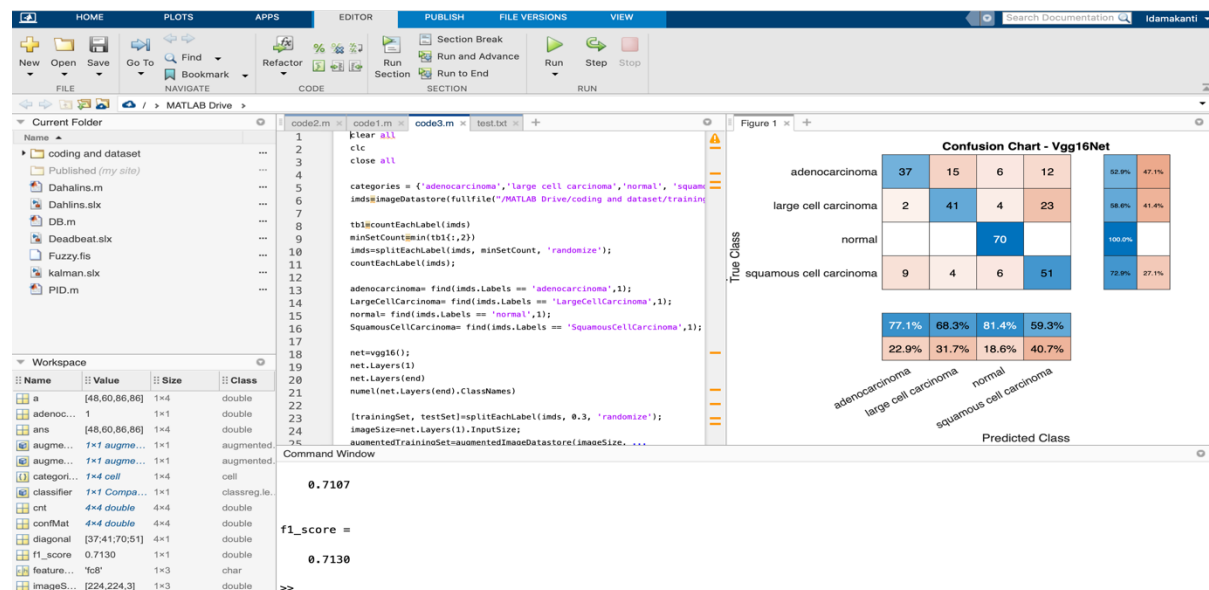


Fig:5.6: Confusion matrix for Vgg16

While running 50 greyscale images in Vgg16 network, the required classification is obtained. The output obtained is in the form of confusion matrix. In the confusion matrix the predicted class percentage for adenocarcinoma, squamous cell carcinoma, large cell carcinoma and normal lung is shown. The predicted class percentage is 77.0%, 68.3%, 81.4%, 59.3% for adenocarcinoma, large cell carcinoma, normal lung and squamous cell carcinoma respectively.

5.1.6 Output

Adenocarcinoma, squamous cell carcinoma, big cell carcinoma, and a normal lung are being used to extract characteristics. It is now checked with previously preserved 30% data using these categorised data.

Finally, it is contrasted and classed according to its characteristics.

The project's outcome is represented through a confusion matrix.

5.2 DISCUSSION

As per the literature survey, many researchers used GLCM and SVM classifier to examine lung CT images to diagnose whether the lung is diseased or not. Later few scientists came up with an idea to predict carcinoma stages of a specific lung cancer. In this project I used CNN, GOOGLNET and VGG16 to predict accuracy level for adenocarcinoma, squamous cell carcinoma, large cell carcinoma and normal lung.

CHAPTER 6

SUMMARY AND CONCLUSIONS

6.1 SUMMARY

It has been universally acknowledged that the problem of lung cancer is escalating at an alarming rate in current scenario. This problem has been a depressing magnitude. Although a plethora of reasons are there yet some sensible solutions are available to curb this grave concern. There is no denying the fact that the use of CNN, Google net and Vgg 16 is ubiquitous across the globe due to its paramount importance. At the certain conditions when large number of CT images are to be processed in a time, the method of deep learning will make the work easier. As the globe is changing into telehealth, it is much essential for a person/doctor to examine the results without the help of an expertise. In this project, I used CNN, Google net and Vgg 16 to classify CT images. The main advantage of this project is that, it is helpful to run a maximum of 300-500 images at a time and classify them accordingly. The second advantage is that to run and classify an expertise is not required. The output represented in the confusion matrix gives the obtained predicted class percentage. Use of MATLAB to run a program is also easier. To recapitulate, unquestionably, the problem is lung malignancy classification has become a burning situation. To conclude without any doubt, the use of these deep learning networks has many positives than the negatives.

6.2 CONCLUSIONS

The method for classifying medical photos is created in this paper. The traits are retrieved from CT pictures of healthy lungs, as well as cancer-affected lungs. Despite the fact that each illness type has its own features and patterns, there are certain commonalities across these categories that make creating a classifier with a suitable decision boundary problematic. As a result, feature selection is a difficult problem that must be solved through a meticulous trial and error procedure. Furthermore, efficient feature selection remains a challenge in medical images, but it may be handled in the future in a more effective way to get better outcomes. The suggested techniques are shown to be suitable for diagnosing cancer illness based on the classification accuracy of the Google Net classifier. In the future, the optimization technique is improved for utilizing the operations in both training and

testing processes. The CNN model is modified to provide better efficiency than this system. It can be improved by including other hyperparameters.

REFERENCES

- 1) Ali, Imdad Efficient Lung Nodule Classification Using Transferable Texture Convolutional Neural Network. IEEE Access, (2020) 8(), 175859–175870. doi:10.1109/ACCESS.2020.3026080
- 2) Asuntha, Srinivasan and Andy Deep learning for lung Cancer detection and classification. Multimedia Tools and Applications, (2020), – . doi:10.1007/s11042-019-08394-3
- 3) Bhandary Deep-Learning Framework to Detect Lung Abnormality – A study with Chest X-Ray and Lung CT Scan Images. Pattern Recognition Letters, (2019), S0167865519303277– . doi:10.1016/j.patrec.2019.11.013
- 4) Chenwei Deng, Weisi Lin & Jianfei Cai, Content-Based Image Compression for Arbitrary-Resolution Display Devices, IEEE Trans on M media,(2012)1127-1139, 4
- 5) Coccia and Mario Deep learning technology for improving cancer care in society: New directions in cancer imaging driven by artificial intelligence. Technology in Society, (2019), 101198–. doi:10.1016/j.techsoc.101198
- 6) Coudray Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. Nature Medicine, (2018), –.doi:10.1038/s41591-018-0177-5
- 7) Davis L, Johns S and Aggarwal J, “Texture analysis using generalized co-occurrence matrices” , IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1(1979) ,pp 251-259
- 8) Gemma paella, Adaptive lifting schemes combining semi norms for lossless image compression, IEEE Int conf on image proc, (2005)1753-756
- 9) Ginneken B V, Romeny B M and Viergever M A , Computer-aided diagnosis in chest radiography survey, IEEE Tran on med image, (2001),1228 – 1241,20
- 10) Han and Guo-Guang Chen, Maximum kurtosis principle for the parameter selection of Gabor wavelet and its application to ultrasonic signal processing , Russ. J. Nondestr. Test , (2009), Volume 45, Number 6, Page 436.

- 11)Haralick R M, Shaunmugam K and Dinstein I, "Textural features for image classification",IEEE Transactions on Systems, Man, and Cybernetics: Systems,(1973) SMC 3,pp 610-621.
- 12)Kanavati Weakly-supervised learning for lung carcinoma classification using deep learning. Scientific Reports,(2020) 10(1), 9297–. doi:10.1038/s41598-020-66333-x
- 13)Lakshmanaprabu Optimal deep learning model for classification of lung cancer on CT images. Future Generation Computer Systems, (2018), S0167739X18317011–. doi:10.1016/j.future.2018.10.009
- 14)Minh N Do and Countourlet transform An efficient directional multi resolutional image representation, IEEE Trans on image proc,(2003),14
- 15)Naik,Amrita and Damodar Reddy Lung Nodule Classification on Computed Tomography Images Using Deep Learning. Wireless Personal Communications, (2020), –. doi:10.1007/s11277-020-07732-1
- 16)Pandian R ,Vigneswaran T and Lalitha Kumari S Characterization of CT Cancer Lung Image Using Image Compression Algorithms and Featu Extraction, Journal of Scientific & Industrial Research, (2016), 75, 747-751.
- 17)Pandian R and Vigneswaran T, Adaptive wavelet packet basis selection for zerotree image coding, Int J of Sig and Imag Sys Eng 9 ,(2016), 388-392
- 18)Pandian R Evaluation of image compression algorithms, IEEE Und Tech (UT) NIOT, (2015) 1-3.
- 19)Riquelme,Diego and Moulay A. Deep Learning for Lung Cancer Nodules Detection and Classification in CT Scans. AI,(2020) 1(1), 28–67. doi:10.3390/ai1010003
- 20)Weszka, Joan S, Dyer, Charles R, Rosenfeld and Azriel, "A Comparative Study of Texture Measures for Terrain Classification",IEEE Transactions on Systems, Man, and Cybernetics: Systems,1976, SMC-6 , 269-285.